Monotone Local Search with Sample-Dependent Running Times

Serge Gaspers
UNSW Sydney, Australia
serge.gaspers@unsw.edu.au

Abstract

Monotone Local Search turns an algorithm that extends a given partial solution into an algorithm that finds a solution from scratch: sample a few elements, hope that they lie in an unknown solution, and complete them. Its known analyses bound every call of the extension algorithm by a worst-case time that depends only on the number of sampled elements, and Esmer, Kulik, Marx, Neuen and Sharma (SODA 2024) showed that no algorithm that uses the extension algorithm only as a black box with this worst-case running time beats the original form by more than a polynomial factor. We let the running time of the extension algorithm depend on the actual sample. If it decides in time $O^*(2^{g(I,X,k)})$, for a polynomial-time computable g, whether a sample X can be completed to a solution with at most k elements, our main theorem gives a randomized algorithm with one-sided error in time $O^*(2^{\psi n})$ that accepts with probability at least $1/2$ whenever, for some solution S of size k in a universe of n elements and some t, the *exponential moment* $\sum_{X\subseteq S,|X|=t} 2^{-g(I,X,k)}$ is at least $2^{-\psi n}\binom{n}{t}$. A corollary, multivariate Monotone Local Search, generalizes the theorems of Fomin, Gaspers, Lokshtanov and Saurabh (STOC 2016 & J. ACM 2019) and of Gaspers and Lee (ICALP 2017).

For Max Leaf Spanning Tree and, equivalently, Connected Dominating Set, the previous fastest known algorithms run in time $O^*(1.8619^n)$ in polynomial space and $O^*(1.8139^n)$ in exponential space. Completing a sample is a Steiner tree problem, solved in time exponential in the number of terminals, which varies strongly with the sample. We obtain a randomized polynomial-space algorithm with one-sided error in time $O(1.6991^n)$ and a deterministic polynomial-space algorithm in time $(\sqrt{5}-\frac{1}{2})^n 2^{o(n)} = O(1.7361^n)$. For connectivity problems in which the running time of the extension algorithm doubles with every component of the sample, an analysis through a partition function gives randomized polynomial-space algorithms in time $O^*(1.8^n)$ for Connected Feedback Vertex Set and $O(1.6444^n)$ for Connected Vertex Cover, both also with vertex weights.

Companion site (reading version, ancillary programs and certificates, Lean formalization):
https://serggasp.github.io/sample-MLS-site/

Contents

1 Introduction **1**

2 Overview of the key ideas **8**

3 Monotone Local Search with sample-dependent running times **14**

4 Corollaries: mean exponents, linear savings and product form **23**

5 Connected Dominating Set and Max Leaf Spanning Tree **29**

6 Warm-up: sampling a connected dominating set with the mean running-time exponent **34**

7 Exponential moments of the Steiner extension: Algorithm RS-θ **37**

8 Closure sampling under exponential moments: Algorithm RSC **42**

9 Derandomization **47**

10 Deterministic algorithms in polynomial space **53**

11 Component-count running times **55**

12 Further applications **68**

13 Conclusion and open problems **75**

A Other levers **76**

B Related mechanisms **80**

C Linear savings, product-form running times and Max Leaf Spanning Tree **82**

D Deferred proofs **85**

E Audit of the applications **162**

F Computer certificates **165**

References **174**

1 Introduction

Many exact exponential-time algorithms [FK10] solve a *subset problem*: given an instance whose solutions are subsets of a universe of n elements, find a solution of a given size k. We write $O^*(f(n))$ for $f(n)$ times a factor polynomial in the input length. When the decimal base of our running times is rounded up from a strictly smaller constant, the big-O notation absorbs the polynomial factor, so we write $O(1.7204^n)$ rather than $O^*(1.7204^n)$.

A standard way to solve a subset problem faster than 2^n combines a parameterized algorithm with enumeration. Suppose that solutions of size k can be found in time $O^*(c^k)$ with $c < 4$. For each k, run this algorithm or enumerate all $\binom{n}{k}$ candidate sets, whichever is faster; the running time is $O^*\left(\max_k \min\left(c^k, \binom{n}{k}\right)\right)$. At $k = n/2$ the two terms are $c^{n/2}$ and about 2^n, and if $c < 4$ then for every fraction k/n one of the two terms is exponentially smaller than 2^n.

Monotone Local Search [FGLS19] beats 2^n whenever the *extension* version of the subset problem is solved in time $O^*(c^{k-t})$, for *any* constant c. In the extension version, a size-t part of the solution is given and the question is whether the part can be completed to a full solution of size k. Monotone Local Search draws a random set X of t elements, hopes that it lies inside an unknown solution S, and asks an extension algorithm whether X can be completed by at most $k-t$ further elements. If the extension algorithm runs in time $O^*(c^{k-t})$, the resulting algorithm runs in time $O^*((2-1/c)^n)$. Gaspers and Lee [GL17] allowed a further factor b^{n-t} in the running time of the extension algorithm, giving running time $O^*((b+1-1/c)^n)$. We show that an extension algorithm with running time $O^*(b^{n-t}c^{k-t}d^t)$, for constants $b, c, d \geq 1$ with $d \leq bc$, gives the multivariate bound $O^*((b+d-d/c)^n)$ (Corollary 4.6). In all these versions the running-time bound of the extension algorithm depends only on n, t and k: every sample of a given size is charged the same, worst-case, running time.

As long as the extension algorithm is used only through such a worst-case running time, the bound $(2-1/c)^n$ cannot be improved, and by a remark in [EKM+24] neither can $(b+1-1/c)^n$. [EKM+24] consider algorithms whose only problem-specific access is a polynomial-time feasibility test and an extension algorithm with running time $O^*(c^{k-|X|})$, and they show that every randomized algorithm of this kind takes time at least $(2-1/c)^n$ divided by a polynomial in n; their concluding section remarks that the same technique shows the bound of Gaspers and Lee to be optimal. To go below these bounds, an algorithm must therefore use more information about the extension algorithm than its worst-case running time (Section 1.3).

This paper lets the running time of the extension algorithm depend on the sample. Our main theorem (Theorem 3.5) applies to every extension algorithm that, given (I, X, k), decides whether some set of at most $k - |X|$ further elements completes X to a solution, in time $O^*(2^{g(I,X,k)})$, for some *running-time exponent* g that is computable in polynomial time before the extension algorithm is called (Definition 3.2). For a solution S of size k and a sample size t, the *exponential moment* of S is

$$\mu_t(S) = \sum_{X \in \binom{S}{t}} 2^{-g(I,X,k)}.$$

The theorem requires one solution S whose exponential moment, for some t, is large compared with the number $\binom{n}{t}$ of possible samples.

Theorem 1.1 (informal version of Theorem 3.5)**.** *Suppose that the extension algorithm runs in time $O^*(2^{g(I,X,k)})$ for a running-time exponent g that is computable in polynomial time, and let $\psi \geq 0$ be a rational constant. There is a randomized algorithm that, given I and k, runs in time $O^*(2^{\psi n})$, and in polynomial space if the extension algorithm uses polynomial space; it rejects if I has no solution of size at most k, and it accepts with probability at least $1/2$ if some solution S of size k satisfies $\mu_t(S) \geq \binom{n}{t}2^{-\psi n}$ for some t.*

Theorem 1.1 follows from Theorem 3.5 with $\Gamma = \kappa = 1$. The algorithm is simple: it tries every sample size and every integer threshold θ, and for each pair it draws about $2^{\psi n - \theta}$ samples and calls the extension algorithm only on those with $g(I, X, k) \leq \theta$. It never needs to know which sample size or threshold is the right one. For one sample size, the probabilistic core of the theorem is the counterpart of Scaman's analysis of restart strategies [Sca23], and the mean-exponent form (Theorem 4.1) is the counterpart of Zamir's Markov cutoff [Zam23] (Appendix B).

What is new compared with restarts and cutoffs. One might summarize the algorithm as "run the extension algorithm only on samples whose running time is below a threshold, and optimize the restart schedule". Five features separate our setting from restart and cutoff strategies for Las Vegas algorithms, and the analyses in this paper use all five.

(i) The running-time exponent $g(I, X, k)$ is computed in polynomial time *before* the extension algorithm is called, so a sample above the threshold is skipped at polynomial cost instead of being run and cut off.

(ii) A call can succeed only if $X \subseteq S$ for an unknown solution S, so the moment $\mu_t(S)$ sums $2^{-g(I,X,k)}$ over the subsets of S only; a sample outside every solution plays the role of a run that never succeeds. The algorithm thus balances the number of samples inside S with a small running-time exponent against the running times of their extension calls.

(iii) The running-time exponent depends on the structure of the sample X, not only on its size.

(iv) The generating function W_S defined below collects the moments of all sample sizes, so no sample size has to be guessed.

(v) W_S is bounded from below by combinatorial recursions over the structure of the sampled subsets of S, such as dynamic programming along a spanning tree of S; all our applications are analyzed in this way.

Using the exponential moment instead of the mean of g can gain an exponential factor. If half of the t-subsets of S have running-time exponent $n/4$ and the other half have exponent $3n/4$, then the worst case charges running time $2^{3n/4}$ for a sample in S and the mean charges $2^{n/2}$, whereas the exponential moment charges at most $2^{n/4+1}$, so the faster half determines it. Section 2.1 gives the extreme case of exponents 0 and n. In general Jensen's inequality shows that the exponential moment never certifies a larger running time than the mean. Within the class of algorithms that sample uniformly, threshold on g and are charged 2^θ for a call at threshold θ, exponential moments measure, up to polynomial factors, exactly what thresholded sampling can use (Theorem 3.12), and in a black-box model the theorem cannot in general be derandomized without exponential loss (Section 9.2). These optimality statements are limited to the models in which they are proved; they are not lower bounds against other algorithms for the problems themselves. For concrete extension algorithms, we bound exponential moments from below via generating functions along a spanning tree of the solution. Adapting an entropy argument of Csikvári and Lin [CL14] to weighted counts, we show that paths are the worst connected solutions for the tree partition function that bounds these generating functions from below, up to a constant factor.

1.1 Why sample-dependent running times

The motivating example is Max Leaf Spanning Tree (MaxLeaf): given a connected graph G, find a spanning tree with as many leaves as possible. For $n \geq 3$, the maximum number of leaves is

$n - \gamma_c(G)$, where $\gamma_c(G)$ is the minimum size of a connected dominating set, and optimal solutions of MaxLeaf and of Connected Dominating Set, the problem of finding a connected dominating set of minimum size, translate into each other in polynomial time (Lemma 5.1). So a running-time bound for one of them holds for the other up to a polynomial factor, and we state our results for both. Table 1 lists the previous bounds for the two problems, in polynomial and in exponential space, together with ours; the fastest previously published polynomial-space bound, 1.8619^n, dates from 2011, setting aside a claimed k-leaf algorithm that is deemed incorrect (Section 1.3).

Completing a sampled set X to a connected dominating set is a Steiner tree problem, as in the algorithm of Abu-Khzam, Mouawad and Liedloff [AKML11]: the terminals are the vertices that X does not dominate and the components of $G[X]$. Inclusion–exclusion over branching walks [Ned13] solves it in time $2^{|\mathrm{T}(J_X)|}n^{O(1)}$ and polynomial space through a directed Steiner encoding (Lemmas 5.3 and 5.6), where J_X is the residual instance left by X (Section 5.2). Denoting the closed neighborhood of X by $N[X]$ and the number of connected components of $G[X]$ by $\mathrm{comp}(G[X])$, the number of terminals is

$$|\mathrm{T}(J_X)| = n - |N[X]| + \mathrm{comp}(G[X])$$

by Corollary 5.8. This running time depends on the sample, and a single sampled vertex can be worthless: if its closed neighborhood is already dominated and it touches exactly one component of $G[X]$, adding it to X leaves the number of terminals unchanged. Accounting that charges every sample of a given size its worst case therefore credits sampling with nothing. Sampling leaves of the spanning tree instead meets another obstacle: the extension problem of MaxLeaf parameterized by the number ℓ' of additional leaves is W[1]-hard (Theorem C.2), so it has no algorithm with running time $O^*(c^{\ell'})$ for a constant c unless FPT = W[1], and for running times $b^{n-|X|}d^{|X|}$ (the product form with $c = 1$) the main theorem certifies, for the worst solution size, nothing below a single call of the extension algorithm on the empty sample (Appendix C.2).

An average sample, by contrast, is valuable. If S is a connected dominating set of size $s = \sigma n$ and X is a uniform t-subset of S with $t \geq 1$, then

$$\mathbb{E}\,|\mathrm{T}(J_X)| \leq n - 1 - \frac{t(n-2)}{s}$$

by Lemma 5.11: a sampled vertex removes on average about $1/\sigma$ terminals, which divides the running time of the extension algorithm by about $2^{1/\sigma}$, while the requirement that it lie in S lowers the success probability only by a factor of about $1/\sigma$. As $2^{1/\sigma} > 1/\sigma$, an average sampled vertex pays for itself, and sample-dependent running times turn this average gain into a faster algorithm.

1.2 Our results

We make three contributions. First, sample-dependent Monotone Local Search: the main theorem with exponential moments and its generating-function form (Theorem 3.5 and Lemma 3.9). Second, Algorithm RSC with graded closure sampling, which solves Max Leaf Spanning Tree and Connected Dominating Set in time $O(1.6991^n)$, together with a deterministic algorithm (Theorem 1.2). Third, the component-count principle: for extension algorithms whose running time doubles with every component of the sample, paths are the worst connected solutions up to a constant factor (Lemma 11.2), which gives our algorithms for Connected Feedback Vertex Set and Connected Vertex Cover (Theorem 1.3). Further consequences of these techniques are collected in Table 2. The companion site [Gas26] has a reading version of this paper together with the ancillary programs, the certificates and the Lean development.

Monotone Local Search with exponential moments. Classical Monotone Local Search with sample size t draws about $\binom{n}{t}/\binom{k}{t}$ samples before one of them lies inside S, and it budgets each call of the extension algorithm its worst-case running time 2^γ, so its running time is about $\binom{n}{t}\binom{k}{t}^{-1}2^\gamma$. In $\mu_t(S)$, each t-subset X of S contributes $2^{-g(I,X,k)}$, the reciprocal of the running time of the extension algorithm on X. If g takes the same value γ on all these subsets, then $\mu_t(S) = \binom{k}{t}2^{-\gamma}$ and the running time $\binom{n}{t}/\mu_t(S)$ of the main theorem is the classical one. In general, let γ be the largest value of g on these subsets. A subset X with $g(I,X,k) = \gamma - j$, whose running-time bound is 2^j times smaller, contributes 2^j times as much to $\mu_t(S)$ as a subset with $g(I,X,k) = \gamma$. So the subsets on which the extension algorithm is fast can make $\mu_t(S)$ exponentially larger than $\binom{k}{t}2^{-\gamma}$, and the running time $\binom{n}{t}/\mu_t(S)$ exponentially smaller than the classical one.

The main theorem (Theorem 3.5) is stated for arbitrary implicit set systems, real-valued running-time exponents, and a polynomial loss factor. Its condition $(\star)$ asks that $\mu_t(S) \geq \binom{n}{t}2^{-\psi n}$ for some t, up to that factor. To avoid a separate analysis for each t, Lemma 3.9 collects all sample sizes in one polynomial in a variable y, the *generating function* $W_S(y) = \sum_t \mu_t(S)\, y^t = \sum_{X\subseteq S} y^{|X|}2^{-g(I,X,k)}$: up to a polynomial factor, the smallest running time $2^{\psi n}$ for which the theorem applies to a solution S is $\inf_{y\geq 0}(1+y)^n/W_S(y)$. Our analyses of the applications bound $W_S(y)$ from below. Exponential moments are the right quantity for this class of algorithms. Condition $(\star)$ is necessary, up to a polynomial factor, for the algorithm to call the extension algorithm on any subset of S at all (Proposition 3.10), and no algorithm that draws uniform samples, chooses sample sizes and thresholds θ on g, even adaptively, and calls the extension algorithm only on samples with $g \leq \theta$, is faster on any instance by more than a polynomial factor, when each trial is counted with the running time 2^θ that its threshold allows (Theorem 3.12). The theorem extends to randomized trials, such as samples from any polynomial-time samplable distribution and extension algorithms with one-sided error; it needs only that a trial at threshold θ runs in time at most 2^θ and succeeds whenever a random variable attached to it is at most θ (Theorems 3.14 and A.2 and Corollary 3.15).

Corollaries. If only a bound λ on the *mean* of g over the t-subsets of every solution is known, Jensen's inequality turns it into a bound on the exponential moment, and the running time is $O^*\left(\binom{n}{t}\binom{k}{t}^{-1}2^\lambda\right)$ (Theorem 4.1). This bound also has a simple, direct proof, by a cutoff at $\lambda+1$ and Markov's inequality (Remark 4.2). If the mean decreases by ρ bits per sampled element, the exponent has a closed form, and sampling pays exactly when $\rho > \log(1/\sigma)$ at solution fraction $\sigma = k/n$ (Corollary 4.3). For running times of product form $b^{n-|X|}c^{k-|X|}d^{|X|}$ the generating function has a closed form, which gives the multivariate bound $\left(b + \min(d, bc)(1-1/c)\right)^n$ for all $b, c, d \geq 1$, and hence the theorems of [FGLS19] ($b = d = 1$) and [GL17] ($d = 1$); conversely, the main theorem certifies nothing better for such running times (Proposition 4.5 and Corollary 4.6). Every gain in this paper therefore comes from the dependence of the running time on the samples.

Applications. Tables 1 and 2 compare our running-time bounds with the previous ones. The rest of this subsection describes the algorithms and the ideas behind them. All our algorithms use polynomial space, and the randomized ones have one-sided error.

Max Leaf Spanning Tree and Connected Dominating Set. All four algorithms complete a sample with the Steiner tree algorithm described above, whose running time is exponential in the number of terminals. The randomized ones never output a value below $\gamma_c(G)$, and they output $\gamma_c(G)$ with constant probability.

Theorem 1.2 (Theorems 8.1 and 10.4)**.** Max Leaf Spanning Tree *and* Connected Dominating Set *can be solved in polynomial space by a randomized algorithm with one-sided error in time* $O(1.6991^n)$*, and by a deterministic algorithm in time* $(\sqrt{5}-\frac{1}{2})^n 2^{o(n)} = O(1.7361^n)$*.*

- Algorithm RS uses only the mean number of terminals of a sample (Lemma 5.11), through the linear-savings corollary, and runs in time $O(1.7449^n)$ (Theorem 6.4).
- Algorithm RS-θ is the same algorithm run with a smaller target running time $2^{\psi n}$, that is, a smaller *rate* ψ; the improvement lies in the analysis. Exponential moments are not additive, so the mean bound no longer suffices. Instead we bound the number of terminals by a potential that is local along a spanning tree of the solution, evaluate the generating function of this potential by a three-state recursion, and bound the recursion from below by a convex region that the merge rule maps into itself, following Rote [Rot19] and Rosenfeld [Ros21] (Section 2.2). This gives $O(1.7204^n)$ (Theorem 7.2). On the uniform $(2,2)$-lobsters, and on graphs whose unique maximum-leaf spanning tree is such a lobster, the algorithm needs time $\Omega(1.7192^n)$, so the analysis is nearly tight for this algorithm.
- Algorithm RSC adds *closure sampling*: every vertex with at least two sampled neighbors is added to the sample independently, with probability 0.07 if it has exactly two and 0.077 if it has more. In the tight case of the mean bound the solution induces a star, and its center is the only vertex that can have two or more sampled neighbors. The budgeted form of the main theorem applies to these random trials and gives $O(1.6991^n)$ (Theorem 8.1).
- The deterministic algorithm uses more than the values of the running-time exponent. In a black-box model, every deterministic algorithm is, on some instances, slower than the randomized one by a factor exponential in n, or has cost $\binom{n}{k}$ up to a polynomial factor, even for the mean form of the main theorem (if the mean bound, at least 1, is below $\log\binom{k}{t}$ by a linear amount; Section 9 and Appendix D.6). But the number of Steiner terminals is at most an affine function of the sample with nonnegative integer weights, an *integer affine certificate*. Balanced set-inclusion families, which we list deterministically in polynomial space, derandomize the exponential moments of such certificates (Theorem 9.10). This gives a deterministic polynomial-space algorithm in time $O(1.7361^n)$ (Theorem 10.4).

Connected Feedback Vertex Set. Our extension algorithm is a Cut & Count algorithm, built on the $O^*(3^k)$ algorithm of [CNP+22], that, given a sample X, runs in time $O^*(3^{k-|X|}2^{\mathrm{comp}(G[X])})$: its running time doubles with every component of the sample, so its product form $(b,c,d) = (1,3,2)$ gives nothing. A random subset of a connected solution, however, has far fewer components than vertices once the sample is a sizeable part of the solution. We bound the generating function W_S from below by a partition function on a spanning tree of S (Corollary 11.5), and a weighted form of a lower bound of Csikvári and Lin [CL14] on the number of homomorphisms of a tree into a graph shows that, up to a constant factor, this partition function is smallest on paths (Lemma 11.2). The main theorem then gives time $O^*(1.8^n)$ (Theorem 11.8), against the baseline $O(1.9526^n)$ obtained by balancing $O^*(3^k)$ against enumeration. The analysis is tight for this algorithm: on a family of graphs, every base below 1.8 leads to success probability $o(1)$ (Proposition 11.10).

Other problems with component-count running times. Let $c > 1$ be a constant, and consider a problem on a graph G whose solutions induce connected subgraphs of G. For every extension algorithm that runs in time $O^*(c^{k-|X|}2^{\mathrm{comp}(G[X])})$, the Gibbs-type inequality and the

Bound	Type	Space	Year	Source	Method
1.9407^n	deterministic	polynomial	2008	[FGK08]	measure and conquer
1.8966^n	deterministic	polynomial	2011	[FKK+11]	measure and conquer
1.8962^n	deterministic	polynomial	2014	[BRF14]	measure and conquer
1.8619^n	deterministic	polynomial	2011	[AKML11]	branching, Steiner tree
1.8139^n	deterministic	exponential	2012	[BRF12]	algorithm for digraphs
1.7449^n	randomized	polynomial		Theorem 6.4	mean number of terminals
$\mathbf{1.7361^n}$	**deterministic**	polynomial		Theorem 10.4	affine certificate
1.7204^n	randomized	polynomial		Theorem 7.2	exponential moments
$\mathbf{1.6991^n}$	**randomized**	polynomial		Theorem 8.1	closure sampling

Table 1: Bounds for Connected Dominating Set and Max Leaf Spanning Tree on n-vertex graphs. The 1.8139^n algorithm is for digraphs and applies to G with each edge replaced by two opposite arcs. Randomized algorithms have one-sided error. Our four bases rest on computations checked in exact or ball arithmetic (Appendix F); Table 4 maps the proof of the base 1.6991.

main theorem give an algorithm with running time $O^*(((4c-3)/(2c-1))^n)$ (Corollary 11.6). The case $c = 2$, where a sharper analysis gives 1.6444 (Lemma 11.12), covers two problems:

- Connected Vertex Cover. With Cygan's weighted $O^*(2^k)$ algorithm [Cyg12] as the extension step, we find a minimum-weight connected vertex cover, for polynomially bounded integer weights, in time $O(1.6444^n)$ (Theorem 11.15).
- Graph Motif. Our extension algorithm contracts each component of the sample to one vertex with a new color and then runs the $O^*(2^k)$ algorithm of Björklund, Kaski and Kowalik [BKK16], in time $O^*(2^{k-|X|}2^{\mathrm{comp}(G[X])})$. The exponential moments give $O(1.6444^n)$ (Theorem 11.13).

For polynomially bounded integer weights, the same analysis gives $O(1.6444^n)$ for Maximum-Weight Connected k-Subgraph and $O^*(1.8^n)$ for vertex-weighted Connected Feedback Vertex Set (Theorems 11.16 and 11.17). For the two problems that we consider the main applications of the component-count principle, we obtain the following.

Theorem 1.3 (Theorems 11.8, 11.15 and 11.17)**.** (Weighted) Connected Feedback Vertex Set *and* (Weighted) Connected Vertex Cover *can be solved in polynomial space by randomized algorithms with one-sided error in time* $O^*(1.8^n)$ *and* $O(1.6444^n)$*, respectively, for vertex weights in* $\{1, \ldots, w_{\max}\}$ *with* $w_{\max} \leq n^{O(1)}$*.*

Further applications. The analyses of Algorithm RS-θ and of the deterministic algorithm use only a spanning tree of the solution and an assignment of each vertex outside the solution to a solution vertex that dominates it, so they extend to Connected r-Dominating Set and to vertex-weighted Connected Dominating Set and weighted MaxLeaf, with randomized $O(1.7204^n)$ and deterministic $O(1.7361^n)$ running times, and to Weakly Connected Dominating Set and Directed Max Leaf Spanning Tree, with a deterministic $O(1.7361^n)$ running time (Theorems 12.2 to 12.5); closure sampling does not extend in this way (Remark 12.6). The inclusion–exclusion formula for vertex-weighted directed Steiner trees behind these extensions (Lemma 5.3) is also the extension algorithm for Directed Steiner Tree, which gives randomized

Problem	Previous	New	Type	Source
Main applications of the component-count principle				
Connected Feedback Vertex Set	1.9526^n [CNP+22]*	1.8^n	rand.	Thm. 11.8
Connected Vertex Cover	1.7088^n [GHK18]	1.6444^n	rand.	Thm. 11.15
weighted	1.7088^n [Cyg12]*	1.6444^n	rand.	Thm. 11.15
Further consequences				
Graph Motif	1.7088^n [BKK16]*	1.6444^n	rand.	Thm. 11.13
Connected r-Dominating Set, $r \geq 2$	1.8619^n [AKML11]†	1.7361^n	det.	Thm. 12.2
		1.7204^n	rand.	Thm. 12.2
Connected Dominating Set, weighted	1.9896^n [AKFG+25]†	1.7361^n	det.	Thm. 12.3
		1.7204^n	rand.	Thm. 12.3
Weakly Connected Dominating Set	1.8619^n [AKML11]†	1.7361^n	det.	Thm. 12.4
Directed Max Leaf Spanning Tree	1.9044^n [BRF12]	1.7361^n	det.	Thm. 12.5
Directed Steiner Tree	$2^{n/2}$ (see §12.5)	1.3727^n	rand.	Thm. 12.8
Paired Domination	1.7159^n [ABK+24]	1.6302^n	rand.	Thm. 12.10

Table 2: Component-count applications and further consequences, on n-vertex graphs. All bounds in the table use polynomial space, and the randomized ones have one-sided error. Every new base except 1.8 rests on a computation checked in exact or ball arithmetic (Appendix F). Weighted problems have polynomially bounded integer weights. *: baseline derived by balancing the cited algorithm against enumeration. †: by a reduction to the cited algorithm (Sections 12.1 to 12.3). Other previous bounds: 1.3533^n for undirected Steiner Tree [FGK+13, Theorem 5], and 1.8139^n in exponential space for Directed Max Leaf Spanning Tree [BRF12].

$O(1.3727^n)$ (Theorem 12.8). For Paired Domination, a Tutte-matrix finishing step gives randomized $O(1.6302^n)$ (Theorem 12.10); both bounds also hold with polynomially bounded integer vertex weights.

1.3 Techniques and related work

Monotone Local Search. Section 4.3 recalls classical and multivariate Monotone Local Search [FGLS19, GL17]. [EKM+22, EKM+26] developed approximate Monotone Local Search, after Lee's thesis [Lee21] had applied Monotone Local Search with exact and approximate extension algorithms to approximation and computed the running times numerically, and they showed it to be optimal in the extension-oracle model described earlier [EKM+24]. Their oracle lower bounds use the same planting idea as our black-box barrier (Section 9.2). Esmer and Kulik [EK25] sample with a black box in which the number of correct samples is random, but the running time of each black-box call is fixed by the algorithm. Theorem 3.5 reads a sample-dependent running-time exponent before the extension call, which is information outside the oracle model of [EKM+24], so their lower bounds do not apply; for product-form running times it gives the classical bound up to a polynomial factor (Proposition 4.5), so it does not contradict them.

Restarts, cutoffs and exponential moments. The probabilistic core of Theorem 3.5 has counterparts outside Monotone Local Search: restart strategies with cutoffs for Las Vegas algorithms [LSZ93, AGM+96], the Markov cutoff of Zamir [Zam23], which corresponds to the mean-

exponent theorem (Theorem 4.1), the bounds of Scaman [Sca23] on restart strategies through the mean of the reciprocal running time, and the exponential moment of the number of guessed variables in the analysis of [PPSZ05]. [FGLS19, GL17] already used generating functions in Monotone Local Search, for product-form running times. Appendix B compares these with our setting.

Certificates. The running times of RS-θ and RSC follow from lower bounds on the generating function W_S of a solution S, which obeys a recursion along a spanning tree of S. We prove these lower bounds by antinorm certificates in a cumulative cone. This adapts the method that Rote [Rot19] introduced to count minimal dominating sets in trees, generalized by Rosenfeld [Ros21], from upper to lower bounds; for lower bounds, antinorms in the sense of Guglielmi and Protasov [GP13] take the place of norms. Measure-and-conquer analyses of branching algorithms [FGK09] bound the number of leaves of a search tree from above; our certificates bound generating functions from below.

A caveat. Koutis and Williams [KW09] claimed a randomized $O^*(2^k)$ algorithm that decides whether a graph has a spanning tree with at least k leaves; balanced against enumeration, it would give randomized time $O(1.7088^n)$. This algorithm is deemed incorrect [BRF14, footnote (i)][Zeh15, footnote 1], and the k-leaf application does not appear in the journal version [KW16]. We know of no repair; our 1.6991^n is below 1.7088^n in any case.

Derandomization. [FGLS19] derandomize product-form Monotone Local Search with set-inclusion families, which multiplies the running time by $2^{o(n)}$. For sample-dependent running-time exponents this route fails in general. Consider the class of instances whose running-time exponents satisfy the mean bounds that Algorithm RS for Connected Dominating Set uses. On this class, every deterministic algorithm that may use the running-time exponent and the extension algorithm only as oracles must spend running time at least $1.7547^n/\operatorname{poly}(n)$ in that model, where an extension call on a sample X takes time $2^{g(I,X,k)}$ and every other oracle call takes unit time, whereas the randomized algorithm of the linear-savings corollary needs time $O^*(\sqrt{3}^n)$ when the solution size is $n/2$ (Corollary 9.3; the proofs are in Appendix D.6). Our deterministic algorithms use the affine structure of the running-time exponent instead, so the bound 1.7547^n, which concerns the class defined by the mean bounds alone, does not apply to them. [EKM$^+$22, EKM$^+$26] generalize set-inclusion families to set-intersection families and to known weights; in their extension-oracle model a query costs c^ℓ for an extension budget ℓ, whatever the sample, and deterministic and randomized running times match up to a factor $2^{o(n)}$ [EKM$^+$24, EKM$^+$26]. The lopsided universal families of [FLPS16] also give set-inclusion families, but in general exponentially larger ones.

2 Overview of the key ideas

This section explains informally why the main theorem has the form it has, and which ideas give the bounds for Connected Dominating Set and for the problems with component-count running times. Every statement in the first four subsections is proved later, at the place cited. Section 2.5 reports recent claims about the Exponential Time Hypothesis together with our assessment of what they imply for the problems of this paper, supported by informal arguments that are not developed into proofs here.

2.1 Why exponential moments

Fix an instance with a solution S of size k in a universe of n elements, and a sample size t. A uniformly random t-subset X of the universe lies in S with probability $\binom{k}{t}/\binom{n}{t}$. In classical Monotone Local Search every extension call has the same running time $O^*(c^{k-t})$, so it finds S in about $\binom{n}{t}/\binom{k}{t}$ trials of running time $O^*(c^{k-t})$ each, and optimizing over t gives the running time $O^*((2-1/c)^n)$. When the running time $2^{g(X)}$ of an extension call depends on X, where we write $g(X)$ for the running-time exponent of the sample X when the target size k is fixed, the natural first attempt runs the extension algorithm on every sample. Its expected running time per trial is the mean $\mathbb{E}[2^{g(X)}]$, which the slowest samples dominate.

The algorithm of Theorem 3.5 instead runs the extension algorithm only on samples whose running-time exponent, computed in polynomial time beforehand, is at most a threshold θ, and discards the other samples. A trial of the pair (t,θ) then runs in time $O^*(2^\theta)$, and it succeeds with probability $p_\theta = \Pr[X \subseteq S \text{ and } g(X) \le \theta]$, so this pair finds S in expected time about $2^\theta/p_\theta$. The best threshold is characterized by the following two-sided estimate (Lemma 3.8). Write $\xi = g(X)$ if $X \subseteq S$ and $\xi = \infty$ otherwise. Since $2^{-\theta}\mathbf{1}[\xi \le \theta] \le 2^{-\xi}$ pointwise, every threshold satisfies $2^{-\theta}p_\theta \le \mathbb{E}[2^{-\xi}]$. Conversely, splitting $\mathbb{E}[2^{-\xi}]$ according to the integer part of ξ shows that one of the thresholds up to about $\log(1/\mathbb{E}[2^{-\xi}])+1$ attains $2^{-\theta}p_\theta \ge \mathbb{E}[2^{-\xi}]/\operatorname{poly}(n)$. Moreover,

$$\mathbb{E}[2^{-\xi}] = \frac{1}{\binom{n}{t}} \sum_{X \in \binom{S}{t}} 2^{-g(X)} = \frac{\mu_t(S)}{\binom{n}{t}},$$

the normalized exponential moment of the introduction. So the best threshold for sample size t finds S in time $O^*\left(\binom{n}{t}/\mu_t(S)\right)$. The algorithm knows neither the right t nor the right θ, but there are only polynomially many pairs, and it gives each the same running-time budget $2^{\psi n}$; it succeeds as soon as $\binom{n}{t}/\mu_t(S) = O^*(2^{\psi n})$ for some t, which is condition $(\star)$.

Thresholding can be exponentially better than paying the mean. Suppose that half of the t-subsets of S have running-time exponent 0 and the other half n. Running the extension algorithm on every sample in S takes time about 2^{n-1} on average, the mean-exponent theorem (Theorem 4.1) charges $2^{\mathbb{E}[g]} = 2^{n/2}$ per sample, and the exponential moment charges $1/\mathbb{E}[2^{-g}] \le 2$. In general, Jensen's inequality gives $\mathbb{E}[2^{-g}] \ge 2^{-\mathbb{E}[g]}$, so the exponential moment never certifies a worse running time than the mean, and it certifies a better one whenever the running-time exponent varies.

The same estimate shows that nothing in this class of algorithms does better. Any algorithm that draws uniform samples, picks the sample size and the threshold of each trial, possibly adaptively, and is charged 2^θ for a trial with threshold θ, succeeds with probability at most its total charge times $\max_{t,\theta} 2^{-\theta}\Pr[X \text{ extends to a solution and } g(X) \le \theta]$, and by the estimate above this maximum is at most $\max_t \mu_t/\binom{n}{t}$, where μ_t sums $2^{-g(X)}$ over all t-sets X that extend to a solution (Theorem 3.12). The exponential moments are therefore not an artifact of the analysis: they measure exactly what thresholded sampling can exploit.

One generating function instead of $k+1$ moments. To verify condition $(\star)$ we use the generating function $W_S(y) = \sum_{X \subseteq S} y^{|X|}2^{-g(X)} = \sum_t \mu_t(S)y^t$. Comparing it termwise with $(1+y)^n = \sum_t \binom{n}{t}y^t$ shows that the running time certified for S is $\inf_{y \ge 0}(1+y)^n/W_S(y)$, up to a factor $n+1$ (Lemma 3.9). Upper bounds on running times thus become lower bounds on a sum over the subsets of S, for a single well-chosen value of y; this is the form in which all our applications are analyzed. For a product-form running time, $2^{-g(X)} = b^{-(n-|X|)}c^{-(k-|X|)}d^{-|X|}$, the sum is $W_S(y) = b^{-n}c^{-k}(1+ybc/d)^k$, and at $y = (c-1)d/(bc)$ the second factor becomes c^k; this gives $(1+y)^n/W_S(y) = (b+d-d/c)^n$. This is the multivariate bound when $d \le bc$; for $d > bc$ the

choice $y = 0$ gives at most $(bc)^n$, which is smaller, hence the $\min(d, bc)$ in Corollary 4.6. This bound subsumes the bounds of [FGLS19] ($b = d = 1$) and of Gaspers and Lee [GL17] ($d = 1$) (Proposition 4.5 and Corollary 4.6). For such running times the main theorem certifies nothing better, so the gains below come only from the dependence of the running time on the sample.

2.2 Connected Dominating Set: from averages to the shape of the solution

The extension problem and its running time. Suppose we have a sample X that is a subset of an unknown minimum connected dominating set S. The extension algorithm must add a set Y of vertices that dominates the vertices outside $N[X]$ and such that $G[X \cup Y]$ is connected: the added vertices must join the components of $G[X]$ into one connected subgraph. This is a Steiner tree problem with one terminal for each undominated vertex and one for each component of $G[X]$, which inclusion–exclusion over branching walks [Ned13] solves in polynomial space and in time $O^*(2^{|\mathrm{T}(J_X)|})$, where $|\mathrm{T}(J_X)| = n - |N[X]| + \mathrm{comp}(G[X])$ is the number of terminals (Lemma 5.6 and Corollary 5.8).

An affine bound and the mean. Take a spanning tree F of G in which all vertices outside S are leaves, and give each $u \in S$ the weight $w(u) = d_F(u) - 1$. The weights add up to $n - 2$, and counting the vertices and edges of F around a nonempty sample gives the pointwise bound $|\mathrm{T}(J_X)| \leq n - 1 - w(X)$ (Lemma 5.10). Each vertex of S lies in a uniform t-subset of S with probability $t/|S|$, so on average a sampled vertex removes $(n-2)/|S| \approx 1/\sigma$ terminals, where $\sigma = |S|/n$ (Lemma 5.11). Each further sampled vertex multiplies the probability that the sample lies in S by about σ, so it multiplies the expected number of trials by about $1/\sigma = 2^{\log(1/\sigma)}$; the terminals it removes divide the running time of the extension algorithm by about $2^{1/\sigma}$. Since $1/\sigma > \log(1/\sigma)$ for every $\sigma \in (0, 1]$, the second factor wins at every solution size; the linear-savings corollary (Corollary 4.3) turns this into Algorithm RS and time $O(1.7449^n)$ (Theorem 6.4). The bound on the mean is tight, for instance, when the solution induces a star and every outside vertex has exactly one neighbor in it (Proposition 5.12).

Exponential moments along a tree. The mean cannot take advantage of the variation of the number of terminals between samples, but the exponential moments reward this variation. To use it, we bound the number of terminals from above by a potential $\Pi_{F,\nu}(X)$ along a spanning tree F of $G[S]$ (Lemma 7.3). Each vertex outside S is assigned a neighbor in S, its *owner*, and $\nu : S \to \mathbb{Z}_{\geq 0}$ is the map that gives, for each solution vertex v, the number ν_v of outside vertices that v owns. The potential adds up three counts: the outside vertices whose owner is not sampled, the solution vertices that are neither sampled nor adjacent in F to a sampled vertex, and the components of $F[X]$. The potential is *local* along F: since $\mathrm{comp}(F[X]) = |X| - e(F[X])$ for the forest $F[X]$, where $e(F[X])$ is the number of edges of F with both ends in X, the potential is a sum over the vertices $v \in S$ of terms that each depend only on ν_v and on which vertices of the closed neighborhood $N_F[v]$ are sampled. Because of this, its generating function $\sum_{X \subseteq S} y^{|X|} 2^{-\Pi_{F,\nu}(X)}$ can be computed by dynamic programming from the leaves of F to its root, with a vector of three numbers per vertex and a fixed bilinear rule that merges a child into its parent (Lemma 7.4). What we need is a lower bound on the result that holds for every tree at once. Following the method of Rote [Rot19] and Rosenfeld [Ros21] for counting dominating sets in trees, which we turn from upper into lower bounds, we look for a convex region $\mathcal{P}$ in $\mathbb{R}^3$ that is mapped into itself by the merge rule and contains the vector of every single vertex v up to a factor $e_1 e_2^{\nu_v}$, where ν_v is the number of outside vertices that v owns and $e_1, e_2 > 0$ are two constants chosen together with $\mathcal{P}$: a scale for each solution vertex and a scale for each outside vertex. Then the vector of every tree on the s solution

vertices lies in $e_1^s e_2^{n-s}\mathcal{P}$, and a lower bound of order $e_1^s e_2^{n-s}$ on the generating function follows (Lemma 7.5). The region is the convex hull of finitely many points, enlarged by a cone that the merge rule preserves; for lower bounds this plays the role that a norm plays in the method for upper bounds, and is an antinorm in the sense of Guglielmi and Protasov [GP13]. Two such regions for the middle solution sizes, with the closed form of RS elsewhere, give Algorithm RS-θ, which is RS with a smaller rate, and time $O(1.7204^n)$ (Theorem 7.2). On a family of lobsters it needs time at least 1.7192^n, so little remains to be gained for this algorithm (Proposition 7.9).

Changing the samples. The tight case of the mean bound, a star, suggests a change of the algorithm. A sample of leaves of the star dominates the hub but leaves every sampled leaf in a component of its own; adding the hub would merge these components into one. Algorithm RSC therefore adds every vertex with at least two sampled neighbors to the sample independently, with probability 0.07, or 0.077 if it has three or more sampled neighbors. The resulting samples are no longer uniform sets of a given size, but the budgeted form of the main theorem (Theorem 3.14) only needs trials that are fast, sound, and accept with probability at least $\Pr[\xi \le \theta]$ for a random variable ξ whose distribution does not depend on θ; then $\mathbb{E}[2^{-\xi}]$ averages over the random closure. An eight-state version of the tree recursion, which bounds the generating function from below, and a region of the same kind give time $O(1.6991^n)$ (Theorem 8.1).

Derandomization. Classical Monotone Local Search is derandomized by replacing random samples with a family of t-sets that has a member inside every k-set [FGLS19]. With sample-dependent running times this is not enough, since the member inside the solution may be one of the slowest samples. In a black-box model, where an algorithm sees the running-time exponent and the extension algorithm only through queries, an adversary can make slow exactly the samples that a deterministic algorithm queries. For target size $k = n/2$ and the instances that satisfy the mean bounds of Connected Dominating Set, this forces every deterministic algorithm to incur cost at least $1.7547^n/\operatorname{poly}(n)$, when an extension call on X takes time $2^{g(X)}$ and every other query unit time, while the randomized algorithm has cost only $O^*(\sqrt{3}^n)$ (Corollary 9.3). The affine bound $|\mathrm{T}(J_X)| \le n - 1 - w(X)$ above has more structure than the mean: the weights are unknown, but their total is known. A *balanced* family contains, inside every k-set and for every nonnegative weighting of it, a sample whose weight is at least the average; such a family plays the role of the random sample in the mean-exponent analysis. We construct balanced families deterministically and in polynomial space (Theorem 9.6). For integer weights, the exponential moments of the affine bound, $2^{-(n-1)}\prod_{u\in S}(1 + y2^{w(u)})$ in generating-function form, can also be achieved deterministically, with two-class balanced families, and this gives a deterministic polynomial-space algorithm in time $O(1.7361^n)$ (Theorem 10.4).

Other domination variants. These arguments use the graph only through a spanning tree of the solution and the assignment of every outside vertex to a solution vertex that dominates it. They therefore carry over to distance-r domination, vertex weights, weakly connected domination and directed graphs, and similar finishing steps give the bounds for Directed Steiner Tree and Paired Domination (Section 12).

2.3 Component-count running times: paths are the worst case

Consider Connected Feedback Vertex Set, Graph Motif and Connected Vertex Cover. When given a sample X of a connected solution S of size k, an extension algorithm runs in time $O^*(c^{k-|X|}2^{\operatorname{comp}(G[X])})$, with $c = 2$ or $c = 3$. A sample of size t can have up to t

components, so the product-form bound is $c^{k-t}2^t$, which gives nothing better than enumeration. Let F be a spanning tree of $G[S]$. Every edge of F inside X merges two components, so $\mathrm{comp}(G[X]) \le |X| - e(F[X])$, where $e(F[X])$ is the number of edges of F with both endpoints in X, and therefore

$$W_S(y) \;\ge\; c^{-k} \sum_{X \subseteq V(F)} \zeta^{|X|}\, 2^{e(F[X])} = c^{-k}\, \mathcal{Z}_F(\zeta), \qquad \zeta = cy/2,$$

a partition function on the tree in which every sampled vertex has weight ζ and every edge between two sampled vertices has weight 2 (Corollary 11.5). On a path with k vertices, $\mathcal{Z}_F(\zeta)$ is computed by the 2×2 transfer matrix $M_\zeta = \left(\begin{smallmatrix} 1 & \zeta \\ 1 & 2\zeta \end{smallmatrix}\right)$, whose rows and columns are indexed by the states "not sampled" and "sampled" of a vertex, and it grows like $\lambda_{\max}(M_\zeta)^k$, where $\lambda_{\max}(M_\zeta)$ is the largest eigenvalue of M_ζ. The key lemma (Lemma 11.2) shows that no tree does worse: $\mathcal{Z}_F(\zeta) \ge \min(1, \zeta)\lambda_{\max}(M_\zeta)^{k-1}$ for every tree with k vertices. The proof, a weighted form of an argument of Csikvári and Lin [CL14] for homomorphisms of trees, bounds $\ln \mathcal{Z}_F$ from below by the Gibbs variational inequality, using as test distribution the distribution of the labels given by a reversible two-state Markov chain run from the root down the tree, started in its stationary distribution. Then every vertex and every edge has the same label distribution, whatever the shape of the tree, and the bound is the same as for a path. Concretely, the Gibbs inequality bounds $\ln \mathcal{Z}_F(\zeta)$ from below by the entropy of the test distribution plus the expected logarithm of the weight $\zeta^{|X|}2^{e(F[X])}$. For the Markov chain both terms split into a term for the root and one term for each of the $k-1$ edges, and since every edge has the same joint label distribution, whichever endpoint is the parent, all edge terms are equal. The chain is built from the Perron eigenvector of a symmetric matrix with the same eigenvalues as M_ζ, and this choice makes each edge term exactly $\ln \lambda_{\max}(M_\zeta)$. The lemma is a statement about the partition function $\mathcal{Z}_F$, which bounds W_S from below; it does not say that paths are the hardest instances of the problems. With paths identified as the worst case of this lower bound, the choice of y is a one-dimensional optimization. It gives $O^*(((4c-3)/(2c-1))^n)$ in closed form for every $c > 1$ (Corollary 11.6), which is $O^*(1.8^n)$ for Connected Feedback Vertex Set, and no smaller base works for this algorithm (Proposition 11.10). For $c = 2$, further optimization gives $O(1.6444^n)$ for Graph Motif and Connected Vertex Cover (Theorems 11.13 and 11.15).

2.4 Computer certificates

The following bounds rest on computations: the bases 1.7449, 1.7204, 1.6991 and 1.7361 of RS, RS-θ, RSC and the deterministic algorithm (Theorems 6.4, 7.2, 8.1 and 10.4), the bases 1.6444 for $c = 2$, 1.3727 for Directed Steiner Tree and 1.6302 for Paired Domination (Theorems 11.13, 12.8 and 12.10), and the lower bound 1.7192^n for RS-θ on the uniform lobsters. For RS-θ and RSC, the points that span the regions $\mathcal{P}$ of Section 2.2 are found by computer: the certificate of RSC has 1024 rational points. Each certificate lists its points and is checked by programs that verify the finitely many inclusions in exact rational arithmetic. The two certificates of RS-θ are checked by three independent programs and the certificate of RSC by two; the second program for RSC was written from the definitions without reusing code of the first, re-derives the recursion and rejects perturbed copies of the certificate (Appendices F.2 and F.4). Both RS-θ certificates, and the reduction of the bound for RSC to the finite conditions of its certificate, are also checked in Lean 4 (Appendix F.9); there the conditions of the RSC certificate are checked as well, by compiled evaluation, which also trusts the Lean compiler. The lobster bound and the three certificates for Directed Steiner Tree are also checked in Lean, by the kernel alone, without compiled evaluation. The rates of these certificates, the constant of Theorem 6.4, the optimization over y behind the base 1.6444 for $c = 2$, and the finitely many cases behind the base 1.6302 are checked in exact

rational arithmetic, or in ball arithmetic with outward rounding where logarithms, exponentials and square roots occur. Appendix F says which statement rests on which program or certificate. The programs, the certificates and the Lean development are on the companion site [Gas26], together with a reading version of the paper that links each statement to them. The base 1.8 for Connected Feedback Vertex Set has a proof without computation (Corollaries 11.5 and 11.6), and the proof of the base $\sqrt{5} - \frac{1}{2}$ needs one algebraic identity and ten rational inequalities checked in exact arithmetic (Lemma 10.1).

2.5 The Exponential Time Hypothesis

Recent preprints by OpenAI [Ope26] and Claude (Anthropic) [Cla26] claim to refute the Exponential Time Hypothesis. This would give subexponential time algorithms for 3-SAT and imply subexponential time algorithms for every problem with a SERF reduction to 3-SAT [IPZ01]. This includes all problems in SNP and size-constrained SNP; for example, Vertex Cover, Independent Set and k-Set Cover. We are not aware of SERF reductions to 3-SAT for any of the problems in Tables 1 and 2, nor of subexponential-time algorithms for them under either claim. The obstacle that we see is a global constraint on the solution. All these problems except Paired Domination require the solution to be connected, and so they are not evidently in SNP: universal first-order constraints on a witness are preserved under induced substructures, whereas connectivity is not, since deleting the middle vertex of a path on three vertices disconnects it. Moreover, a natural CNF encoding of connectivity, by parent pointers of a spanning tree, uses $\Theta(n \log n)$ variables, and with $\Theta(n \log n)$ variables the claimed bounds do not give subexponential time. For Paired Domination, the subgraph induced by the solution must have a perfect matching, and a natural encoding by a partner pointer for each vertex also uses $\Theta(n \log n)$ variables.

2.6 Organization

Section 3 sets up implicit set systems, running-time exponents and exponential moments, and proves the main theorem and its general forms. It also shows that the algorithm is optimal in its class. Section 4 derives the mean-exponent theorem, the linear-savings corollary and multivariate Monotone Local Search. The applications are grouped by problem family. Section 5 prepares the applications to Connected Dominating Set and MaxLeaf and states the vertex-weighted directed Steiner lemma that the later sections use; Sections 6 to 8 present Algorithms RS, RS-θ and RSC; Section 9 states the black-box barrier and gives the derandomization for affine certificates; and Section 10 gives the deterministic algorithm. Section 11 treats component-count running times: it proves that paths are the worst case, up to a constant factor, of the partition function that bounds the exponential moments from below, and then treats Connected Feedback Vertex Set, Graph Motif, Connected Vertex Cover and their variants, one problem at a time. The section on further applications (Section 12) transfers the algorithms for Connected Dominating Set to connected r-domination, vertex weights, weak connectivity and directed graphs, and treats Directed Steiner Tree and Paired Domination (Sections 12.5 and 12.6). Section 13 lists open problems. The appendices treat samplable distributions and other levers (Appendix A), related mechanisms outside Monotone Local Search (Appendix B), linear savings, product-form running times and the hardness of MaxLeaf-Extension (Appendix C), and the computer certificates (Appendix F). Appendix D contains the proofs that the main text defers: optimality in its class (Appendix D.1), linear savings and product-form running times (Appendix D.2), the Steiner computations (Appendix D.3), the analysis of RS-θ and its lower bound (Appendix D.4), the closure bound of RSC (Appendix D.5), the black-box barrier (Appendix D.6), the balanced families and the

count of trees across block partitions (Appendix D.7), the deterministic algorithms (Appendix D.8), the extension algorithm and the tightness results for Connected Feedback Vertex Set (Appendix D.9), the weighted finishing steps (Appendix D.10), and the further applications, among them Directed Steiner Tree and Paired Domination with their vertex-weighted versions (Appendix D.11); Appendix E lists, for every application, the imported results, the randomness, the weights and the output.

3 Monotone Local Search with sample-dependent running times

This section proves the main theorem, on which the rest of the paper is built. It concerns an extension algorithm whose running time is exponential in a quantity that depends on the sample, and it says that sampling and completing finds a solution quickly as soon as the extension algorithm is fast on enough subsets of a single solution. "Enough" is measured by the sum of 2^{-x}, for running time 2^x, over the subsets of the solution of a given size, which we call an exponential moment, compared with the number of subsets of the universe of that size.

Section 3.1 fixes the model. Section 3.2 presents Algorithm MLS-θ, the main theorem and its proof. Sections 3.3 to 3.5 show that the condition of the theorem can be phrased through a generating function, that this condition is necessary for the algorithm, and that the algorithm is optimal, up to polynomial factors, among all algorithms that draw uniform samples, run the extension algorithm only on samples whose running-time exponent is at most a chosen threshold, and are charged 2^θ for a trial with threshold θ. Section 3.6 gives the general form that later sections use: budgeted trials, which include one-sided-error extension algorithms. Appendix B compares the main theorem with restart strategies, the Markov cutoff and earlier uses of exponential moments. Appendix A treats arbitrary samplable distributions and records which further changes of the trials can and cannot lower the running time by more than a polynomial factor. The mean-exponent theorem, linear mean savings and multivariate Monotone Local Search are derived from Theorem 3.5 in Section 4.

3.1 Setting

We write log for the logarithm to base 2 and ln for the natural logarithm. For a finite set S and an integer t, $\binom{S}{t}$ is the family of t-element subsets of S. For a statement or event A, $\mathbf{1}[A]$ is 1 if A holds and 0 otherwise. We write $0^0 = 1$ and $0 \log 0 = 0$. For $x \in [0,1]$, $H(x) = -x \log x - (1-x)\log(1-x)$ is the binary entropy, so that $2^{H(x)} = x^{-x}(1-x)^{-(1-x)}$. We use the standard entropy estimates for binomial coefficients, valid for integers $0 \le j \le m$:

$$\frac{2^{mH(j/m)}}{m+1} \;\le\; \binom{m}{j} \;\le\; 2^{mH(j/m)}. \tag{1}$$

Throughout the paper, $O^*(f)$ denotes $f \cdot N^{O(1)}$, where N is the length of the input. Every decimal base of a running time of ours is rounded up from a constant that the analysis proves to be strictly smaller, so the polynomial factors are absorbed and we write $O(c^n)$. Exact and parametric bases of ours, such as 1.8^n, $2^{n/2}$ or c^k, are stated with the O^*-notation. So are the two bases of ours that are only computed numerically, the mean-exponent bound for Graph Motif after Theorem 11.13 and the multivariate bound for Max Leaf Spanning Tree in Appendix C.2; no proof uses them. A *randomized algorithm with one-sided error* never answers yes on a no-instance; its error probability is the probability of answering no on a yes-instance. We assume that a randomized algorithm can

draw a uniform integer from $\{1, \ldots, M\}$ in time polynomial in $\log M$; so it draws a uniform t-element subset of an n-element set, or a Bernoulli variable with a rational probability, in time polynomial in n or in the bit length of the probability. (With fair coins alone, such draws take this time in expectation but not in the worst case.)

The framework is that of [FGLS19]: a problem is given by a polynomial-time membership test for its solutions.

Definition 3.1 (Implicit set systems)**.** An *implicit set system* Φ assigns to every instance $I \in \{0,1\}^*$ a universe U_I with $n := |U_I| \le N := |I|$ and a family $\mathcal{F}_I \subseteq 2^{U_I}$, such that U_I can be listed and membership in $\mathcal{F}_I$ decided in time $N^{O(1)}$. The members of $\mathcal{F}_I$ are the *solutions* of I.

- Φ-*Subset*: given I, decide whether $\mathcal{F}_I \neq \emptyset$.
- Φ-*Extension*: given I, a set $X \subseteq U_I$ and an integer $k \ge 0$, decide whether some $Y \subseteq U_I \setminus X$ with $|Y| \le k - |X|$ satisfies $X \cup Y \in \mathcal{F}_I$, that is, whether X lies in a solution with at most k elements. For $k < |X|$ the answer is NO; the extension algorithms of this paper reject such inputs in polynomial time, and their running-time bounds are stated for $k \ge |X|$.

All theorems below are stated for the decision versions. If the extension algorithm returns a witness Y when it accepts, the algorithms below return the solution $X \cup Y$.

Sample-dependent running-time exponents. In classical and multivariate Monotone Local Search (Corollary 4.6) the running-time bound of an extension call depends on the sample X only through $|X|$; Section 1 explains why such a bound gains nothing from sampling for Connected Dominating Set. The model below bounds the running time of the extension algorithm on each input separately, by a function that the subset algorithm can evaluate in advance.

Definition 3.2 (Running-time exponent)**.** Let $\mathcal{A}$ be an algorithm for Φ-Extension. A *running-time exponent* for $\mathcal{A}$ is a function g from inputs (I, X, k) with $k \ge |X|$ to $[0, \infty)$ such that $g(I, X, k)$ can be computed in time $N^{O(1)}$ and $\mathcal{A}$ runs in time $2^{g(I,X,k)} N^{O(1)}$ on every such input (I, X, k).

The running-time bound must hold on every input, including samples X that are contained in no solution. A *product-form* running-time exponent is $g(I, X, k) = (n - |X|) \log b + (k - |X|) \log c + |X| \log d$ with constants $b, c, d \ge 1$; it depends on X only through $|X|$, and it expresses the running-time bound $b^{n-|X|} c^{k-|X|} d^{|X|} N^{O(1)}$ of classical and multivariate Monotone Local Search. An example of a sample-dependent running-time exponent is one computed from the number of Steiner terminals that a sample of vertices leaves (Section 5). Once I and a target size k are fixed, we write $g(X) = g(I, X, k)$ for $|X| \le k$.

Comparisons with thresholds. The algorithms below compare $g(I, X, k)$ with thresholds that are integers or rationals of polynomial bit length, and Definition 3.2 guarantees that such a comparison is decided in time $N^{O(1)}$. This holds for all running-time exponents of this paper.

The main theorem needs, for one solution, the following sums.

Definition 3.3 (Exponential moments and generating function)**.** Fix an instance I, a running-time exponent g, and a solution $S \in \mathcal{F}_I$ with $|S| = k$. For $0 \le t \le k$ and real $y \ge 0$ let

$$\mu_t(S) = \sum_{X \in \binom{S}{t}} 2^{-g(I,X,k)}, \qquad W_S(y) = \sum_{t=0}^{k} \mu_t(S)\, y^t = \sum_{X \subseteq S} y^{|X|}\, 2^{-g(I,X,k)}.$$

If X is drawn uniformly from $\binom{U_I}{t}$, then $\mu_t(S)/\binom{n}{t} = \mathbb{E}[\mathbf{1}[X \subseteq S]\, 2^{-g(I,X,k)}]$: the normalized moment is the expected value of 2^{-g} on the event that the sample lies inside S. Every $\mu_t(S)$ is positive, because g is finite, and $\mu_t(S) \le \binom{k}{t}$, because $g \ge 0$.

3.2 The algorithm and the main theorem

How the algorithm works. The algorithm monitors one quantity: the running-time exponent $g(I, X, k)$ of each sample X, which it computes in polynomial time before deciding whether to call the extension algorithm. It works through all pairs (t, θ) of a sample size $t \in \{0, \dots, k\}$ and an integer threshold $\theta \ge 0$. A *trial* of the pair (t, θ) draws a uniformly random t-set X. If $g(I, X, k) \le \theta$, the trial runs $\mathcal{A}$ on (I, X, k), in time $O^*(2^\theta)$; otherwise it discards X in polynomial time. Low thresholds make trials fast but rarely successful, and high thresholds do the opposite. To balance the two, the algorithm gives every pair the same running-time budget $O^*(2^{\psi n})$, where $2^{\psi n}$ is the target running time: the pair (t, θ) receives about $2^{\psi n - \theta}$ trials. Thresholds above a cap $\Theta \approx \psi n$ are not used, since a single such trial could exceed the budget. As there are only polynomially many pairs, the total time is $O^*(2^{\psi n})$.

The algorithm has three parameters. The *rate* $\psi \ge 0$ is the exponent of the target running time $2^{\psi n}$. The *loss factor* $\Gamma \ge 1$ weakens the success condition $(\star)$ of Theorem 3.5 to $\mu_t(S) \ge \binom{n}{t} 2^{-\psi n}/\Gamma$, so that an application may prove a lower bound on $\mu_t(S)$ that is smaller by a factor Γ, typically polynomial in n, without changing ψ; the cap and the trial counts grow with Γ, and the running time stays $2^{\psi n} N^{O(1)}$ as long as $\Gamma \le N^{O(1)}$. The *confidence* $\kappa > 0$ sets the bound $e^{-\kappa}$ on the error probability in item (iii) of Theorem 3.5.

Algorithm 3.4 (MLS-$\theta(\psi, \Gamma, \kappa)$)

Input: an instance I of Φ and an integer $k \le n$.

Parameters: ψ, Γ, κ, an integer cap $\Theta \ge 0$ and integer trial counts $m_{t,\theta}$ such that

$$x + 1 \le \Theta \le x + 3, \qquad M_{t,\theta} \le m_{t,\theta} \le 2M_{t,\theta} + 1, \\ \text{where } x = \psi n + \log \Gamma \text{ and } M_{t,\theta} = 4\kappa\Gamma(\Theta + 1)\, 2^{\psi n - \theta}. \tag{2}$$

For every $t \in \{0, \dots, k\}$ and every integer $\theta \in \{0, \dots, \Theta\}$, repeat $m_{t,\theta}$ times:

1. draw X uniformly from $\binom{U_I}{t}$ and compute $g(I, X, k)$;
2. if $g(I, X, k) \le \theta$, run $\mathcal{A}$ on (I, X, k), and if it accepts, answer YES.

After all trials, answer NO.

When x and $M_{t,\theta}$ are rational, $\Theta = \lceil x \rceil + 1$ and $m_{t,\theta} = \lceil M_{t,\theta} \rceil$ qualify. In general, integers in these ranges are computable in time $N^{O(1)}$ from rational approximations: $\Theta = \lceil \tilde{x} + \frac{1}{2} \rceil + 1$ for a rational $\tilde{x}$ with $|\tilde{x} - x| < \frac{1}{2}$, and $m_{t,\theta} = \lceil \hat{m} \rceil$ for a rational $\hat{m}$ with $M_{t,\theta} \le \hat{m} \le 2M_{t,\theta}$. The ranges in (2) are part of the statement of the theorem, so no separate rounding argument is needed in the applications.

Theorem 3.5 (Exponential-moment Monotone Local Search)**.** *Let Φ be an implicit set system and let $\mathcal{A}$ solve Φ-Extension with running-time exponent g. Let $\psi \ge 0$, $\Gamma \ge 1$ and $\kappa > 0$ be numbers that can be computed from (I, k) in time $N^{O(1)}$ and whose values are at most $N^{O(1)}$. With any cap Θ and trial counts $m_{t,\theta}$ satisfying (2), Algorithm* MLS-$\theta(\psi, \Gamma, \kappa)$*, given I and $k \le n$,*

(i) *runs in time* $2^{\psi n} N^{O(1)}$ *and in space* $N^{O(1)}$ *plus the space of* $\mathcal{A}$*;*

(ii) *answers* NO *if* $\mathcal{F}_I$ *contains no set of size at most* k*;*

(iii) *answers* YES *with probability at least* $1 - e^{-\kappa}$ *if* $\mathcal{F}_I$ *contains a set* S *with* $|S| = k$ *that satisfies*

$$\max_{0 \le t \le k} \frac{\mu_t(S)}{\binom{n}{t}} \;\ge\; \frac{2^{-\psi n}}{\Gamma}. \tag{$\star$}$$

Condition ($\star$) *holds in particular if* $W_S(y) \ge (1+y)^n \, 2^{-\psi n}/\Gamma$ *for some* $y \ge 0$*.*

By items (ii) and (iii), Algorithm MLS-θ is a randomized algorithm with one-sided error. Two remarks complete the statement.

Optimization problems. For an optimization problem, we say that the optimum can be computed (or that the problem can be solved) by a randomized algorithm with one-sided error, with success probability p, if the following holds on every input. On every outcome of its coins, the algorithm stops within the stated time and space bounds. Its output is never better than the optimum, which means never below it for a minimization problem and never above it for a maximization problem. A report that the input has no solution counts as the value $+\infty$ for a minimization problem and $-\infty$ for a maximization problem. With probability at least p, the output is the optimum, or a correct report that there is no solution. Where the algorithm also outputs a solution, that solution is feasible on every outcome of the coins and its value is the output value. Unless a statement names another probability, $p = 1 - e^{-\kappa}$ for an arbitrary constant $\kappa > 0$ fixed in advance, and the time bound holds for every such κ. The wrappers that provide this form are Lemma 5.2, which outputs the first accepted size, Lemma D.11.6, which outputs the least accepted budget, and equivalent loops over the value and the size, such as the one in the proof of Theorem 11.15.

Remark 3.6 (Decision problem). Running the algorithm for $k = 0, 1, \ldots, n$, with rates $\psi = \psi(I, k)$, decides Φ-Subset with one-sided error at most $e^{-\kappa}$ in time $N^{O(1)} \max_k 2^{\psi(I,k) n}$, provided every instance with $\mathcal{F}_I \neq \emptyset$ has a solution S that satisfies ($\star$) with $k = |S|$.

Remark 3.7 (Real-valued running-time exponents and parameters). If a running-time exponent g can only be approximated, as the product-form exponent with irrational $\log b$, $\log c$ or $\log d$, replace it by a rational $\hat{g}(I, X, k)$ with $g \le \hat{g} \le g + 1$, computable in time $N^{O(1)}$ from an approximation of g to within $\frac{1}{2}$. Then $\hat{g}$ is again a running-time exponent for $\mathcal{A}$, and $2^{-\hat{g}} \ge 2^{-g}/2$, so every exponential moment $\mu_t(S)$ and every value $W_S(y)$ of the generating function (Definition 3.3) is at least half of its value for g, and the mean of $\hat{g}$ over any set of samples exceeds that of g by at most 1. All statements of the paper therefore hold for $\hat{g}$ once their success conditions are weakened accordingly: the loss factor Γ of Theorem 3.5 is replaced by 2Γ, and an upper bound on the mean running-time exponent of a random sample, as assumed in the mean-exponent theorem (Theorem 4.1), is increased by 1. Parameters of the algorithms that are real numbers, such as a bound on the mean or the constants of the affine certificates of Section 9, are likewise replaced by rational approximations from above with slack at most 1; this multiplies the running-time bound only by a constant factor.

Proof overview. The proof has three steps. *Running time:* the trials of every pair (t, θ) take time $2^{\psi n} N^{O(1)}$, because there are about $2^{\psi n - \theta}$ of them and each takes time at most $2^{\theta} N^{O(1)}$. *A good threshold exists below the cap:* if $\mu_t(S) \ge \binom{n}{t} 2^{-\psi n}/\Gamma$, some threshold $\theta \le \Theta$ is passed by at least $2^{\theta} \mu_t(S)/(4(\Theta + 1))$ sets $X \in \binom{S}{t}$. The sets $X \in \binom{S}{t}$ with running-time exponent above the cap contribute at most $\mu_t(S)/2$ to $\mu_t(S)$. *Success:* a trial of that pair succeeds with probability

at least $2^{\theta-\psi n}/(4\Gamma(\Theta+1))$, and the number $m_{t,\theta}$ of trials of the pair is at least κ divided by this probability. The second step is the following lemma, stated for a general random variable so that Section 3.6 can use it as well. In this section the random variable is the running-time exponent of a uniformly random sample of size t, set to ∞ when the sample is not inside the solution under consideration. We write $2^{-\infty}=0$.

Lemma 3.8 (Threshold selection)**.** *Let ξ be a random variable with values in $[0,\infty]$ and $\Pr[\xi<\infty]>0$, so that $\mathbb{E}[2^{-\xi}]>0$.*

(a) *If $\Theta\geq 0$ is an integer with $2^{-\Theta}\leq \mathbb{E}[2^{-\xi}]/2$, then some $\theta\in\{0,\dots,\Theta\}$ satisfies $\Pr[\xi\leq\theta]\,2^{-\theta}\geq \mathbb{E}[2^{-\xi}]/(4(\Theta+1))$.*

(b) *Conversely, $\Pr[\xi\leq\theta]\,2^{-\theta}\leq \mathbb{E}[2^{-\xi}]$ for every integer θ.*

Proof. (a) On the event $\xi>\Theta$ we have $2^{-\xi}<2^{-\Theta}$, so $\mathbb{E}[2^{-\xi}\mathbf{1}[\xi>\Theta]]\leq 2^{-\Theta}\leq \mathbb{E}[2^{-\xi}]/2$ and hence $\mathbb{E}[2^{-\xi}\mathbf{1}[\xi\leq\Theta]]\geq \mathbb{E}[2^{-\xi}]/2$. On the event $\xi\leq\Theta$ put $\theta_\xi=\lceil\xi\rceil$; then $\theta_\xi\in\{0,\dots,\Theta\}$, because $0\leq\xi\leq\Theta$ and Θ is an integer. Moreover $2^{-\xi}<2^{1-\theta_\xi}$ and $\{\theta_\xi=\theta\}\subseteq\{\xi\leq\theta\}$. Hence

$$\frac{\mathbb{E}[2^{-\xi}]}{2}\leq\sum_{\theta=0}^{\Theta}2^{1-\theta}\Pr[\xi\leq\Theta,\ \theta_\xi=\theta]\leq 2\sum_{\theta=0}^{\Theta}2^{-\theta}\Pr[\xi\leq\theta]\leq 2(\Theta+1)\max_{0\leq\theta\leq\Theta}2^{-\theta}\Pr[\xi\leq\theta].$$

(b) $\mathbb{E}[2^{-\xi}]\geq\mathbb{E}[2^{-\xi}\mathbf{1}[\xi\leq\theta]]\geq 2^{-\theta}\Pr[\xi\leq\theta]$. □

Proof. Throughout, a *trial* of the pair (t,θ) is one execution of steps 1 and 2 of Algorithm MLS-θ with these values.

(ii) Soundness. If $\mathcal{A}$ accepts (I,X,k), then $X\cup Y\in\mathcal{F}_I$ for some $Y\subseteq U_I\setminus X$ with $|Y|\leq k-t$, and this set has size at most $t+(k-t)=k$.

(i) Time and space. Drawing X, computing $g(I,X,k)$ and comparing it with θ take time $N^{O(1)}$. If $\mathcal{A}$ is run, then $g(I,X,k)\leq\theta$, so the trial takes time $2^{g(I,X,k)}N^{O(1)}\leq 2^\theta N^{O(1)}$. The trials of a pair (t,θ) therefore take time at most

$$m_{t,\theta}2^{\theta+1}N^{O(1)}\leq(2M_{t,\theta}+1)2^{\theta+1}N^{O(1)}\leq(16\kappa\Gamma(\Theta+1)2^{\psi n}+2^{\Theta+1})N^{O(1)},$$

and $\Theta\leq\psi n+\log\Gamma+3$ gives $2^\Theta\leq 8\Gamma 2^{\psi n}$. There are $(k+1)(\Theta+1)$ pairs. Since κ, Γ and ψ are at most $N^{O(1)}$ and $n\leq N$, also $\Theta+1\leq N^{O(1)}$, and the total time is $2^{\psi n}N^{O(1)}$. Besides the space of $\mathcal{A}$, the algorithm stores one sample, the current pair and a trial counter of $O(\Theta+\log(\kappa\Gamma))$ bits.

(iii) Success probability. Let $S\in\mathcal{F}_I$ with $|S|=k$ satisfy $(\star)$, and fix t with $\mu_t(S)\geq\binom{n}{t}2^{-\psi n}/\Gamma$. Let X be uniform in $\binom{U_I}{t}$, and put $\xi=g(I,X,k)$ if $X\subseteq S$ and $\xi=\infty$ otherwise. Then $\mathbb{E}[2^{-\xi}]=\mu_t(S)/\binom{n}{t}>0$, and for an integer θ, $\Pr[\xi\leq\theta]=q_\theta/\binom{n}{t}$, where q_θ is the number of $X\in\binom{S}{t}$ with $g(I,X,k)\leq\theta$. The hypothesis of Lemma 3.8(a) holds, because $\Theta\geq\psi n+\log\Gamma+1$ gives

$$2^{-\Theta}\leq\frac{2^{-\psi n}}{2\Gamma}\leq\frac{\mu_t(S)}{2\binom{n}{t}}.$$

So some $\theta\in\{0,\dots,\Theta\}$ satisfies

$$p:=\frac{q_\theta}{\binom{n}{t}}=\Pr[\xi\leq\theta]\ \geq\ \frac{2^\theta\mu_t(S)}{4(\Theta+1)\binom{n}{t}}\ \geq\ \frac{2^{\theta-\psi n}}{4\Gamma(\Theta+1)}.$$

A trial of the pair (t,θ) draws an $X\in\binom{S}{t}$ with $g(I,X,k)\leq\theta$ with probability p. Then $\mathcal{A}$ is run on (I,X,k), and it accepts, because $Y=S\setminus X$ satisfies $|Y|=k-t$ and $X\cup Y=S\in\mathcal{F}_I$. The

$m_{t,\theta} \geq M_{t,\theta} = 4\kappa\Gamma(\Theta+1)2^{\psi n-\theta} \geq \kappa/p$ trials of this pair are independent, so all of them fail with probability at most $(1-p)^{m_{t,\theta}} \leq e^{-pm_{t,\theta}} \leq e^{-\kappa}$.

The generating-function condition. For $y \geq 0$,

$$W_S(y) = \sum_{t=0}^{k} \binom{n}{t} y^t \, \frac{\mu_t(S)}{\binom{n}{t}} \leq \max_t \frac{\mu_t(S)}{\binom{n}{t}} \sum_{t=0}^{n} \binom{n}{t} y^t = \max_t \frac{\mu_t(S)}{\binom{n}{t}} \, (1+y)^n,$$

so $W_S(y) \geq (1+y)^n 2^{-\psi n}/\Gamma$ implies $(\star)$. □

Mean versus exponential moments. By Jensen's inequality, $\mathbb{E}[2^{-g(X)}] \geq 2^{-\mathbb{E}[g(X)]}$ for X uniform in $\binom{S}{t}$, so condition $(\star)$ is never more demanding than a hypothesis on the mean of g (Theorem 4.1), and it can certify a smaller running time when g varies between samples: for Connected Dominating Set it lowers the base from 1.7449 to 1.7204 (Theorems 6.4 and 7.2). The same inequality, $\log(1/\mathbb{E}[2^{-\xi}]) \leq \mathbb{E}[\xi]$, shows that the condition on Θ in Lemma 3.8(a) holds whenever $\Theta \geq \mathbb{E}[\xi] + 1$. The converse fails, and the gap can be large: if $\xi = 0$ and $\xi = 2m$ with probability $\frac{1}{2}$ each, the condition holds for $\Theta \geq 2$, whereas $\mathbb{E}[\xi] + 1 = m + 1$.

3.3 The generating function

Condition $(\star)$ involves a maximum over sample sizes. The generating function W_S replaces it by an optimization over one real variable, which is how the later sections bound the exponential moments (Sections 7 and 8). The next lemma shows that the two forms agree up to a factor $n+1$.

Lemma 3.9 (Generating function versus exponential moments)**.** *For every instance I, running-time exponent g and solution S with $|S| = k$, the moments $\mu_t(S)$ with $0 \leq t \leq k$ are positive, since g is finite, and*

$$\min_{0\leq t\leq k} \frac{\binom{n}{t}}{\mu_t(S)} \;\leq\; \inf_{y\geq 0} \frac{(1+y)^n}{W_S(y)} \;\leq\; (n+1) \min_{0\leq t\leq k} \frac{\binom{n}{t}}{\mu_t(S)}.$$

Proof. The first inequality is the last display of the proof of Theorem 3.5, read as $(1+y)^n/W_S(y) \geq 1/\max_t(\mu_t(S)/\binom{n}{t})$; note that $W_S(y) \geq \mu_0(S) > 0$.

For the second, fix t; we exhibit $y \geq 0$ with $(1+y)^n/W_S(y) \leq (n+1)\binom{n}{t}/\mu_t(S)$, and the minimum over t gives the claim. If $t = 0$, then $y = 0$ and $(1+y)^n/W_S(y) = 1/\mu_0(S) = \binom{n}{0}/\mu_0(S)$. If $0 < t < n$, put $y = t/(n-t)$ and $\tau = t/n$, so that $y/(1+y) = \tau$ and $1/(1+y) = 1-\tau$. Since $W_S(y) \geq \mu_t(S)\, y^t$,

$$\frac{(1+y)^n}{W_S(y)} \leq \frac{(1+y)^n}{\mu_t(S)\, y^t} = \frac{1}{\mu_t(S)\, \tau^t (1-\tau)^{n-t}} \leq \frac{(n+1)\binom{n}{t}}{\mu_t(S)}.$$

The last step uses $\binom{n}{t}\tau^t(1-\tau)^{n-t} \geq 1/(n+1)$, which holds because $\tau^t(1-\tau)^{n-t} = 2^{-nH(\tau)}$ and $\binom{n}{t} \geq 2^{nH(\tau)}/(n+1)$ by (1).

If $t = n \geq 1$, then $k = n$ and $y = n$. Now $W_S(n) \geq \mu_n(S)\, n^n$, and $(1+n)^n/n^n = (1+1/n)^n \leq \min(e, 2^n) \leq n+1$, so $(1+y)^n/W_S(y) \leq (n+1)/\mu_n(S) = (n+1)\binom{n}{n}/\mu_n(S)$. □

The infimum in Lemma 3.9 need not be attained: for $k = n$ the ratio may decrease toward $1/\mu_n(S)$ as $y \to \infty$; Theorem 3.5 uses the generating-function condition only at a specific y.

3.4 The criterion is necessary for the algorithm

Exponential moments are the right quantity for Algorithm MLS-θ: if the left-hand side of $(\star)$ is smaller than $2^{-\psi n}$ by more than a polynomial factor, namely $4P_{\mathrm{nec}}$ with the polynomial factor P_{nec} defined in the next proposition, then the algorithm runs $\mathcal{A}$ on a subset of S with probability at most $1/4$. It might still succeed through another solution, or through a sample that is not inside S; the class-optimality result of Section 3.5 accounts for those as well. Proposition 3.10 is used in Section 7 to show that no analysis of Algorithm RS-θ can lower its base below 1.7192.

The proposition also covers runs of the algorithm on (I,k) with k larger than the size of S, which the tightness results of Section 11 need. In such a run, a trial of sample size t runs $\mathcal{A}$ on (I,X,k), with the size bound k rather than $|S|$. For $S \in \mathcal{F}_I$, an integer $k \geq |S|$ and $0 \leq t \leq |S|$ we therefore put

$$\mu_t^{(k)}(S) = \sum_{X \in \binom{S}{t}} 2^{-g(I,X,k)}. \tag{3}$$

For $|S| = k$ this is the exponential moment $\mu_t(S)$ of Definition 3.3.

Proposition 3.10 (Necessity of $(\star)$)**.** *Let $k \leq n$ be an integer, fix $S \in \mathcal{F}_I$ with $|S| \leq k$ and run Algorithm* MLS-$\theta(\psi,\Gamma,\kappa)$ *on (I,k). The expected number of trials that run $\mathcal{A}$ on a subset of S is at most*

$$P_{\mathrm{nec}} \cdot 2^{\psi n} \max_{0 \leq t \leq |S|} \frac{\mu_t^{(k)}(S)}{\binom{n}{t}},$$

where $P_{\mathrm{nec}} = 8\Gamma(k+1)(\Theta+1)\big(\kappa(\Theta+1)+1\big) \leq N^{O(1)}$ and Θ is the cap used. In particular, if $\max_{t \leq |S|} \mu_t^{(k)}(S)/\binom{n}{t} \leq 2^{-\psi n}/(4P_{\mathrm{nec}})$, the probability that some trial runs $\mathcal{A}$ on a subset of S is at most $1/4$.

Proof. A trial of (t,θ) draws a t-set X and runs $\mathcal{A}$ on (I,X,k) exactly when $g(I,X,k) \leq \theta$. So it runs $\mathcal{A}$ on a subset of S exactly when $t \leq |S|$ and it draws $X \in \binom{S}{t}$ with $g(I,X,k) \leq \theta$; for $t > |S|$ this never happens. Let $t \leq |S|$. With X uniform in $\binom{U_I}{t}$ and $\xi = g(I,X,k)$ if $X \subseteq S$, $\xi = \infty$ otherwise, we have $\mathbb{E}[2^{-\xi}] = \mu_t^{(k)}(S)/\binom{n}{t}$, so by Lemma 3.8(b) the event has probability $\Pr[\xi \leq \theta] \leq 2^{\theta}\mu_t^{(k)}(S)/\binom{n}{t}$. Here $m_{t,\theta} \leq 2M_{t,\theta}+1 = 8\kappa\Gamma(\Theta+1)2^{\psi n-\theta}+1$ and $2^{\theta} \leq 2^{\Theta} \leq 8\Gamma 2^{\psi n}$ by (2), so $m_{t,\theta}2^{\theta} \leq 8\Gamma(\kappa(\Theta+1)+1)2^{\psi n}$. By linearity of expectation, the expected number of trials that run $\mathcal{A}$ on a subset of S is at most the sum of $m_{t,\theta}2^{\theta}\mu_t^{(k)}(S)/\binom{n}{t}$ over the pairs (t,θ) with $t \leq |S|$. There are at most $(k+1)(\Theta+1)$ such pairs, and each term is at most $8\Gamma(\kappa(\Theta+1)+1)2^{\psi n}\max_{t' \leq |S|}\mu_{t'}^{(k)}(S)/\binom{n}{t'}$; this gives the bound on the expectation. By Markov's inequality, the probability that some trial runs $\mathcal{A}$ on a subset of S is at most this expectation, which gives the bound $1/4$. □

The moments with a larger size bound are easy to bound for the running-time exponents of this paper, which grow with the size bound. If $g(I,X,k)$ is nondecreasing in k, then for every $S \in \mathcal{F}_I$, every $k \geq |S|$ and every $t \leq |S|$, each term of (3) satisfies $2^{-g(I,X,k)} \leq 2^{-g(I,X,|S|)}$; hence

$$\mu_t^{(k)}(S) \leq \mu_t^{(|S|)}(S) = \mu_t(S), \tag{4}$$

where $\mu_t(S)$ is the exponential moment of Definition 3.3 for S and its own size $|S|$. For such running-time exponents, Proposition 3.10 therefore bounds the runs on (I,k) for all $k \geq |S|$ by the moments of S for $k = |S|$.

3.5 Optimality in its class

Proposition 3.10 concerns one algorithm and one solution. We now show that Algorithm MLS-θ is optimal, up to polynomial factors and on every instance, among all algorithms that sample uniformly, run $\mathcal{A}$ only on samples whose running-time exponent is at most a chosen threshold θ, and are charged 2^θ per trial (the class $\mathcal{C}_{\mathrm{thr}}$ defined below). Consequently a running time below the one certified by Theorem 3.5, by more than a polynomial factor, needs trials outside this class (Appendix A); a sharper analysis of Algorithm MLS-θ cannot give it.

Fix I, $k \le n$ and a correct extension algorithm $\mathcal{A}$ with running-time exponent g, and write $U = U_I$ and, as declared after Definition 3.2, $g(X) = g(I, X, k)$.

Definition 3.11 (Extendable sets and the class optimum)**.** The family $\mathcal{E}$ of *extendable sets* consists of the sets $X \subseteq U$ with $|X| \le k$ such that $X \cup Y \in \mathcal{F}_I$ for some $Y \subseteq U \setminus X$ with $|Y| \le k - |X|$; put $\mathcal{E}_t = \mathcal{E} \cap \binom{U}{t}$. Since $\mathcal{A}$ is correct, it accepts (I, X, k) exactly when $X \in \mathcal{E}$, and $\binom{S}{t} \subseteq \mathcal{E}_t$ for every $S \in \mathcal{F}_I$ with $|S| = k$. For an integer $\theta \ge 0$ let $q_\theta(t)$ be the number of $X \in \mathcal{E}_t$ with $g(X) \le \theta$, let $\mu_t(\mathcal{E}) = \sum_{X \in \mathcal{E}_t} 2^{-g(X)}$, and define the *class optimum*

$$\pi^* = \max_{0 \le t \le k,\ \theta \in \mathbb{Z}_{\ge 0}} \frac{q_\theta(t)\, 2^{-\theta}}{\binom{n}{t}}.$$

The maximum exists, because for each t the count $q_\theta(t)$ is constant for $\theta \ge \max g$, so only finitely many values of θ are candidates. Note that $\mu_t(\mathcal{E}) \ge \mu_t(S)$ for every $S \in \mathcal{F}_I$ with $|S| = k$.

Let $\mathcal{C}_{\mathrm{thr}}$ be the class of algorithms that perform a sequence of trials of the following kind. Before trial i the algorithm chooses $t_i \in \{0, \dots, k\}$ and an integer $\theta_i \ge 0$, as any, possibly randomized, function of everything observed so far; negative thresholds would pass nothing, as $g \ge 0$. Trial i draws X_i uniformly from $\binom{U}{t_i}$, independently of the past. If $g(X_i) \le \theta_i$ it runs $\mathcal{A}$ on (I, X_i, k), and it answers YES if $\mathcal{A}$ accepts; these are the only YES answers of the algorithm. Trial i is *charged* 2^{θ_i}, whether or not $\mathcal{A}$ is run. Algorithm MLS-θ belongs to $\mathcal{C}_{\mathrm{thr}}$, and its running time is its total charge times $N^{O(1)}$. Under this charge the threshold filter is the filter with the largest success probability: a filter must reject every X with $g(X) > \theta$ to keep the running-time bound of the trial within $2^\theta N^{O(1)}$, and rejecting more samples only lowers the success probability.

Theorem 3.12 (Optimality in its class)**.** *Let I, k, $\mathcal{A}$ and g be as fixed above, and $\mathcal{E}_t$, $q_\theta(t)$, $\mu_t(\mathcal{E})$ and π^* as in Definition 3.11. Then:*

(a) *every algorithm in $\mathcal{C}_{\mathrm{thr}}$ whose total charge is at most B on every run answers YES with probability at most $\mathsf{B}\pi^*$;*

(b) *if $\pi^* \ge 2^{-\psi n}/\Gamma$, then Algorithm MLS-$\theta(\psi, \Gamma, \kappa)$ answers YES with probability at least $1 - e^{-\kappa}$, and it runs in time $2^{\psi n} N^{O(1)}$ by Theorem 3.5(i);*

(c) *for every t and every integer θ, $q_\theta(t) 2^{-\theta} \le \mu_t(\mathcal{E})$. If, for an integer $g_{\max} \ge 0$, g takes integer values in $\{0, \dots, g_{\max}\}$ on $\mathcal{E}_t$, then $\max_{0 \le \theta \le g_{\max}} q_\theta(t) 2^{-\theta} \ge \mu_t(\mathcal{E})/(g_{\max} + 1)$.*

The proof is in Appendix D.1. For (a) it bounds the success probability of each trial by π^* times its charge and sums over the trials; for (b) it uses the trials of a pair (t, θ) that attains π^*; and (c) compares each threshold with the running-time exponents below it.

By (a) and (b), Algorithm MLS-θ with $\Gamma = 1$ and $2^{\psi n} = 1/\pi^*$ is optimal in $\mathcal{C}_{\mathrm{thr}}$ up to polynomial factors; in a worst-case analysis, $2^{\psi n}$ is an upper bound on $1/\pi^*$ proven for a whole family of instances. By (c), $1/\pi^*$ equals $\min_t \binom{n}{t}/\mu_t(\mathcal{E})$ up to a factor $g_{\max} + 1$ for integer running-time exponents bounded by $g_{\max}$. For real-valued g, Lemma 3.8(a) takes the place of the integer case

of (c): with $\xi = g(X)$ if $X \in \mathcal{E}_t$ and $\xi = \infty$ otherwise, for X uniform in $\binom{U}{t}$, the definitions of $\mu_t(\mathcal{E})$ and $q_\theta(t)$ give $\mathbb{E}[2^{-\xi}] = \mu_t(\mathcal{E})/\binom{n}{t}$ and $\Pr[\xi \le \theta] = q_\theta(t)/\binom{n}{t}$. Since Algorithm MLS-$\theta$ succeeds on every extendable sample, (b) and (c) also show that Theorem 3.5(iii) remains true when $\mu_t(S)$ in $(\star)$ is replaced by the sum $\mu_t(\mathcal{E})$ over the t-subsets of all solutions of size at most k: for Algorithm MLS-$\theta(\psi, (g_{\max}+1)\Gamma, \kappa)$ when g takes integer values in $\{0, \dots, g_{\max}\}$ on $\mathcal{E}$ for an integer $g_{\max} \le N^{O(1)}$, and for Algorithm MLS-$\theta(\psi, \Gamma, \kappa)$ itself, for any g, when the whole left-hand side of $(\star)$ is replaced by π^*.

3.6 General forms

The proof of Theorem 3.5 uses only two properties of a trial: it runs in time at most $2^\theta N^{O(1)}$, and it succeeds whenever a random quantity attached to it is at most θ. This subsection isolates that structure in a version for arbitrary budgeted trials, which covers randomized extension algorithms and the closure sampling of Section 8.

The general form rests on Lemma 3.8. The next theorem replaces the trials of Algorithm MLS-θ by arbitrary randomized procedures. A trial indexed by ι with threshold θ must run in time $2^\theta N^{O(1)}$ and must succeed at least when a random variable ξ_ι, which the analysis attaches to it, is at most θ. The variable ξ_ι is only used in the analysis; it may depend on an unknown solution.

Algorithm 3.13 (Budgeted trials)

Input: an instance I and an integer $k \le n$.

Parameters: a randomized algorithm Tr with inputs (I, k, ι, θ) and, for each (I, k), a set $\mathcal{T} = \mathcal{T}_{I,k}$ of indices; for $\iota \in \mathcal{T}$ and an integer $\theta \ge 0$, the run of Tr on (I, k, ι, θ) is the *trial* $\mathrm{Tr}_{\iota,\theta}$. Further ψ, Γ and κ, with a cap Θ and trial counts $m_{\iota,\theta}$ as in (2) with ι in place of t.

For every $\iota \in \mathcal{T}$ and $\theta \in \{0, \dots, \Theta\}$, run $m_{\iota,\theta}$ independent trials $\mathrm{Tr}_{\iota,\theta}$, and answer YES as soon as one accepts. Answer NO after all trials.

Theorem 3.14 (Budgeted trials)**.** *Let Φ be an implicit set system, and let* Tr *and the index sets $\mathcal{T} = \mathcal{T}_{I,k}$ be as in Algorithm 3.13, such that $\mathcal{T}_{I,k}$ can be listed in time $N^{O(1)}$. Suppose that there is a polynomial p such that for every instance I, every integer $k \le n$, every $\iota \in \mathcal{T}_{I,k}$ and every integer $\theta \ge 0$*

(T1) *$\mathrm{Tr}_{\iota,\theta}$ runs in time $2^\theta p(N)$ and in space $p(N)$ plus* sp*, where* sp ≥ 0 *depends only on I (for example, the space of an extension algorithm on I), and*

(T2) *$\mathrm{Tr}_{\iota,\theta}$ rejects if $\mathcal{F}_I$ contains no set of size at most k.*

Fix I and k, let $S \in \mathcal{F}_I$ with $|S| = k$, and suppose that for each $\iota \in \mathcal{T}$ there is a random variable ξ_ι with values in $[0, \infty]$, whose distribution may depend on S but not on θ, such that

(T3) *$\Pr[\mathrm{Tr}_{\iota,\theta}$ accepts$] \ge \Pr[\xi_\iota \le \theta]$ for every integer $\theta \ge 0$.*

Let ψ, Γ, κ be as in Theorem 3.5. Then Algorithm 3.13, run with these trials and parameters, runs in time $2^{\psi n} N^{O(1)}$ and space $N^{O(1)} +$ sp*, answers* NO *if $\mathcal{F}_I$ contains no set of size at most k, and answers* YES *with probability at least $1 - e^{-\kappa}$ if*

$$\max_{\iota \in \mathcal{T}} \mathbb{E}[2^{-\xi_\iota}] \;\ge\; \frac{2^{-\psi n}}{\Gamma}.$$

Proof. Time, space and soundness follow from (T1) and (T2) exactly as items (i) and (ii) of Theorem 3.5; the number of pairs is now $|\mathcal{T}|(\Theta+1) \le N^{O(1)}$, as $\mathcal{T}$ is listed in time $N^{O(1)}$. Fix ι with $\mathbb{E}[2^{-\xi_\iota}] \ge 2^{-\psi n}/\Gamma$. Then $2^{-\Theta} \le 2^{-\psi n}/(2\Gamma) \le \mathbb{E}[2^{-\xi_\iota}]/2$, so Lemma 3.8(a) gives $\theta \le \Theta$ with $\Pr[\xi_\iota \le \theta] \ge 2^{\theta}\mathbb{E}[2^{-\xi_\iota}]/(4(\Theta+1)) \ge 2^{\theta-\psi n}/(4\Gamma(\Theta+1))$. By (T3) each trial of (ι,θ) accepts with probability $\pi_{\iota,\theta} \ge \Pr[\xi_\iota \le \theta]$, the trial count satisfies $m_{\iota,\theta} \ge M_{\iota,\theta} \ge \kappa/\pi_{\iota,\theta}$, and by independence all these trials fail with probability at most $(1-\pi_{\iota,\theta})^{m_{\iota,\theta}} \le e^{-\kappa}$. □

In each application, (T1) and (T2) follow directly from the description of the trial, and only (T3) needs an argument. Theorem 3.5 is the instance $\mathcal{T} = \{0,\dots,k\}$, in which a trial of (t,θ) is steps 1–2 of Algorithm MLS-θ, and $\xi_t = g(I,X,k)$ if $X \subseteq S$ and $\xi_t = \infty$ otherwise; (T3) holds because $\mathcal{A}$ accepts when $X \subseteq S$, and $\mathbb{E}[2^{-\xi_t}] = \mu_t(S)/\binom{n}{t}$. The closure algorithm RSC of Section 8 is another instance: its trials add a random closure to a uniform sample before the Steiner computation, so the uniform-sample form of Theorem 3.5 does not apply literally, but Theorem 3.14 does. The next instance allows extension algorithms with one-sided error, as used for Connected Feedback Vertex Set in Section 11.3.

Corollary 3.15 (One-sided-error extension algorithms)**.** *Let $\mathcal{A}$ be a randomized algorithm for Φ-Extension that never accepts a* no*-instance, accepts every* yes*-instance with probability at least $1/2$ using its own independent coins, and runs in time $2^{g(I,X,k)}N^{O(1)}$ for every input and every outcome of its coins, where g is polynomial-time computable. Then Theorem 3.5 holds for $\mathcal{A}$ with Γ replaced by 2Γ in the cap Θ and in the trial counts $m_{t,\theta}$.*

Proof. Apply Theorem 3.14 with $\mathcal{T} = \{0,\dots,k\}$ and the trials of Algorithm MLS-θ, now with the randomized $\mathcal{A}$. Property (T1) holds because the time bound holds for every outcome of the coins, and (T2) because $\mathcal{A}$ never accepts a no-instance. Let ξ_t be as above, and let ξ'_t equal ξ_t if an independent fair coin shows heads and ∞ otherwise. If $X \subseteq S$ and $g(I,X,k) \le \theta$, then $\mathcal{A}$ is run on a yes-instance and accepts with probability at least $1/2$, independently of X. Hence

$$\Pr[\mathrm{Tr}_{t,\theta} \text{ accepts}] \ge \tfrac{1}{2}\Pr[\xi_t \le \theta] = \Pr[\xi'_t \le \theta], \qquad \mathbb{E}[2^{-\xi'_t}] = \tfrac{1}{2}\mathbb{E}[2^{-\xi_t}] = \frac{\mu_t(S)}{2\binom{n}{t}}.$$

If S satisfies $(\star)$ with Γ, then $\max_t \mathbb{E}[2^{-\xi'_t}] \ge 2^{-\psi n}/(2\Gamma)$, and Theorem 3.14 with 2Γ in place of Γ gives the claim. □

Other levers. Changes of the trials outside the class $\mathcal{C}_{\mathrm{thr}}$ of Theorem 3.12 are examined in Appendix A. Theorem 3.14 covers every sampling distribution that can be sampled in polynomial time (Theorem A.2); non-uniform sampling can lower the running time by an exponential factor (Proposition A.5), whereas sampling in several adaptive stages cannot (Proposition A.6).

4 Corollaries: mean exponents, linear savings and product form

This section derives earlier and intermediate forms of Monotone Local Search from Theorem 3.5: the mean-exponent theorem (Section 4.1), which is the reference point of the black-box barrier (Section 9) and of the comparisons in Section 11; a closed form for linear mean savings (Section 4.2), used for Algorithm RS (Section 6); and multivariate Monotone Local Search, for whose product-form running-time exponents the main theorem certifies no smaller bound (Section 4.3). Appendix C explains why Max Leaf Spanning Tree needs sample-dependent running times.

Recall the notation of Section 3: Φ is an implicit set system, I an instance with $N = |I|$, universe U_I with $n = |U_I| \le N$ and solution family $\mathcal{F}_I$; the algorithm $\mathcal{A}$ solves Φ-Extension with

running-time exponent g (Definition 3.2); and for $S \in \mathcal{F}_I$ with $|S| = k$ the exponential moments and the generating function are

$$\mu_t(S) = \sum_{X \in \binom{S}{t}} 2^{-g(I,X,k)} \quad (0 \le t \le k), \qquad W_S(y) = \sum_{t=0}^{k} \mu_t(S)\, y^t \quad (y \ge 0),$$

as in Definition 3.3. By the first inequality of Lemma 3.9, the *generating-function condition* $W_S(y) \ge (1+y)^n 2^{-\psi n}/\Gamma$, for a single $y \ge 0$, implies condition $(\star)$ of Theorem 3.5, and by the second inequality the converse holds up to a factor $n+1$: with

$$M(S) = \min_{0 \le t \le k} \frac{\binom{n}{t}}{\mu_t(S)} \qquad \text{and} \qquad G(S) = \inf_{y \ge 0} \frac{(1+y)^n}{W_S(y)},$$

we have $M(S) \le G(S) \le (n+1)M(S)$. Note that $(\star)$ holds exactly when $2^{\psi n}\Gamma \ge M(S)$; thus, with $\Gamma = 1$, $M(S)$ is the smallest running-time bound $2^{\psi n}$ (up to the factor $N^{O(1)}$) that Theorem 3.5 certifies for the solution S, and $G(S)$ is the smallest one that the generating-function condition certifies. The binary entropy H and the entropy estimates (1) are as in Section 3.1.

4.1 Mean exponents: the Markov cutoff

The simplest hypothesis on a sample-dependent running-time exponent is a bound λ on its mean over the t-subsets of a solution, for one sample size t. We derive the resulting bound from the main theorem by Jensen's inequality; Remark 4.2 describes the direct algorithm, the Markov-cutoff mechanism of Lemma 2 of Zamir [Zam23], which discards samples whose running-time exponent exceeds λ by more than one.

In Theorem 4.1, $t(I,k)$ is the sample size and $\lambda(I,k)$ the bound on the mean running-time exponent of samples of this size inside a solution of size k; the running time is the expected number $\binom{n}{t}/\binom{k}{t}$ of uniform t-sets drawn until one lies inside a fixed solution, times the running-time bound $2^{\lambda(I,k)}$ of an extension call at the mean running-time exponent.

Theorem 4.1 (Mean-exponent Monotone Local Search)**.** *Let $t(I,k) \in \{0,\dots,k\}$ and $\lambda(I,k) \ge 0$ be computable in time $N^{O(1)}$, with values $\lambda(I,k) \le N^{O(1)}$, such that for every instance I, every $k \le n$ and every $S \in \mathcal{F}_I$ with $|S| = k$, writing $t = t(I,k)$,*

$$\mathbb{E}_X\big[g(I,X,k)\big] \le \lambda(I,k), \qquad X \text{ uniform in } \binom{S}{t}. \tag{5}$$

Then there is a randomized algorithm that, given I and $k \le n$, runs in time

$$\frac{\binom{n}{t}}{\binom{k}{t}}\, 2^{\lambda(I,k)}\, N^{O(1)}$$

and in space $N^{O(1)}$ plus the space of $\mathcal{A}$. It answers NO *if $\mathcal{F}_I$ contains no set of size at most k, and it answers* YES *with probability at least $1 - 1/e$ if $\mathcal{F}_I$ contains a set of size exactly k. Running it for $k = 0, 1, \dots, n$ decides Φ-Subset as in Remark 3.6.*

Proof. Fix I and k, and write $t = t(I,k)$ and $\lambda = \lambda(I,k)$. If $n = 0$, then $\mathcal{F}_I \subseteq \{\emptyset\}$ and $k = 0$, and the membership test for $\mathcal{F}_I$ decides the question in time $N^{O(1)}$. So let $n \ge 1$. Define ψ by

$$2^{\psi n} = \frac{\binom{n}{t}}{\binom{k}{t}}\, 2^{\lambda},$$

and take $\Gamma = 1$ and $\kappa = 1$. Since $\binom{k}{t} \le \binom{n}{t} \le 2^n$ and $\binom{k}{t} \ge 1$, we have $0 \le \psi \le 1 + \lambda/n \le N^{O(1)}$, and ψ is computable from (I, k) in time $N^{O(1)}$ because t and λ are. Let $S \in \mathcal{F}_I$ with $|S| = k$. The function $x \mapsto 2^{-x}$ is convex, so Jensen's inequality and (5) give

$$\mu_t(S) = \binom{k}{t} \mathbb{E}_X\left[2^{-g(I,X,k)}\right] \;\ge\; \binom{k}{t} 2^{-\mathbb{E}_X[g(I,X,k)]} \;\ge\; \binom{k}{t} 2^{-\lambda} = \binom{n}{t} 2^{-\psi n},$$

with X uniform in $\binom{S}{t}$. Hence $(\star)$ holds with $\Gamma = 1$, and Theorem 3.5 gives the time and space bounds, soundness, and success probability $1 - e^{-1}$. □

Remark 4.2 (The Markov cutoff). The algorithm of this proof is Algorithm MLS-θ, which tries every sample size and every threshold. A direct algorithm uses only the sample size $t = t(I, k)$ and the single threshold $\lambda + 1$, where $\lambda = \lambda(I, k)$: it repeats the following trial $m = \lceil (\lambda+1)\binom{n}{t}\binom{k}{t}^{-1} \rceil$ times and answers NO if no trial succeeds. A trial draws X uniformly from $\binom{U_I}{t}$, computes $g(I, X, k)$, and, if $g(I, X, k) \le \lambda + 1$ (with a rational threshold in $[\lambda + 1, \lambda + 2]$ if needed, Remark 3.7), runs $\mathcal{A}$ on (I, X, k) and answers YES if $\mathcal{A}$ accepts. Every YES answer is correct, each trial takes time $2^{\lambda+2}N^{O(1)}$, so the total time is the bound of the theorem, and the space is that of $\mathcal{A}$ plus $N^{O(1)}$. For completeness, let $S \in \mathcal{F}_I$ with $|S| = k$. A trial has $X \subseteq S$ with probability $\binom{k}{t}/\binom{n}{t}$; conditioned on this event X is uniform in $\binom{S}{t}$, and by Markov's inequality and (5) the running-time exponent exceeds $\lambda + 1$ with probability at most $\lambda/(\lambda + 1)$. Otherwise the trial runs $\mathcal{A}$, which accepts because $Y = S \setminus X$ extends X to S. So a trial succeeds with probability at least $p_{\text{trial}} = \binom{k}{t}/((\lambda + 1)\binom{n}{t}) \ge 1/m$, and all m trials fail with probability at most $(1 - p_{\text{trial}})^m \le e^{-1}$. This is the Markov-cutoff mechanism of Lemma 2 of Zamir [Zam23].

4.2 Linear mean savings

In many applications the bound on the mean running-time exponent of a sample decreases roughly linearly in the number of sampled elements: for samples X of size t inside a solution of size k, the mean of $g(I, X, k)$ is at most $\beta n - \rho t + C$ for reals $\beta, \rho, C \ge 0$, so that each sampled element of the solution lowers this bound by ρ. The next corollary gives the resulting exponent of the running time in closed form. At solution fraction $\sigma = k/n$, a uniformly random element lies in a fixed solution with probability σ, so each further sampled element divides the probability that the sample lies inside the solution by about $1/\sigma$. This multiplies the number of trials by about $1/\sigma$; in exchange it divides the running-time bound $2^{\beta n - \rho t + C}$ of an extension call by 2^ρ. We show (Corollary 4.3(2)) that sampling lowers the exponent exactly when $\rho > \log(1/\sigma)$.

For $\sigma \in (0, 1)$, reals $\beta, \rho \ge 0$ and $p = 2^{-\rho}$ define

$$F^*(\sigma) = \begin{cases} \beta & \text{if } \sigma \le p, \\ \beta + \sigma \log \dfrac{p}{\sigma} + (1 - \sigma) \log \dfrac{1 - p}{1 - \sigma} = \beta - \sigma\rho + H(\sigma) + (1 - \sigma)\log(1 - p) & \text{if } \sigma > p. \end{cases} \tag{6}$$

For $\sigma > p$ the corresponding base is

$$2^{F^*(\sigma)} = 2^\beta \left(\frac{p}{\sigma}\right)^\sigma \left(\frac{1 - p}{1 - \sigma}\right)^{1-\sigma} \le 2^\beta \left(\sigma \cdot \frac{p}{\sigma} + (1 - \sigma) \cdot \frac{1 - p}{1 - \sigma}\right) = 2^\beta$$

by the weighted arithmetic–geometric mean inequality, with equality only if $p/\sigma = (1 - p)/(1 - \sigma)$, that is $\sigma = p$. So $F^*(\sigma) \le \beta$, with equality exactly when $\sigma \le p$.

In the corollary, $\beta n + C$ bounds the mean running-time exponent of the unsampled call ($t = 0$), ρ is the decrease of this bound per sampled element, and C is an additive slack.

Corollary 4.3 (Linear mean savings)**.** *Let $\beta = \beta(I,k)$, $\rho = \rho(I,k)$ and $C = C(I,k)$ be nonnegative reals computable from (I,k) in time $N^{O(1)}$, with $\beta n + C \le N^{O(1)}$. Let I be an instance and k an integer with $\sigma = k/n \in (0,1)$ such that for every $t \in \{0,\dots,k\}$ and every $S \in \mathcal{F}_I$ with $|S| = k$,*

$$\mathbb{E}_X\big[g(I,X,k)\big] \le \beta n - \rho t + C, \qquad X \text{ uniform in } \binom{S}{t}. \tag{7}$$

Let $F^*(\sigma)$ be as in (6)*, and let $\kappa > 0$ be as in Theorem 3.5. Then:*

(1) Every such S satisfies $G(S) \le 2^{C + nF^*(\sigma)}$. Algorithm MLS-$\theta$ *with $\Gamma = 1$, confidence κ and a rate computable from (I,k) in time $N^{O(1)}$ gives a randomized algorithm that, given I and k, runs in time $2^{\max(0,\, C + nF^*(\sigma))} N^{O(1)}$, answers* NO *if $\mathcal{F}_I$ contains no set of size at most k, and answers* YES *with probability at least $1 - e^{-\kappa}$ if $\mathcal{F}_I$ contains a set of size k.*

(2) $F^*(\sigma) < \beta$ if and only if $\rho > \log(1/\sigma)$.

The corollary uses the bound (7) for every t, and ρ only has to be the decrease of a bound on the *mean* running-time exponent over the samples inside a solution, not a decrease of the running-time exponent guaranteed for every sampled element. Remark 4.4 compares this with the algorithm of Theorem 4.1 at a single sample size.

The proof is in Appendix D.2.1. It bounds the moments $\mu_t(S)$ from below by Jensen's inequality, as in the proof of Theorem 4.1, sums them by the binomial theorem, and minimizes the resulting bound on $(1+y)^n / W_S(y)$ over y; the minimum lies at $y = 0$ if $\sigma \le p$ and at $y_0 = (\sigma - p)/(1 - \sigma)$ otherwise.

Remark 4.4 (One sample size). The algorithm of Theorem 4.1 at a single sample size t with the mean bound $\beta n - \rho t + C$ of (7) runs in time $\binom{n}{t}\binom{k}{t}^{-1} 2^{\beta n - \rho t + C} N^{O(1)}$, and

$$\binom{n}{t}\binom{k}{t}^{-1} 2^{\beta n - \rho t} = 2^{\beta n} \prod_{i=0}^{t-1} \frac{n-i}{2^{\rho}(k-i)}. \tag{8}$$

The factor with index i weighs the two effects of sampling one more element: the probability that the sample lies inside a fixed solution of size k is divided by $(n-i)/(k-i)$, and the running-time bound $2^{\beta n - \rho t + C}$ of the extension call is divided by 2^{ρ}. As $n \ge k$, the factors are nondecreasing in i, and with $p = 2^{-\rho}$ the factor with index i is below 1 exactly when $(1-p)i < k - pn$. So the best sample size is the number of factors below 1, and sampling at all beats $t = 0$ exactly when the first factor $n/(2^{\rho}k) = p/\sigma$ is below 1, that is $\sigma > p$, the condition of Corollary 4.3(2).

By the entropy estimates (1), the running time above is at most $(k+1)2^{C} 2^{n f_\sigma(\tau)} N^{O(1)}$ with $\tau = t/n$ and

$$f_\sigma(\tau) = H(\tau) - \sigma H(\tau/\sigma) + \beta - \rho\tau \qquad (0 \le \tau \le \sigma), \qquad p = 2^{-\rho}. \tag{9}$$

Expanding the entropies gives $H(\tau) - \sigma H(\tau/\sigma) = -(1-\tau)\log(1-\tau) + (\sigma - \tau)\log(\sigma - \tau) - \sigma \log \sigma$, so on $[0, \sigma)$ the derivative is $f'_\sigma(\tau) = \log \frac{1-\tau}{\sigma - \tau} - \rho$, the logarithm of the factor of (8) with index $i = \tau n$. It is strictly increasing in τ, so f_σ is convex on $[0, \sigma]$. If $\sigma \le p$ (which includes $\rho = 0$), then $f'_\sigma(\tau) \ge f'_\sigma(0) = \log \frac{1}{\sigma} - \rho \ge 0$, and the minimum of f_σ over $[0,\sigma]$ is $f_\sigma(0) = \beta = F^*(\sigma)$. If $\sigma > p$ (so $p < 1$), then f'_σ vanishes exactly at $\tau^* = \frac{\sigma - p}{1-p} \in (0, \sigma)$, the unique minimizer. With $A = \frac{1-\sigma}{1-p}$ we have $1 - \tau^* = A$, $\sigma - \tau^* = pA$ and $-\rho = \log p$, so

$$f_\sigma(\tau^*) = -(1-p)A \log A + pA \log p + \tau^* \log p - \sigma \log \sigma + \beta = \beta + \sigma \log \frac{p}{\sigma} + (1-\sigma) \log \frac{1-p}{1-\sigma} = F^*(\sigma),$$

using $(1-p)A = 1-\sigma$ and $pA+\tau^* = \sigma$. In both cases the minimum of f_σ over $[0,\sigma]$ is $F^*(\sigma)$. Since $H(\tau) - \sigma H(\tau/\sigma)$ is nondecreasing on $[0,\sigma]$ (its derivative is $\log\frac{1-\tau}{\sigma-\tau} \geq 0$), the single-size algorithm run at $t = \lfloor \tau^* n \rfloor$ (with $\tau^* = 0$ if $\sigma \leq p$) attains the exponent $C + nF^*(\sigma)$ of Corollary 4.3(1) up to the factors 2^ρ and $k+1$. Algorithm RS (Section 6), the black-box barrier (Appendix D.6) and the two-class family of Appendix A use f_σ in this form.

4.3 Product-form running times and multivariate Monotone Local Search

In the classical forms of Monotone Local Search the running-time bound of the extension algorithm depends on the sample X only through $|X|$. [FGLS19] showed that an extension algorithm running in time $c^{k-|X|}N^{O(1)}$ on input (I, X, k) gives a randomized algorithm for Φ-Subset running in time $(2-1/c)^n N^{O(1)}$; Gaspers and Lee [GL17] allowed a further factor $b^{n-|X|}$ in the running time of the extension algorithm and obtained the running time $(1+b-1/c)^n N^{O(1)}$; and the multivariate version, with a further factor $d^{|X|}$, gives $(b+d-d/c)^n N^{O(1)}$ when $d \leq bc$ (Corollary 4.6). We now derive all three from the main theorem and show that for such running-time exponents the main theorem certifies no smaller running-time bound.

For constants $b, c, d \geq 1$, the *product-form* running-time exponent is

$$g(I, X, k) = (n - |X|)\log b + (k - |X|)\log c + |X|\log d. \tag{10}$$

It is nonnegative on inputs with $k \geq |X|$, the domain of Definition 3.2, because $b, c, d \geq 1$. Put

$$\mathrm{base}(b,c,d) = b + \min(d, bc)\Big(1 - \frac{1}{c}\Big) = \begin{cases} b+d-d/c & \text{if } d \leq bc, \\ bc & \text{if } d \geq bc. \end{cases}$$

Since $b + d - d/c - bc = (c-1)(d-bc)/c$, also $\mathrm{base}(b,c,d) = \min(b+d-d/c,\, bc)$. If $S \in \mathcal{F}_I$ with $|S| = k$ and g has the form (10) on all inputs (I, X, k) with $X \subseteq S$, then every $X \in \binom{S}{t}$ has $2^{g(I,X,k)} = b^{n-t}c^{k-t}d^t$, so

$$\mu_t(S) = \binom{k}{t} b^{-(n-t)} c^{-(k-t)} d^{-t} \qquad (0 \leq t \leq k). \tag{11}$$

Thus $M(S)$ and $G(S)$ depend only on n, k, b, c and d; we write M_k and G_k for them (with n, b, c, d fixed), and define them by (11) for every integer $0 \leq k \leq n$.

The evaluation of the product form through a generating function goes back to Gaspers and Lee [GL17] and [FGLS19] (see Section 1.3). The generating function W_S used here is a different series, but for product-form running times it has the following closed form.

Proposition 4.5 (Product form)**.** *Let $b, c, d \geq 1$, let $0 \leq k \leq n$, and let μ_t, M_k and G_k be given by* (11)*. Then:*

(a) The generating function is $W(y) = \sum_{t=0}^{k} \mu_t y^t = b^{-n}c^{-k}(1 + ybc/d)^k$, and

$$M_k = \min_{0 \leq t \leq k} \frac{\binom{n}{t}}{\binom{k}{t}} b^{n-t} c^{k-t} d^t.$$

(b) $M_k \leq G_k \leq \mathrm{base}(b,c,d)^n$. More precisely, $y^* = (c-1)d/(bc)$ gives $(1+y^*)^n/W(y^*) = (b+d-d/c)^n$, and $y = 0$ gives $(1+0)^n/W(0) = b^n c^k \leq (bc)^n$.

(c) Conversely, the following hold:

- *if $c = 1$, then $G_0 = M_0 = b^n = \mathrm{base}(b,c,d)^n$;*
- *if $c > 1$ and $d < bc$, let $\sigma^* = cd/(bc+cd-d)$; then $\sigma^* \in (0,1)$; if $\sigma^* n$ is an integer, then $G_{\sigma^* n} = (b+d-d/c)^n$; and for every $n \geq 1$,*

$$\max_{0\leq k\leq n} G_k \;\geq\; (b+d-d/c)^n \Big(\frac{b}{b+d-d/c}\Big)^{1/\sigma^*};$$

- *if $c > 1$ and $d \geq bc$, then $G_n = M_n = (bc)^n = \mathrm{base}(b,c,d)^n$.*

Consequently, with $\mathsf{K} = \big((b+d-d/c)/b\big)^{1/\sigma^*}$ in the second case and $\mathsf{K} = 1$ otherwise,

$$\frac{\mathrm{base}(b,c,d)^n}{(n+1)\mathsf{K}} \;\leq\; \max_{0\leq k\leq n} M_k \;\leq\; \mathrm{base}(b,c,d)^n.$$

The expression for M_k in (a) is the classical Monotone Local Search expression. So for product-form running-time exponents, $\max_{0\leq k\leq n} M_k$, the smallest running-time bound $2^{\psi n}$ that Theorem 3.5 with $\Gamma = 1$ certifies for every solution size, is $\mathrm{base}(b,c,d)^n$ up to the factor $(n+1)\mathsf{K}$, where K depends only on b, c, d.

Proof. Parts (a) and (b) are direct computations. For (c) we treat the solution size as a real parameter x: the minimization over y in (a) then defines a concave function f of x, whose value at $x = \sigma^* n$ is exactly $n\log(b+d-d/c)$, and concavity transfers this value to the integer $\lfloor \sigma^* n \rfloor$, losing in $G_{\lfloor \sigma^* n \rfloor}$ only the factor K, which does not depend on n.

(a) Grouping by t and using the binomial theorem,

$$W(y) = \sum_{t=0}^{k} \binom{k}{t} y^t\, b^{-(n-t)} c^{-(k-t)} d^{-t} = b^{-n}c^{-k} \sum_{t=0}^{k} \binom{k}{t} \Big(\frac{ybc}{d}\Big)^t = b^{-n}c^{-k}\Big(1+\frac{ybc}{d}\Big)^k.$$

The formula for $M_k = \min_t \binom{n}{t}/\mu_t$ is immediate from (11).

(b) The inequality $M_k \leq G_k$ is the first inequality of Lemma 3.9: for every $y \geq 0$, $W(y) \leq (1+y)^n/M_k$. At $y^* \geq 0$ we have $1 + y^* bc/d = c$, so $W(y^*) = b^{-n}$ and

$$\frac{(1+y^*)^n}{W(y^*)} = \big(b(1+y^*)\big)^n = \Big(b + \frac{(c-1)d}{c}\Big)^n = \Big(b+d-\frac{d}{c}\Big)^n.$$

At $y = 0$ the ratio is $1/W(0) = b^n c^k \leq (bc)^n$, because $c \geq 1$ and $k \leq n$. Hence $G_k \leq \min(b+d-d/c,\, bc)^n = \mathrm{base}(b,c,d)^n$.

Part (c) and the final inequalities are proved in Appendix D.2.2. □

With the product form evaluated, the multivariate theorem follows; its cases $d = 1$ and $b = d = 1$ are the two classical statements. In the running-time bound $b^{n-|X|}c^{k-|X|}d^{|X|}N^{O(1)}$ of the extension algorithm, b is the factor per element outside the sample, c the factor per element that may still be added, and d the factor per sampled element.

Corollary 4.6 (Multivariate Monotone Local Search)**.** *Let $b, c, d \geq 1$ be constants such that $\log b$, $\log c$ and $\log d$ are computable to polynomially many bits in polynomial time. If Φ-Extension can be solved in time $b^{n-|X|}c^{k-|X|}d^{|X|}N^{O(1)}$ on every input (I, X, k), then Φ-Subset can be decided by a randomized algorithm with one-sided error at most $1/e$ in time*

$$\Big(b + \min(d, bc)\Big(1-\frac{1}{c}\Big)\Big)^n N^{O(1)} = \begin{cases} (b+d-d/c)^n N^{O(1)} & \text{if } d \leq bc, \\ (bc)^n N^{O(1)} & \text{if } d \geq bc, \end{cases}$$

and in space $N^{O(1)}$ plus the space of the extension algorithm.

If the extension algorithm returns a completion when it accepts, the algorithm returns a solution. As $bc \geq 1$, the case $d = 1$ gives $(1+b-1/c)^n N^{O(1)}$, the theorem of Gaspers and Lee [GL17], and the case $b = d = 1$ gives $(2-1/c)^n N^{O(1)}$, the randomized theorem of [FGLS19]. The case $d \geq bc$ needs no sampling: a single call of the extension algorithm with $X = \emptyset$ takes time $b^n c^k N^{O(1)} \leq (bc)^n N^{O(1)}$.

Proof. The product-form exponent (10) is nonnegative, computable in time $N^{O(1)}$, and by hypothesis a running-time exponent for the given extension algorithm. Fix $k \leq n$, and put $2^\psi = \mathrm{base}(b, c, d)$, $\Gamma = 1$ and $\kappa = 1$; then $0 \leq \psi \leq \log(bc)$ is a constant (if $\log b$, $\log c$ or $\log d$ is irrational, g is rounded up as in Remark 3.7, which replaces Γ by 2; if ψ is irrational, a rational rate in $[\psi, \psi + 1/n]$ is used; both change the running time by a constant factor). Let $S \in \mathcal{F}_I$ with $|S| = k$. On the inputs (I, X, k) with $X \subseteq S$ the running-time exponent has the form (10), so W_S is the function W of Proposition 4.5(a), and by Proposition 4.5(b) the generating-function condition holds with $2^{\psi n} = \mathrm{base}(b, c, d)^n$: at $y = y^*$ if $d \leq bc$, and at $y = 0$ if $d \geq bc$. By the first inequality of Lemma 3.9, S satisfies $(\star)$. So Theorem 3.5 gives an algorithm that runs in time $\mathrm{base}(b, c, d)^n N^{O(1)}$, answers NO if $\mathcal{F}_I$ has no set of size at most k, and answers YES with probability at least $1 - 1/e$ if $\mathcal{F}_I$ has a set of size k. Running it for $k = 0, \ldots, n$ decides Φ-Subset as claimed (Remark 3.6). A witness Y returned by the extension algorithm gives the solution $X \cup Y$. For $d = 1$ we have $\min(1, bc) = 1$, since $bc \geq 1$, so $\mathrm{base}(b, c, 1) = 1 + b - 1/c$; for $b = d = 1$ this is $2 - 1/c$. □

Max Leaf Spanning Tree. For Max Leaf Spanning Tree, the extension problem that the original Monotone Local Search needs is W[1]-hard. Product-form running-time exponents certify nothing below a single unsampled call when $k/n \leq d/(bc)$, nor, when $c = 1$, for the worst solution size. Both statements and their proofs are in Appendix C.

5 Connected Dominating Set and Max Leaf Spanning Tree

This section prepares the applications of Sections 6 to 8 and 10: the equivalence of Max Leaf Spanning Tree and Connected Dominating Set (Lemma 5.1), a vertex-weighted directed Steiner tree computation by inclusion–exclusion (Lemma 5.3), the Steiner extension algorithm built on it, whose running-time exponent is the number of terminals that the sample leaves (Corollary 5.8), and the mean bound on this running-time exponent (Lemma 5.11).

All graphs are finite, simple and undirected. Throughout, G is a graph with vertex set $V = V(G)$ and $n = |V|$ vertices; when G is fixed we write $N(v)$, $N[v]$, $N[X] = \bigcup_{x \in X} N[x]$ and $N(X) = N[X] \setminus X$ for $N_G(v)$, $N_G[v]$, $N_G[X]$ and $N_G(X)$. For a subgraph $F \subseteq G$ we keep the subscript: $N_F(v)$, $N_F[X]$ and $d_F(v)$ are neighborhoods and degrees in F. For a graph, $\mathrm{comp}(\cdot)$ denotes the number of its connected components; in particular $\mathrm{comp}(G[\emptyset]) = 0$. A *connected dominating set* of G is a nonempty set $S \subseteq V$ with $N[S] = V$ such that $G[S]$ is connected; a graph that has one is connected, and for a connected graph G we write $\gamma_c(G)$ for the minimum size of a connected dominating set. Connected Dominating Set asks for $\gamma_c(G)$, and Max Leaf Spanning Tree (MaxLeaf) asks for the maximum number of leaves of a spanning tree of a connected graph G.

The following equivalence of the two problems is folklore; it is stated, for example, by Caro, West and Yuster [CWY00], by Fomin, Grandoni and Kratsch [FGK08], by [FKK+11, Section 1] and by Abu-Khzam, Mouawad and Liedloff [AKML11, Section 1]. We include its short proof because the construction in the second half of the proof is reused for the mean bound later in this section.

Lemma 5.1 (Max leaf and connected domination)**.** *Let G be a connected graph with n vertices. If $n \geq 3$, the maximum number of leaves of a spanning tree of G equals $n - \gamma_c(G)$, and a spanning tree*

with at least $n - |S|$ leaves can be computed in polynomial time from any connected dominating set S. If $n \leq 2$, the only spanning tree of G is G itself, with no leaf if $n = 1$ and two leaves if $n = 2$.

Proof. For $n \leq 2$ the claim is immediate, as a connected graph on at most two vertices is a tree. Let $n \geq 3$, let F be a spanning tree of G and let $\mathrm{leaves}(F)$ be its set of leaves. Since $n \geq 3$, two leaves of F are never adjacent in F (they would form a component K_2 of F), so the set $S = V \setminus \mathrm{leaves}(F)$ of non-leaves is nonempty and contains the neighbor in F of every leaf; hence $N[S] = V$. Deleting leaves from a tree leaves a tree, so $F - \mathrm{leaves}(F)$ is a spanning tree of $G[S]$ and $G[S]$ is connected. Thus S is a connected dominating set and $|\,\mathrm{leaves}(F)| = n - |S| \leq n - \gamma_{\mathrm{c}}(G)$.

Conversely, let S be a connected dominating set. Take a spanning tree of $G[S]$ and attach every vertex of $V \setminus S$ by one edge to a neighbor in S, which exists because $N[S] = V$. The result is a spanning tree of G in which every vertex of $V \setminus S$ is a leaf, so it has at least $n - |S|$ leaves, and it is computed in polynomial time. With $|S| = \gamma_{\mathrm{c}}(G)$ this gives the reverse inequality. □

All algorithms of Sections 6 to 8 and 10 and of Section 12 run, for $s = 1, \ldots, n$ in turn, a procedure that rejects s if no solution of size at most s exists and accepts the optimum size (with probability bounded away from 0, for the randomized ones), and output the first accepted s. The following lemma records what this wrapper needs from the inner procedure; it is stated for an arbitrary implicit set system (Definition 3.1).

Lemma 5.2 (First accepted size)**.** *Let Φ be an implicit set system and let I be an instance of Φ with a universe of n elements whose family $\mathcal{F}_I$ of solutions is nonempty and does not contain the empty set; let* $\mathrm{opt}(I) \in \{1, \ldots, n\}$ *be the least size of a solution. Let* Dec *be a procedure which, on input (I, s) with $1 \leq s \leq n$, runs in time T and space* Sp *and rejects if $\mathcal{F}_I$ contains no set of size at most s. Then running* Dec *on $(I, 1), (I, 2), \ldots, (I, n)$ and outputting the first accepted s, or n if none is accepted, runs in time nT and space* Sp *and never outputs a value below* $\mathrm{opt}(I)$*; if* Dec *accepts $(I, \mathrm{opt}(I))$ with probability at least p, the output is* $\mathrm{opt}(I)$ *with probability at least p (deterministic form: $p = 1$). For Φ_{CDS} (Definition 5.7), a connected graph G and* $\mathrm{opt}(G) = \gamma_{\mathrm{c}}(G)$*, the same procedure solves* Max Leaf Spanning Tree *within the same bounds, by Lemma 5.1, where graphs with $n \leq 2$ vertices are handled directly in constant time.*

Proof. The time and space bounds are clear. An accepted s satisfies $s \geq \mathrm{opt}(I)$, and the default output n satisfies $n \geq \mathrm{opt}(I)$ because every solution is a subset of the universe; so the output is never below $\mathrm{opt}(I)$. If the run on $(I, \mathrm{opt}(I))$ accepts, which happens with probability at least p, then, as no smaller s is ever accepted, the output is $\mathrm{opt}(I)$. For Max Leaf Spanning Tree, output $n - \gamma_{\mathrm{c}}(G)$ leaves for $n \geq 3$ by Lemma 5.1; for $n \leq 2$, output 0 if $n = 1$ and 2 if $n = 2$ without running Dec, which is correct by the last sentence of Lemma 5.1. □

5.1 Vertex-weighted directed Steiner trees

The extension algorithm of Section 5.2, the variants of Section 12 and Directed Steiner Tree in Section 12.5 all use one Steiner computation, on a digraph with vertex weights. This subsection states it, Lemma 5.3; its proof is in Appendix D.3.

Let $\vec{G}$ be a digraph (finite, without loops or parallel arcs) with n vertices, let $r \in V(\vec{G})$ be a *root*, let $K \subseteq V(\vec{G}) \setminus \{r\}$ be a set of *terminals*, and let $\varphi\colon V(\vec{G}) \to \{0, 1, \ldots, \varphi_{\max}\}$ be vertex weights; zero weights are allowed. A set $D \subseteq V(\vec{G}) \setminus (\{r\} \cup K)$ is *feasible* if every terminal is reachable from r in $\vec{G}[\{r\} \cup K \cup D]$, and we put

$$\mathrm{OPT}_{\varphi}(\vec{G}, r, K) = \min\{\varphi(D) : D \text{ feasible}\},$$

with $\min \emptyset = \infty$. Equivalently, $\mathrm{OPT}_\varphi(\vec{G}, r, K)$ is the least weight of the non-root, non-terminal vertices of an out-tree of $\vec{G}$ rooted at r that contains K. The weights of the root and of the terminals play no role.

Lemma 5.3 (Vertex-weighted directed Steiner tree)**.** *For $\vec{G}$, n, r, K and φ as above, $\mathrm{OPT}_\varphi(\vec{G}, r, K)$ can be computed deterministically in time $2^{|K|}(n + \varphi_{\max})^{O(1)}$ and space $(n + \varphi_{\max})^{O(1)}$.*

The method is Nederlof's [Ned13]: count, by inclusion–exclusion over the terminals, the branching walks from the root that visit every terminal, and read the optimum off the least length at which this count is nonzero. Nederlof states it for undirected graphs with edge weights; Lokshtanov and Nederlof obtain the same bound, $2^{|K|}$ times a factor polynomial in n and in the largest edge weight, from the Dreyfus–Wagner recurrence [LN10, Theorem 6.3], and [FKL$^+$19] remove the dependence on the weights at the price of a larger base, in time $O(7.97^{|K|} n^4 \log W)$ and polynomial space for edge weights at most W. Two changes give the lemma: arcs are used only in their own direction, and the weight of a vertex, multiplied by n, is added to the length of every arc that enters it, so that one unit of weight contributes more to a length than all arcs of an out-tree together. The directed case is not new: [JLR$^+$17, Lemma 3.5] state it for a minimum-size set of non-terminals and attribute it to Nederlof and to [MPR$^+$12], who prove it for Directed Steiner Out-Tree, with the number of vertices of the out-tree as the objective; the length encoding of vertex weights is a routine device. We regard the lemma as folklore and include a proof (Appendix D.3) because we need vertex weights, including zero weights, on a digraph.

5.2 Red–blue states and the Steiner encoding

Abu-Khzam, Mouawad and Liedloff [AKML11] solve Connected Dominating Set by reducing it to a connected red–blue domination problem, which they finish with a Steiner tree computation. We use their formulation, extended by *dominating components*, the connected pieces of a partial solution, which the encoding contracts to single vertices.

Definition 5.4 (State)**.** A *state* of a graph G is a triple $J = (\mathcal{D}, B, R)$ in which $\mathcal{D}$ is a family of pairwise disjoint nonempty vertex sets, each inducing a connected subgraph of G, with no edge of G between two distinct members, and, writing $V(\mathcal{D}) = \bigcup_{C \in \mathcal{D}} C$, the sets B and R satisfy $B \subseteq V \setminus V(\mathcal{D})$ and $R \subseteq V \setminus N[V(\mathcal{D})]$. The vertices in B are *blue* (still available), those in R are *red* (still to be dominated), and the members of $\mathcal{D}$ are the *dominating components*. The *solutions* of J are the sets

$$Y \subseteq B \quad \text{with} \quad R \subseteq N[Y], \quad G[V(\mathcal{D}) \cup Y] \text{ connected and } \quad V(\mathcal{D}) \cup Y \neq \emptyset,$$

and $\mathrm{OPT}(J)$ is the minimum size of a solution ($\mathrm{OPT}(J) = \infty$ if there is none). The *terminals* of J are its red vertices and its dominating components; their set is $\mathrm{T}(J) = R \cup \mathcal{D}$, so $|\mathrm{T}(J)| = |R| + |\mathcal{D}|$.

The initial state $J_0 = (\emptyset, V, V)$ has the connected dominating sets of G as its solutions, so $\mathrm{OPT}(J_0) = \gamma_{\mathrm{c}}(G)$ for connected G. For a nonempty set $X \subseteq V$ let $\mathcal{D}_X$ be the family of vertex sets of the components of $G[X]$, and put

$$J_X = (\mathcal{D}_X,\ V \setminus X,\ V \setminus N[X]), \tag{12}$$

and $J_\emptyset = J_0$. This is a state: the components of $G[X]$ are disjoint, connected and pairwise non-adjacent, $V(\mathcal{D}_X) = X$, and $V \setminus N[X]$ avoids $N[X]$. Unwinding the definition shows that the

solutions of J_X are exactly the sets $Y \subseteq V \setminus X$ for which $X \cup Y$ is a connected dominating set (for $X \neq \emptyset$ use $N[X \cup Y] = N[X] \cup N[Y]$; for $X = \emptyset$ this is the statement about J_0). The number of terminals is

$$|\mathrm{T}(J_X)| = n - |N[X]| + \mathrm{comp}(G[X]) \qquad \text{for every } X \subseteq V, \tag{13}$$

including $X = \emptyset$, where both sides equal n. Since $\mathrm{comp}(G[X]) \leq |X| \leq |N[X]|$, this is an integer in $\{0, \ldots, n\}$; if X is a connected dominating set, then $|\mathrm{T}(J_X)| = 1$.

The Steiner encoding turns a state into an instance of the directed Steiner problem of Section 5.1. Adjacencies among blue vertices and dominating components become pairs of opposite arcs, and each red vertex becomes a sink that can be entered only from a blue vertex dominating it, so that red terminals are leaves of every out-tree. Abu-Khzam, Mouawad and Liedloff [AKML11, Lemma 2] use an undirected encoding instead, in which heavy edge weights force the red terminals to be leaves, and solve it with Nederlof's algorithm [Ned13]; this is the classical route, and the directed encoding needs no heavy weights. We allow weights on the blue vertices, which Section 12.2 uses; for the unweighted problem all weights are 1.

Definition 5.5 (Directed Steiner encoding)**.** For a state $J = (\mathcal{D}, B, R)$ of G let $\vec{G}_J$ be the digraph with a vertex $\hat{C}$ for every $C \in \mathcal{D}$, a vertex $\hat{b}$ for every $b \in B$ and a vertex $\check{r}$ for every $r \in R$ (a red vertex r may lie in B, and then $\hat{r}$ and $\check{r}$ are different vertices), with the following arcs:

- $\hat{b}\hat{b}'$ and $\hat{b}'\hat{b}$ for every edge bb' of G with $b, b' \in B$;
- $\hat{C}\hat{b}$ and $\hat{b}\hat{C}$ for every $C \in \mathcal{D}$ and every $b \in N(C) \cap B$;
- $\hat{b}\check{r}$ for every $r \in R$ and every $b \in N[r] \cap B$.

For weights $\varphi\colon B \to \{0, 1, \ldots, \varphi_{\max}\}$ let $\mathrm{OPT}_\varphi(J)$ be the minimum of $\varphi(Y)$ over the solutions Y of J (∞ if there is none), and give $\hat{b}$ the weight $\varphi(b)$ and every $\hat{C}$ and $\check{r}$ the weight 0. With $\varphi \equiv 1$, $\mathrm{OPT}_\varphi(J) = \mathrm{OPT}(J)$.

The members of $\mathcal{D}$ are disjoint nonempty subsets of $V \setminus B$, so $|\mathcal{D}| + |B| \leq n$ and $\vec{G}_J$ has at most $2n$ vertices. The vertices of $\check{R} = \{\check{r} : r \in R\}$ have no outgoing arcs. The proofs of the next lemma and of Corollary 5.8 are in Appendix D.3; the first turns solutions of a state into feasible sets of its encoding, and back, and the second is the first with unit weights.

Lemma 5.6 (Directed Steiner encoding)**.** *Let $J = (\mathcal{D}, B, R)$ be a state of an n-vertex graph G and $\varphi\colon B \to \{0, 1, \ldots, \varphi_{\max}\}$. If $C_1 \in \mathcal{D}$, then $\mathrm{OPT}_\varphi(J) = \mathrm{OPT}_\varphi(\vec{G}_J, \hat{C}_1, K_1)$ for $K_1 = \{\hat{C} : C \in \mathcal{D} \setminus \{C_1\}\} \cup \check{R}$. Consequently, for every state J of G, $\mathrm{OPT}_\varphi(J)$ can be computed deterministically in time $2^{|\mathrm{T}(J)|}(n + \varphi_{\max})^{O(1)}$ and space $(n + \varphi_{\max})^{O(1)}$.*

The extension algorithm used in all the applications is this computation with unit weights.

Definition 5.7 (Set system and extension algorithm)**.** Let Φ_{CDS} be the implicit set system whose instances are graphs G, with universe $V(G)$ and with the connected dominating sets of G as solutions. Let $\mathcal{A}_{\mathrm{St}}$ be the algorithm that on input (G, X, k) rejects at once if $k < |X|$, and otherwise computes $\mathrm{OPT}(J_X)$ and accepts if and only if $\mathrm{OPT}(J_X) \leq k - |X|$.

Corollary 5.8 (Steiner extension)**.** *For every state J of an n-vertex graph, $\mathrm{OPT}(J)$ can be computed deterministically in time $2^{|\mathrm{T}(J)|}n^{O(1)}$ and polynomial space. Consequently, the algorithm $\mathcal{A}_{\mathrm{St}}$ of Definition 5.7 solves Φ_{CDS}-Extension in polynomial space, and*

$$g(G, X, k) = |\mathrm{T}(J_X)| = n - |N[X]| + \mathrm{comp}(G[X])$$

is a running-time exponent for it in the sense of Definition 3.2, with values in $\{0, \ldots, n\}$.

This running-time exponent does not depend on the size bound k. For a fixed graph G and target size k, the shorthand of Definition 3.2 therefore reads $g(X) = g(G, X, k) = |\mathrm{T}(J_X)|$, and the later sections use it in this form.

The running-time exponent depends on the sample itself, not only on its size, and adding a vertex to a sample need not reduce it: if $x \notin X$ has $N[x] \subseteq N[X]$ and is adjacent to exactly one component of $G[X]$, then $|\mathrm{T}(J_{X\cup\{x\}})| = |\mathrm{T}(J_X)|$. So product-form running times cannot credit a sampled vertex with a guaranteed reduction (Section 4); the mean number of terminals, however, does decrease with the sample size, as we show next.

5.3 Sampling inside a connected dominating set

The main result of this subsection is the mean bound of Lemma 5.11. It follows from a pointwise inequality for trees, applied to a spanning tree of G in which the vertices outside the solution are leaves.

For a tree F and a set $X \subseteq V(F)$, let $F\{X\}$ denote the subgraph of F with vertex set $N_F[X]$ and the edges of F with at least one end in X. It records how the components of $F[X]$ are joined to the neighbors of X; the inequality below is an equality exactly when it is connected.

Lemma 5.9 (Tree inequality)**.** *Let F be a tree and let $X \subseteq V(F)$ be nonempty. Then*

$$|N_F[X]| - \mathrm{comp}(F[X]) \;\geq\; 1 + \sum_{x\in X} \big(d_F(x) - 1\big),$$

with equality if and only if $F\{X\}$ is connected.

Proof. Let e_X be the number of edges of $F[X]$ and e_{XY} the number of edges of F with exactly one end in X; the other end of such an edge lies in $Y = N_F(X)$. Then $\sum_{x\in X} d_F(x) = 2e_X + e_{XY}$, and $\mathrm{comp}(F[X]) = |X| - e_X$ because $F[X]$ is a forest. The graph $F\{X\}$ is a forest with $|X| + |Y|$ vertices and $e_X + e_{XY}$ edges, so $e_X + e_{XY} \leq |X| + |Y| - 1$, with equality if and only if $F\{X\}$ is connected. Therefore

$$|N_F[X]| - \mathrm{comp}(F[X]) = |Y| + e_X \;\geq\; 2e_X + e_{XY} + 1 - |X| = 1 + \sum_{x\in X} \big(d_F(x) - 1\big),$$

with equality exactly when $e_X + e_{XY} = |X| + |Y| - 1$. □

Lemma 5.10 (Pointwise form)**.** *Let G be a connected graph with $n \geq 2$ vertices and let S be a connected dominating set of G. Let F be a spanning tree of G in which every vertex of $V \setminus S$ is a leaf; one exists, by the construction in the proof of Lemma 5.1, which needs only $n \geq 2$. Put $w(u) = d_F(u) - 1$ for $u \in S$. Then $\sum_{u\in S} w(u) = n - 2$, and for every nonempty $X \subseteq S$,*

$$|N[X]| - \mathrm{comp}(G[X]) \;\geq\; 1 + \sum_{x\in X} w(x).$$

Proof. The tree F has $n - 1$ edges and the $n - |S|$ vertices outside S have degree 1 in F, so $\sum_{u\in S} d_F(u) = 2(n-1) - (n - |S|)$ and $\sum_{u\in S} w(u) = n - 2$. For nonempty $X \subseteq S$ we have $N[X] \supseteq N_F[X]$ because $F \subseteq G$, and $\mathrm{comp}(G[X]) \leq \mathrm{comp}(F[X])$ because $F[X]$ is a spanning subgraph of $G[X]$. Now Lemma 5.9, applied to F and X, gives the inequality. □

Averaging the pointwise form over a uniform t-subset of the solution gives the mean bound.

Lemma 5.11 (Mean number of terminals)**.** *Let G be a connected graph with $n \geq 2$ vertices, let S be a connected dominating set of G with $|S| = s$, let $1 \leq t \leq s$ and let X be uniform in $\binom{S}{t}$. Then*

$$\mathbb{E}[|N[X]| - \mathrm{comp}(G[X])] \;\geq\; 1 + \frac{t(n-2)}{s}, \qquad \textit{equivalently} \qquad \mathbb{E}\,|\mathrm{T}(J_X)| \;\leq\; n - 1 - \frac{t(n-2)}{s}.$$

Proof. Take F and w as in Lemma 5.10. As $t \geq 1$, the set X is nonempty, so Lemma 5.10 applies to it. Each $u \in S$ lies in X with probability t/s, so by linearity of expectation

$$\mathbb{E}[|N[X]| - \mathrm{comp}(G[X])] \;\geq\; 1 + \frac{t}{s}\sum_{u \in S} w(u) = 1 + \frac{t(n-2)}{s}.$$

The second form follows from (13). □

The assumption $t \geq 1$ of Lemma 5.11 is needed: for $X = \emptyset$ the expectation $\mathbb{E}[|N[X]| - \mathrm{comp}(G[X])]$ is 0. The bound of Lemma 5.11 cannot be improved in general:

Proposition 5.12 (Tightness for stars)**.** *Let S be a connected dominating set of a connected graph G with $|S| = s \geq 2$. Suppose that $G[S]$ is a star $K_{1,s-1}$ and that every vertex of $V \setminus S$ has exactly one neighbor in S (edges between vertices of $V \setminus S$ are allowed). Then equality holds in Lemma 5.10 for every nonempty $X \subseteq S$, and in Lemma 5.11 for every $1 \leq t \leq s$.*

An example for Proposition 5.12 is the *star of stars*: a star on S in which every vertex of S carries h pendant vertices, so that $n = s(h+1)$.

Proof. First, the tree F of Lemma 5.10 is unique. Deleting the leaves of F that lie outside S leaves a spanning tree of $G[S]$, which must be the star $G[S]$ itself. If $V \setminus S$ is nonempty, then every $v \notin S$ is a leaf of F, and its neighbor in F lies in S: otherwise that neighbor is a leaf as well and F would be a single edge, whereas $|V(F)| \geq s + 1 \geq 3$. So v is attached to its unique neighbor in S.

For $X \subseteq S$ we have $G[X] = F[X]$, and $N[X] = N_F[X]$ because the only edges of G between S and $V \setminus S$ are edges of F. So equality in Lemma 5.10 is equality in Lemma 5.9, and it remains to show that $F\{X\}$ is connected for nonempty $X \subseteq S$. Let z be the center of the star. If $z \in X$, then $F[X]$ is connected, as every other vertex of X is adjacent to z, and every vertex of $N_F(X)$ is joined to X by an edge of $F\{X\}$. If $z \notin X$, then the vertices of X are all adjacent to $z \in N_F(X)$, and every other vertex of $N_F(X)$ lies outside S and is adjacent to one of them. In both cases $F\{X\}$ is connected. Equality in Lemma 5.11 follows, because its proof uses only Lemma 5.10 and linearity of expectation. □

6 Warm-up: sampling a connected dominating set with the mean running-time exponent

This section combines the linear mean savings of Corollary 4.3 with the Steiner extension (Corollary 5.8) and the mean bound of Lemma 5.11. The resulting Algorithm RS is Algorithm MLS-θ for connected dominating sets, run with a constant rate: it samples vertices of an unknown connected dominating set and completes the sample by a Steiner tree computation. We prove that it runs in randomized time $O(1.7449^n)$ and polynomial space. At each solution fraction, the certified rate has a closed form (Proposition 6.1), and at solution fraction $\frac{1}{2}$ the corresponding base is exactly $\sqrt{3}$. RS shows the mechanism in its simplest form: its analysis uses, through Jensen's inequality, only the mean number of terminals of the samples inside a solution. Section 7 runs the same algorithm

with a smaller rate and bounds the exponential moments directly on part of the range of solution sizes.

Throughout, $\sigma = s/n$ denotes the solution fraction of a candidate size s. The *certified rate* at σ is the infimum of the rates ψ (Section 3) for which our analysis gives condition $(\star)$ for every connected dominating set of size $s = \sigma n$. Algorithm RS uses one rate, ψ_{RS}, above the maximum of the certified rate over σ. The certified rate is governed by the function

$$F^*(\sigma) = H(\sigma) + (1-\sigma)\log(1 - 2^{-1/\sigma}) \qquad (0 < \sigma < 1), \tag{14}$$

which is the function F^* of (6) with $\beta = 1$ and $\rho = 1/\sigma$: since $2^x > x$ for every real x, we have $2^{1/\sigma} > 1/\sigma$, that is $p = 2^{-1/\sigma} < \sigma$, so the second case of (6) applies, and $\sigma\rho = 1$ turns its last expression into (14). The corresponding base is

$$2^{F^*(\sigma)} = \frac{\left(1 - 2^{-1/\sigma}\right)^{1-\sigma}}{\sigma^\sigma (1-\sigma)^{1-\sigma}},$$

whose denominator is $2^{-H(\sigma)}$, where $2^{nH(\sigma)}$ is, up to a polynomial factor, the number of candidate sets of size σn, and whose numerator is the saving. We put $F^*(1) = 0$, the value of the right-hand side of (14) at $\sigma = 1$ and its limit as $\sigma \to 1$. By Remark 4.4, for $\sigma < 1$, $F^*(\sigma)$ is the minimum over the sample fraction $\tau \in [0, \sigma]$ of

$$f_\sigma(\tau) = H(\tau) - \sigma H(\tau/\sigma) + 1 - \frac{\tau}{\sigma},$$

whose first part is, up to an additive $O(\log n)/n$ by (1), the loss $\frac{1}{n}\log\left(\binom{n}{t}/\binom{s}{t}\right)$ of requiring a uniform t-subset of V to lie inside a fixed connected dominating set of size s, and whose second part, $1 - \tau/\sigma$, is the mean bound of Lemma 5.11 on the number of terminals, divided by n. A single sample size and the Markov cutoff of Remark 4.2 attain this exponent up to the factors $2^{1/\sigma}$ and $s + 1$ (Remark 4.4); Algorithm MLS-θ attains it without rounding.

How RS decides. For each candidate size s, RS runs Algorithm MLS-θ with a fixed rate. For each sample X it monitors the number of terminals $|\mathrm{T}(J_X)| = n - |N[X]| + \mathrm{comp}(G[X])$, which is computed in polynomial time, and compares it with the current threshold θ (the thresholds of Algorithm MLS-θ, Section 3.2). A sample at or below the threshold is completed by the Steiner tree algorithm, in time at most $2^\theta n^{O(1)}$; a sample above it is discarded in polynomial time.

The next proposition evaluates the certified rate at a given solution fraction: it gives the value at $\sigma = \frac{1}{2}$ and an upper bound on the maximum over σ, together with two bounds on ranges of σ that Section 7 uses.

Proposition 6.1 (Certified rate of RS)**.** *Let F^* be the function of* (14)*.*

(a) For every $\sigma \in (0,1)$, $F^*(\sigma) < H(\sigma) \le 1$, and the minimum of f_σ over $[0,\sigma]$ is attained exactly at $\tau^*(\sigma) = \dfrac{\sigma - 2^{-1/\sigma}}{1 - 2^{-1/\sigma}} \in (0,\sigma)$.

(b) At $\sigma = \frac{1}{2}$: $\tau^* = \frac{1}{3}$ and $F^*(\frac{1}{2}) = \frac{1}{2}\log 3$, so the base $2^{F^*(1/2)}$ is exactly $\sqrt{3}$.

(c) $F^*(\sigma) \le \log 1.744835$ for every $\sigma \in (0,1]$. Moreover $F^*(\sigma) \le 0.7807$ for $\sigma \in (0, 0.31]$ and $F^*(\sigma) \le 0.7802$ for $\sigma \in [0.54, 1]$.

Proof. (a) The first claim holds because $\log(1-2^{-1/\sigma}) < 0$ and $H \le 1$. The second is Remark 4.4 with $p = 2^{-1/\sigma} < \sigma$.

(b) Here $2^{-1/\sigma} = \frac{1}{4}$ and $\tau^* = \frac{1/2-1/4}{3/4} = \frac{1}{3}$, and $2^{F^*(1/2)} = (\frac{3}{4})^{1/2}/\frac{1}{2} = \sqrt{3}$.

(c) The certificate of Appendix F.1 gives $F^*(\sigma) \le 0.8030906143 < \log 1.744835$ for every $\sigma \in [0.2, 0.8]$, $F^* \le 0.7807$ on $[0.15, 0.31]$ and $F^* \le 0.7802$ on $[0.54, 0.8]$. For $\sigma \in (0,1) \setminus [0.2, 0.8]$, (a) gives $F^*(\sigma) \le H(\sigma) \le H(0.2) = \log 5 - 1.6 < 0.73$, because H is increasing on $[0, \frac{1}{2}]$ and symmetric about $\frac{1}{2}$, and $5^{100} < 2^{233}$; and $F^*(1) = 0$. As $0.73 < 0.7802$, this bound covers the rest of the two ranges. □

The closed form bounds the exponential moments of Definition 3.3 for every connected dominating set; the corollary is used again in Section 7.

Corollary 6.2 (Closed-form moment bound)**.** *Let G be a connected graph with n vertices and let S be a connected dominating set of G with $|S| = s$ and $\sigma = s/n$. Then, for the running-time exponent $g(X) = |\mathrm{T}(J_X)|$ of Corollary 5.8,*

$$\max_{0 \le t \le s} \frac{\mu_t(S)}{\binom{n}{t}} \;\ge\; \frac{2^{-nF^*(\sigma)}}{2}.$$

Proof. If $s = n$, then $S = V$, $\mu_n(S) = 2^{-|\mathrm{T}(J_V)|} = \frac{1}{2}$ by (13), and $F^*(1) = 0$. Otherwise $n \ge 2$ and $\sigma \in (0,1)$. For $1 \le t \le s$, Lemma 5.11 gives

$$\mathbb{E}_{X \in \binom{S}{t}} |\mathrm{T}(J_X)| \le n - 1 - \frac{t(n-2)}{s} = n - \frac{t}{\sigma} + \frac{2t}{s} - 1 \le n - \frac{t}{\sigma} + 1,$$

as $t \le s$, and for $t = 0$ we have $|\mathrm{T}(J_\emptyset)| = n \le n + 1$. So (7) holds with $\beta = 1$, $\rho = 1/\sigma$ and $C = 1$ for every connected dominating set of size s, and Corollary 4.3(1) gives $G(S) \le 2^{1+nF^*(\sigma)}$ with F^* as in (14). The first inequality of Lemma 3.9 gives $\min_{0 \le t \le s} \binom{n}{t}/\mu_t(S) = M(S) \le G(S)$. □

We can now state the algorithm.

Algorithm 6.3 (RS)

Input: a connected graph G with n vertices.

Parameters: $\psi_{\mathrm{RS}} = 0.80309062$ and a constant $\kappa > 0$.

For $s = 1, 2, \ldots, n$ in turn, run Algorithm MLS-$\theta(\psi_{\mathrm{RS}}, 2, \kappa)$ (Algorithm 3.4) on (G, s) for the implicit set system Φ_{CDS}, the extension algorithm $\mathcal{A}_{\mathrm{St}}$ and the running-time exponent g of Corollary 5.8, written $g(X) = g(G, X, s) = |\mathrm{T}(J_X)|$ as declared after that corollary. Output the first s for which the run answers YES, and output n if there is none.

Theorem 6.4 (Algorithm RS)**.** *For every constant $\kappa > 0$, Algorithm RS runs in time $O(1.7449^n)$ and polynomial space. It never outputs a value below $\gamma_{\mathrm{c}}(G)$, and it outputs $\gamma_{\mathrm{c}}(G)$ with probability at least $1 - e^{-\kappa}$. Consequently* Connected Dominating Set *and* Max Leaf Spanning Tree *(by Lemma 5.1, and directly when $n \le 2$) can be solved by a randomized algorithm with one-sided error (one-sided Monte Carlo) in time $O(1.7449^n)$ and polynomial space.*

Proof. We verify the hypotheses of Lemma 5.2 for the run on (G, s). By Corollary 5.8, $\mathcal{A}_{\mathrm{St}}$ solves Φ_{CDS}-Extension in polynomial space with the running-time exponent g, and the parameters ψ_{RS}, $\Gamma = 2$ and κ are constants. So Theorem 3.5 applies: the run takes time $2^{\psi_{\mathrm{RS}} n} n^{O(1)}$ and polynomial space, and it answers NO if $\gamma_{\mathrm{c}}(G) > s$.

Completeness. Let $s = \gamma_c(G)$ and let S be a connected dominating set with $|S| = s$. By Corollary 6.2 and Proposition 6.1(c),

$$\max_{0\le t\le s} \frac{\mu_t(S)}{\binom{n}{t}} \;\ge\; \frac{2^{-n\log 1.744835}}{2} \;\ge\; \frac{2^{-\psi_{\mathrm{RS}} n}}{2},$$

as $\psi_{\mathrm{RS}} > \log 1.744835$; that is, S satisfies $(\star)$ with $\Gamma = 2$. By Theorem 3.5(iii), the run on $(G, \gamma_c(G))$ answers YES with probability at least $1 - e^{-\kappa}$.

Lemma 5.2 now gives the statements about the output, and the total time is $n \cdot 2^{\psi_{\mathrm{RS}} n} n^{O(1)} = O(1.7449^n)$, because $2^{\psi_{\mathrm{RS}}} = 1.744835\ldots < 1.7449$, where the constant 1.744835 comes from Proposition 6.1(c). □

7 Exponential moments of the Steiner extension: Algorithm RS-θ

The analysis of Algorithm RS in Section 6 uses, through Jensen's inequality, only the mean number of terminals of the samples inside a solution. By Proposition 5.12 the bound of Lemma 5.11 on this mean cannot be improved, but how it is used can: when the number of terminals fluctuates, Jensen's inequality $\mathbb{E}[2^{-|\mathrm{T}(J_X)|}] \ge 2^{-\mathbb{E}|\mathrm{T}(J_X)|}$ is strict, and the samples with few terminals account for the difference. Algorithm RS-θ is Algorithm RS run with a smaller rate: it is Algorithm MLS-θ (Algorithm 3.4) for connected dominating sets with the Steiner extension of Corollary 5.8, and its success condition in Theorem 3.5 is condition $(\star)$ on the exponential moments $\mu_t(S)$ of the number of terminals, which we now bound directly. We prove that it runs in randomized time $O(1.7204^n)$ and polynomial space (Theorem 7.2).

The analysis needs a lower bound on these moments for *every* connected dominating set. Since $\mathbb{E}[2^{-|\mathrm{T}(J_X)|}]$ is not linear in $|\mathrm{T}(J_X)|$, the linearity-of-expectation argument behind Lemma 5.11 does not apply. We bound the number of terminals by a potential that is local along a spanning tree of the solution (Lemma 7.3), evaluate its generating function by a three-state recursion (Lemma 7.4), and bound the recursion from below, uniformly over all trees, by an antinorm certificate (Lemma 7.5), following Rote [Rot19] and Rosenfeld [Ros21]. The proofs of these three steps are in Appendix D.4. Condition $(\star)$ is then verified at every solution fraction by two certificates or by the closed form of Section 6 (Table 3). On the uniform $(2,2)$-lobsters the analysis is nearly tight for this algorithm (Proposition 7.9).

For a connected graph G and the running-time exponent g of Corollary 5.8, written $g(X) = g(G, X, s) = |\mathrm{T}(J_X)|$ in the shorthand of Definition 3.2 with target size $k = s$, the quantities of Definition 3.3 for a connected dominating set S with $|S| = s$ are

$$\mu_t(S) = \sum_{X\in\binom{S}{t}} 2^{-|\mathrm{T}(J_X)|} \quad (0 \le t \le s), \qquad W_S(y) = \sum_{X\subseteq S} y^{|X|}\, 2^{-|\mathrm{T}(J_X)|} \quad (y \ge 0); \tag{15}$$

the running-time exponent does not depend on the size bound k. Recall from (13) that $|\mathrm{T}(J_X)|$ is an integer in $\{0,\ldots,n\}$, that $|\mathrm{T}(J_\emptyset)| = n$ and that $|\mathrm{T}(J_S)| = 1$.

7.1 The algorithm and the result

Algorithm 7.1 (RS-θ)

Input: a connected graph G with n vertices.

Parameters: $\psi^* = 0.7827016$ and a constant $\kappa > 0$.

For $s = 1, 2, \ldots, n$ in turn, run Algorithm MLS-$\theta(\psi^*, 2, \kappa)$ on (G, s) for the implicit set system Φ_{CDS}, the extension algorithm $\mathcal{A}_{\mathrm{St}}$ and the running-time exponent $g(X) = |\mathrm{T}(J_X)|$ of Corollary 5.8, as in Algorithm RS. Output the first accepted s, or n if no s is accepted.

So RS-θ differs from RS only in its rate, and the improvement lies in the analysis. The analysis of RS bounds the moments $\mu_t(S)$ through the mean of Lemma 5.11, whereas the analysis below chooses, for each minimum connected dominating set S, a pair (t, θ) at which the number of samples $X \subseteq S$ of size t with $|\mathrm{T}(J_X)| \le \theta$ is, up to a polynomial factor, at least $2^{\theta}\mu_t(S)$ (Lemma 3.8). For instance, the pair $t = s$, $\theta = 1$ replaces enumeration of the s-sets: each of its trials draws a uniform s-set X, and the threshold test passes exactly when X is a connected dominating set. The main result of this section is the following.

Theorem 7.2 (Algorithm RS-θ)**.** *For every constant $\kappa > 0$, Algorithm RS-θ runs in time $O^*(2^{\psi^* n})$, where $2^{\psi^*} = 1.7203\ldots < 1.7204$, and in polynomial space. It never outputs a value below $\gamma_{\mathrm{c}}(G)$, and it outputs $\gamma_{\mathrm{c}}(G)$ with probability at least $1 - e^{-\kappa}$. Consequently* Connected Dominating Set*, and* Max Leaf Spanning Tree *(by Lemma 5.1, and directly when $n \le 2$), can be solved by a randomized algorithm with one-sided error (one-sided Monte Carlo) in time $O(1.7204^n)$ and polynomial space.*

The proof is given at the end of Section 7.5; it rests on Proposition 7.8, which verifies condition ($\star$) for every connected dominating set. The constant 1.7204 rests on Proposition 7.7 and Proposition 6.1(c).

7.2 Reduction to a tree with an owner map

The first step replaces the graph by a tree. Let S be a connected dominating set of G. An *owner map* is a map $o : V \setminus S \to S$ with $o(u) \in N(u)$ for every $u \notin S$; it exists because S is dominating. For $v \in S$ let $\nu_v = |o^{-1}(v)|$ be the number of vertices that v owns, so that $\sum_{v \in S} \nu_v = n - |S|$. For a tree F with $V(F) = S$, a map $\nu : S \to \mathbb{Z}_{\ge 0}$ and $X \subseteq S$ put

$$\Pi_{F,\nu}(X) = \sum_{v \in S \setminus X} \nu_v \;+\; \bigl|\{v \in S : N_F[v] \cap X = \emptyset\}\bigr| \;+\; \mathrm{comp}(F[X]),$$
$$W_{F,\nu}(y) = \sum_{X \subseteq S} y^{|X|}\, 2^{-\Pi_{F,\nu}(X)}. \tag{16}$$

The three terms count, in order: the owned vertices whose owner is not sampled, the solution vertices not dominated by a sample within the tree, and the components of the sample within the tree. We call $\Pi_{F,\nu}$ the *potential*; the next lemma shows that it is at least the running-time exponent $|\mathrm{T}(J_X)|$ of every sample $X \subseteq S$.

Lemma 7.3 (Tree reduction)**.** *Let G be a connected graph, let S be a connected dominating set of G, let F be a spanning tree of $G[S]$ and let o be an owner map with counts ν. Then for every $X \subseteq S$,*

$$|\mathrm{T}(J_X)| \;\le\; \Pi_{F,\nu}(X).$$

Equality holds for every $X \subseteq S$ if $G[S] = F$ and every vertex of $V \setminus S$ has exactly one neighbor in S. Consequently $W_S(y) \ge W_{F,\nu}(y)$ for every $y \ge 0$.

7.3 A three-state recursion

The second step computes $W_{F,\nu}(y)$ by dynamic programming over the tree. Root F at a vertex r; for $v \in V(F)$ let F_v be the subtree formed by v and its descendants. A *message* is a vector $P = (P_1, P_2, P_3) \in \mathbb{R}^3$ whose coordinates are sums of weights $y^{|X|}2^{-\Pi^{\mathrm{in}}(X)}$, where $\Pi^{\mathrm{in}}(X)$ is the part of the potential of X that is determined inside the subtree (made precise in the proof of Lemma 7.4); the sums run over the subsets X of that subtree in each of three *states* of its top vertex v: in state 1, $v \in X$; in state 2, $v \notin X$ and some child of v lies in X; in state 3, $v \notin X$ and no child of v lies in X. For $P, Q \in \mathbb{R}^3$ let

$$P \circ Q = \Big(P_1(2Q_1 + Q_2 + Q_3),\quad P_2(Q_1 + Q_2 + \tfrac{Q_3}{2}) + P_3 Q_1,\quad P_3(Q_2 + \tfrac{Q_3}{2})\Big), \tag{17}$$

and let $\mathrm{fin}(P) = P_1 + P_2 + \frac{1}{2}P_3$ and $\mathrm{Init}_y(j) = (\frac{y}{2}, 0, 2^{-j})$ for integers $j \geq 0$; the argument j of Init_y will be the number ν_v of vertices owned by v. The operation $\circ$ is bilinear, and it adds a completed child with message Q to a partially built message P. For $y \geq 0$, order the children of every vertex arbitrarily, and for $v \in V(F)$ with children $w_1, \ldots, w_d$ in this order define the messages from the leaves upwards by

$$\mathrm{msg}_v = \Big(\cdots\big(\mathrm{Init}_y(\nu_v) \circ \mathrm{msg}_{w_1}\big) \circ \cdots\Big) \circ \mathrm{msg}_{w_d}. \tag{18}$$

The output of this recursion is as follows.

Lemma 7.4 (Tree recursion)**.** *Let $y \geq 0$, and let the messages msg_v be defined by (18) for any order of the children. Then every msg_v has nonnegative entries and $W_{F,\nu}(y) = \mathrm{fin}(\mathrm{msg}_r)$.*

7.4 Antinorm certificates in the cumulative cone

The third step bounds $\mathrm{fin}(\mathrm{msg}_r)$ from below for all trees at once. The idea is to find a convex region that the merge operation maps into itself up to a known scale; the scale then multiplies along the tree. The region must be closed upwards in an order that the recursion respects. Let

$$\mathcal{K}_{\mathrm{cum}} = \{P \in \mathbb{R}^3 :\ P_1 \geq 0,\ P_1 + P_2 \geq 0,\ P_1 + P_2 + P_3 \geq 0\},$$

the *cumulative cone*. It is the preimage of $\mathbb{R}^3_{\geq 0}$ under the cumulative map $P \mapsto (P_1, P_1 + P_2, P_1 + P_2 + P_3)$, so it is a convex cone generated by $(1, -1, 0)$, $(0, 1, -1)$ and $(0, 0, 1)$, and it contains $\mathbb{R}^3_{\geq 0}$. For a finite nonempty set $Z \subseteq \mathbb{R}^3_{\geq 0}$ let $\mathcal{P}_Z = \mathrm{conv}(Z) + \mathcal{K}_{\mathrm{cum}}$ and $c_Z = \min_{z \in Z} \mathrm{fin}(z)$. The facts about $\mathcal{K}_{\mathrm{cum}}$ that the certificate argument uses (Lemma D.4.1) and the abstract mechanism behind it (Lemma D.4.2), which Section 8 reuses with another cone, are proved in Appendix D.4.

The next lemma is the core of the third step. If a finite set Z of nonnegative messages is such that $\mathcal{P}_Z$ absorbs the merge of any two of its generators and contains every initial message up to a geometric scale, then that scale multiplies along any tree. Data (y, e_1, e_2, Z) with these two properties are the *certificates* referred to above.

Lemma 7.5 (Antinorm certificate)**.** *Let $y \geq 0$ and $e_1, e_2 > 0$, and let $Z \subseteq \mathbb{R}^3_{\geq 0}$ be finite and nonempty with*

(a) $z \circ z' \in \mathcal{P}_Z$ for all $z, z' \in Z$, and

(b) $\mathrm{Init}_y(j) \in e_1 e_2^j\, \mathcal{P}_Z$ for every integer $j \geq 0$.

Then for every tree F with s vertices and every $\nu : V(F) \to \mathbb{Z}_{\geq 0}$, writing $\nu(F) = \sum_{v \in V(F)} \nu_v$, we have $W_{F,\nu}(y) \geq c_Z\, e_1^s e_2^{\nu(F)}$. If $e_2 \leq 1$, then (b) holds for all j as soon as it holds for $j \leq j_0$ and $(y/2, 0, 0) \in e_1 e_2^{j_0+1} \mathcal{P}_Z$.

range of $\sigma = s/n$	argument	certified exponent
$(0, 0.31]$	closed form (Corollary 6.2, Proposition 6.1(c))	$F^*(\sigma) \le 0.7807$
$[0.31, 0.4172629]$	certificate 0	$\psi_0(\sigma) \le 0.78270152$
$[0.4172629, 0.54]$	certificate 1	$\psi_1(\sigma) \le 0.78270152$
$[0.54, 1]$	closed form (Corollary 6.2, Proposition 6.1(c))	$F^*(\sigma) \le 0.7802$

Table 3: Coverage of the solution fraction σ in Proposition 7.8. The maximum, 0.7827015136 (base $2^{0.7827015136} = 1.7203\ldots$), is attained at $\sigma = 0.4172629$.

Together with Lemma 7.3, a certificate turns into the exponential-moment condition of the main theorem.

Corollary 7.6 (From certificates to exponential moments)**.** *Suppose that (y, e_1, e_2, Z) satisfies the hypotheses of Lemma 7.5. Then for every connected graph G and every connected dominating set S of G with $|S| = s$, writing $\sigma = s/n$,*

$$W_S(y) \ge c_Z\, e_1^s e_2^{n-s} \qquad \text{and} \qquad \max_{0 \le t \le s} \frac{\mu_t(S)}{\binom{n}{t}} \;\ge\; c_Z\, 2^{-n\,\psi_{\mathrm{cert}}(\sigma)},$$

where $\psi_{\mathrm{cert}}(\sigma) = \log(1+y) - \sigma \log e_1 - (1-\sigma)\log e_2$.

So a certificate gives condition $(\star)$ of Theorem 3.5 for S with the rate $\psi = \psi_{\mathrm{cert}}(\sigma)$ and the loss factor $\Gamma = \max(1, 1/c_Z)$; we call ψ_{cert} the *rate function* of the certificate.

7.5 The certificates and the coverage of the solution fraction

Two certificates for Lemma 7.5 cover the middle range of the solution fraction; the closed form of Section 6 covers the rest. We record the certificates first and then combine the two arguments.

Proposition 7.7 (Antinorm certificates)**.** *For $i \in \{0, 1\}$, the data $(y_i, e_{1,i}, e_{2,i}, Z_i)$ of certificate i (Appendix F.2) are rational and satisfy the hypotheses of Lemma 7.5, with $j_0 = 30$ and $e_{2,i} < 1$:*

i	y_i	$e_{1,i}$	$e_{2,i}$	$\lvert Z_i\rvert$	$c_{Z_i} \ge$	$\psi_i(\sigma)$
0	$\frac{69}{200}$	$\frac{747593601}{960952792}$	$\frac{512099210}{652701491}$	27	0.6212	$0.7776061742 + 0.0122113367\,\sigma$
1	$\frac{17}{50}$	$\frac{248231700}{312721579}$	$\frac{568097827}{739289481}$	29	0.6293	$0.8022330021 - 0.0468085913\,\sigma$

Here ψ_i is the rate function ψ_{cert} of Corollary 7.6, with coefficients rounded to ten decimals. The two lines cross at $\sigma_\times = 0.4172629270\ldots$, and

$$\max\Big(\max_{\sigma \in [0.31,\, 0.4172629]} \psi_0(\sigma),\ \max_{\sigma \in [0.4172629,\, 0.54]} \psi_1(\sigma)\Big) = 0.78270151357\ldots < \psi^*.$$

Proof. The hypotheses of Lemma 7.5 are finitely many conditions on the rational data; they are checked for both certificates in Appendix F.2. Since ψ_i is linear in σ, its maximum on an interval is attained at an endpoint, so the displayed maximum is the largest of the values of ψ_0 at 0.31 and 0.4172629 and of ψ_1 at 0.4172629 and 0.54; these four values are bounded from above in Appendix F.2, and each is below $0.78270152 < \psi^*$. □

The cases of Table 3 combine into condition $(\star)$ for every connected dominating set.

Proposition 7.8 (Rate of RS-θ)**.** *For every connected graph G and every connected dominating set S of G,*

$$\max_{0\le t\le |S|} \frac{\mu_t(S)}{\binom{n}{t}} \;\ge\; \frac{2^{-\psi^* n}}{2}.$$

Proof. Let $s = |S|$ and $\sigma = s/n$; we follow Table 3.

Closed form: $\sigma \notin [0.31, 0.54]$. By Corollary 6.2 and Proposition 6.1(c), $\max_t \mu_t(S)/\binom{n}{t} \ge 2^{-nF^*(\sigma)}/2 \ge 2^{-0.7807n}/2 \ge 2^{-\psi^* n}/2$, as $0.7807 < \psi^*$.

Certificates: $\sigma \in [0.31, 0.54]$. Take $i = 0$ if $\sigma \le 0.4172629$ and $i = 1$ otherwise. Corollary 7.6 and Proposition 7.7 give $\max_t \mu_t(S)/\binom{n}{t} \ge c_{Z_i} 2^{-n\psi_i(\sigma)} \ge 0.62 \cdot 2^{-\psi^* n} \ge 2^{-\psi^* n}/2$. □

Proof of Theorem 7.2. We verify the hypotheses of Lemma 5.2. For each s, RS-θ is Algorithm MLS-θ for Φ_{CDS}, $\mathcal{A}_{\mathrm{St}}$ and $g(X) = |\mathrm{T}(J_X)|$, with $k = s$, $\psi = \psi^*$, $\Gamma = 2$ and confidence κ. By Corollary 5.8, $\mathcal{A}_{\mathrm{St}}$ solves Φ_{CDS}-Extension in polynomial space and g is a running-time exponent for it, and the parameters are constants. So Theorem 3.5 applies: the run for s takes time $2^{\psi^* n} n^{O(1)}$ and polynomial space, and it rejects if $\gamma_{\mathrm{c}}(G) > s$. For $s = \gamma_{\mathrm{c}}(G)$ let S be a minimum connected dominating set; Proposition 7.8 gives condition ($\star$) for S with $\Gamma = 2$, so s is accepted with probability at least $1 - e^{-\kappa}$. Finally, $2^{\psi^*} = 2^{0.7827016} = 1.7203\ldots < 1.7204$. □

7.6 A lower bound for RS-θ: uniform lobsters

The analysis above is close to tight for Algorithm RS-θ itself. Recall that a *caterpillar* is a tree that becomes a path when its leaves are deleted, and a *lobster* is a tree that becomes a caterpillar when its leaves are deleted. The *uniform* $(2,2)$*-lobster* Λ_L has a spine path $x_1 x_2 \cdots x_L$; every spine vertex has exactly two neighbors off the spine, its *legs*, and every leg has exactly two further neighbors, both of degree 1, its *claws*. So the legs are the vertices at distance 1 from the spine, the claws are the vertices at distance 2 and the leaves of Λ_L, and $n = 7L$. The proposition identifies the minimum solution of Λ_L, bounds the acceptance probability of RS-θ on it in terms of the normalized moments, and bounds these moments by a certificate.

Proposition 7.9 (Uniform $(2,2)$-lobsters)**.** *Let $L \ge 1$, let $n = 7L$ be the number of vertices of Λ_L, and let S^* be the set of its spine vertices and legs. Then:*

(a) S^* is the unique minimum connected dominating set of Λ_L; so $\gamma_{\mathrm{c}}(\Lambda_L) = 3L$ and $\sigma = 3/7$.

(b) Run Algorithm RS-θ with any rate $\psi \ge 0$ in place of ψ^* (and the corresponding cap and trial counts). For $s = \gamma_{\mathrm{c}}(\Lambda_L)$, the probability that s is accepted is at most $P_{\mathrm{nec}}\, 2^{\psi n} \max_t \mu_t(S^*)/\binom{n}{t}$, where $P_{\mathrm{nec}} \le n^{O(1)}$ is the polynomial factor of Proposition 3.10 (with $\Gamma = 2$).

(c) For all $L \ge 1$ and $0 \le t \le n$, $\mu_t(S^*)/\binom{n}{t} \le (n+1)\, 1.71927^{-n}$. Conversely, $\min_t \binom{n}{t}/\mu_t(S^*) \le C \cdot 1.71928^n$ for a constant C and all $L \ge 1$. So $\left(\min_t \binom{n}{t}/\mu_t(S^*)\right)^{1/n}$ lies in $[1.71927, 1.71928]$ up to factors $(n+1)^{1/n}$ and $C^{1/n}$; numerically its limit is $1.7192725\ldots$, attained near $y^* = 0.3453$ in the generating-function form.

Consequently, RS-θ run with a rate ψ in place of ψ^* outputs $\gamma_{\mathrm{c}}(\Lambda_L)$ with probability bounded away from 0 on all Λ_L only if $2^{\psi} \ge 1.71927$; so the certified base 1.7204 exceeds the least base with which this algorithm succeeds on all Λ_L by at most 0.0012.

The proposition holds for every graph whose unique maximum-leaf spanning tree is Λ_L with $L \ge 2$ (Appendix D.4.3; parts (b) and (c) with their proofs verbatim, part (a) by a separate argument), so the lower bound does not rest on instances on which Max Leaf Spanning Tree is easy.

8 Closure sampling under exponential moments: Algorithm RSC

The analysis of Section 7 is nearly tight for its algorithm, and the instances on which it is nearly attained have a common feature: some vertices of the solution have many neighbors inside the solution. A random sample of the solution then contains several neighbors of such a vertex, and the vertex stands out among the vertices with at least two sampled neighbors. Algorithm RSC adds each such *candidate* to the sample independently with a small probability, before it computes the number of terminals and decides whether to call the Steiner tree algorithm. The probability is *graded*: a candidate with exactly two sampled neighbors is added with probability p_2, a candidate with three or more with probability $p_3 \geq p_2$. The main theorem in its budgeted-trial form (Theorem 3.14) applies to these trials directly, and the exponential moment in its success condition becomes an average over the random closure as well as over the sample. For $(p_2, p_3) = (0.07, 0.077)$ we prove the following bound, the best one in this paper.

Theorem 8.1 (Algorithm RSC)**.** *For every constant $\kappa > 0$, Algorithm RSC (Algorithm 8.2) runs in time $O^*(\varrho^n)$, where $\varrho = 1.6990\ldots < 1.6991$, and in polynomial space, on every connected graph G with n vertices. It never outputs a value below $\gamma_c(G)$, and it outputs $\gamma_c(G)$ with probability at least $1 - e^{-\kappa}$. Consequently* Connected Dominating Set*, and* Max Leaf Spanning Tree *(by Lemma 5.1, and directly when $n \leq 2$), can be solved by a randomized algorithm with one-sided error (one-sided Monte Carlo) in time $O(1.6991^n)$ and polynomial space.*

The proof of the value of ϱ rests on Proposition D.5.23. The base is $\varrho = (1+y)/\hat{e}$ for the rationals $y = 31/100$ and $\hat{e} = 145729958/189014575$ of the certificate, so the running time is $2^{\psi n} n^{O(1)}$ with $\psi = \log \varrho = 0.76476681\ldots$.

Route of the proof. Section 8.1 describes the algorithm, and Section 8.2 shows that Theorem 3.14 reduces Theorem 8.1 to a lower bound $W_S^{\mathrm{cl}}(y) \geq 0.6060\,\hat{e}^n$ on a generating function W_S^{cl} of the closure trials (Definition 8.3), for every *minimum* connected dominating set S. We prove this lower bound (Proposition 8.6) by extending the analysis of Section 7 in three steps.

(1) *Tree model* (Appendix D.5.1). Lemma D.5.2 replaces the graph by a spanning tree F of $G[S]$, keeping for every vertex outside S a factor h_u that records whether it is dominated and whether it was a candidate, and keeping of the closure only what F certifies. Every outside vertex is assigned to an owner in S (Definition D.5.3), and Lemma D.5.4 compares h_u with a factor of the owner; the comparison depends on the *type* of u, that is, on whether u has two or more neighbors in S. Most factors h_u are then bounded below by owner factors times an *exposure*, a factor $q_\tau = 1 - p_\tau$ whenever the owner is in the closed sample, together with a *credit* $\varpi_\tau \geq 1$ on the factor of an owner outside the closed sample, which records that an exposed vertex is usually dominated by one of its other neighbors (Lemma D.5.13). For the other shared vertices, Jensen's inequality gives a correction term instead (Lemma D.5.6), which the pointwise comparison bounds from below.

(2) *Light owners* (Appendices D.5.2 and D.5.3). The tree model is evaluated by a ten-state recursion along F (Lemma D.5.8). Inequalities between the coordinates of its messages (Lemma D.5.9) show that the Jensen correction of an outside vertex is nonnegative when its owner has few owned vertices and small degree in F (Lemma D.5.11), and they bound the odds that a vertex of S is sampled from below (Lemma D.5.12), which is what the credits rest on. The other owners are exposed; minimality of S limits the exposures at the leaves of F (Lemma D.5.14).

(3) *Certificate* (Appendix D.5.4). An antinorm in an eight-dimensional poset cone, in the style of Lemma 7.5, bounds a relaxation of the recursion from below for every tree that can arise (Lemma D.5.20); the relaxation (Lemma D.5.18) forgets whether a vertex has more than two sampled children except where the eight states can tell, and it never overestimates the tree model. The certificate consists of 1024 rational points and staged conditions for exposed vertices, and Proposition D.5.23 states the conditions it satisfies.

The three steps are carried out in Appendix D.5, and Section 8.3 records how tight the analysis is and how the parameters were chosen.

Proof map. Table 4 lists the ingredients of Theorem 8.1, with the kind of argument behind each and the machine checks that exist for it. Apart from the Lean kernel, the bound therefore rests

Ingredient	Statements	Argument	Machine checks
Steiner extension in time $2^{\lvert \mathrm{T}(J_X)\rvert} n^{O(1)}$ and polynomial space	Lemmas 5.3 and 5.6 and Corollary 5.8	inclusion–exclusion over branching walks [Ned13]; proofs in Appendix D.3	Lean: the computed value, not running time or space
Budgeted trials: running time, rejection, success probability	Theorem 3.14, Lemma 8.4, and Corollary 8.5	mathematical proof	Lean: rejection and success probability, not running time
Graph to tree model, owners, exposures, credits, ten-state recursion, light owners	Lemma D.5.2 to Corollary D.5.17	mathematical proof	Lean
Finitely many conditions imply the bound for every tree	Lemmas D.5.18 to D.5.20	invariant region and induction from the leaves	Lean
Light pairs (condition (C3)) and credit inequalities	Lemmas D.5.21 and D.5.22	rational bounds on logarithms	exact program; Lean
Certificate data: (C1), (C2), (C4), (C5), (C6) and $c_Z > 0.6060$	Proposition D.5.23	1024 rational points and about 3.7 million relations, each checked exactly	two independently written programs, exact rational arithmetic; Lean with compiled evaluation
Base $\varrho < 1.6991$	proof of Theorem 8.1	comparison of rationals	Lean
Max Leaf Spanning Tree from Connected Dominating Set	Lemma 5.1	mathematical proof	Lean

Table 4: Proof map of Theorem 8.1. "Lean" refers to the statements listed in Appendix F.9; the programs and their running times are described in Appendix F.4.

on the mathematical proofs of the running time and space bounds, including those of the Steiner extension, which are not formalized, and on the two programs that check the data of the certificate; the Lean check of these data (Appendix F.9) is a third check, which also trusts the Lean compiler. The two programs are independent: the second was written from the definitions of this section without reusing code of the first, it re-derives the recursion, and it rejects perturbed copies of the certificate (Appendix F.4).

Throughout this section $0 < p_2 \le p_3 < \frac{1}{2}$, $q_2 := 1 - p_2$ and $q_3 := 1 - p_3$, so $q_2 \ge q_3 > \frac{1}{2}$; for an integer $k \ge 2$ we write $p_k := p_2$ if $k = 2$ and $p_k := p_3$ if $k \ge 3$, and $q_k := 1 - p_k$. For a connected graph G with vertex set V and $X' \subseteq V$ the *candidate set* is

$$C(X') = \{v \in V \setminus X' : \ |N(v) \cap X'| \ge 2\}.$$

With $p_2 = p_3$ the closure is the Bernoulli closure of a single probability; the bound of the theorem rests on the graded one (see Remark 8.7).

Esmer and Kulik [EK25] also use problem-specific random steps before a black-box call, for parameterized approximation of vertex deletion problems: each of their sampling steps returns a vertex whose removal lowers the optimum with at least a fixed probability, and the running time of the black-box call is fixed by the algorithm. Here the closure is added to a uniform sample inside Monotone Local Search, its probabilities depend on the sample, and it changes the running-time exponent of the call, which the exponential moments account for.

8.1 The algorithm

How RSC decides. For each candidate size s, RSC runs the trials of RS-θ (Algorithm 7.1), indexed by a sample size $t \in \{0, \dots, s\}$ and an integer threshold θ, with one change, which takes polynomial time per trial: after drawing a uniform t-set X', it puts each candidate $v \in C(X')$ into a set Υ independently with probability $p_{|N(v)\cap X'|}$, that is, with probability $p_2 = 0.07$ if v has exactly two sampled neighbors and $p_3 = 0.077$ if it has more, and it then treats the *closed sample* $X = X' \cup \Upsilon$ as RS-θ treats its sample, also discarding X if $|X| > s$. Fix a solution S. The closure never increases the number of terminals (Lemma D.5.1), and an added candidate with a neighbor outside $N[X']$ lowers it; but the closed sample leaves S when the closure adds a vertex outside S, which happens with probability $1 - \prod_{v \in C(X')\setminus S} q_{|N(v)\cap X'|}$. The small values of p_2 and p_3 keep this probability small; the analysis needs small probabilities, and p_3 only slightly above p_2.

Algorithm 8.2 (RSC)

Input: a connected graph G with $n \ge 1$ vertices.

Parameters: $p_2 = 0.07$, $p_3 = 0.077$, $y = 31/100$ and $\hat{e} = 145729958/189014575$; $\psi = \log \varrho$ with $\varrho = (1+y)/\hat{e}$; a constant $\kappa > 0$ and $\Gamma = 2$; a cap Θ and trial counts $m_{t,\theta}$ within the ranges (2).

For $s = 1, 2, \dots, n$, decide whether to accept s as follows, and output the first accepted s (and n if there is none). For every $t \in \{0, \dots, s\}$ and every integer $\theta \in \{0, \dots, \Theta\}$, repeat $m_{t,\theta}$ times:

(a) draw X' uniformly from $\binom{V}{t}$ and compute $C(X')$;

(b) put each $v \in C(X')$ into Υ independently with probability $p_{|N(v)\cap X'|}$, and let $X = X' \cup \Upsilon$;

(c) if $|X| \le s$ and $|\mathrm{T}(J_X)| = n - |N[X]| + \mathrm{comp}(G[X]) \le \theta$, compute $\mathrm{OPT}(J_X)$ by Corollary 5.8, and accept s if $\mathrm{OPT}(J_X) \le s - |X|$.

The trial counts $m_{t,\theta}$ should not be confused with the numbers ν_v of owned vertices used in the analysis.

8.2 Applying the budgeted-trials theorem

The trials of RSC are not those of Algorithm MLS-θ: the extension algorithm is called on the closed sample $X' \cup \Upsilon$, whose distribution is not uniform on sets of a given size. The budgeted form of the

main theorem (Theorem 3.14) needs only that a trial of threshold θ runs in time $2^\theta n^{O(1)}$, rejects if no solution of size at most k exists, and accepts whenever a random variable attached to it is at most θ. The exponential moment in its success condition is then a generating function that averages over the closure.

Definition 8.3 (Closure generating function)**.** Let G be a connected graph, S a connected dominating set of G, $0 \le p_2 \le p_3 \le 1$ and $y \ge 0$. For $X' \subseteq V$ let Υ be the random subset of $C(X')$ that contains each candidate v independently with probability $p_{|N(v)\cap X'|}$. Put

$$W_S^{\mathrm{cl}}(y) = \sum_{X' \subseteq S} y^{|X'|}\, \mathbb{E}_\Upsilon\Big[\mathbf{1}[\Upsilon \subseteq S]\, 2^{-|\mathrm{T}(J_{X'\cup\Upsilon})|}\Big].$$

For $p_2 = p_3 = 0$ the set Υ is empty and W_S^{cl} is the generating function W_S of (15).

Lemma 8.4 (The trials of RSC)**.** *Let G be a connected graph with n vertices and let $1 \le s \le n$. For $t \in \{0,\dots,s\}$ and an integer $\theta \ge 0$, let $\mathrm{Tr}_{t,\theta}$ be one execution of steps (a)–(c) of Algorithm 8.2 for s, with any rational $0 \le p_2 \le p_3 \le 1$.*

(a) $\mathrm{Tr}_{t,\theta}$ runs in time $2^\theta r_0(n)$ and space $r_0(n)$ for a polynomial r_0 that depends only on p_2 and p_3, not on G, s, t or θ, and it rejects if $\gamma_{\mathrm{c}}(G) > s$.

(b) Let S be a connected dominating set of G with $|S| = s$. Let $\xi_t = |\mathrm{T}(J_X)|$ if $X \subseteq S$ and $\xi_t = \infty$ otherwise, where $X = X' \cup \Upsilon$ is the closed sample of the trial. Then the distribution of ξ_t does not depend on θ, $\Pr[\mathrm{Tr}_{t,\theta}$ accepts$] \ge \Pr[\xi_t \le \theta]$ for every θ, and $W_S^{\mathrm{cl}}(y) \le (1+y)^n \max_{0\le t\le s} \mathbb{E}[2^{-\xi_t}]$ for every $y \ge 0$.

Proof. (a) Steps (a) and (b) and the test in (c) take polynomial time, by the convention on random draws in Section 3.1, since p_2 and p_3 are rational. The Steiner extension is called only if $|\mathrm{T}(J_X)| \le \theta$, and then it takes time $2^{|\mathrm{T}(J_X)|} n^{O(1)} \le 2^\theta n^{O(1)}$ and polynomial space by Corollary 5.8, where the exponent of n is a fixed constant. Together these give one polynomial r_0 as stated. If the trial accepts, then some $Y \subseteq V \setminus X$ with $|Y| \le s - |X|$ makes $X \cup Y$ a connected dominating set, by the description of the solutions of J_X in Section 5; this set has at most s vertices.

(b) The trial's random choices do not involve θ. Suppose that $\xi_t \le \theta$. Then $X \subseteq S$, so $|X| \le s$ and $|\mathrm{T}(J_X)| \le \theta$; the trial computes $\mathrm{OPT}(J_X)$, and since $X \cup (S \setminus X) = S$ is a connected dominating set, $S \setminus X$ is a solution of J_X and $\mathrm{OPT}(J_X) \le |S \setminus X| = s - |X|$. So the trial accepts. For the last inequality, $X \subseteq S$ holds if and only if $X' \subseteq S$ and $\Upsilon \subseteq S$, and X' is uniform in $\binom{V}{t}$, so

$$\mathbb{E}\big[2^{-\xi_t}\big] = \binom{n}{t}^{-1} \sum_{X' \in \binom{V}{t}} \mathbf{1}[X' \subseteq S]\, \mathbb{E}_\Upsilon\Big[\mathbf{1}[\Upsilon \subseteq S]\, 2^{-|\mathrm{T}(J_{X'\cup\Upsilon})|}\Big].$$

Grouping the terms of Definition 8.3 by $t = |X'|$ (only $t \le s$ occur) and using $\sum_{t=0}^{n} \binom{n}{t} y^t = (1+y)^n$ gives $W_S^{\mathrm{cl}}(y) = \sum_{t=0}^{s} \binom{n}{t} \mathbb{E}[2^{-\xi_t}]\, y^t \le (1+y)^n \max_t \mathbb{E}[2^{-\xi_t}]$. □

The following corollary of Theorem 3.14 and Lemma 5.2 packages the wrapper around budgeted trials for connected dominating sets.

Corollary 8.5 (Budgeted trials for connected domination)**.** *Let $\psi \ge 0$ be a real constant that can be approximated to within 2^{-b} in time $b^{O(1)}$ for every integer $b \ge 1$, let $\Gamma_n \ge 1$ be computable in time $n^{O(1)}$ with $\Gamma_n \le n^{O(1)}$, and let $\kappa > 0$ be a constant. Suppose that there are a randomized algorithm Tr with inputs (G, s, t, θ), written $\mathrm{Tr}_{t,\theta}$ when G and s are fixed, and a polynomial r_0 such that for every connected graph G with n vertices, every $s \in \{1,\dots,n\}$, every $t \in \{0,\dots,s\}$ and every integer $\theta \ge 0$:*

(T1) $\mathrm{Tr}_{t,\theta}$ *runs in time* $2^{\theta} r_0(n)$ *and space* $r_0(n)$,

(T2) $\mathrm{Tr}_{t,\theta}$ *rejects if* $\gamma_{\mathrm{c}}(G) > s$, *and*

(T3) *if* $s = \gamma_{\mathrm{c}}(G)$, *then there are random variables* ξ_t *with values in* $[0,\infty]$, *whose distributions do not depend on* θ, *such that* $\Pr[\mathrm{Tr}_{t,\theta}\ \textit{accepts}] \geq \Pr[\xi_t \leq \theta]$ *for all* t *and* θ, *and* $\max_{0\leq t\leq s} \mathbb{E}[2^{-\xi_t}] \geq 2^{-\psi n}/\Gamma_n$.

Then the algorithm that, for $s = 1, \ldots, n$ *in turn, runs Algorithm 3.13 on* (G, s) *with the index set* $\{0, \ldots, s\}$, *these trials, rate* ψ, *loss factor* Γ_n, *confidence* κ *and a cap and trial counts within the ranges* (2), *and outputs the first accepted* s, *or* n *if none, runs in time* $2^{\psi n} n^{O(1)}$ *and polynomial space, never outputs a value below* $\gamma_{\mathrm{c}}(G)$, *outputs* $\gamma_{\mathrm{c}}(G)$ *with probability at least* $1 - e^{-\kappa}$, *and, when* $n \geq 3$, *solves* Max Leaf Spanning Tree *within the same bounds.*

Proof. Fix s. The cap and the trial counts are computed in polynomial time: the number $x = \psi n + \log \Gamma_n$ of (2) is approximated to within $\frac{1}{2}$, and $M_{t,\theta} = 4\kappa\Gamma_n(\Theta + 1)2^{\psi n - \theta}$ to within a factor 2, by rationals obtained from an approximation of ψ and from Γ_n, which suffices, as noted after (2); the values of ψ, Γ_n and κ are at most $n^{O(1)}$. Extend Tr to all other inputs, that is, to instances that do not encode a connected graph and to values of s or t outside the ranges above, by rejecting at once after a test that takes time $r_1(N)$ for a polynomial r_1. We may take r_0 nondecreasing; as $n \leq N$, conditions (T1) and (T2) of Theorem 3.14 then hold for every instance and every $k \leq n$, with the polynomial $r_0 + r_1$ and space bound $\mathrm{sp} = 0$. By Theorem 3.14 for the implicit set system Φ_{CDS} of Definition 5.7, with $k = s$, the index sets $\mathcal{T}_{I,k} = \{0, \ldots, k\}$ (empty for $k < 0$), which are listed in time $O(N)$, the run for s takes time $2^{\psi n} n^{O(1)}$ and polynomial space by (T1), rejects if $\gamma_{\mathrm{c}}(G) > s$ by (T2), and, when $s = \gamma_{\mathrm{c}}(G)$, accepts with probability at least $1 - e^{-\kappa}$ by (T3), with a minimum connected dominating set as the set S of size k in Theorem 3.14. Lemma 5.2, with success probability $1 - e^{-\kappa}$, gives the rest, the total time being $n \cdot 2^{\psi n} n^{O(1)}$. □

The analysis uses the solution S only through W_S^{cl}, and it needs S to be a minimum connected dominating set. The proof of the following proposition, given at the end of Appendix D.5.4 in Appendix D.5, uses the certificate of Proposition D.5.23.

Proposition 8.6 (Closure generating function of a minimum solution)**.** *Let* $p_2 = 0.07$, $p_3 = 0.077$, $y = 31/100$ *and* $\hat{e} = 145729958/189014575$. *For every connected graph* G *with* n *vertices and every connected dominating set* S *of* G *with* $|S| = \gamma_{\mathrm{c}}(G)$,

$$W_S^{\mathrm{cl}}(y) \;\geq\; 0.6060\, \hat{e}^{\,n}.$$

Proof of Theorem 8.1, given Proposition 8.6. The steps of Algorithm 8.2 for s are Algorithm 3.13 with the trials $\mathrm{Tr}_{t,\theta}$ of Lemma 8.4, rate $\psi = \log \varrho$, loss factor $\Gamma_n = 2$ and confidence κ, so Theorem 8.1 follows from Corollary 8.5, with $n \leq 2$ handled directly; the rate ψ, the logarithm of a rational number, can be approximated to any precision in polynomial time. Properties (T1) and (T2) are Lemma 8.4(a). For (T3), let $s = \gamma_{\mathrm{c}}(G)$, let S be a minimum connected dominating set, and take the variables ξ_t of Lemma 8.4(b); their distributions do not depend on θ and $\Pr[\mathrm{Tr}_{t,\theta}\ \text{accepts}] \geq \Pr[\xi_t \leq \theta]$, and by Lemma 8.4(b) and Proposition 8.6,

$$\max_{0\leq t\leq s} \mathbb{E}\big[2^{-\xi_t}\big] \;\geq\; \frac{W_S^{\mathrm{cl}}(y)}{(1+y)^n} \;\geq\; 0.6060\Big(\frac{\hat{e}}{1+y}\Big)^n = 0.6060 \cdot 2^{-\psi n} \;\geq\; \frac{2^{-\psi n}}{2}.$$

So RSC runs in time $2^{\psi n} n^{O(1)} = O^*(\varrho^n)$ and polynomial space, never outputs a value below $\gamma_{\mathrm{c}}(G)$, outputs $\gamma_{\mathrm{c}}(G)$ with probability at least $1 - e^{-\kappa}$, and, for $n \geq 3$ by Corollary 8.5, solves Max Leaf

Spanning Tree within the same bounds; for $n \le 2$ the maximum leaf spanning tree is G itself (Lemma 5.1). Finally $\varrho = (1+y)/\hat{e} = 1.6990\ldots < 1.6991$, by exact evaluation of the rational number $\frac{131}{100} \cdot \frac{189014575}{145729958} = \frac{990436373}{582919832}$, so $O^*(\varrho^n) \subseteq O(1.6991^n)$. □

8.3 Tightness of the analysis

The following remark is a numerical observation; it is not used in the proof of Theorem 8.1.

Remark 8.7 (How tight the analysis is). On the uniform $(2,2)$-lobster Λ_L of Proposition 7.9, the generating function W_S^{cl} of the unique minimum connected dominating set can be evaluated by a transfer matrix. With $(p_2, p_3) = (0.07, 0.077)$, the running-time bound $\inf_{y>0}(1+y)^n / W_S^{\mathrm{cl}}(y)$ that Lemma 8.4 and Theorem 3.14 give is about 1.6976^n (numerical, floating point), so the certified running time $O(1.6991^n)$ is within a factor $2^{0.0013n}$ of the best that this route can give for this algorithm. The grading is worth about 0.0008 in the base: with $p_2 = p_3 = 0.07$ an antinorm certificate of the same kind certifies about 1.6999^n.

The values of p_2, p_3 and y come from a numerical exploration, which the following remark records; as with Remark 8.7, nothing in it is used in a proof.

Remark 8.8 (Choice of the closure probabilities). Without closure ($p_2 = p_3 = 0$), Algorithm RSC is RS-θ, which needs time $\Omega(1.7192^n)$ on the uniform lobsters (Proposition 7.9). A candidate with k sampled neighbors in different components of the sample can merge up to k components into one and lower the number of terminals by up to $k-1$ if it lies in S, whereas the trial is lost if it lies outside S, whatever k is; this is why a larger probability for candidates with three or more sampled neighbors can pay. The analysis, however, charges a factor $q_\tau = 1 - p_\tau$ for every exposed outside vertex, and the light-owner tests of Definition D.5.10 hold only for small probabilities. Numerically, the smallest base that the exact tree model of Appendix D.5 can certify, over (p_2, p_3, y), is attained near $(0.07, 0.077, 0.31)$, where it is about 1.6983; the eight-state relaxation cannot go below about 1.6985 there, and the certificate gives 1.69910. The window for p_3 is narrow: at $(p_2, p_3) = (0.07, 0.15)$, owners of two or more outside vertices fail the test for type-3 vertices and the limit of the tree model rises to about 1.749.

The decimal 1.6991 needs no numerical slack: $\varrho = 990436373/582919832 = 1.699095\ldots$ is an exact rational, and every condition of the certificate is checked exactly, so the rounding up to four decimals is the only step between ϱ and the stated base.

9 Derandomization

The algorithms of Theorems 3.5 and 4.1 are randomized. Classical Monotone Local Search is derandomized by [FGLS19] at the price of a factor $2^{o(n)}$ in the running time. For sample-dependent running-time exponents the answer has two parts. If the algorithm may use the running-time exponent and the extension algorithm only as oracles, no deterministic algorithm matches the mean-exponent theorem in general: when the mean bound, at least 1, is below $\log \binom{k}{t}$ by a linear amount, where t is the sample size, its cost exceeds that of the randomized algorithm by an exponential factor unless it reaches the cost $\binom{n}{k}$ of testing every k-set (Theorems D.6.1 and D.6.3 and Corollary D.6.2), and for the exponential-moment theorem the lower bound can come close to $\binom{n}{k}$ (Theorem 9.4); for the mean bounds of Connected Dominating Set the gap is $1.7547^n / \operatorname{poly}(n)$ against $\sqrt{3}^n \operatorname{poly}(n)$ (Corollary 9.3). If, however, the mean bound comes from a pointwise affine majorant $g(X) \le \alpha - w(X)$, where $g(X) = g(I, X, k)$ is the shorthand declared after Definition 3.2 and w are unknown nonnegative weights of known total, then balanced set-inclusion families, which we list deterministically in polynomial space (Theorem 9.6), derandomize, for integer weights and

with a factor $2^{O(n(\log\log n)^2/\log n)}$, the exponential-moment bound of the affine *surrogate* exponent $\alpha - w(X)$ (Theorem 9.10). This is not a derandomization of Theorem 3.5 for the true running-time exponent g. Section 10 derives the consequences for Connected Dominating Set.

Throughout we use the setting of Definitions 3.1 to 3.3. An implicit set system Φ assigns to an instance I with $N = |I|$ a universe U_I with $n = |U_I| \le N$ and a family $\mathcal{F}_I$ of solutions; an extension algorithm $\mathcal{A}$ decides, given (I, X, k), whether some $Y \subseteq U_I \setminus X$ with $|Y| \le k - |X|$ has $X \cup Y \in \mathcal{F}_I$, in time $2^{g(I,X,k)}N^{O(1)}$, where the running-time exponent $g \ge 0$ is computable in polynomial time. For a solution S with $|S| = k$ we write $\mu_t(S) = \sum_{X \in \binom{S}{t}} 2^{-g(I,X,k)}$, and H is the binary entropy, with the estimates (1).

9.1 Set-inclusion families and what changes with sample-dependent running times

Classical Monotone Local Search samples a uniform t-subset X of the universe and calls the extension algorithm on it. The running-time bound $c^{k-t}N^{O(1)}$ of the extension call depends only on $|X|$, so a sample X of size t can fail only by not being contained in the solution. [FGLS19] replace the random samples by a deterministic family that contains a t-subset of every k-set.

Definition 9.1 (Set-inclusion family [FGLS19])**.** Let $|U| = n$ and $0 \le t \le k \le n$. A family $\mathcal{J} \subseteq \binom{U}{t}$ is an (n, k, t)*-set-inclusion family* if every $S \in \binom{U}{k}$ contains some member of $\mathcal{J}$.

Every member of $\binom{U}{t}$ lies in $\binom{n-t}{k-t}$ sets of $\binom{U}{k}$, so double counting gives $|\mathcal{J}| \ge \binom{n}{k}/\binom{n-t}{k-t} = \binom{n}{t}/\binom{k}{t}$. [FGLS19, Section 3] construct families of size $\binom{n}{t}\binom{k}{t}^{-1}2^{o(n)}$ in time of the same order: they hash the universe into about $\log n$ buckets with a pairwise independent family of functions, build a near-optimal family on each bucket by greedy Set Cover, and combine the bucket families. The bucket families have size $2^{O(n/\log n)}$ and are stored, and the paper notes that the deterministic algorithm needs an additional $2^{o(n)}$ space. Looping over such a family instead of sampling gives a deterministic $(2 - \frac{1}{c})^{n+o(n)}N^{O(1)}$ algorithm from a $c^k N^{O(1)}$ extension algorithm.

A sample-dependent running-time exponent changes the picture. The family must now contain, inside every solution, a sample of small running-time exponent, and the running-time exponents of the subsets of the solution are visible only through evaluations of g. An adversary can give large running-time exponents to exactly the evaluated subsets; this planting argument gives the lower bounds of Section 9.2. For running-time exponents bounded by an affine function of unknown nonnegative weights, a family with a stronger property suffices (Section 9.3): it contains, inside every solution, a sample of at least average weight. This *balance* is strictly stronger than inclusion: $\{\{1\}, \{2\}, \{3\}\}$ is a $(4, 2, 1)$-set-inclusion family, but for $S = \{1, 4\}$ and a weight concentrated on the element 4, no member inside S has positive weight.

9.2 The black-box barrier

The randomized algorithms of Theorems 3.5 and 4.1 use the instance only through evaluations of g and calls of $\mathcal{A}$. The barrier concerns deterministic algorithms with the same access, and with membership tests added.

Definition 9.2 (Black-box model)**.** An *instance* is a pair $(\mathcal{F}, g)$ with $\mathcal{F} \subseteq 2^U$, $|U| = n$, and $g\colon 2^U \times \mathbb{N} \to \mathbb{R}_{\ge 0}$. The algorithm knows n, k and the parameters of the hypothesis (t and λ, $\lambda_0, \ldots, \lambda_k$, or ψ and Γ, below). It may call

- $\mathrm{COST}(X, \ell)$, for a set $X \subseteq U$ and an integer $\ell \ge 0$, which returns $g(X, \ell)$, at cost 1;

- EXT(X, ℓ), which returns whether some $Y \subseteq U \setminus X$ with $|Y| \le \ell - |X|$ has $X \cup Y \in \mathcal{F}$, at cost $2^{g(X,\ell)}$;
- MEM(X), which returns whether $X \in \mathcal{F}$, at cost 1.

Computation is free. A deterministic algorithm is *correct for a class* $\mathcal{C}$ of instances if it answers NO if $\mathcal{F}$ contains no set of size at most k, and it answers YES on every instance of $\mathcal{C}$ in which $\mathcal{F}$ contains a set of size exactly k. Its *cost* on an instance is the total cost of its calls. With $\mathbb{E}_{X \in \binom{S}{t}}$ the mean over $X \in \binom{S}{t}$, the classes are:

- $\mathcal{C}_{\text{mean}}(t, \lambda)$: every $S \in \mathcal{F}$ with $|S| = k$ satisfies $\mathbb{E}_{X \in \binom{S}{t}} g(X, k) \le \lambda$;
- $\mathcal{C}^{\text{all}}_{\text{mean}}(\lambda_0, \dots, \lambda_k)$: every $S \in \mathcal{F}$ with $|S| = k$ satisfies $\mathbb{E}_{X \in \binom{S}{j}} g(X, k) \le \lambda_j$ for every $j \in \{0, \dots, k\}$;
- $\mathcal{C}_{\exp}(\psi, \Gamma)$: some $S \in \mathcal{F}$ with $|S| = k$ has $\max_j \mu_j(S)/\binom{n}{j} \ge 2^{-\psi n}/\Gamma$, where $\mu_j(S) = \sum_{X \in \binom{S}{j}} 2^{-g(X,k)}$.

The model is generous to the algorithm: computation is free, every evaluation of g costs one step, and $\mathcal{F}$ need not be monotone. The randomized algorithms of Theorem 4.1 (on $\mathcal{C}_{\text{mean}}(t, \lambda)$) and Theorem 3.5 (on $\mathcal{C}_{\exp}(\psi, \Gamma)$) run in this model within their stated bounds, since they only evaluate g and call the extension algorithm. Write

$$\mathsf{R}_t = \binom{n}{t}\binom{k}{t}^{-1} 2^{\lambda}$$

for the running-time bound of the mean-exponent theorem at sample size t, without its polynomial factor; Theorem 4.1 decides the class $\mathcal{C}_{\text{mean}}(t, \lambda)$ with one-sided error in time R_t up to a factor polynomial in N and λ (the Markov-cutoff algorithm of Remark 4.2, with the exact threshold $\lambda+1$, runs $\lceil (\lambda+1)\binom{n}{t}\binom{k}{t}^{-1} \rceil$ trials of cost at most $1+2^{\lambda+1}$ each, a total cost of at most $6(\lambda+1)\mathsf{R}_t$).

The idea. All lower bounds use one adversary, a form of the planting argument of the oracle lower bounds of [EKM+22, EKM+24], where a solution is hidden among the sets that a run leaves unqueried. It runs a deterministic algorithm on a no-instance, $\mathcal{F} = \emptyset$, whose running-time exponents depend only on the size of the sample. Call a set *touched* if the run evaluates its running-time exponent or calls the extension algorithm on it, and call a k-set *clean* if the run neither tests its membership nor calls the extension algorithm on one of its subsets. A clean k-set can be planted as the only solution, with running-time exponent 0 on its untouched subsets, without changing any answer of the run, so the algorithm still answers NO. A cheap run leaves a random k-set clean with few touched subsets, and then the planted instance satisfies the hypothesis of the class, which contradicts correctness. The proofs are in Appendix D.6.

Mean exponents. The planting argument gives lower bounds for the hypothesis of Theorem 4.1 at one sample size and for mean bounds $\lambda_0, \dots, \lambda_k$ at all sample sizes; in both cases a deterministic black-box algorithm has cost about $\binom{n}{t}$ instead of R_t, up to the cap $\binom{n}{k}$ of testing every k-set, so the gap to the randomized algorithm is exponential whenever $\log \binom{k}{t}$ exceeds the mean bound by a linear amount. The statements are in Appendix D.6.1 (Theorems D.6.1 and D.6.3 and Corollary D.6.2). We evaluate Theorem D.6.3 on the mean bounds $\lambda_0, \dots, \lambda_k$ that CONNECTED DOMINATING SET

provides. By the mean number of terminals (Lemma 5.11), a uniform j-subset of a connected dominating set of size k leaves on average at most $n-1-j(n-2)/k$ terminals; with the empty sample, which leaves all n vertices as terminals, this gives the bounds $\lambda_0 = n$ and $\lambda_j = n - 1 - j(n-2)/k$ for $1 \le j \le k$.

Corollary 9.3 (The mean bounds of Connected Dominating Set)**.** *Let n be even, $k = n/2$, $\lambda_0 = n$ and $\lambda_j = n - 1 - j(n-2)/k$ for $1 \le j \le k$. Every deterministic black-box algorithm that is correct for $\mathcal{C}^{\mathrm{all}}_{\mathrm{mean}}(\lambda_0, \dots, \lambda_k)$ has cost at least*

$$\frac{2^{H(1/4)n}}{6(n+1)(n+2)} = \frac{(4 \cdot 3^{-3/4})^n}{6(n+1)(n+2)} \ge \frac{1.7547^n}{6(n+1)(n+2)}$$

on some no-instance, while Algorithm MLS-θ *with the parameters of Corollary 4.3(1), which calls only* COST *and* EXT *and needs only the mean bounds $\lambda_0, \dots, \lambda_k$ as parameters, decides the same class with one-sided error at cost $\sqrt{3}^n n^{O(1)}$.*

Corollary 9.3 is a statement about the abstract class, not about Connected Dominating Set: over all solution sizes, the randomized algorithm of Theorem 6.4 runs in time $O(1.7449^n)$, and Section 10 solves Connected Dominating Set deterministically in time $O(1.7361^n)$ (Theorem 10.4), by using the affine structure of the running-time exponent and not only its values.

Exponential moments. For the exponential-moment theorem the lower bound is much larger: the cost of a deterministic black-box algorithm can come close to the cost $\binom{n}{k}$ of testing every k-set with MEM.

Theorem 9.4 (Black-box lower bound for exponential moments)**.** *Let $\psi \ge 0$ and $\Gamma \ge 1$, and let $\mathcal{H}$ be a deterministic black-box algorithm that is correct for $\mathcal{C}_{\exp}(\psi, \Gamma)$. Call a level $t \in \{0, \dots, k\}$* feasible *if $J_t := \lceil \binom{n}{t} 2^{-\psi n} / \Gamma \rceil \le \binom{k}{t}/2$. Then for every feasible t, the cost of $\mathcal{H}$ on the no-instance $\mathcal{F} = \emptyset$, $g \equiv n$, is at least $\frac{1}{3} \min\{\binom{n}{k}, \binom{n}{t}\}$.*

The strongest bound comes from the largest feasible t; for Algorithm RS-θ at the solution fraction $k/n \approx 0.4173$, where its running-time bound is attained (Table 3), it is numerically about 1.9440^n (Appendix F), against $O(1.7204^n)$ for the randomized algorithm; this value is not used later.

We conjecture that the same planting argument, combined with Yao's principle, shows that Theorem 4.1 is optimal up to polynomial factors also for *randomized* black-box algorithms, in analogy with the oracle lower bounds of [EKM+24, EKM+26].

9.3 Balanced set-inclusion families

The structured derandomization replaces random samples by a family that contains, inside every solution, a sample of at least average weight, for every nonnegative weighting at once. We allow several classes of solution elements, because Theorem 9.10 needs two.

Definition 9.5 (Balanced family)**.** Let $|U| = n$ and $\mathsf{r} \ge 1$, let $k_1, \dots, k_{\mathsf{r}} \ge 0$ be integers with $\sum_j k_j \le n$, and let t_j be integers with $0 \le t_j \le k_j$; put $t = \sum_j t_j$. A family $\mathcal{B} \subseteq \binom{U}{t}$ is $(n; k; t)$-*balanced* if for all pairwise disjoint $S_1, \dots, S_{\mathsf{r}} \subseteq U$ with $|S_j| = k_j$ and every $w\colon \bigcup_j S_j \to \mathbb{R}_{\ge 0}$, some $X \in \mathcal{B}$ satisfies $X \subseteq \bigcup_j S_j$, $|X \cap S_j| = t_j$ for all j, and

$$w(X \cap S_j) \;\ge\; \frac{t_j}{k_j}\, w(S_j) \qquad \text{for every } j \text{ with } k_j \ge 1.$$

For $\mathsf{r} = 1$ we say (n, k, t)-*balanced*.

An (n,k,t)-balanced family is in particular an (n,k,t)-set-inclusion family (Definition 9.1).

The construction partitions the universe into blocks of size about $\log n/\log\log n$, guesses how the unknown solution classes meet the blocks, distributes the targets proportionally, and covers each block greedily by subsets that capture their proportional share of every weighting; the details and the proof of the following theorem are in Appendix D.7.

Theorem 9.6 (Balanced families in polynomial space)**.** *Fix* $\mathsf{r} \geq 1$. *Let* $n \geq 16$ *and* $\iota = \max(2, \lfloor \log n/\log\log n \rfloor)$. *A deterministic algorithm, given* $k_1,\dots,k_{\mathsf{r}}$ *and* $t_1,\dots,t_{\mathsf{r}}$ *as in Definition 9.5, lists (possibly with repetitions) the members of an* $(n;k;t)$*-balanced family* $\mathcal{B}$ *with*

$$|\mathcal{B}| \;\leq\; \iota^{12\mathsf{r}(n/\iota+1)}\,\frac{\binom{n}{t}}{\prod_j \binom{k_j}{t_j}} \;=\; \frac{\binom{n}{t}}{\prod_j \binom{k_j}{t_j}}\,2^{O(\mathsf{r}n(\log\log n)^2/\log n)}.$$

It runs in time $n^{O(1)}$ *times this bound, plus* $n^{O(1)}$*, and in space* $n^{O(1)}$.

For $n < 16$ the family $\binom{U}{t}$ is balanced (for each class take the t_j elements of S_j of largest weight) and has at most 2^{15} members, so the theorem holds for all n after a constant change in the bound. For $\mathsf{r} = 1$ the size is optimal up to the factor $2^{O(n(\log\log n)^2/\log n)}$, since every (n,k,t)-set-inclusion family has at least $\binom{n}{t}/\binom{k}{t}$ members (Section 9.1).

9.4 Exponential moments of integer affine certificates

The balanced families apply to running-time exponents with an *affine certificate*: a solution $S \in \mathcal{F}_I$ with $|S| = k$ for which some weights $w\colon S \to \mathbb{R}_{\geq 0}$, which may depend on S and need not be computable, and a constant α satisfy $g(I,X,k) \leq \alpha - w(X)$ for the samples $X \subseteq S$ in question, where only α and a bound Ξ on the total weight $w(S)$ are known to the algorithm. The exponential-moment theorem can exploit such a certificate more strongly than the mean-exponent theorem, because $\mu_t(S)$ sees the whole distribution of $w(X)$ and not only its mean. We first show that if the weights may be arbitrary nonnegative reals, then in the worst case this certifies no smaller running-time bound than the mean-exponent theorem, and then that for integer weights it certifies a smaller one, and that this smaller bound can be achieved deterministically.

Forms of the certificate. Theorem 9.10 below tries every sample size $t \geq 1$ and needs $g(I,X,k) \leq \alpha - w(X)$ for every nonempty $X \subseteq S$. The empty sample is excluded because the algorithm never lists it and because the inequality can fail there: for Connected Dominating Set, $\alpha = n-1$ while the empty sample leaves all n vertices as terminals (Lemma 5.10 and (13)). It needs $w(S) = \Xi$ rather than $w(S) \geq \Xi$. This is no restriction, since lowering integer weights until the total is Ξ keeps it a certificate. And it needs integer weights, because only for them does the exponential-moment bound of the surrogate gain over its mean-exponent bound (Proposition 9.8).

Let S with $|S| = k$ have an affine certificate $g(I,X,k) \leq \alpha - w(X)$ for all nonempty $X \subseteq S$, with $w(S) = \Xi$. The *surrogate exponent* $\alpha - w(X)$ is defined for every $X \subseteq S$, including $X = \emptyset$, and the generating function of Definition 3.3 for the surrogate exponent is

$$W_S(y) = \sum_{X\subseteq S} y^{|X|}2^{-\alpha+w(X)} = 2^{-\alpha}\prod_{u\in S}(1+y2^{w(u)}).$$

Lemma 9.7 finds, for every y at once, the weight profile that minimizes this product, and hence $W_S(y)$.

Lemma 9.7 (Chord lemma)**.** *Let $k \geq 1$, let $\Xi \geq 0$ be an integer, $a = \lfloor \Xi/k \rfloor$, $k_{a+1} = \Xi - ak$ and $k_a = k - k_{a+1}$. Let $v \in \mathbb{R}^k$ have $\sum_u v_u = \Xi$ and no entry in the open interval $(a, a+1)$. Then for every $y > 0$,*

$$\prod_u (1 + y2^{v_u}) \;\geq\; \mathsf{P}^*(y) := (1 + y2^a)^{k_a} (1 + y2^{a+1})^{k_{a+1}}.$$

Without the restriction on the entries, $\prod_u (1 + y2^{v_u}) \geq (1 + y2^{\Xi/k})^k$.

With the chord lemma we can compare the two theorems on the surrogate exponent.

Proposition 9.8 (Real weights give only the mean-exponent bound)**.** *Let S be a k-subset of a universe of size n, with $k \geq 1$, and let α be real and $\Xi \geq 0$. For $w \colon S \to \mathbb{R}_{\geq 0}$ with $w(S) = \Xi$ let $\mu_t^w(S) = \sum_{X \in \binom{S}{t}} 2^{-\alpha + w(X)}$ for $0 \leq t \leq k$, the exponential moments of the surrogate exponent $\alpha - w(X)$, and put $\mathsf{R}(w) = \min_{0 \leq t \leq k} \binom{n}{t} / \mu_t^w(S)$. Let $\bar{w} \equiv \Xi/k$ be the uniform profile. Then $\mathsf{R}(\bar{w}) = \min_{0 \leq t \leq k} \binom{n}{t} \binom{k}{t}^{-1} 2^{\alpha - t\Xi/k}$, and $\mathsf{R}(w) \leq (n+1)\,\mathsf{R}(\bar{w})$ for every such w. If Ξ is an integer and w^* is the profile of Lemma 9.7, then $\mathsf{R}(w) \leq (n+1)\,\mathsf{R}(w^*)$ for every such w with integer values.*

Here $\mathsf{R}(w)$ is the smallest running-time bound $2^{\psi n}$ that Theorem 3.5 with $\Gamma = 1$ certifies for S through the surrogate exponent, and $\mathsf{R}(\bar{w})$ is the running-time bound of Theorem 4.1 with mean bound $\alpha - t\Xi/k$, without its polynomial factor.

So in the worst case over real weights the affine surrogate certifies, through Theorem 3.5, no smaller running-time bound than the mean-exponent theorem. For Connected Dominating Set the weights of Lemma 5.10, the degrees minus one of the vertices of the solution in a spanning tree, are integers, and for the worst integer profile the running-time bound certified through Theorem 3.5 is exponentially smaller than that of the mean-exponent theorem (Section 10). The following algorithm derandomizes the exponential-moment bound for the worst integer profile. It splits the unknown solution into a *light* class (weights at most a) and a *heavy* class (weights at least $a+1$), guesses the sizes and total weights of the two classes, and runs a two-class balanced family for each guess.

Algorithm 9.9 (Integer affine certificates)

Input: (I, k) together with a rational α and an integer $\Xi \geq 0$.

Put $a = \lfloor \Xi/k \rfloor$. If $\alpha < \Xi$, answer NO. Otherwise run the following for every *guess* (k^-, Ξ^-) of integers with $1 \leq k^- \leq k$ and $0 \leq \Xi^- \leq ak^-$, where $k^+ := k - k^-$ and $\Xi^+ := \Xi - \Xi^-$, subject to

$$\Xi^+ \geq (a+1)k^+ \;\text{ if } k^+ \geq 1, \qquad \Xi^+ = 0 \;\text{ if } k^+ = 0.$$

(1) Put $\bar{w}^- = \Xi^-/k^-$ and, if $k^+ \geq 1$, $\bar{w}^+ = \Xi^+/k^+$. (If $k^+ = 0$ the heavy class is omitted, $t^+ = 0$, and $\bar{w}^+$ is not used.)

(2) Choose integers $0 \leq t^- \leq k^-$ and $0 \leq t^+ \leq k^+$ with $t := t^- + t^+ \geq 1$ that minimize $\binom{n}{t} / \left(\binom{k^-}{t^-} \binom{k^+}{t^+} \right) \cdot 2^\theta$, where $\theta := \alpha - t^-\bar{w}^- - t^+\bar{w}^+$, up to a factor 2, by taking a minimizer of approximations of the logarithms of the candidates to within $\frac{1}{4}$.

(3) List the two-class $(n; k^-, k^+; t^-, t^+)$-balanced family of Theorem 9.6 (a one-class family if $k^+ = 0$).

(4) For each listed X, compute $g(I, X, k)$; if $g(I, X, k) \leq \theta$, run $\mathcal{A}$ on (I, X, k) and answer YES if it accepts.

After all guesses, answer NO.

The condition $\Xi^+ = 0$ for $k^+ = 0$ matters: without it, a guess with $k^+ = 0$ and $\Xi^- < \Xi$ would run a one-class family with mean $\bar{w}^- < \Xi/k$, and hence a larger threshold θ. Since every guess is run, this guess, which takes time about 2^n for the parameters of Section 10, would dominate the total. Algorithm 9.9 has the following guarantee.

Theorem 9.10 (Deterministic exponential moments of integer affine certificates)**.** *Let $k \geq 1$, a rational $\alpha = \alpha(I,k)$ and an integer $\Xi = \Xi(I,k)$ with $0 \leq \Xi \leq N^{O(1)}$ be computable in time $N^{O(1)}$. Say that $S \in \mathcal{F}_I$ with $|S| = k$ has an* integer certificate *if some $w\colon S \to \mathbb{Z}_{\geq 0}$ with $w(S) = \Xi$ satisfies*

$$g(I,X,k) \leq \alpha - w(X) \qquad \text{for every nonempty } X \subseteq S.$$

Put $a = \lfloor \Xi/k \rfloor$, $k_{a+1} = \Xi - ak$, $k_a = k - k_{a+1}$ and $\mathsf{P}^*(y) = (1+y2^a)^{k_a}(1+y2^{a+1})^{k_{a+1}}$. Algorithm 9.9 is deterministic, answers NO *if $\mathcal{F}_I$ has no set of size at most k, and answers* YES *if some $S \in \mathcal{F}_I$ with $|S| = k$ has an integer certificate. It runs in time*

$$N^{O(1)}\, 2^{O(n(\log\log n)^2/\log n)}\, \max\Bigl(1,\ \inf_{y>0} \frac{((1+y)^n - 1)\, 2^\alpha}{\mathsf{P}^*(y) - 1}\Bigr)$$

and in space $N^{O(1)}$ plus the space of $\mathcal{A}$.

The maximum with 1 in the time bound covers the early answer NO and the polynomial bookkeeping; the infimum alone can be smaller than 1 when $\alpha < \Xi$. The hypothesis that α is rational makes the test $\alpha < \Xi$ and the comparisons with θ exact; a real α is replaced by a rational $\hat{\alpha}$ with $\alpha \leq \hat{\alpha} \leq \alpha + 1$, which preserves every integer certificate and at most doubles the time.

The proof of Theorem 9.10 is in Appendix D.8.1, with those of Lemma 9.7 and Proposition 9.8. It shows that the guess formed by the light and heavy classes of an integer certificate is tried, that the balanced family for this guess contains a nonempty sample whose weight is at least its proportional share, and it bounds the running time of each guess by expanding the product over a two-valued weight profile and applying Lemma 9.7.

10 Deterministic algorithms in polynomial space

The algorithms of Sections 6 to 8 are randomized. With the balanced set-inclusion families of Section 9, we derandomize the exponential moments of an integer affine surrogate of the number of terminals (Theorem 10.4). The resulting algorithm runs in polynomial space, below the $O^*(1.8619^n)$ of Abu-Khzam, Mouawad and Liedloff [AKML11]. It escapes the black-box barrier of Corollary 9.3 because it uses an affine upper bound on the running-time exponent, the pointwise tree inequality of Lemma 5.10, and not only the values of the exponent. It does not reach the randomized bounds of Sections 7 and 8: their analyses bound the running-time exponent through tests along an unknown spanning tree of the solution, which are not separable over the fixed block partitions of the balanced families.

Setting. All graphs are connected, with n vertices. We use the implicit set system Φ_{CDS} of connected dominating sets, the Steiner extension algorithm $\mathcal{A}_{\mathrm{St}}$ and its running-time exponent (Definition 3.2, with the graph G as the instance I)

$$g(G,X,k) = |\mathrm{T}(J_X)| = n - |N[X]| + \mathrm{comp}(G[X])$$

from Corollary 5.8: $\mathcal{A}_{\mathrm{St}}$ decides whether X extends to a connected dominating set with at most k vertices, deterministically, in time $2^{|\mathrm{T}(J_X)|}n^{O(1)}$ and polynomial space. The running-time exponent

does not depend on k. The certificate is the pointwise form of the tree inequality: if S is a connected dominating set and F a spanning tree of G in which every vertex outside S is a leaf, the integer weights $w(v) = d_F(v) - 1 \geq 0$ on S satisfy

$$w(S) = n - 2 \qquad \text{and} \qquad |\mathrm{T}(J_X)| \leq n - 1 - w(X) \ \text{ for every nonempty } X \subseteq S, \tag{19}$$

by Lemma 5.10 and (13) (for $n \geq 2$). In the language of Section 9, (19) is an integer certificate (Theorem 9.10) with $\alpha = n - 1$ and total weight $\Xi = n - 2$.

The algorithm decides, for $s = 1, \ldots, n$, whether G has a connected dominating set of size s, by enumeration for small and large s and by Algorithm 9.9 in between (Lemma 10.3), and outputs the least s at which it answers YES. By Lemma 5.1, every bound below holds for Max Leaf Spanning Tree as well.

10.1 Two weight classes

For the integer weights of (19), we express the running-time bound of Theorem 9.10 through the following function. For $u \in (0, 1]$ put $a = \lfloor 1/u \rfloor$ and, for $y > 0$,

$$\chi_y(u) = \log(1 + y) + 1 - \big((a + 1)u - 1\big) \log\big(1 + y2^a\big) - (1 - au) \log\big(1 + y2^{a+1}\big).$$

Here u will be s/Ξ, the ratio of the solution size to the total weight Ξ ($\Xi = n - 2$ here, $\Xi = n - 1$ in Section 12.3), and the two coefficients are the fractions (relative to Ξ) of solution vertices of weight a and $a + 1$ in the balanced integer profile of Lemma 9.7. The next lemma bounds $\inf_{y>0} \chi_y(u)$ on the range of u that the algorithm meets, with one explicit y for each range of u.

Lemma 10.1 (Rate certificate)**.** *For every $u \in [0.15, 0.85]$ there is $y \in \{\frac{1+\sqrt{5}}{8}, \frac{1}{2}, \frac{5}{16}, \frac{1}{4}, \frac{27}{128}, \frac{29}{160}\}$ with $\chi_y(u) \leq \log(\sqrt{5} - \frac{1}{2})$.*

Further optimization over y does not appear to lower this bound (Appendix F); the proof does not use this.

With the rate certificate in hand, the integer-affine algorithm gives the deterministic bound.

The running time of the integer-affine algorithm with a certificate of this shape depends only on n, s, α and Ξ. We record it once, for the two parameter pairs that occur in this paper ($\Xi = n - 2$ here, $\Xi = n - 1$ for weak connectivity in Section 12.3).

Lemma 10.2 (Running time of the integer-affine algorithm for tree-type certificates)**.** *Let $n \geq 34$, $0.15n < s < 0.8n$, $\Xi \in \{n - 2, n - 1\}$ and $\alpha = \Xi + 1$. For every implicit set system, extension algorithm and running-time exponent with values in $\{0, \ldots, n\}$, Algorithm 9.9 on $(I, k = s, \alpha, \Xi)$ runs in time*

$$(\sqrt{5} - \tfrac{1}{2})^n \, 2^{O(n(\log\log n)^2/\log n)} \, N^{O(1)}.$$

The next lemma packages one run of the integer-affine algorithm, together with enumeration for the solution sizes that Lemma 10.2 excludes, as a deterministic decision procedure for one solution size; Section 12 uses it for the variants.

Lemma 10.3 (Deterministic decision for one size)**.** *Let Φ be an implicit set system, let $\mathcal{A}$ be a deterministic algorithm that solves Φ-Extension with a running-time exponent g with values in $\{0, \ldots, n\}$, and let $(\alpha, \Xi) \in \{(n - 1, n - 2), (n, n - 1)\}$. For an instance I with a universe of n elements and $1 \leq s \leq n$, let $\mathrm{Det}(I, s)$ be the following procedure: if $n < 34$, or $s \leq 0.15n$, or $s \geq 0.8n$, test every s-subset of U_I for membership in $\mathcal{F}_I$ and answer YES if and only if one of them is a solution; otherwise run Algorithm 9.9 on $(I, k = s, \alpha, \Xi)$ with $\mathcal{A}$ and g. Then* Det *is*

deterministic, runs in time $(\sqrt{5}-\frac{1}{2})^n 2^{O(n(\log\log n)^2/\log n)} N^{O(1)}$ *and in space* $N^{O(1)}$ *plus the space of* $\mathcal{A}$*, answers* NO *if* $\mathcal{F}_I$ *has no solution of size at most* s*, and answers* YES *whenever* $\mathcal{F}_I$ *has a solution* S *of size exactly* s *which, in the case* $n \geq 34$ *and* $0.15n < s < 0.8n$*, has an integer certificate for* (α, Ξ) *in the sense of Theorem 9.10.*

The proofs of Lemmas 10.1 to 10.3 and of the following theorem are in Appendix D.8.2. The theorem runs $\mathrm{Det}(G, s)$ of Lemma 10.3 for $s = 1, \ldots, n$ with $(\alpha, \Xi) = (n-1, n-2)$, for which the weights of (19) are an integer certificate of every minimum connected dominating set.

Theorem 10.4 (Deterministic Connected Dominating Set and MaxLeaf)**.** Connected Dominating Set*, and hence* Max Leaf Spanning Tree *(by Lemma 5.1, and directly when* $n \leq 2$*), can be solved by a deterministic algorithm in time*

$$O\Big((\sqrt{5}-\tfrac{1}{2})^n\, 2^{O(n(\log\log n)^2/\log n)}\Big) = O(1.7361^n)$$

and polynomial space, where $\sqrt{5}-\frac{1}{2} = 1.7360\ldots$.

What Theorem 10.4 derandomizes. By Proposition 9.8, the randomized exponential-moment theorem (Theorem 3.5) applied to the surrogate exponent $\alpha - w(X)$ certifies, in the worst case over integer certificates of total Ξ, the running-time bound $\inf_y (1+y)^n 2^\alpha / \mathsf{P}^*(y)$ up to polynomial factors; whenever the infimum may be taken over y with $\mathsf{P}^*(y) \geq 2$, this agrees with the bound of Theorem 9.10 up to a factor 2. So Theorems 9.10 and 10.4 derandomize the exponential-moment bound of the *integer affine surrogate*, here $n-1-w(X)$, of the number of terminals, up to a factor $2^{o(n)}$ in the running time; they do *not* derandomize Theorem 3.5 for the true running-time exponent, whose exponential moments can be much larger. The surrogate bound 1.7361^n already improves on the mean-exponent analysis of Algorithm RS (1.7449^n), but not on Algorithm RS-θ (1.7204^n, Theorem 7.2), whose tree potential captures correlations that the surrogate discards, nor on RSC (1.6991^n, Theorem 8.1). The factor $2^{o(n)}$ that the balanced families contribute is large for practical n, about 2^{12718} at $n = 1000$ (Appendix D.7).

11 Component-count running times

This section treats problems whose solutions are vertex sets S of a graph G with $G[S]$ connected, and whose extension algorithms run in time exponential in the number of components of the sample. Given a sample X of size t from a solution of size k, with $\mathrm{comp}(G[X])$ the number of components of $G[X]$, the extension algorithm runs in time

$$O^*\big(c^{k-t}\, 2^{\mathrm{comp}(G[X])}\big),$$

with $c = 3$ for Connected Feedback Vertex Set (a Cut & Count algorithm) and $c = 2$ for Graph Motif and Connected Vertex Cover (the algorithms of Björklund, Kaski and Kowalik and of Cygan). A sample of size t can have t components, so a running-time bound that depends on X only through t is the product-form bound $c^{k-t}2^t$, for which multivariate Monotone Local Search (Corollary 4.6 with $(b, c, d) = (1, c, 2)$) gives the base $3 - 2/c \geq 2$ when $c \geq 2$, no better than enumeration. Our improvements come from the fact that a random part of a connected solution has few components once it is a sizeable fraction of the solution.

We use this fact through a lower bound on $W_S(y)$ by a partition function of a spanning tree of the solution (Definition 11.1), which paths minimize up to a factor independent of k (Section 11.1).

Section 11.2 turns this into the running time $O^*(((4c-3)/(2c-1))^n)$ for Algorithm MLS-θ; the applications follow in Sections 11.3 to 11.5.

Throughout, H is the binary entropy of Section 3.1, with the logarithm to base 2, and for a probability distribution $\mathbb{P}$ on a finite set, $H_e(\mathbb{P}) = -\sum_z \mathbb{P}(z) \ln \mathbb{P}(z)$ is its entropy with the natural logarithm. For a graph F and $X \subseteq V(F)$ we write $e(F[X]) = |E(F[X])|$.

11.1 The partition function of a tree

Definition 11.1 (Tree partition function)**.** For a tree F and a real $\zeta \geq 0$ let

$$\mathcal{Z}_F(\zeta) = \sum_{X \subseteq V(F)} \zeta^{|X|}\, 2^{e(F[X])},$$

let $M_\zeta = \left(\begin{smallmatrix} 1 & \zeta \\ 1 & 2\zeta \end{smallmatrix}\right)$, and let $\lambda_{\max}(M_\zeta)$ be its largest eigenvalue, that is,

$$\lambda_{\max}(M_\zeta) = \tfrac{1}{2}\Big(1 + 2\zeta + \sqrt{(1+2\zeta)^2 - 4\zeta}\Big) = \tfrac{1}{2}\Big(1 + 2\zeta + \sqrt{1 + 4\zeta^2}\Big).$$

Both eigenvalues of M_ζ are real and nonnegative, since the discriminant $1 + 4\zeta^2$ is positive and their sum $1 + 2\zeta$ and product ζ are nonnegative. So $\lambda_{\max}(M_\zeta)$ is also the largest absolute value of an eigenvalue of M_ζ, its spectral radius.

The function $\zeta \mapsto \lambda_{\max}(M_\zeta)$ is continuous and increasing on $[0, \infty)$, with $\lambda_{\max}(M_0) = 1$, and $\lambda_{\max}(M_\zeta) > 1$ for $\zeta > 0$. Since $\lambda_{\max}(M_\zeta)$ is the larger root of $x^2 - (1+2\zeta)x + \zeta$, it is at least half the trace: $\lambda_{\max}(M_\zeta) \geq (1 + 2\zeta)/2$. On a path, M_ζ is the transfer matrix of $\mathcal{Z}_F(\zeta)$: the state of a vertex is 1 if it lies in X and 0 otherwise, a vertex in state 1 has weight ζ, and an edge between two vertices in state 1 has weight 2. The first lemma says that no tree on k vertices has a partition function smaller than $\lambda_{\max}(M_\zeta)^{k-1}$, up to a factor depending only on ζ.

The quantity $\mathcal{Z}_F(\zeta)$ is a weighted count of homomorphisms from F to a target with two vertices, 0 and 1, with vertex weights $w_0 = 1$ and $w_1 = \zeta$ and the symmetric edge weights $A = \left(\begin{smallmatrix} 1 & 1 \\ 1 & 2 \end{smallmatrix}\right)$ (a loop of weight 1 at vertex 0, an edge of weight 1 between the two vertices, and a loop of weight 2 at vertex 1), as in the proof below. For graphs without loops or weights, Csikvári and Lin [CL14, Theorem 4.3] proved that every tree T with m vertices has $\hom(T, G) \geq e^{\eta} \lambda^{m-1}$ homomorphisms into a connected graph G, where λ is the largest adjacency eigenvalue of G and $\eta \geq 0$ is the natural-logarithm entropy of the probability vector formed by the squared entries of a unit Perron eigenvector. They proved it by running a stationary Markov chain on G down the tree, and the proof below follows theirs, with the vertex weights absorbed into a symmetric matrix. Whether the path minimizes such counts exactly depends on the target. Sidorenko [Sid94] proved that among the trees with a given number of vertices the star has the most homomorphisms into every graph, whereas for some graphs some trees have fewer homomorphisms than the path [GMNR25]. Galvin, Marmorino, McMillon, Nir and Redlich [GMM$^+$25] give criteria for paths to be the minimizers, for targets without edge weights (with weights on the vertices only); our target carries the weight 2 on a loop, and we know of no exact result for it. Our analysis needs only the bound up to the factor $\min(1, \zeta)$.

Lemma 11.2 (Gibbs bound, after Csikvári and Lin)**.** *For every tree F with $k \geq 1$ vertices and every $\zeta > 0$,*

$$\mathcal{Z}_F(\zeta) \;\geq\; \min(1, \zeta)\, \lambda_{\max}(M_\zeta)^{k-1}.$$

Proof. We write $\mathcal{Z}_F(\zeta)$ as a sum of positive weights $f(z)$ over labelings z of $V(F)$, construct a probability distribution $\mathbb{P}$ on the labelings from a reversible two-state Markov chain run down the

tree, and apply the Gibbs inequality $\ln \mathcal{Z}_F(\zeta) \geq \mathbb{E}_{\mathbb{P}} \ln f + H_e(\mathbb{P})$; for this $\mathbb{P}$ the right-hand side is $(k-1)\ln \lambda_{\max}(M_\zeta)$ up to a term depending only on ζ.

Write $w_0 = 1$, $w_1 = \zeta$, $A_{ij} = 2$ if $i = j = 1$ and $A_{ij} = 1$ otherwise, for $i, j \in \{0, 1\}$. A subset X of $V(F)$ is a labeling $z\colon V(F) \to \{0,1\}$, and $\mathcal{Z}_F(\zeta) = \sum_z f(z)$ with

$$f(z) = \prod_{v \in V(F)} w_{z_v} \prod_{uv \in E(F)} A_{z_u z_v} > 0.$$

A reversible chain. Let $\mathbf{K}_{ij} = \sqrt{w_i}\, A_{ij} \sqrt{w_j}$, that is, $\mathbf{K} = \left(\begin{smallmatrix} 1 & \sqrt{\zeta} \\ \sqrt{\zeta} & 2\zeta \end{smallmatrix}\right)$. This is a symmetric matrix with the same characteristic polynomial $x^2 - (1+2\zeta)x + \zeta$ as M_ζ, so its largest eigenvalue is $\lambda_{\max}(M_\zeta)$. The vector $(\sqrt{\zeta}, \lambda_{\max}(M_\zeta) - 1)$ is an eigenvector for $\lambda_{\max}(M_\zeta)$, by the first row of $\mathbf{K} - \lambda_{\max}(M_\zeta)\,\mathrm{Id}$, with Id the identity matrix, and both entries are positive, since $\lambda_{\max}(M_\zeta) > 1$. Let $\mathbf{v}$ be this vector normalized to $\mathbf{v}_0^2 + \mathbf{v}_1^2 = 1$, and put

$$\pi_i = \mathbf{v}_i^2, \qquad p_{ij} = \frac{\mathbf{K}_{ij}\mathbf{v}_j}{\lambda_{\max}(M_\zeta)\mathbf{v}_i} \qquad (i, j \in \{0,1\}).$$

Then $p_{ij} > 0$ and $\sum_j p_{ij} = (\mathbf{K}\mathbf{v})_i / (\lambda_{\max}(M_\zeta)\mathbf{v}_i) = 1$, so (p_{ij}) is a stochastic matrix. Moreover $\pi_{ij} := \pi_i p_{ij} = \mathbf{v}_i \mathbf{K}_{ij} \mathbf{v}_j / \lambda_{\max}(M_\zeta)$ is symmetric in i and j, so $\sum_i \pi_i p_{ij} = \sum_i \pi_j p_{ji} = \pi_j$: the probability vector π is stationary for (p_{ij}).

A labeling of the tree. Root F at a vertex r. Let $\mathbb{P}$ be the distribution of the random labeling that gives r a label drawn from π and then, going down the tree, gives every child v of a vertex u a label drawn from the row $p_{z_u,\cdot}$, independently of everything else:

$$\mathbb{P}(z) = \pi_{z_r} \prod_{u \to v} p_{z_u z_v} > 0,$$

the product over the $k-1$ pairs of a parent u and a child v. By stationarity and induction on the depth, every vertex has label distribution π. Hence every edge, with parent u and child v, has joint label distribution $\Pr[z_u = i, z_v = j] = \pi_{ij}$, which is symmetric and has both marginals equal to π.

The Gibbs inequality. For any probability distribution $\mathbb{P}$ on the labelings, Jensen's inequality for the concave function ln gives

$$\mathbb{E}_{\mathbb{P}} \ln f + H_e(\mathbb{P}) = \sum_{z\colon \mathbb{P}(z) > 0} \mathbb{P}(z) \ln \frac{f(z)}{\mathbb{P}(z)} \leq \ln \sum_z f(z) = \ln \mathcal{Z}_F(\zeta),$$

where $H_e(\mathbb{P}) = -\sum_z \mathbb{P}(z) \ln \mathbb{P}(z)$. We evaluate the left-hand side for our $\mathbb{P}$. By the product form of $\mathbb{P}$ and the marginals above,

$$H_e(\mathbb{P}) = H_e(\pi) - (k-1) \sum_{i,j} \pi_{ij} \ln p_{ij}, \qquad \mathbb{E}_{\mathbb{P}} \ln f = k \sum_i \pi_i \ln w_i + (k-1) \sum_{i,j} \pi_{ij} \ln A_{ij}.$$

Since $\ln A_{ij} = \ln \mathbf{K}_{ij} - \frac{1}{2} \ln w_i - \frac{1}{2} \ln w_j$ and both marginals of (π_{ij}) are π, we have $\sum \pi_{ij} \ln A_{ij} = \sum \pi_{ij} \ln \mathbf{K}_{ij} - \sum_i \pi_i \ln w_i$. Since $\ln p_{ij} = \ln \mathbf{K}_{ij} + \ln \mathbf{v}_j - \ln \mathbf{v}_i - \ln \lambda_{\max}(M_\zeta)$ and (π_{ij}) is symmetric, the terms in $\mathbf{v}$ cancel, and $-\sum \pi_{ij} \ln p_{ij} = \ln \lambda_{\max}(M_\zeta) - \sum \pi_{ij} \ln \mathbf{K}_{ij}$. Adding up, the terms in $\mathbf{K}$ cancel as well, and

$$\ln \mathcal{Z}_F(\zeta) \;\geq\; \sum_i \pi_i \ln w_i + H_e(\pi) + (k-1) \ln \lambda_{\max}(M_\zeta) = \pi_1 \ln \zeta + H_e(\pi) + (k-1) \ln \lambda_{\max}(M_\zeta).$$

Finally $H_e(\pi) \geq 0$ and $\pi_1 \in [0,1]$, so $\pi_1 \ln \zeta \geq \ln \min(1, \zeta)$. □

Paths attain the bound of Lemma 11.2 up to a factor depending only on ζ, by the next lemma. This is why the tightness results below (Proposition 11.9) use solutions whose tree is a path.

Lemma 11.3 (Paths)**.** *For a path F with $k \geq 1$ vertices and every $\zeta \geq 0$, $\mathcal{Z}_F(\zeta) \leq (1+\zeta)\,\lambda_{\max}(M_\zeta)^{k-1} \leq 2\lambda_{\max}(M_\zeta)^k$.*

Proof. Let $v_1, \ldots, v_k$ be the vertices of the path in order, and $\mathbf{d} = (1, \sqrt{\zeta})$. With $\mathbf{K}$ as in the proof of Lemma 11.2 (also for $\zeta = 0$), each labeling z contributes $\prod_v w_{z_v} \prod_{i<k} A_{z_{v_i} z_{v_{i+1}}} = \mathbf{d}_{z_{v_1}} \prod_{i<k} \mathbf{K}_{z_{v_i} z_{v_{i+1}}}\, \mathbf{d}_{z_{v_k}}$, because every inner vertex lies on two edges and each end on one. Hence $\mathcal{Z}_F(\zeta) = \mathbf{d}^{\mathsf{T}} \mathbf{K}^{k-1} \mathbf{d}$. As $\mathbf{K}$ is symmetric with eigenvalues in $[0, \lambda_{\max}(M_\zeta)]$ (their product is $\zeta \geq 0$ and their sum is positive), this is at most $\|\mathbf{d}\|^2 \lambda_{\max}(M_\zeta)^{k-1} = (1+\zeta)\lambda_{\max}(M_\zeta)^{k-1}$. Finally $1 + \zeta \leq 1 + 2\zeta \leq 2\lambda_{\max}(M_\zeta)$. □

11.2 Component-count running times and exponential-moment Monotone Local Search

We now state the property of a running-time exponent that the applications share, and the running time that Theorem 3.5 gives for it. Fix an implicit set system Φ and an algorithm $\mathcal{A}$ for Φ-Extension with running-time exponent g (Definition 3.2); recall $W_S(y) = \sum_{X \subseteq S} y^{|X|} 2^{-g(I,X,k)}$ for a solution S of size k (Definition 3.3).

Definition 11.4 (Component-count bound)**.** Let $c \geq 1$ and $\gamma \geq 0$ be constants. The running-time exponent g has a *component-count bound with constants* (c, γ) if for every instance I and every solution $S \in \mathcal{F}_I$ with $|S| = k \geq 1$ there is a tree F with vertex set S such that

$$g(I, X, k) \;\leq\; (k - |X|)\log c + |X| - e(F[X]) + \gamma \qquad \text{for every } X \subseteq S. \tag{20}$$

The tree F is used only in the analysis. Typically the solutions induce connected subgraphs of a graph G, the running-time exponent is $(k - |X|)\log c + \mathrm{comp}(G[X])$, and F is a spanning tree of $G[S]$: then $F[X]$ is a spanning forest of a subgraph of $G[X]$, so $\mathrm{comp}(G[X]) \leq \mathrm{comp}(F[X]) = |X| - e(F[X])$, by the edge count for forests. For $\sigma \in [0,1]$ and $y \geq 0$ put

$$\phi_{c,\sigma}(y) = \log(1+y) - \sigma \log \frac{\lambda_{\max}(M_{cy/2})}{c}. \tag{21}$$

For fixed y this is an affine function of σ, so a bound on it over an interval of values of σ follows from its values at the two ends; the certificate of Section 11.4.2 uses this.

Corollary 11.5 (Generating function of a component-count running time)**.** *Let g have a component-count bound with constants (c, γ), let $S \in \mathcal{F}_I$ with $|S| = k \geq 1$, let F be a tree for S as in Definition 11.4, and put $\sigma = k/n$.*

(a) *For every $y > 0$, with $\zeta = cy/2$,*

$$W_S(y) \;\geq\; 2^{-\gamma} c^{-k} \mathcal{Z}_F(\zeta) \;\geq\; 2^{-\gamma}\, \frac{\min(1,\zeta)}{\lambda_{\max}(M_\zeta)} \Big(\frac{\lambda_{\max}(M_\zeta)}{c} \Big)^k,$$

and therefore

$$\frac{(1+y)^n}{W_S(y)} \leq 2^{\gamma}\, \frac{\lambda_{\max}(M_\zeta)}{\min(1,\zeta)}\, 2^{n\phi_{c,\sigma}(y)}.$$

(b) *Put* $y^* = \frac{2(c-1)}{2c-1}$ *and* $\zeta^* = cy^*/2 = \frac{c(c-1)}{2c-1}$. *Then* $\lambda_{\max}(M_{\zeta^*}) = c$, *and if* $c > 1$,

$$W_S(y^*) \;\geq\; 2^{-\gamma}\,\frac{\min(1,\zeta^*)}{c}, \qquad \frac{(1+y^*)^n}{W_S(y^*)} \leq \frac{2^\gamma c}{\min(1,\zeta^*)}\Big(\frac{4c-3}{2c-1}\Big)^n.$$

(c) $W_S(0) = \mu_0(S) \geq 2^{-\gamma}c^{-k}$ *and* $\mu_k(S) \geq 2^{-1-\gamma}$.

Proof. (a) By (20), each term of $W_S(y)$ satisfies $y^{|X|}2^{-g(I,X,k)} \geq 2^{-\gamma}c^{-k}(cy/2)^{|X|}2^{e(F[X])}$. Summing over $X \subseteq S$ gives the first inequality, and Lemma 11.2 the second. The last form follows by dividing $(1+y)^n$ by this bound and writing $(c/\lambda_{\max}(M_\zeta))^k = 2^{-n\sigma\log(\lambda_{\max}(M_\zeta)/c)}$.

(b) The number c is a root of $x^2-(1+2\zeta^*)x+\zeta^*$, because $c^2-c-\zeta^*(2c-1) = c^2-c-c(c-1) = 0$. The other root is $\zeta^*/c = (c-1)/(2c-1) < c$, so $\lambda_{\max}(M_{\zeta^*}) = c$. Now (a) gives $W_S(y^*) \geq 2^{-\gamma}(\min(1,\zeta^*)/c)\cdot 1^k$, and $1+y^* = (4c-3)/(2c-1)$.

(c) For $X = \emptyset$ the right-hand side of (20) is $k\log c+\gamma$; for $X = S$ it is $k-(k-1)+\gamma = 1+\gamma$, because F has $k-1$ edges. □

Part (b) uses the same y^* for every solution size and, through Corollary 11.6 below, gives the closed-form base $(4c-3)/(2c-1)$, which is 1.8 for $c = 3$ and 5/3 for $c = 2$. Part (a) allows y to depend on σ, and part (c) covers the two ends: no sampling ($t = 0$) and enumeration ($t = k$). Together they give the following consequence of Theorem 3.5.

Corollary 11.6 (Algorithm MLS-θ for component-count running times)**.** *Let* $\mathcal{A}$ *solve* Φ*-Extension, either deterministically with running-time exponent* g*, or with one-sided error as in Corollary 3.15, and let* g *have a component-count bound with constants* (c,γ). *Let* $\psi > 0$ *and* $\Gamma_0 \geq 1$ *be constants such that every* $\sigma \in (0,1]$ *satisfies at least one of*

(R1) $\sigma\log c \leq \psi$;

(R2) $H(\sigma) \leq \psi$;

(R3) $\phi_{c,\sigma}(y) \leq \psi$ *for some* $y > 0$ *with* $\lambda_{\max}(M_{cy/2})/\min(1,cy/2) \leq \Gamma_0$.

Then for every $k \geq 1$, *every instance* I *and every* $\kappa > 0$ *as in Theorem 3.5, Algorithm* MLS-$\theta(\psi,\Gamma,\kappa)$ *with* $\Gamma = 2^{1+\gamma}\Gamma_0$, *given* (I,k), *runs in time* $2^{\psi n}N^{O(1)}$ *and in space* $N^{O(1)}$ *plus the space of* $\mathcal{A}$, *answers* NO *if* $\mathcal{F}_I$ *contains no set of size at most* k, *and answers* YES *with probability at least* $1-e^{-\kappa}$ *if* $\mathcal{F}_I$ *contains a set of size exactly* k. *In particular this holds with* $\psi = \log((4c-3)/(2c-1))$ *for every* $c > 1$.

Proof. Items (i) and (ii) of Theorem 3.5 give the time, space and soundness; for a randomized $\mathcal{A}$ use Corollary 3.15, which doubles Γ in the cap and in the trial counts. The rate ψ is irrational in general, for instance $\psi = \log 1.8$; the cap and the trial counts are then any integers in the ranges of (2), which are found in time $N^{O(1)}$ from an approximation of ψ to within $1/(4n)$. For item (iii), let $S \in \mathcal{F}_I$ with $|S| = k$ and $\sigma = k/n$. If (R1) holds, then by Corollary 11.5(c) $W_S(0) \geq 2^{-\gamma}2^{-\sigma n\log c} \geq (1+0)^n 2^{-\psi n}/\Gamma$. If (R2) holds, then $\binom{n}{k} \leq 2^{nH(\sigma)} \leq 2^{\psi n}$, and Corollary 11.5(c) gives $\mu_k(S)/\binom{n}{k} \geq 2^{-1-\gamma}2^{-\psi n} \geq 2^{-\psi n}/\Gamma$, which is $(\star)$. If (R3) holds with y, then Corollary 11.5(a) gives $(1+y)^n/W_S(y) \leq 2^\gamma\Gamma_0 2^{\psi n} \leq \Gamma 2^{\psi n}$. In the first and third case the condition $W_S(y) \geq (1+y)^n 2^{-\psi n}/\Gamma$ of Theorem 3.5 holds, with $y = 0$ in the first case, and it implies $(\star)$. For the last claim, (R3) holds for every σ with $y = y^*$ and $\Gamma_0 = c/\min(1,\zeta^*)$, by Corollary 11.5(b). □

The parameters of Corollary 11.6 have the following roles. The constant ψ is the rate: the target running time is $2^{\psi n}$. Condition (R1) covers the solution fractions σ for which running $\mathcal{A}$ without a sample ($t = 0$) is fast enough, (R2) those for which enumerating k-subsets ($t = k$) is, and (R3) those for which the generating-function bound of Corollary 11.5(a) is. The constant Γ_0 bounds the factor $\lambda_{\max}(M_\zeta)/\min(1,\zeta)$ of that bound; together with the factor $2^{1+\gamma}$ it enters only the loss factor Γ of condition ($\star$), not the rate. Real-valued running-time exponents are handled as in Remark 3.7. The decision problem is solved by running the algorithm for every k (Remark 3.6); membership of $\emptyset$ in $\mathcal{F}_I$ is tested directly.

The best rate for a given c. The best rate ψ that Corollary 11.6 can certify for a given c, when y in (R3) may be chosen freely, is

$$\psi_c = \max_{0\le\sigma\le 1} \min\Big(\sigma\log c,\ H(\sigma),\ \inf_{y>0}\phi_{c,\sigma}(y)\Big).$$

By Lemma 11.3, solutions whose tree F is a path with equality in (20) attain this rate up to polynomial factors by the argument of Proposition 11.9(a), which makes this precise for $c = 3$; and Lemma 11.2 shows that no other tree has a smaller partition function $\mathcal{Z}_F(\zeta)$, up to a factor depending only on ζ. For $c = 3$ the closed form is optimal: $\psi_3 \le \log 1.8$ by Corollary 11.5(b), and $\psi_3 \ge \log 1.8$, since at $\sigma = 26/45$ the infimum equals $\log 1.8$ (Lemma D.9.8) and both $\sigma\log 3$ (by the proof of Lemma D.9.8) and $H(\sigma)$ are larger: as $1/2 \le 26/45 < 3/5$ and H decreases on $[1/2,1]$, $H(\sigma) \ge H(\frac{3}{5}) = \log 5 - \frac{3}{5}\log 3 - \frac{2}{5} > 2\log 3 - \log 5 = \log 1.8$, which is equivalent to $5^{10} = 9765625 > 6377292 = 4\cdot 3^{13}$. For $c = 2$ it is not: the base $5/3$ is larger than $2^{\psi_2} \approx 1.6443$, because at the solution fraction σ that attains the maximum in ψ_2 the infimum over y is not attained at y^*. The following table compares, for a few c, three bases: the base $\max_\sigma \min(c^\sigma, 2^{H(\sigma)})$ obtained by running $\mathcal{A}$ without a sample in time $O^*(c^k)$ or enumerating all k-subsets, whichever is faster; the base given by the mean-exponent theorem (Theorem 4.1), for which the expected number of components of a uniformly random t-subset X of a connected k-set S is at most $t - t(t-1)/k$ (each of the $k-1$ edges of a spanning tree of $G[S]$ lies in $G[X]$ with probability $t(t-1)/(k(k-1))$); and 2^{ψ_c}. All entries are floating-point evaluations and are not used in any proof, except 1.8, which is exact; for $c = 2$, Lemma 11.12 certifies $2^{\psi_2} \le 1.6444$, and for $c = 4$ and $c = 6$ the entries in the column 2^{ψ_c} agree to five decimals with the closed-form upper bounds $13/7 = 1.857142\ldots$ and $21/11 = 1.909090\ldots$ of Corollary 11.5(b).

c	$\max_\sigma \min(c^\sigma, 2^{H(\sigma)})$	mean exponent	2^{ψ_c}
2	1.70871	1.64877	1.64429
3	1.95258	1.80802	1.8
4	2	1.86182	1.85714
6	2	1.91121	1.90909

11.3 Connected Feedback Vertex Set

A *connected feedback vertex set* of a graph G is a set $S \subseteq V(G)$ such that $G - S$ is a forest and $G[S]$ is connected. We write $\mathrm{OPT}(G)$ for the minimum size of a connected feedback vertex set of G, if one exists; Connected Feedback Vertex Set asks, given G, for one of size $\mathrm{OPT}(G)$. We are not aware of a published exact algorithm for Connected Feedback Vertex Set that runs in time $O((2-\varepsilon)^n)$ for a constant $\varepsilon > 0$. Without the connectivity constraint, [FGPR08] find a minimum feedback vertex set in time $O(1.7548^n)$ and show that a graph with n vertices has at

most 1.8638^n minimal feedback vertex sets. [MPR$^+$12] gave the first fixed-parameter algorithm for Connected Feedback Vertex Set, in time $c^k n^{O(1)}$ for a constant c. [CNP$^+$22] solve the problem by a Monte Carlo algorithm in time $3^k n^{O(1)}$ and polynomial space, where k is the solution size. For each k, running this algorithm or enumerating all k-subsets, whichever is faster, gives time $\max_{0\le\sigma\le 1} \min(3^\sigma, 2^{H(\sigma)})^n n^{O(1)}$; a floating-point evaluation of the maximum gives the base 1.9526, at $\sigma \approx 0.6091$. This combination is folklore; we do not know of it in print, and it is the bound that we improve.

We sample X, force it into the solution and complete it by a Cut & Count algorithm that, after contracting each component of $G[X]$ to a forced vertex, runs in time $3^{k-t}2^{\mathrm{comp}(G[X])}n^{O(1)}$ (Appendices D.9.2 and D.9.3 and Section 11.3.2), and Section 11.3.3 analyzes the sampling with the exponential moments.

11.3.1 Problem and the role of the sample

Graphs are finite and simple. We use the convention that the graph with no vertices is connected, so that $\emptyset$ is a connected feedback vertex set of every forest. It suffices to treat connected graphs that are not forests. If G is a forest, then $\mathrm{OPT}(G) = 0$. If G is disconnected, a connected feedback vertex set lies inside one component of G, so all other components must be forests; hence G has a connected feedback vertex set if and only if at most one component G_1 of G contains a cycle, and then the connected feedback vertex sets of G that are nonempty are exactly those of G_1. Both checks take polynomial time. For a connected graph G, the set $V(G)$ itself is a connected feedback vertex set, so $\mathrm{OPT}(G)$ exists and is at most n.

For a connected graph G that is not a forest, the connected feedback vertex sets form an implicit set system in the sense of Definition 3.1: the instance is G, the universe is $V(G)$ with $n = |V(G)|$, and membership is decided by testing that $G - S$ is acyclic and $G[S]$ is connected. Every solution is nonempty. Φ-Extension for this system (Definition 3.1), which we call Connected Feedback Vertex Set Extension, asks, given a connected graph G, a set $X \subseteq V(G)$ and an integer $k \ge 0$, whether some $Y \subseteq V(G) \setminus X$ with $|Y| \le k - |X|$ makes $X \cup Y$ a connected feedback vertex set of G.

Why the improvement needs a sample-dependent running-time exponent. The following three facts show that the improvement does not come from classical or multivariate Monotone Local Search. First, the extension problem is not fixed-parameter tractable in the number $k - |X|$ of further vertices under standard assumptions (Lemma D.9.1), so the theorem of [FGLS19], which needs an extension algorithm with running time $c^{k-|X|}N^{O(1)}$, does not apply. Second, the product-form running-time bound that our extension algorithm satisfies in the worst case $\mathrm{comp}(G[X]) = |X|$ is $3^{k-|X|}2^{|X|}$, that is, $(b, c, d) = (1, 3, 2)$, and Corollary 4.6 turns it into $(1 + 2 - 2/3)^n = (7/3)^n$, worse than enumeration. Third, under the Set Cover Conjecture, no extension algorithm with a running-time bound $c^{k-|X|}d^{|X|}N^{O(1)}$ gives a base below 2 through Corollary 4.6 (Proposition D.9.2). The running-time exponent we use instead is

$$g(G, X, k) = (k - |X|)\log 3 + \mathrm{comp}(G[X]), \tag{22}$$

which depends on X through the number of components of $G[X]$. The proofs of the first and the third fact are in Appendix D.9 (Lemma D.9.1 and Proposition D.9.2).

11.3.2 The extension algorithm

The extension algorithm contracts each component of the sample to a forced vertex and then follows the iterative compression of [CNP$^+$11, CNP$^+$22] along a sequence of edge contractions, with a Cut

& Count test that keeps the forced vertices in the solution; its running time is exponential in the number of forced vertices, which is $\mathrm{comp}(G[X])$. The construction and the proof of the following lemma are in Appendix D.9.

Lemma 11.7 (The extension algorithm)**.** *There is a randomized algorithm $\mathcal{A}$ for* Connected Feedback Vertex Set Extension *that, given a connected graph G with n vertices, a set $X \subseteq V(G)$ and an integer $k \geq |X|$, runs in polynomial space and, for every outcome of its coins, in time $3^{k-|X|}2^{\mathrm{comp}(G[X])}n^{O(1)}$. If it accepts, it outputs a set $Y \subseteq V(G) \setminus X$ with $|Y| \leq k - |X|$ such that $X \cup Y$ is a connected feedback vertex set of G. On a* yes*-instance it accepts with probability at least $3/4$. In particular, the running-time exponent* (22) *bounds its running time.*

11.3.3 Exponential moments: the base 1.8

We now run Algorithm MLS-θ with the extension algorithm of Section 11.3.2 and analyze it through the exponential moments of the running-time exponent (22).

Theorem 11.8 (Connected Feedback Vertex Set)**.** *There is a randomized algorithm that, given a connected graph G with n vertices, outputs a connected feedback vertex set of G in time $O^*(1.8^n)$ and polynomial space. The output always has size at least $\mathrm{OPT}(G)$, and it has size exactly $\mathrm{OPT}(G)$ with probability at least $1-1/e$. For the running-time exponent* (22)*, the rate $\log 1.8$ of Algorithm* MLS-θ *cannot be lowered by a positive constant (Proposition 11.10).*

Proof. If G is a forest, output $\emptyset$. Otherwise the connected feedback vertex sets form an implicit set system whose solutions are nonempty (Section 11.3.1). The extension algorithm of Lemma 11.7 has one-sided error, accepts a yes-instance with probability at least $3/4$, and runs in time $2^{g(G,X,k)}n^{O(1)}$ for every outcome of its coins, with $g(G, X, k) = (k-|X|)\log 3 + \mathrm{comp}(G[X])$ as in (22); comparisons of g with the thresholds of Algorithm MLS-θ reduce to integer comparisons (Section 3.1). Every solution S induces a connected subgraph, so a spanning tree F of $G[S]$ gives the component-count bound (20) with constants $(3, 0)$. By Corollary 11.6 with $c = 3$, $y^* = 4/5$, $\zeta^* = 6/5$ and $\psi = \log 1.8$, Algorithm MLS-θ with $\kappa = 1$, run for $k = 1, \ldots, n$, returns a solution of size at most k only if one exists, and returns one with probability at least $1 - 1/e$ for $k = \mathrm{OPT}(G)$; each run takes time $1.8^n\, n^{O(1)}$ and polynomial space. Output the smallest set found, and $V(G)$, which is a connected feedback vertex set because G is connected, if none is found. Every output is a connected feedback vertex set, so its size is at least $\mathrm{OPT}(G)$, and with probability at least $1 - 1/e$ the run for $k = \mathrm{OPT}(G)$ returns a set of size at most $\mathrm{OPT}(G)$, which is then the output. □

By Corollary 11.5(b) with $\gamma = 0$, every solution satisfies $W_S(4/5) \geq 1/3$. The next proposition turns the path bound of Lemma 11.3 into an upper bound on the probability that a run of Algorithm MLS-θ runs the extension algorithm on a subset of a solution S with $|S| = 26n/45$ whose running-time exponent is bounded below as for a path, for every run on (I, k) with $k \geq |S|$. Its proof, in Appendix D.9.5, uses that at the critical fraction $\sigma^* = 26/45$ the function ϕ_{3,σ^*} of (21) has $\inf_{y \geq 0} \phi_{3,\sigma^*}(y) = \log 1.8$, attained at $y = 4/5$ (Lemma D.9.8), and combines this with the necessity of $(\star)$ (Proposition 3.10), in the form for budgets $k \geq |S|$, and with the bound (4) on the moments $\mu_t^{(k)}(S)$ of (3).

Proposition 11.9 (Tightness of 1.8)**.** *Let I be an instance and $S \in \mathcal{F}_I$ with $|S|/n = 26/45$. Suppose that $g(I, X, k)$ is nondecreasing in k for every $X \subseteq S$, and that, for a path F with vertex set S and a constant $\gamma' \geq 0$,*

$$g(I, X, |S|) \;\geq\; (|S| - |X|)\log 3 + |X| - e(F[X]) - \gamma' \qquad \text{for every } X \subseteq S.$$

(a) *The exponential moments of S (Definition 3.3, with $k = |S|$) satisfy*

$$\max_{0 \le t \le |S|} \frac{\mu_t(S)}{\binom{n}{t}} \le 2^{\gamma'+1}(n+1)(5/9)^n.$$

(b) *For every integer k with $|S| \le k \le n$, a run of Algorithm* MLS-$\theta(\psi, \Gamma, \kappa)$ *on (I, k) runs $\mathcal{A}$ on a subset of S with probability at most $P_{\text{nec}} \cdot 2^{\gamma'+1}(n+1)\big(2^{\psi} \cdot 5/9\big)^n$, with $P_{\text{nec}} \le N^{O(1)}$ as in Proposition 3.10.*

(c) *Suppose that $\mathcal{A}$ never accepts a* NO*-instance and that S is the only set in $\mathcal{F}_I$ of size at most $|S|$. Then a run on (I, k) with $k \le |S|$ answers* YES*, and a run on (I, k) with any $k \le n$ returns a solution of size at most $|S|$, only if it runs $\mathcal{A}$ on a subset of S.*

(d) *Under the hypotheses of (c), let $\varepsilon > 0$ and $\psi \le \log 1.8 - \varepsilon$, and run Algorithm* MLS-$\theta(\psi, \Gamma, \kappa)$ *on (I, k) for $k = 1, \ldots, n$, with any Γ and κ as in Theorem 3.5 (for a one-sided-error $\mathcal{A}$, with 2Γ as in Corollary 3.15). The probability that some run answers* YES *at a $k \le |S|$, or that some run returns a set of size at most $|S|$, is at most $nP_{\text{nec}}2^{\gamma'+1}(n+1)2^{-\varepsilon n}$, where $P_{\text{nec}} \le N^{O(1)}$ is the largest of the factors of Proposition 3.10 over these runs.*

Families of instances that satisfy all hypotheses of Proposition 11.9 exist for Connected Feedback Vertex Set.

Proposition 11.10 (A tight family for Connected Feedback Vertex Set)**.** *Let $\varepsilon > 0$ be a constant. There is an infinite family of graphs on which the algorithm of the proof of Theorem 11.8, run with rates $\psi \le \log 1.8 - \varepsilon$ in place of $\log 1.8$ and with any Γ and κ as in Theorem 3.5, outputs a minimum connected feedback vertex set with probability $o(1)$ as the number n of vertices grows.*

11.4 Graph Motif

The method of Section 11.3 needs a finishing step whose running time is exponential in the number of components of the sample, and a solution that is connected. Graph Motif has both, with a simpler finishing step: the randomized $O^*(2^k)$ algorithm of Björklund, Kaski and Kowalik [BKK16], used as a black box, after the components of the sample have been contracted and given fresh colors.

An instance of Graph Motif is a graph G with n vertices, a coloring $\text{col} : V(G) \to C$ and a multiset M over C with $|M| = k$. A *solution* is a set $S \subseteq V(G)$ such that $G[S]$ is connected and $\text{col}(S) = M$ as multisets; every solution has size exactly k. Björklund, Kaski and Kowalik [BKK16] solve the problem in randomized time $O^*(2^k)$ and polynomial space. They also show [BKK16, Theorem 6] that if, for some constant $\varepsilon > 0$, the problem can be solved in time $O((2-\varepsilon)^k)$, even only on instances in which every color occurs at most once in M, then there is a constant $\varepsilon' > 0$ such that Set Cover on a universe of u elements can be solved in time $O((2-\varepsilon')^u)$. (They state this for Maximum Graph Motif, which asks for a connected set S with $|S| = k$ and $\text{col}(S) \subseteq M$; when $|M| = k$, as in their reduction, this is the problem above.) Running their algorithm or enumerating all k-subsets, whichever is faster, gives $\max_{0 \le \sigma \le 1} \min(2^{\sigma}, 2^{H(\sigma)})^n = (1.7087\ldots)^n \le 1.7088^n$, at $\sigma \approx 0.7729$.

11.4.1 Finishing with forced vertices

The finishing step contracts each component of the sample to one vertex and gives that vertex a fresh color which the motif demands exactly once, so that the black-box algorithm must use every contracted component.

Lemma 11.11 (Extension)**.** *Given an instance (G, col, M) with $|M| = k$ and a set $X \subseteq V(G)$ with $|X| = t \le k$, a randomized algorithm can decide whether some solution S contains X. It runs in time $2^{(k-t)+\mathrm{comp}(G[X])}n^{O(1)}$ for every outcome of its coins and in polynomial space, never accepts if there is no such solution, and accepts with probability at least $1/2$ if there is one.*

Proof. If $\mathrm{col}(X) \not\subseteq M$ as multisets, reject. Otherwise let $X_1, \dots, X_q$ be the vertex sets of the components of $G[X]$, $q = \mathrm{comp}(G[X])$. Form G' from G by contracting each X_i into one vertex d_i (adjacent to every vertex outside X with a neighbor in X_i; the vertices d_i are pairwise non-adjacent, since the X_i are components of $G[X]$), give d_i a fresh color c_i, and let $M' = (M \setminus \mathrm{col}(X)) + \{c_1, \dots, c_q\}$, a multiset of size $k - t + q$.

We claim that the solutions $S \supseteq X$ of the original instance correspond to the connected sets S' of G' with $\mathrm{col}(S') = M'$, by $S \mapsto (S \setminus X) \cup \{d_1, \dots, d_q\}$. Such an S' contains every d_i, because c_i occurs once in M' and only d_i has that color. Let S' be such a set. Then $S = (S' \setminus \{d_1, \dots, d_q\}) \cup X$ induces a connected subgraph of G, since each vertex of the connected graph $G'[S']$ expands to a connected set, and an edge of G' at a vertex d_i comes from an edge of G at X_i. Its colors are $(M' - \{c_1, \dots, c_q\}) + \mathrm{col}(X) = M$. Conversely, if $S \supseteq X$ is a solution, the image of S in G' is connected and has the colors M'.

If $q \ge 1$ and the vertices d_i do not all lie in one component of G', reject. Otherwise restrict G' to the component that contains the vertices d_i (to each component in turn if $q = 0$), so that the host graph is connected. On this graph run the algorithm of Björklund, Kaski and Kowalik [BKK16, Theorem 4 and Section 3.7] with motif M'. It decides the existence of a connected induced subgraph S' with $|S'| = |M'|$ and $\mathrm{col}(S') \subseteq M'$ as multisets, that is, with $\mathrm{col}(S') = M'$, in time $2^{|M'|}n^{O(1)}$ and polynomial space; it never answers YES on a NO-instance and answers YES on a YES-instance with probability at least $1/2$. As $|M'| = (k - t) + \mathrm{comp}(G[X])$, the lemma follows. □

11.4.2 Exponential moments: the base 1.6444

For $c = 2$ the closed form of Corollary 11.5(b) gives the base $5/3$, which is not optimal, so we optimize y for each solution fraction and certify the result. Since $\phi_{2,\sigma}(y)$ is affine in σ for fixed y, a finite list of intervals in σ suffices, and no rounding of a sample size has to be absorbed: y is a real parameter of the analysis only.

Lemma 11.12 (Rate certificate for $c = 2$)**.** *Every $\sigma \in (0, 1]$ satisfies* (R1)*,* (R2) *or* (R3) *of Corollary 11.6 with $c = 2$, $\psi = 0.71755$ and $\Gamma_0 = 16$. Moreover $0.71755 < \log 1.6444$, so $2^{0.71755n}n^{O(1)} = O(1.6444^n)$.*

Proof. Let $\psi = 0.71755$, $\sigma_0 = 0.7175$ and $\sigma_1 = 0.81$. The certificate of Appendix F.6 gives (o) $\psi < \log 1.6444$; (i) $\sigma_0 \le \psi$; (ii) $H(\sigma_1) < \psi$ and $\sigma_1 \ge 1/2$; (iii) for 925 consecutive intervals $[\sigma_j^-, \sigma_j^+]$ of width 10^{-4} covering $[\sigma_0, \sigma_1]$, with explicit decimals $y_j > 0$, we have $\max(\phi_{2,\sigma_j^-}(y_j), \phi_{2,\sigma_j^+}(y_j)) < \psi$ and $\lambda_{\max}(M_{y_j})/\min(1, y_j) \le 16$. If $\sigma \le \sigma_0$, then (R1) holds, as $\log c = 1$. If $\sigma \ge \sigma_1$, then (R2) holds, as H is decreasing on $[1/2, 1]$. Otherwise $\sigma \in [\sigma_j^-, \sigma_j^+]$ for some j, and since $\phi_{2,\sigma}(y_j)$ is affine in σ, it is at most the maximum of its values at σ_j^- and σ_j^+, so (R3) holds with $y = y_j$. □

Theorem 11.13 (Graph Motif)**.** Graph Motif *can be solved by a randomized algorithm with one-sided error in time $O(1.6444^n)$ and polynomial space.*

The constant 1.6444 rests on Lemma 11.12; the algorithm of Björklund, Kaski and Kowalik [BKK16] is used as a black box. The mean-exponent theorem (Theorem 4.1) with the same finishing step, run with the sample size that minimizes its bound, gives, by a numerical evaluation, a running time of $O^*(1.6488^n)$ (the entry 1.64877 in the table of Section 11.2).

Proof. The solutions of an instance (G, col, M) with $|M| = k$ are the sets S with $G[S]$ connected and $\mathrm{col}(S) = M$; they have size $k \geq 1$ (for $k = 0$ the answer is trivial). By Lemma 11.11 the extension problem is solved with one-sided error, success probability at least $1/2$ and polynomial space, in time $2^{g(X)}n^{O(1)}$ for every outcome of the coins, with the integer running-time exponent $g(X) = g(I, X, k) = (k - |X|) + \mathrm{comp}(G[X])$ (shorthand of Definition 3.2, for the instance $I = (G, \mathrm{col}, M)$). A spanning tree of $G[S]$ gives the component-count bound with constants $(2, 0)$. Corollary 11.6 with Lemma 11.12, that is, with the rate $\psi = 0.71755$, gives running time $2^{0.71755n}n^{O(1)} = O(1.6444^n)$; only the value $k = |M|$ is run. □

Numerically, the base $2^{\psi_2} \approx 1.64429$ is attained at $\sigma \approx 0.71746$, where the running-time bound $2^{\sigma n}$ of the plain algorithm equals the bound $2^{n\phi_{2,\sigma}(y)}$ of (R3) with $y \approx 1.294$; the best strategy jumps there from no sampling to large samples. Lemma 11.12 proves the upper bound 1.6444; it is not sharp enough for the bound 1.6443. The analysis needs paths to minimize $\mathcal{Z}_F$ up to a constant factor (Lemma 11.2). We do not know whether the exact inequality $\mathcal{Z}_F \geq \mathcal{Z}_{\mathrm{path}}$ holds for all trees on the same number of vertices.

Related work. Graph motif problems were introduced by Lacroix, Fernandes and Sagot [LFS06] for the analysis of metabolic networks. [FFHV11] showed that Graph Motif is fixed-parameter tractable for the parameter k, and a series of improvements [BvBF$^+$11, GS13, Kou12] led to the $O^*(2^k)$ algorithm of Björklund, Kaski and Kowalik [BKK16]. When M is a set (Colorful Graph Motif), the problem can be solved in time $O(2^{n-k}|E(G)|)$ [BvBF$^+$11, Theorem 3]; together with the $O^*(2^k)$ algorithm this gives $O^*(2^{n/2})$, below the bound of Theorem 11.13. For multisets this route is closed: Graph Motif is W[1]-hard for the parameter $n - k$, even when M has only two colors [BvBF$^+$11, Theorem 4] (see also [FK20]).

11.5 Connected Vertex Cover and further applications

Each application below needs only a finishing step whose running-time exponent has a component-count bound; the sampling analysis is Corollary 11.6, with Lemma 11.12 for $c = 2$ and the closed form of Corollary 11.5(b) for $c = 3$. For optimization problems with weights we use the following standard reduction. If the weights are integers in $\{-w_{\max}, \ldots, w_{\max}\}$ with $w_{\max} \leq n^{O(1)}$, we guess the optimal value θ_w among $O(nw_{\max})$ candidates and the size k of one optimal solution S^*, and decide Φ-Subset for the implicit set system Φ whose solutions are the feasible sets of value at least (or at most) θ_w; its membership test is polynomial. The sampling analysis concerns only S^*, and the weights never enter the running-time exponent.

11.5.1 Connected Vertex Cover, weighted and unweighted

A *connected vertex cover* of a graph G is a set $S \subseteq V(G)$ such that $G - S$ has no edges and $G[S]$ is connected. In Weighted Connected Vertex Cover the vertices carry weights $w\colon V(G) \to \{1, \ldots, w_{\max}\}$, $w_{\max} \leq n^{O(1)}$, and we seek a connected vertex cover of minimum weight; unit weights give Connected Vertex Cover. We may assume that G is connected and has an edge: otherwise a connected vertex cover exists only if at most one component has an edge, and the other components can be discarded. The finishing step uses Cygan's algorithm as a black box [Cyg12]: given G, positive integer weights bounded by a polynomial and an integer b, it computes a minimum-weight connected vertex cover of size at most b (or reports that none exists), deterministically, in time $2^b n^{O(1)}$ and polynomial space (his Theorem 3).

Lemma 11.14 (Finishing step)**.** *Given a connected graph G with at least one edge, weights as above, a set $X \subseteq V(G)$ and integers $k \geq |X|$ and θ_w, one can decide deterministically, in polynomial space and in time $2^{k-|X|+\mathrm{comp}(G[X])}n^{O(1)}$, whether some $Y \subseteq V(G) \setminus X$ with $|Y| \leq k - |X|$ makes $X \cup Y$ a connected vertex cover of weight at most θ_w, and output such a Y if so.*

Proof. Put $k' = k - |X|$. If $X = \emptyset$, run Cygan's algorithm with $b = k'$; accept if it returns a cover of weight at most θ_w, and output that cover as Y. Otherwise let G_X be the contracted graph of Appendix D.9.2, with forced set $\hat{X} = \{x_1, \ldots, x_q\}$, $q = \mathrm{comp}(G[X])$. For $Y \subseteq V(G) \setminus X$, the set $X \cup Y$ is a connected vertex cover of G if and only if $\hat{X} \cup Y$ is one of G_X: the proof of Lemma D.9.3(2) applies verbatim, because $G - (X \cup Y) = G_X - (\hat{X} \cup Y)$, and a set is a vertex cover exactly when the graph without it has no edges. If G_X has one vertex, $q = 1$, $Y = \emptyset$ is the only candidate, and the algorithm accepts if and only if $w(X) \leq \theta_w$, with output $Y = \emptyset$. Otherwise G_X is connected with at least two vertices. Let G_X^+ be G_X with a new pendant vertex r' attached to each $r \in \hat{X}$, and give the vertices of $\hat{X}$ and the new vertices weight 1.

Every connected vertex cover C of G_X^+ contains $\hat{X}$. Indeed, if $r \in \hat{X} \setminus C$, then $r' \in C$, to cover rr', and the other end u of an edge ru of G_X lies in C; but the only neighbor of r' is $r \notin C$, so $G_X^+[C]$ is disconnected. Hence each $r' \in C$ is a leaf of $G_X^+[C]$, and removing all of them leaves a connected vertex cover of G_X that contains $\hat{X}$, of no larger size and weight. Conversely, a connected vertex cover of G_X that contains $\hat{X}$ is one of G_X^+. So the minimum weight of a connected vertex cover of G_X^+ of size at most $k' + q$ equals the minimum of $w(Y) + q$ over the sets Y with $|Y| \leq k'$ such that $\hat{X} \cup Y$ is a connected vertex cover of G_X. Compute it with Cygan's algorithm, $b = k' + q$, and accept if it is at most $\theta_w - w(X) + q$. In that case the cover C returned by Cygan's algorithm gives the output $Y = C \setminus (\hat{X} \cup \{r' : r \in \hat{X}\})$: by the above, $\hat{X} \cup Y$ is a connected vertex cover of G_X with $|Y| \leq |C| - q \leq k'$ and $w(Y) \leq w(C) - q \leq \theta_w - w(X)$. □

Theorem 11.15 (Connected Vertex Cover)**.** *Let $w_{\max} \leq n^{O(1)}$ and let $\kappa > 0$ be a constant. There is a randomized algorithm that, given a connected graph G with $n \geq 2$ vertices and weights $w \colon V(G) \to \{1, \ldots, w_{\max}\}$, outputs a connected vertex cover of G in time $O(1.6444^n)$ and polynomial space. On every outcome of its coins the output is a connected vertex cover of G, so its weight is at least the minimum; with probability at least $1 - e^{-\kappa}$ its weight is the minimum. In particular, this holds for* Connected Vertex Cover*, and by the reduction above it extends to graphs that are not connected or have no edge.*

The constant rests on Lemma 11.12; Cygan's algorithm is used as a black box.

Proof. For each guess of $\theta_w \in \{1, \ldots, nw_{\max}\}$ and each $k \in \{1, \ldots, n\}$, run Algorithm MLS-θ for the connected vertex covers of weight at most θ_w and the size k, and output the witness $X \cup Y$ of an accepting run with the smallest θ_w. If no run accepts, output $V(G)$, which is a connected vertex cover because G is connected. An accepting run returns a connected vertex cover of weight at most its θ_w (Lemma 11.14), so on every outcome of the coins the output is a connected vertex cover. The extension problem for these implicit set systems is solved by Lemma 11.14, with running-time exponent $g(G, X, k) = k - |X| + \mathrm{comp}(G[X])$, deterministically. Every solution induces a connected subgraph, so a spanning tree of $G[S]$ gives the component-count bound with constants $(2, 0)$, and Corollary 11.6 with Lemma 11.12 applies for every k, with the rate $\psi = 0.71755$; there are polynomially many runs, each in time $2^{0.71755n}n^{O(1)}$, which is $O(1.6444^n)$ in total by Lemma 11.12. A run for the optimal θ_w and $k = |S^*|$ succeeds with probability at least $1 - e^{-\kappa}$, and no run accepts a value below the optimum. □

Comparison with published bounds. For Connected Vertex Cover the best published bound in terms of n is $O(1.7088^n)$ [GHK18, Corollary 1], from Cygan's 2^k algorithm and enumeration. For the weighted problem, enumerating the minimal connected vertex covers gives $O^*(1.8668^n)$ [GHK18, Theorem 3] (with positive weights a minimum-weight connected vertex cover is minimal), and balancing Cygan's Theorem 3 [Cyg12] against enumeration gives $O(1.7088^n)$, a combination we have not seen stated. Our algorithm is randomized through the sampling only; the finishing step is deterministic.

11.5.2 Maximum-Weight Connected k-Subgraph

Given a graph G, integer vertex weights $w\colon V(G) \to \{-w_{\max}, \ldots, w_{\max}\}$ with $w_{\max} \geq 1$ and an integer $1 \leq k \leq n$, Maximum-Weight Connected k-Subgraph asks for a set S with $|S| = k$ and $G[S]$ connected that maximizes $w(S)$; we call the sets S with $|S| = k$ and $G[S]$ connected, of any weight, *connected k-sets*. We know of no exact bound in terms of n besides enumeration of all k-sets; for the version without the size constraint, Buchanan, Wang and Butenko [BWB18] give $O(1.5875^n)$. The finishing step below with $X = \emptyset$ runs in time $2^k(nw_{\max})^{O(1)}$; balanced against enumeration it gives $O(1.7088^n)$ for $w_{\max} \leq n^{O(1)}$, the same base as for Graph Motif, and this folklore combination is the bound we compare with.

The finishing step. As for Graph Motif (Lemma 11.11), the components $X_1, \ldots, X_q$ of the sample are contracted to vertices $d_1, \ldots, d_q$ of weight 0, and each new vertex receives a fresh color that must occur exactly once, while all other vertices share one color of multiplicity $k - |X|$. The connected k-sets $S \supseteq X$ then correspond to the connected sets of size $k - |X| + q$ of the contracted graph that contain all vertices d_i, with $w(S) = w(X)$ plus the weight of the corresponding set. We detect these sets with the algebraic sieve of Björklund, Kaski and Kowalik [BKK16], to which we add one indeterminate whose exponent records the weight of the set, and we recover the polynomial in this indeterminate by interpolation. Their cost-constrained lemma lets the coloring vary, so we derive the variant with a fixed coloring and a cost on each vertex from their proofs. A witness is found by deleting vertices one at a time. The finishing step takes time $2^{(k-|X|)+\mathrm{comp}(G[X])}(nw_{\max})^{O(1)}$ and polynomial space and has one-sided error.

Theorem 11.16 (Maximum-Weight Connected k-Subgraph)**.** *Let $1 \leq w_{\max} \leq n^{O(1)}$. There is a randomized algorithm that, given a graph G with n vertices, weights $w\colon V(G) \to \{-w_{\max}, \ldots, w_{\max}\}$ and an integer $1 \leq k \leq n$, either reports correctly that G has no connected k-set or outputs a connected k-set, in time $O(1.6444^n)$ and polynomial space. The output has maximum weight with probability at least $1 - 1/e$. In particular, the decision version ("is there a connected k-set of weight at least θ?") is solved with one-sided error.*

The constant rests on Lemma 11.12. The proof is in Appendix D.10. The variant without the size constraint is not improved by this method.

11.5.3 Vertex-weighted Connected Feedback Vertex Set

In the vertex-weighted version of Connected Feedback Vertex Set the vertices carry weights $w\colon V(G) \to \{1, \ldots, w_{\max}\}$, and we seek a connected feedback vertex set of minimum weight. We know of no published algorithm in time $O((2-\varepsilon)^n)$ for a constant $\varepsilon > 0$; the Cut & Count algorithm of [CNP+22] is stated for the cardinality version.

The extension algorithm. After contracting the sample (Lemma D.9.3(2)) and giving the forced vertices weight 0, the weight of a completion is $w(X)$ plus the weight of its non-forced part. The weighted extension algorithm first runs the unweighted extension algorithm of Lemma 11.7, only to obtain a connected feedback vertex set of small excess, which serves as a structured modulator; so the intermediate graphs of the iterative compression never need weights. It then counts the objects of (CC1), through the proved Lemmas D.10.8 and D.10.9 rather than the cited dynamic program, with a weight function that adds to each isolating weight of a forest vertex a large multiple of its vertex weight, so that the weight of a counted candidate determines both its isolating weight and its vertex weight, and it finds a solution by self-reduction. It has the running-time exponent (22) of the unweighted algorithm. The two properties of the counting that it needs are proved in Appendix D.10.2, so they are not imported. They are a parity identity for the counted cuts (Lemma D.10.8) and a dynamic program over the forest outside the modulator (Lemma D.10.9). That program has four states per vertex for every weight function, and the weights change only the range of its weight index.

Theorem 11.17 (Vertex-weighted Connected Feedback Vertex Set)**.** *Let $w_{\max} \le n^{O(1)}$. There is a randomized algorithm that, given a graph G with n vertices and weights $w\colon V(G) \to \{1, \dots, w_{\max}\}$, reports correctly that G has no connected feedback vertex set or outputs one, in time $O^*(1.8^n)$ and polynomial space. The output has minimum weight with probability at least $1 - 1/e$. In particular, the decision version ("is there a connected feedback vertex set of weight at most θ?") is solved with one-sided error.*

The proof is in Appendix D.10.

12 Further applications

The deterministic algorithm of Section 10 and Algorithm RS-θ of Section 7 use the problem through two ingredients: an extension algorithm whose running time is $O^*(2^{g(X)})$ for a computable running-time exponent g, written with the shorthand $g(X) = g(I, X, k)$ declared after Definition 3.2; and a pointwise affine bound $g(X) \le \alpha - w(X)$ for the samples X of a solution S, with a number α and nonnegative integer weights $w\colon S \to \mathbb{Z}_{\ge 0}$ whose total $w(S)$ is known (or, for RS-θ, the tree potential of Lemma 7.3). This section transfers them to four variants of Connected Dominating Set and then treats Directed Steiner Tree and Paired Domination (Sections 12.5 and 12.6). The four variants are distance-r domination (Section 12.1), vertex weights (Section 12.2), weak connectivity (Section 12.3) and directed graphs (Section 12.4). All algorithms use polynomial space. All four extension algorithms rest on the vertex-weighted directed Steiner lemma of Section 5.1 (Lemma 5.3): the first three through the directed Steiner encoding of Lemma 5.6, applied to states of red–blue graphs (Lemma D.11.1), and the directed variant through the extension algorithm for Directed Steiner Tree (Lemma 12.7). For each variant we state the problem and the results, and say what changes relative to Connected Dominating Set and why the analysis transfers; the lemmas and proofs are in Appendix D.11.

Directed Steiner Tree and Paired Domination share the feature of the earlier applications: the running time of the finishing step is exponential in a count that a sample can reduce, while a global constraint on the solution does not enter the exponent. In Directed Steiner Tree the count is the number of source components left by a sample, and the finishing step is inclusion–exclusion over them. In Paired Domination the count is the number of undominated vertices, and the global constraint, a perfect matching inside the solution, is handled by a Tutte matrix at no increase of the exponent. For both problems Algorithm MLS-θ runs in polynomial

space, with one-sided error, and below the best previous bound that we know (Table 2); the analysis is tight for Paired Domination (Proposition 12.11) and appears close to tight, numerically, for Directed Steiner Tree.

12.1 Connected r-domination

The two ingredients of Section 12 transfer to *domination relations*, which include distance-r domination.

Definition 12.1 (Domination relations)**.** Let G be a graph. A *domination relation* on G assigns to every vertex v a set $\mathcal{N}[v] \subseteq V$, the vertices that *dominate* v, such that $N[v] \subseteq \mathcal{N}[v]$ for every v, and such that $\mathcal{N}[v]$ can be computed from G (and the description of $\mathcal{N}$) in time polynomial in their total length. A set $S \subseteq V$ is a *connected $\mathcal{N}$-dominating set* if it is nonempty, $G[S]$ is connected, and $\mathcal{N}[v] \cap S \neq \emptyset$ for every $v \in V$; for a connected graph we write $\gamma_c^{\mathcal{N}}(G)$ for the minimum size of one. For $X \subseteq V$ let $\mathrm{und}_{\mathcal{N}}(X) = \{v \in V : \mathcal{N}[v] \cap X = \emptyset\}$ be the vertices not dominated by X.

With $\mathcal{N}[v] = N[v]$ this is connected domination. With $\mathcal{N}[v] = \{u : \mathrm{dist}_G(u, v) \leq r\}$ for an integer $r \geq 1$, which may be part of the input, it is *connected r-domination*, and Connected r-Dominating Set asks for $\gamma_c^{\mathcal{N}}(G)$. Every superset of an $\mathcal{N}$-dominating set is $\mathcal{N}$-dominating, so a connected $\mathcal{N}$-dominating set exists if G is connected (V itself is one), and for distance-r domination one exists only if G is connected. As in Section 5, G is connected and has $n \geq 2$ vertices.

What changes. Both ingredients change, and each keeps its constants. First, the number of terminals $|\mathrm{T}(J_X)|$ of (13) is replaced by $g_{\mathcal{N}}(X) = |\,\mathrm{und}_{\mathcal{N}}(X)| + \mathrm{comp}(G[X])$, and the Steiner extension of Corollary 5.8 transfers with this running-time exponent (Lemma D.11.2): the completions of X are the solutions of a state (Definition 5.4) of a red–blue graph in which the red copy of v is joined to the blue copies of the vertices of $\mathcal{N}[v]$, while connectivity is still measured by adjacency in G (Lemma D.11.1). Second, every vertex outside the solution is assigned to a solution vertex that dominates it, which need not be a neighbor, and this assignment is treated as a pendant edge of a virtual tree, the *anchored tree*, in which the solution is a connected dominating set. The running-time exponent of a sample is at most its number of terminals in the anchored tree (Lemma D.11.3), so the rate of Proposition 7.8 and the certificate (19), applied to the anchored tree, bound the exponential moments and the integer certificate for $g_{\mathcal{N}}$ with the same constants as for connected domination, and the analyses of Theorems 7.2 and 10.4 transfer pointwise.

Theorem 12.2 (Connected $\mathcal{N}$-domination)**.** *Let $\mathcal{N}$ be a domination relation. For connected n-vertex graphs G, the value $\gamma_c^{\mathcal{N}}(G)$, in particular the minimum size of a connected r-dominating set for every $r \geq 1$ (with r part of the input), can be computed*

(a) by a randomized algorithm with one-sided error in time $O(1.7204^n)$ and polynomial space;

(b) deterministically in time $O((\sqrt{5} - \frac{1}{2})^n\, 2^{O(n(\log\log n)^2/\log n)}) = O(1.7361^n)$ and polynomial space.

The bases are those of Theorems 7.2 and 10.4. The proof is in Appendix D.11.2; Remark 12.6 lists what does not transfer.

Related work. Fomin, Grandoni and Kratsch [FGK08] and Abu-Khzam, Mouawad and Liedloff [AKML11] solve Connected Dominating Set by branching on closed neighborhoods, and their measures do not apply directly to $r \geq 2$. The red–blue graph $H_{\mathcal{N}}$ of Lemma D.11.2, on

$2n$ vertices, reduces the problem to Connected Red-Blue Dominating Set, so the algorithm of Abu-Khzam, Mouawad and Liedloff [AKML11] computes $\gamma_c^{\mathcal{N}}(G)$ deterministically in time $O^*(1.3645^{2n}) = O^*(1.8619^n)$ and polynomial space. For $r \geq 2$ we know of no published exact algorithm below 2^n for Connected r-Dominating Set, also known as connected distance-r or d-hop connected domination; the reduction above is the only one known to us.

12.2 Vertex-weighted connected domination and weighted MaxLeaf

In this subsection G is a connected graph with $n \geq 2$ vertices and $\varphi\colon V \to \{0, 1, \ldots, \varphi_{\max}\}$ is a weight function; we write $\varphi(X) = \sum_{x \in X} \varphi(x)$. To keep the notation of the paper, the weights are part of the instance in unary, so that the input length N is at least $n + \varphi_{\max}$ and O^* hides factors polynomial in $n + \varphi_{\max}$. Vertex-Weighted Connected Dominating Set asks for $\gamma_c^{\varphi}(G) = \min\{\varphi(S) : S \text{ a connected dominating set of } G\}$, and Weighted Max Leaf Spanning Tree asks for the maximum of $\varphi(\text{leaves}(F))$ over the spanning trees F of G, where leaves(F) is the set of leaves of F.

As in the unweighted case, for $n \geq 3$ the maximum leaf weight of a spanning tree equals $\varphi(V) - \gamma_c^{\varphi}(G)$ (Lemma D.11.4).

What changes. The extension now has to find a completion of least weight. This is the minimum weight of a solution of the state J_X of (12), and the directed Steiner encoding of Lemma 5.6, which already allows weights on the blue vertices, computes it in time $2^{|\mathrm{T}(J_X)|}(n + \varphi_{\max})^{O(1)}$ and polynomial space. So the running-time exponent is still $|\mathrm{T}(J_X)|$, and it does not involve the weights. A completion of least weight may be large, so this extension does not control the number of added vertices; the moment and derandomized theorems need only two properties of the extension, which it has (Lemma D.11.5). The algorithms run the analyses of Theorems 7.2 and 10.4 for every weight budget and every solution size (Lemma D.11.6). The tree potential and the certificate (19) concern the solution and not its weight, so they hold unchanged; a minimum-weight solution itself has some size s^*, at which the run for the optimal budget succeeds.

Theorem 12.3 (Vertex-weighted Connected Dominating Set and weighted MaxLeaf)**.** *For connected n-vertex graphs with vertex weights in $\{0, \ldots, \varphi_{\max}\}$, the minimum weight of a connected dominating set, and the maximum leaf weight of a spanning tree (by Lemma D.11.4 for $n \geq 3$; for $n \leq 2$ it is 0 if $n = 1$ and $\varphi(V)$ if $n = 2$), can be computed*

(a) by a randomized algorithm with one-sided error in time $O(1.7204^n)(n + \varphi_{\max})^{O(1)}$ and polynomial space;

(b) deterministically in time $(\sqrt{5} - \frac{1}{2})^n\, 2^{O(n(\log\log n)^2/\log n)}(n + \varphi_{\max})^{O(1)} = O(1.7361^n)(n + \varphi_{\max})^{O(1)}$ and polynomial space.

The bases are those of Theorems 7.2 and 10.4. The proof is in Appendix D.11.3; Remark 12.6 lists what does not transfer.

Related work. The measure-and-conquer algorithms for Connected Dominating Set [FGK08, AKML11] are stated for the cardinality version. With nonnegative weights a minimum-weight connected dominating set can be chosen inclusion-wise minimal, so enumerating all minimal connected dominating sets solves the vertex-weighted problem, and by Lemma D.11.4 weighted MaxLeaf as well. Lokshtanov, Pilipczuk and Saurabh [LPS18] enumerate them in time $2^{(1-\varepsilon)n} n^{O(1)}$ for some $\varepsilon > 10^{-50}$, and [AKFG+25] in time $O^*(1.9896^n)$ and polynomial space. This gives $O^*(1.9896^n)$ for both weighted problems; we know of no faster exact algorithm for them.

12.3 Weakly connected domination

For a set $S \subseteq V$ of a graph G, the *weakly induced subgraph* $\langle S\rangle_{\mathrm{w}}$ is the subgraph of G with vertex set $N[S]$ whose edges are the edges of G with at least one end in S; for a dominating set S it is a spanning subgraph. A *weakly connected dominating set* is a dominating set S for which $\langle S\rangle_{\mathrm{w}}$ is connected; see [DGH+97]. A graph has one if and only if it is connected (V is one), and every connected dominating set is one. Throughout, G is connected with $n \geq 2$ vertices, and $\gamma_{\mathrm{w}}(G)$ is the minimum size of a weakly connected dominating set. For a set $X \subseteq V$ write $G^2[X]$ for the graph on X in which two vertices are adjacent when their distance in G is at most 2.

What changes. Weak connectivity changes the running-time exponent. The number of terminals $|\mathrm{T}(J_X)| = n - |N[X]| + \mathrm{comp}(G[X])$ is replaced by

$$g_{\mathrm{w}}(X) = n - |N[X]| + \mathrm{comp}(G^2[X]),$$

because a set is weakly connected dominating exactly when it is dominating and $G^2[S]$ is connected (Lemma D.11.7); a red–blue graph whose blue copies are adjacent at distance at most 2 turns the completions of X into the solutions of a state, and Lemma D.11.1 gives an extension in time $2^{g_{\mathrm{w}}(X)}n^{O(1)}$ and polynomial space (Lemma D.11.8). The pointwise certificate charges each edge of a spanning tree of $\langle S\rangle_{\mathrm{w}}$ to an end in S (Lemma D.11.9); it has $\alpha = n$ and total weight $\Xi = n - 1$ in place of $\alpha = n - 1$ and $\Xi = n - 2$, and the running time of the integer-affine algorithm is the same for both pairs (Lemma 10.2).

Theorem 12.4 (Weakly Connected Dominating Set)**.** *For connected n-vertex graphs, $\gamma_{\mathrm{w}}(G)$ can be computed deterministically in time $O((\sqrt{5}-\frac{1}{2})^n\, 2^{O(n(\log\log n)^2/\log n)}) = O(1.7361^n)$ and polynomial space.*

The bound rests on Lemma 10.1 and matches the deterministic bound for Connected Dominating Set (Theorem 10.4). The proof is in Appendix D.11.4; Remark 12.6 lists what does not transfer.

Related work. Weakly connected domination is studied in structural graph theory and, for distributed and approximation algorithms, in wireless networks. We know of no published exact algorithm for it; a reduction to Connected Red-Blue Dominating Set and the algorithm of Abu-Khzam, Mouawad and Liedloff [AKML11] give $O^*(1.8619^n)$ (Appendix D.11.4), the only bound below 2^n known to us.

12.4 Directed Max Leaf Spanning Tree

Let $\vec{G}$ be a digraph without loops or parallel arcs (opposite arcs are allowed) with $n \geq 2$ vertices. An *out-branching* rooted at r is a spanning subdigraph that is a tree in which every arc points away from r; its *leaves* are its vertices of out-degree 0. Directed Max Leaf Spanning Tree asks for the maximum number of leaves of an out-branching. Binkele-Raible and Fernau [BRF12] solve it in time $O^*(1.9044^n)$ and polynomial space, and in time $O^*(1.8139^n)$ with exponential space. With the number k of leaves as the parameter, [AFG+09] decide in time $2^{O(k\log^2 k)}n^{O(1)}$ whether a strongly connected digraph has an out-branching with at least k leaves. For all digraphs, Kneis, Langer and Rossmanith [KLR11] decide this in time $4^k n^{O(1)}$, and [DGKY10] in time $3.72^k n^{O(1)}$, which also gives $O(1.9973^n)$ for Directed Max Leaf Spanning Tree. [BRFF+12] give a kernel with $O(k^3)$ vertices when the root is part of the input, and show that without the root there is no polynomial kernel unless coNP $\subseteq$ NP/poly.

For $r \in V(\vec{G})$, a set $S \subseteq V(\vec{G})$ is *r-internal* if $r \in S$, every vertex of S is reachable from r in $\vec{G}[S]$, and every $v \notin S$ has an in-neighbor in S.

By Lemma D.11.11, the maximum number of leaves of an out-branching is $n - \min|S|$, the minimum over all roots r and all r-internal sets S.

What changes. For a fixed root r, the role of the number of terminals is taken by $g_r(X) = |\operatorname{und}_r(X)| + |\operatorname{Src}_r(X)|$, which counts the vertices outside $X \cup \{r\}$ that have no in-neighbor in X, and the source strong components of $\vec{G}[X]$ that do not contain r. The extension problem is an instance of Directed Steiner Tree on a digraph with a blue and a red copy of every vertex, and the extension algorithm of Lemma 12.7 solves it in time $2^{g_r(X)}n^{O(1)}$ and polynomial space (Lemma D.11.12). The certificate is the directed form of the tree inequality: in an out-branching the weight $d_F(u) - 1$ of Lemma 5.10 becomes the number of children, minus 1 at the root, and with these weights the certificate (19) holds with the same constants $\alpha = n - 1$ and $\Xi = n - 2$ (Lemma D.11.13). So the algorithm of Theorem 10.4 applies for each root.

Theorem 12.5 (Directed Max Leaf Spanning Tree)**.** *For digraphs with n vertices, the maximum number of leaves of an out-branching (or the fact that there is none) can be computed deterministically in time $O((\sqrt{5} - \frac{1}{2})^n\, 2^{O(n(\log\log n)^2/\log n)}) = O(1.7361^n)$ and polynomial space.*

The bound rests on Lemma 10.1. The proof is in Appendix D.11.5.

Remark 12.6 (What does not transfer). The closure sampling RSC of Section 8 is claimed for none of the four variants: its closure rule and its penalty for false candidates use the neighborhoods $N(u)$ of vertices outside the sample, which an anchor at distance $r \geq 2$, a weight or an arc direction do not respect; we leave open whether a variant of the closure rule gives 1.6991^n for Connected r-Dominating Set. The exponential-moment analysis of RS-θ (Section 7) is claimed for $\mathcal{N}$-domination and for weights (Theorem 12.2(a) and Theorem 12.3(a)). For weak connectivity we do not expect it to beat the affine certificate of Lemma D.11.9: on spiders that certificate is an equality, and there the moment bound appears to decrease toward $\sqrt{5} - \frac{1}{2}$ (Proposition D.11.10 and the discussion after it). For out-branchings the tree potential of Lemma 7.3 does not apply, since it uses that a sampled vertex dominates its parent in the spanning tree of the solution, while in an out-branching a vertex dominates only its children. The equivalence with Max Leaf Spanning Tree (Lemma 5.1) is special to $r = 1$.

12.5 Directed Steiner Tree

An instance of Directed Steiner Tree is a digraph $\vec{G}$ with n vertices, a root r and a set $K \subseteq V(\vec{G}) \setminus \{r\}$ of terminals. Let $U = V(\vec{G}) \setminus (\{r\} \cup K)$ be the set of non-terminals and $n_U = |U|$. A set $S \subseteq U$ is a *Steiner set* if every terminal is reachable from r in $\vec{G}[\{r\} \cup K \cup S]$; the task is to find a Steiner set of least size. Two algorithms are immediate: inclusion–exclusion over the terminals (Lemma 5.3, with unit weights; see also [MPR$^+$12] and [JLR$^+$17, Lemma 3.5]) takes time $2^{|K|}n^{O(1)}$, and enumeration takes time $2^{n_U}n^{O(1)}$. Since $|K| + n_U = n - 1$, the better of the two runs in time $2^{(n-1)/2}n^{O(1)}$, and we know of no better bound in terms of n for digraphs. Feldmann and Marx [FM23] classify the directed Steiner network problems that are fixed-parameter tractable in the number of terminals. For undirected graphs and unit weights, [FGK$^+$13, Theorem 5] solve Steiner Tree in time $O(1.3533^n)$ and polynomial space, which is below the bound of Theorem 12.8 on this special case; their branching uses a contraction lemma for adjacent terminals that is proved for undirected graphs.

The sampling algorithm works on rooted Steiner sets. Call $S \subseteq U$ *rooted* if every vertex of $K \cup S$ is reachable from r in $\vec{G}[\{r\} \cup K \cup S]$, and let Φ_{DST} be the implicit set system with universe

U whose solutions are the rooted Steiner sets. Every rooted set is a Steiner set, and the vertices of a Steiner set S that are reachable from r in $\vec{G}[\{r\} \cup K \cup S]$ form a rooted Steiner set; so the least size of a solution of Φ_{DST} is the optimum. In particular, every inclusion-minimal Steiner set is rooted, so the restriction to rooted sets loses no minimum solution; it makes the running-time exponent of a sample, defined next, a count that can be read off the digraph.

The running-time exponent of a sample. For $X \subseteq U$ let $\bar{K}_X = \{r\} \cup K \cup X$ and let $\bar{K}^r_X$ be the set of vertices reachable from r in $\vec{G}[\bar{K}_X]$. A *source component* of $\vec{G}[\bar{K}_X \setminus \bar{K}^r_X]$ is a strong component that no arc of $\vec{G}[\bar{K}_X \setminus \bar{K}^r_X]$ enters from outside. For the instance $I = (\vec{G}, r, K)$, a set $X \subseteq U$ and $k \geq 0$, let $g(I, X, k)$ be the number of source components of $\vec{G}[\bar{K}_X \setminus \bar{K}^r_X]$, and 0 if $\bar{K}_X = \bar{K}^r_X$. It does not depend on k, and for fixed I we write $g(X)$, in agreement with the shorthand of Definition 3.2. Every vertex of $\bar{K}_X \setminus \bar{K}^r_X$ is reachable within $\vec{G}[\bar{K}_X \setminus \bar{K}^r_X]$ from some source component, since the condensation of a digraph is acyclic.

Lemma 12.7 (Extension)**.** *There is a deterministic algorithm that, given $(\vec{G}, r, K)$, a set $X \subseteq U$ and weights $w\colon U \to \{0, 1, \ldots, w_{\max}\}$, computes the least weight $w(Y)$ over the sets $Y \subseteq U \setminus X$ for which $X \cup Y$ is a rooted Steiner set (∞ if there is none), in time $2^{g(X)}(n + w_{\max})^{O(1)}$ and polynomial space. With unit weights it decides, for every integer $k \geq 0$, whether some $Y \subseteq U \setminus X$ with $|Y| \leq k - |X|$ makes $X \cup Y$ a rooted Steiner set, in time $2^{g(X)}n^{O(1)}$. Moreover, $g(X)$ is computable in polynomial time, so g is a running-time exponent for this algorithm for Φ_{DST}-Extension.*

Proof. The set $\bar{K}^r_X$, the strong components of $\vec{G}[\bar{K}_X \setminus \bar{K}^r_X]$ and the arcs between them can be computed in polynomial time, and hence so can $g(X)$. Choose one vertex in each source component of $\vec{G}[\bar{K}_X \setminus \bar{K}^r_X]$, and let Q be the set of chosen vertices, so $|Q| = g(X)$. Give every vertex of $\bar{K}_X$ the weight 0 and every vertex $v \in U \setminus X$ the weight $w(v)$, and compute OPT, the least weight of a set $D \subseteq V(\vec{G}) \setminus (\{r\} \cup Q)$ such that every vertex of Q is reachable from r in $\vec{G}[\{r\} \cup Q \cup D]$, by Lemma 5.3 (root r, terminals Q, weights in $\{0, \ldots, w_{\max}\}$). This takes time $2^{g(X)}(n + w_{\max})^{O(1)}$ and polynomial space. We show that OPT is the least $w(Y)$ over the sets Y in question; with unit weights it is the least $|Y|$, and one accepts if and only if $\mathrm{OPT} \leq k - |X|$.

Let $X \cup Y$ be rooted, with $Y \subseteq U \setminus X$. Then every vertex of $\bar{K}_X$ is reachable from r in $\vec{G}[\bar{K}_X \cup Y]$, and $D = (\bar{K}_X \cup Y) \setminus (\{r\} \cup Q)$ is feasible, of weight $w(Y)$. Conversely, let D be feasible, and put $Y = D \cap (U \setminus X)$, so that $w(Y)$ is the weight of D and $\{r\} \cup Q \cup D \subseteq \bar{K}_X \cup Y$. In $\vec{G}_Y = \vec{G}[\bar{K}_X \cup Y]$ every vertex of Q is reachable from r. Each source component contains a vertex of Q and is strongly connected within $\vec{G}[\bar{K}_X]$, so it is reachable in $\vec{G}_Y$; every other vertex of $\bar{K}_X \setminus \bar{K}^r_X$ is reachable from a source component within $\vec{G}[\bar{K}_X \setminus \bar{K}^r_X]$; and $\bar{K}^r_X$ is reachable by definition. So all of $\bar{K}_X$ is reachable from r in $\vec{G}_Y$. Let $Y' \subseteq Y$ be the set of vertices of Y reachable from r in $\vec{G}_Y$. A path from r in $\vec{G}_Y$ visits only reachable vertices, so every vertex of $\bar{K}_X \cup Y'$ is reachable from r in $\vec{G}[\bar{K}_X \cup Y']$, that is, $X \cup Y'$ is rooted, and $w(Y') \leq w(Y)$, as weights are nonnegative. □

The analysis. The proof of the following theorem is in Appendix D.11.6. It fixes an inclusion-minimal rooted Steiner set S and an out-arborescence A of $\vec{G}[\{r\} \cup K \cup S]$ rooted at r, whose leaves other than r are terminals. The running-time exponent $g(X)$ of a sample $X \subseteq S$ is at most the number of trees of the forest $A[\bar{K}_X]$ that do not contain r (Lemma D.11.14). Whether a vertex is counted depends only on whether it and its parent lie in $\bar{K}_X$, so the resulting lower bound on the generating function is computed by a two-state recursion along A (Lemma D.11.15). As for RS-θ (Lemma D.4.2), a region of the plane that the recursion maps into itself, up to known scale factors, bounds the generating function for all arborescences at once (Lemma D.11.16); three such regions,

combined with rational weights, are checked by computer (Proposition D.11.17). The finishing step uses Lemma 5.3.

Theorem 12.8 (Directed Steiner Tree)**.** Directed Steiner Tree *on digraphs with n vertices can be solved by a randomized algorithm with one-sided error in time $2^{0.457n}n^{O(1)} = O(1.3727^n)$ and polynomial space.*

Numerically, the analysis appears close to tight for this algorithm: on a family of digraphs built from a directed path with pendant terminals, the bound that the generating function gives is $\Omega(1.3716^n)$, against the proved $2^{0.457n} < 1.3727^n$ (Appendix F.7).

Vertex weights. With polynomially bounded integer weights on the non-terminals, the same algorithm computes the least weight of a Steiner set within the same time bound, up to a factor polynomial in $w_{\max}$ (Corollary D.11.20, proved in Appendix D.11.8).

12.6 Paired Domination

Let G be a graph with n vertices and no isolated vertex. A *paired dominating set* is a dominating set S such that $G[S]$ has a perfect matching, and $\gamma_{\mathrm{pr}}(G)$ is the least size of one; it exists, as the vertex set of a maximal matching is one, since G has no isolated vertex. [ABK$^+$24] give an exact algorithm in time $O(1.7159^n)$ and polynomial space; 1.7159^n is also the bound of [FGPS08] on the number of minimal dominating sets. We use the implicit set system Φ_{PD} with universe V whose solutions are the paired dominating sets, and for $X \subseteq V$ we write $\mathrm{und}(X) = V \setminus N[X]$ for the set of vertices that X leaves undominated.

The finishing step. The undominated vertices are handled by inclusion–exclusion and the perfect matching by the Tutte matrix [Tut47]. Fix an orientation of the edges of G, and let $\mathbf{T}$ be the skew-symmetric $V \times V$ matrix with $\mathbf{T}_{uv} = x_e = -\mathbf{T}_{vu}$ for every edge $e = uv$ oriented from u to v, where the x_e are distinct indeterminates, and with all other entries 0. For $A \subseteq V$ let $\mathbf{T}_A$ be its principal submatrix on A. Then $\det \mathbf{T}_A = \mathrm{Pf}(\mathbf{T}_A)^2$, and the Pfaffian is $\mathrm{Pf}(\mathbf{T}_A) = \sum_M \pm x_M$, summed over the perfect matchings M of $G[A]$, where $x_M = \prod_{e\in M} x_e$ and $\mathrm{Pf}(\mathbf{T}_\emptyset) = 1$. Distinct matchings give distinct monomials.

Lemma 12.9 (Tutte-matrix finishing)**.** *There is a randomized algorithm that, given G, weights $w\colon V \to \{0,\dots,w_{\max}\}$, a budget $\varpi \geq 0$, $X \subseteq V$ and $k \geq 0$, never accepts if no $Y \subseteq V \setminus X$ with $|Y| \leq k - |X|$ makes $X \cup Y$ a paired dominating set of weight at most ϖ, accepts with probability at least $\frac{1}{2}$ otherwise, and runs in time $2^{|\,\mathrm{und}(X)|}(n + w_{\max})^{O(1)}$ and polynomial space for every outcome of its coins. With $w \equiv 0$ and $\varpi = 0$ this is an algorithm for Φ_{PD}-Extension with the same error bounds, in time $2^{|\,\mathrm{und}(X)|}n^{O(1)}$.*

So $g(I, X, k) = |\,\mathrm{und}(X)| = n - |N[X]|$, for the instance $I = G$ (with or without weights), is a running-time exponent, with integer values in $\{0,\dots,n\}$, for a one-sided-error extension algorithm as in Corollary 3.15; it does not depend on k, and we write $g(X)$ as in Definition 3.2. It is the number of undominated vertices, the exponent of the inclusion–exclusion finishing step for Dominating Set: the perfect-matching requirement does not increase it.

The analysis. The proofs of Lemma 12.9 and of the next two statements are in Appendix D.11.7. They fix a paired dominating set S, a perfect matching M of $G[S]$ and, for every vertex outside S, an anchor in S among its neighbors. A sample $X \subseteq S$ dominates a vertex outside S once it

contains its anchor, and a vertex of S once it contains that vertex or its partner in M; so the lower bound on the generating function that this gives factorizes over the pairs of M (Lemma D.11.18). A pair responsible for $\bar{n}$ vertices, the pair itself and the vertices anchored at it, contributes at most $\bar{n}\, h_{\bar{n}}(y)$ to $\log\bigl((1+y)^n / W_S(y)\bigr)$, where for an integer $\bar{n} \geq 2$

$$h_{\bar{n}}(y) = \log(1+y) - \frac{1}{\bar{n}} \log(y^2 + 4 \cdot 2^{-\bar{n}/2} y + 2^{-\bar{n}}).$$

Let $y^* = (3\sqrt{3} - 1)/16 = 0.2622595\ldots$ and $\psi_{\mathrm{pr}} = h_6(y^*) = 0.7050097\ldots$. The worst pairs are responsible for six vertices (Lemma D.11.19), as is a pair uv with two vertices anchored at each of u and v, as in a double star, and y^* is the minimizer of h_6 (Proposition 12.11).

Theorem 12.10 (Paired Domination)**.** *For graphs G with n vertices and no isolated vertex, $\gamma_{\mathrm{pr}}(G)$ can be computed by a randomized algorithm with one-sided error in time $O(1.6302^n)$ and polynomial space.*

The only randomness in the extension algorithm of Lemma 12.9 is the identity test by the Schwartz–Zippel lemma. A deterministic version cannot rely on generic identity testing alone: Kabanets and Impagliazzo [KI04] show that a deterministic polynomial-time identity test for polynomials given by arithmetic circuits would imply circuit lower bounds that are open.

We show next that this analysis of Algorithm MLS-θ with the running-time exponent $|\operatorname{und}(X)|$ is tight.

Proposition 12.11 (Tightness)**.** *The only critical point of h_6 in $(0, \infty)$ is y^*, and $\min_{y \geq 0} h_6(y) = h_6(y^*)$. Let G_L be the disjoint union of L double stars, each an edge uv with two pendant vertices at u and two at v. Then the centers form the only paired dominating set S^* of G_L of size at most $2L$, and if Algorithm MLS-θ of Theorem 12.10 is run with a rate $\psi' < \psi_{\mathrm{pr}}$, it accepts $k = 2L$ with probability tending to 0 as $L \to \infty$.*

So the base $2^{\psi_{\mathrm{pr}}} = 1.63015\ldots$ is exact for Algorithm MLS-θ with the running-time exponent $|\operatorname{und}(X)|$; a better bound needs a different running-time exponent or extension algorithm.

Vertex weights. With vertex weights in $\{0, \ldots, w_{\max}\}$, written in unary, the least weight of a paired dominating set can be computed within the same time bound, up to a factor polynomial in $w_{\max}$ (Theorem D.11.21, proved in Appendix D.11.8).

13 Conclusion and open problems

Monotone Local Search needs only large exponential moments $\mu_t(S)$ of the running-time exponent over the subsets of one solution (Definition 3.3 and Theorem 3.5), rather than a worst-case running-time bound for every sample of a given size; the criterion is necessary for Algorithm MLS-θ (Proposition 3.10), the algorithm is optimal within the class of Theorem 3.12, and for product-form running-time exponents it certifies the multivariate bound and, up to polynomial factors, nothing better (Corollary 4.6 and Proposition 4.5). The gains come from extension algorithms whose running time depends on the structure of the sample, such as the number of Steiner terminals or of components of the sample. Several questions remain open.

1. *Derandomization.* The black-box barrier of Section 9 rules out a general derandomization, and balanced set-inclusion families derandomize only the bounds that come from integer affine certificates (Section 10). Can RS-θ or RSC be derandomized with a factor $2^{o(n)}$? Their analyses

rest on tests along an unknown spanning tree of the solution; a counting argument shows that against every family of $2^{O(n)}$ partitions into blocks of size $n^{o(1)}$ some such tree has almost all of its edges between blocks, so these tests are not separable over the blocks of a product construction (Proposition D.7.7).

2. *Better trials for Connected Dominating Set.* Algorithm MLS-θ is optimal within the class of Theorem 3.12, so with the same extension algorithm and running-time exponent further gains must come from trials outside this class: a non-uniform sampling distribution (Appendix A), a rule that adds vertices to the sample, as RSC does, or an accounting that charges a trial only for the extension call that it actually makes. For Connected Dominating Set we do not know whether any of these improves on Theorem 8.1.

3. *Component-count running times.* For $c = 3$ the base 1.8 is exact for Algorithm MLS-θ (Propositions 11.9 and 11.10), so further progress for Connected Feedback Vertex Set needs a different extension algorithm or a different trial. For $c = 2$ the proved base 1.6444 is within $2 \cdot 10^{-4}$ of the value $2^{\psi_2} \approx 1.64429$ that Theorem 3.5 certifies for a solution whose spanning tree is a path (Section 11.2), but we have no family of instances of Graph Motif or Connected Vertex Cover on which this value is tight. No deterministic algorithm for Connected Feedback Vertex Set in time $O((2-\varepsilon)^n)$ is known to us; it would need to derandomize both the sampling and the Cut & Count step. Nor do we know of an analysis that bounds a natural branching extension algorithm (branching on whether a vertex on a cycle joins the solution, followed by a Steiner phase) below 2^n.

4. *Closure rules elsewhere.* Does a closure rule, as in RSC, lower the running time for Directed Steiner Tree, or give $O(1.6991^n)$ for Connected r-Dominating Set with $r \geq 2$ (Remark 12.6)?

5. *Other problems and enumeration.* Our method applies whenever the running time of the extension algorithm is exponential in a polynomial-time computable structural parameter of the sample whose exponential moments over the subsets of a solution are large. It would be interesting to characterize the problems for which Theorem 3.5 is exponentially better than every bound from a product-form running-time exponent. [FGLS19] also turn product-form Monotone Local Search into enumeration algorithms; we have not studied enumeration with sample-dependent running times.

Acknowledgments

This research was supported by the Australian Government through the Australian Research Council's Discovery Projects funding scheme (project DP210103849). This research includes computations using the computational cluster Katana supported by Research Technology Services at UNSW Sydney [PVC10], which was used to build and check the certificate for Theorem 8.1 (Appendix F). We thank the UNSW AI Institute for supporting the co-algorithmist framework, a framework for human–AI research collaboration in algorithms and theoretical computer science that is under development.

A Other levers

This appendix extends Theorem 3.5 to every samplable distribution and examines two changes of the class of Theorem 3.12, whose proof is in Appendix D.1. Sampling from a non-uniform distri-

bution can lower the running time by an exponential factor (Proposition A.5), whereas sampling in several adaptive stages cannot (Proposition A.6). Neither has yet given a proven improvement of the running-time bound for Connected Dominating Set, and no result of the main text depends on this appendix. We keep the notation of Section 3, in particular the extendable sets $\mathcal{E}$, the counts $q_\theta(t)$ and the class optimum π^* of Definition 3.11.

A.1 Samplable distributions

Uniform t-sets are one choice of sampling distribution. The theorem extends to every distribution that can be sampled in polynomial time; the success criterion is then the exponential moment of the distribution over the extendable sets of Definition 3.11. For a distribution $\mathcal{X}$ on subsets of $U = U_I$ define

$$M_{\mathcal{X}} = \sum_{X \in \mathcal{E}} \mathcal{X}(X)\, 2^{-g(X)}.$$

Algorithm A.1 (MLS-θ for a distribution $\mathcal{X}$)

Input: an instance I of Φ and an integer $k \le n$.

Parameters: ψ, Γ, κ and a distribution $\mathcal{X} = \mathcal{X}_{I,k}$ on subsets of U_I.

Take a cap Θ and trial counts m_θ as in (2) (with a single index in place of t). For every integer $\theta \in \{0, \dots, \Theta\}$, repeat m_θ times: draw $X \sim \mathcal{X}$; if $|X| \le k$ and $g(X) \le \theta$, run $\mathcal{A}$ on (I, X, k), and answer YES if it accepts. Answer NO after all trials.

Theorem A.2 (Exponential-moment Monotone Local Search for samplable distributions)**.** *Let $\mathcal{A}$ solve Φ-Extension with running-time exponent g, and let $\mathcal{X} = \mathcal{X}_{I,k}$ be a distribution on subsets of U_I that can be sampled exactly in time $N^{O(1)}$. Let ψ, Γ, κ be as in Theorem 3.5. Then Algorithm A.1 is a randomized algorithm that runs in time $2^{\psi n} N^{O(1)}$ and in space $N^{O(1)}$ plus the space of $\mathcal{A}$ and of the sampler, answers* NO *if $\mathcal{F}_I$ contains no set of size at most k, and answers* YES *with probability at least $1 - e^{-\kappa}$ if*

$$M_{\mathcal{X}} \;\ge\; \frac{2^{-\psi n}}{\Gamma},$$

in particular if $\sum_{X \subseteq S} \mathcal{X}(X) 2^{-g(X)} \ge 2^{-\psi n}/\Gamma$ for some $S \in \mathcal{F}_I$ with $|S| = k$.

Proof. This is Theorem 3.14 with $\mathcal{T}$ a single index, the trial of Algorithm A.1 as Tr_θ, and $\xi = g(X)$ if $X \in \mathcal{E}$ and $\xi = \infty$ otherwise, for $X \sim \mathcal{X}$: a trial takes time $N^{O(1)}$ plus at most $2^\theta N^{O(1)}$ ((T1)), an acceptance of $\mathcal{A}$ certifies a solution of size at most k ((T2)), and on the event $\xi \le \theta$ we have $X \in \mathcal{E}$, hence $|X| \le k$ and acceptance by $\mathcal{A}$ ((T3)). Finally $\mathbb{E}[2^{-\xi}] = M_{\mathcal{X}}$, and the sufficient condition holds because every $X \subseteq S$ lies in $\mathcal{E}$. □

The proof uses only that $\mathcal{A}$ accepts on $\mathcal{E}$. The same statement therefore holds in *event form*, with $\mathcal{E}$ replaced by any set of samples on which a possibly different extension algorithm is guaranteed to accept, provided that every acceptance certifies the desired output. If $\mathcal{X}$ is uniform on $\binom{U}{t}$, then $M_{\mathcal{X}} = \mu_t(\mathcal{E})/\binom{n}{t} \ge \mu_t(S)/\binom{n}{t}$, which gives item (iii) of Theorem 3.5 for the algorithm that runs Algorithm A.1 once for each $t \le k$: the run for a value of t that attains the maximum in $(\star)$ alone succeeds with probability at least $1 - e^{-\kappa}$, and the $k+1$ runs multiply the running time by $k + 1 \le n + 1$. If every element is included independently with probability $y/(1+y)$, for a rational $y \ge 0$, then $\mathcal{X}(X) = y^{|X|}/(1+y)^n$ and $M_{\mathcal{X}} \ge W_S(y)/(1+y)^n$, which gives the generating-function condition of Theorem 3.5 directly. The Bernoulli closure of Algorithm RSC (Section 8) is an instance whose distribution is not a product distribution.

A.2 A bound on the success probability

The lower bounds below compare algorithms by their total *charge*, a running-time bound that is fixed for each trial. The following bound is the counterpart of Theorem 3.12(a) for arbitrary trials.

Lemma A.3 (Charge bound)**.** *Let $\pi \geq 0$. Consider an algorithm that performs trials one after another and answers* YES *at the first success, where trial i is any randomized procedure, chosen as a possibly randomized function of the past, that succeeds or fails and is charged an amount $\mathrm{chg}_i \geq 1$ determined by its outcome. Suppose that, conditioned on the history before trial i (including the choice of the trial), trial i succeeds with probability at most $\pi\,\mathbb{E}[\mathrm{chg}_i \mid \textit{history}]$, and that the total charge never exceeds B'. Then the algorithm answers* YES *with probability at most $\pi\mathsf{B}'$.*

Proof. Let $\mathcal{G}_{i-1}$ be the history before trial i together with the choice of trial i, let $s_i = \Pr[\text{trial } i \text{ succeeds} \mid \mathcal{G}_{i-1}]$ and $c_i = \mathbb{E}[\mathrm{chg}_i \mid \mathcal{G}_{i-1}]$, and let $i_{\max}$ be the number of trials. Every charge is at least 1, so $i_{\max} \leq \mathsf{B}'$, and $\{i \leq i_{\max}\}$ is determined by $\mathcal{G}_{i-1}$. By hypothesis $s_i \leq \pi c_i$, so the union bound gives

$$\Pr[\textsc{yes}] \leq \mathbb{E}\Big[\sum_{i \leq i_{\max}} s_i\Big] \leq \pi\,\mathbb{E}\Big[\sum_{i \leq i_{\max}} c_i\Big] = \pi\,\mathbb{E}\Big[\sum_{i \leq i_{\max}} \mathrm{chg}_i\Big] \leq \pi\mathsf{B}',$$

where the equality holds term by term, since $\mathbb{E}[\mathbf{1}[i \leq i_{\max}]\mathrm{chg}_i] = \mathbb{E}[\mathbf{1}[i \leq i_{\max}]c_i]$ by the tower property. □

A.3 Non-uniform sampling

Sampling from a non-uniform distribution can lower the running time when the instance exposes classes of elements on which including an element in the sample has different effects on the probability that the sample lies in a solution and on the running-time exponent. A synthetic family below shows that this can change the worst-case exponent; its analysis uses product distributions.

For a product distribution $\pi_p(X) = \prod_{u \in X} p_u \prod_{u \notin X}(1 - p_u)$ with rational p_u computed from the instance, the condition of Theorem A.2 becomes $W_S^p := \sum_{X \subseteq S} \pi_p(X) 2^{-g(X)} \geq 2^{-\psi n}/\Gamma$ for some $S \in \mathcal{F}_I$ with $|S| = k$. For running-time exponents that are sums of per-element terms on subsets of S, namely $g(X) = \gamma + \sum_{u \in S \setminus X} a_u + \sum_{u \in X} b_u$ for $X \subseteq S$, this sum factorizes. Writing $\pi_p(X) = \prod_{u \notin S}(1-p_u) \cdot \prod_{u \in S} \pi_u(X)$ with $\pi_u(X) = p_u$ if $u \in X$ and $1 - p_u$ otherwise, and expanding the product over $u \in S$ according to whether $u \in X$, gives

$$W_S^p = 2^{-\gamma} \prod_{u \notin S}(1 - p_u) \prod_{u \in S}\big((1 - p_u)2^{-a_u} + p_u 2^{-b_u}\big). \tag{23}$$

In the following family the classes are given with the instance.

Definition A.4 (Two-class family)**.** Let n be even. The *two-class family* consists of instances that expose a partition $U = U_A \sqcup U_B$ with $|U_A| = |U_B| = n/2$ and have a unique solution S, so that the extendable sets are exactly the subsets of S; put $k_A = |S \cap U_A|$ and $k_B = |S \cap U_B|$. The extension algorithm takes two size bounds (ℓ_A, ℓ_B), one for each class, and, for this pair, has running-time exponent

$$g(X) = (\ell_A - |X \cap U_A|) \log c_A + (\ell_B - |X \cap U_B|) \log c_B$$

on samples X with $|X \cap U_A| \leq \ell_A$ and $|X \cap U_B| \leq \ell_B$, with $(c_A, c_B) = (2, 4)$. Algorithms for the family try all of the polynomially many pairs of size bounds.

Proposition A.5 (A two-class family)**.** *Let n be even, and consider the two-class family of Definition A.4. Then:*

(a) *the product distribution with $p_u = 1/3$ on U_A and $p_u = 3/7$ on U_B gives, by Theorem A.2, time $O^*((3/2)^{n/2}(7/4)^{n/2}) = O^*(\sqrt{21/8}^{\,n})$, with $\sqrt{21/8} = 1.6201\ldots$, on every instance of the family;*

(b) *on the instance with $S = U_B$, every algorithm that, trial by trial, chooses t, θ and a pair of size bounds, draws X uniformly from $\binom{U}{t}$, runs the extension algorithm when its running-time exponent is at most θ, and is charged 2^θ per trial, needs total charge at least $\sqrt{3}^{\,n}/(2(n+1))$ for success probability $1/2$; here $\sqrt{3} = 1.7320\ldots$.*

Proof. (a) Fix the correct pair of size bounds (k_A, k_B), and write p_j for the probability on class U_j, $j \in \{A, B\}$. On subsets of S the running-time exponent is a sum of per-element terms, with $a_u = \log c_j$ for $u \in S \cap U_j$, $b_u = 0$ and $\gamma = 0$. The factorization (23) gives

$$W_S^p = \prod_{j\in\{A,B\}} (1-p_j)^{n/2-k_j} \Big(\frac{1-p_j}{c_j} + p_j\Big)^{k_j}.$$

With $p_j = y_j/(1+y_j)$ this equals $\prod_j (1+y_j)^{-n/2} c_j^{-k_j} (1+y_j c_j)^{k_j}$. At $y_j = (c_j-1)/c_j$, that is $p_A = 1/3$ and $p_B = 3/7$, we have $1 + y_j c_j = c_j$, so $W_S^p = \prod_j (1+y_j)^{-n/2} = \prod_j (2 - 1/c_j)^{-n/2} = (21/8)^{-n/2}$ for every (k_A, k_B). Since the extendable sets are the subsets of S, $M_{\mathcal{X}} = W_S^p$, and Theorem A.2 with the correct pair of size bounds, $\Gamma = 1$ and $2^{\psi n} = (21/8)^{n/2}$ gives the claim.

(b) Here $k_A = 0$ and $k_B = n/2$. A sample is accepted under a pair of size bounds (ℓ_A, ℓ_B) only if it is a subset of U_B and $\ell_B \geq n/2$; bounds above the class sizes are not used, so $\ell_B = n/2$, and the running-time exponent is then at least that of the pair $(0, n/2)$. Let $q_\theta(t)$ and π^* be computed for the pair $(0, n/2)$. A trial with t, θ and a pair of size bounds chosen before it succeeds only if its uniform sample $X \in \binom{U}{t}$ is a subset of S and passes the threshold θ under the pair $(0, n/2)$, so, conditioned on the history, it succeeds with probability at most $q_\theta(t)/\binom{n}{t} \leq \pi^* 2^\theta$, while its charge is $2^\theta \geq 1$. By Lemma A.3 with $\pi = \pi^*$, success probability $1/2$ needs total charge at least $1/(2\pi^*)$. By Theorem 3.12(c), $\pi^* \leq \max_t \mu_t(\mathcal{E})/\binom{n}{t}$, and here $\mathcal{E}_t = \binom{S}{t}$, so $\mu_t(\mathcal{E}) = \mu_t(S) = \binom{n/2}{t} 4^{-(n/2-t)}$ for $0 \leq t \leq n/2$. With $\tau = t/n \in [0, \frac{1}{2}]$, the entropy estimates (1) give

$$\frac{\mu_t(S)}{\binom{n}{t}} \leq (n+1)\, 2^{\frac{n}{2}H(2\tau) - nH(\tau) - n + 2\tau n} = (n+1)\, 2^{-n f_{1/2}(\tau)}, \qquad f_{1/2}(\tau) = H(\tau) - \tfrac{1}{2}H(2\tau) + 1 - 2\tau,$$

where $f_{1/2}$ is the function f_σ of (9) with $(\beta, \rho, \sigma) = (1, 2, \frac{1}{2})$, the same function as in the proof of Corollary 9.3. By Remark 4.4, since $p = 2^{-\rho} = \frac{1}{4} < \sigma$, its minimum over $[0, \frac{1}{2}]$ is $F^*(\frac{1}{2}) = 1 + \frac{1}{2}\log\frac{1}{2} + \frac{1}{2}\log\frac{3}{2} = \frac{1}{2}\log 3$ by (6). Hence $\max_t \mu_t(S)/\binom{n}{t} \leq (n+1)\sqrt{3}^{\,-n}$, and the charge is at least $\sqrt{3}^{\,n}/(2(n+1))$. □

The family is synthetic: a sensible algorithm would solve the two parts separately, in time bounded by the larger of the two running times rather than by their product. Its purpose is to show that, for a fixed extension algorithm, the sampling distribution can change the worst-case exponent of Monotone Local Search. For Connected Dominating Set, where degrees are visible, it is open whether importance sampling helps.

A.4 Multi-stage sampling

A *uniform multi-stage process* starts with $D = \emptyset$ and repeatedly either stops, outputting D, or chooses a size $\ell \geq 1$ and adds a uniform ℓ-subset of $U \setminus D$. Its choices may depend on everything seen so far, including earlier draws, running-time exponents of intermediate sets, its own coins and

even an oracle for $\mathcal{E}$; it may also abort without output. The next proposition shows that no such process outputs a given set more often than a uniform draw of the same size does, so multi-stage trials are neutral.

Proposition A.6 (Multi-stage sampling)**.** *For every uniform multi-stage process and every $Y \subseteq U$, $\Pr[\text{output} = Y] \le 1/\binom{n}{|Y|}$. Consequently, suppose each multi-stage trial is charged 2^θ for a threshold θ fixed before its first stage, and runs $\mathcal{A}$ on its output Y when $|Y| \le k$ and $g(Y) \le \theta$. Then the trial succeeds with probability at most $\sum_{t \le k} q_\theta(t)/\binom{n}{t} \le (k+1)2^\theta\pi^*$, and Theorem 3.12(a) holds for algorithms using such trials with $\mathsf{B}\pi^*$ replaced by $(k+1)\mathsf{B}\pi^*$.*

Proof. We prove by induction on $|U \setminus D|$ that, from any point of the process at which the current set is $D \subseteq Y$, the probability of eventually outputting Y is at most $1/\binom{n-|D|}{|Y \setminus D|}$, whatever the history and the later choices. If the process stops, it outputs Y only if $D = Y$, with probability at most $1 = 1/\binom{n-|D|}{0}$. Suppose it adds a uniform ℓ-subset E of $U \setminus D$, and put $n' = n - |D|$ and $j = |Y \setminus D|$. It can output Y only if $E \subseteq Y \setminus D$, which is impossible if $j < \ell$. Otherwise, by the induction hypothesis applied at $D \cup E$, the probability is at most

$$\sum_{E \in \binom{Y \setminus D}{\ell}} \frac{1}{\binom{n'}{\ell}} \cdot \frac{1}{\binom{n'-\ell}{j-\ell}} = \frac{\binom{j}{\ell}}{\binom{n'}{\ell}\binom{n'-\ell}{j-\ell}} = \frac{1}{\binom{n'}{j}},$$

using $\binom{n'}{\ell}\binom{n'-\ell}{j-\ell} = \binom{n'}{j}\binom{j}{\ell}$. A randomized choice of the action mixes these bounds, which are all at most $1/\binom{n'}{j}$, and aborting only lowers the probability. The process terminates, because $|D|$ grows at every step that does not stop. Taking $D = \emptyset$ gives the first claim. For the consequence, sum the pointwise bound over the extendable passing sets Y, layer by layer, and use $q_\theta(t)/\binom{n}{t} \le 2^\theta\pi^*$ and $t \le k$. The pointwise bound holds whatever the history, so conditioned on the history before a multi-stage trial with threshold θ the trial succeeds with probability at most $(k+1)\pi^*2^\theta$, while its charge is $2^\theta \ge 1$; Lemma A.3 with $\pi = (k+1)\pi^*$ gives the last claim. □

Neutrality needs the charge of a trial to be fixed before the trial starts. If the threshold may be chosen after an early stage, the early stages act as a filter that is not charged 2^θ, which we do not analyze further.

B Related mechanisms

This appendix compares the probabilistic core of Theorem 3.5 with related mechanisms outside Monotone Local Search, and its generating function with earlier uses in Monotone Local Search (Section 1.3).

Restarts with a cutoff. Luby, Sinclair and Zuckerman [LSZ93] analyzed restart strategies for Las Vegas algorithms, in which runs are cut off after a prescribed number of steps and restarted; [AGM+96] truncate runs to obtain small tail probabilities. Skipping samples whose running-time exponent exceeds a threshold is this mechanism.

The Markov cutoff. Zamir [Zam23, Lemma 2] shows that a Las Vegas algorithm whose running time, conditioned on a random variable Y, is 2^Y in expectation can be run in expected time $O(2^{\mathbb{E}[Y]} \cdot \mathbb{E}[Y])$ when $\mathbb{E}[Y]$ is known, by repeating runs truncated at the threshold $\mathbb{E}[Y] + 1$ and applying Markov's inequality. The Markov-cutoff algorithm of Remark 4.2, which gives the mean-exponent theorem (Theorem 4.1) directly, is this mechanism transplanted into Monotone Local

Search, with the same polynomial factor in the running time. Zamir's main theorem [Zam23, Theorem 1] removes the polynomial factor and the knowledge of the distribution, giving $O(2^{\mathbb{E}[Y]})$; Zamir states it with base e, and it gives this form when applied to $X = Y \ln 2$, as $2^Y = e^X$. Theorem 3.5 differs in what it is compared with: up to polynomial factors its running-time bound is $1/\mathbb{E}[2^{-g}]$, which is at most $2^{\mathbb{E}[g]}$ by Jensen's inequality and can be exponentially smaller.

Exponential moments for restart strategies. Scaman [Sca23, Lemma 4] shows that for a Las Vegas algorithm with running time $\mathrm{rt} \in [1, \infty]$ the best expected running time of a restart strategy lies between $1/\mathbb{E}[1/\mathrm{rt}]$ and $e^2(1 + \ln(1/\mathbb{E}[1/\mathrm{rt}]))/\mathbb{E}[1/\mathrm{rt}]$. For one sample size t, put $\mathrm{rt} = 2^{g(I,X,k)}$ if a uniform t-subset X of the universe lies in the solution S and $\mathrm{rt} = \infty$ otherwise; then $\mathbb{E}[1/\mathrm{rt}] = \mu_t(S)/\binom{n}{t}$. So Scaman's Lemma 4 is the restart-strategy analogue of the probabilistic core of Theorem 3.5: its lower bound corresponds to Theorem 3.12(a), and its upper bound to Theorem 3.5 for one fixed t, with each run cut off after the running time that its threshold allows in place of a computed running-time exponent. Scaman bounds the expected running time, whereas we bound the success probability within a fixed budget; the two forms convert into each other up to constant factors (by Markov's inequality, expected time T gives success probability at least $1/2$ within budget $2T$; conversely, success probability p within budget B gives expected time at most B/p by independent restarts). His upper bound has the factor $e^2(1 + \ln(1/\mathbb{E}[1/\mathrm{rt}]))$ where our threshold lemma (Lemma 3.8) has a factor linear in the largest threshold; both are polynomial. Scaman's impossibility result for universal strategies, which use one cutoff sequence for every distribution, concerns a different question, since our algorithm is given its target running time $2^{\psi n}$.

Exponential moments in exact algorithms. As Zamir [Zam23] discusses, the success probability of the PPSZ algorithm of [PPSZ05] is an exponential moment $\mathbb{E}[2^{-G}]$ of the number G of guessed variables, which Jensen's inequality bounds from below by $2^{-\mathbb{E}[G]}$. The normalized moments $\mu_t(S)/\binom{n}{t} = \mathbb{E}\left[\mathbf{1}[X \subseteq S]\, 2^{-g(I,X,k)}\right]$, for a uniform t-subset X of the universe and with $\mathbf{1}[X \subseteq S]$ equal to 1 if $X \subseteq S$ and 0 otherwise, have the same form.

Generating functions. [FGLS19] and Gaspers and Lee [GL17] already used generating functions in Monotone Local Search, for product-form running times, in particular a negative-binomial series in $k - t$ followed by the binomial theorem. Our $W_S(y)$ is a different series, the binomial series over the subsets of one solution, and it is defined for arbitrary running-time exponents; the comparison $W_S(y) \le \max_t(\mu_t(S)/\binom{n}{t})(1 + y)^n$ in the proof of Theorem 3.5 is what makes it usable.

Summary. Table 5 summarizes which features of our setting the mechanisms above have. The features are the five of the paragraph "What is new compared with restarts and cutoffs" in Section 1: (i) the running time of a call is computed before the call, so a call above the threshold is skipped rather than cut off; (ii) a call can succeed only if the sample lies in an unknown solution; (iii) the running time depends on the structure of the sample, not only on its size; (iv) one generating function covers all sample sizes; and (v) that generating function is bounded by combinatorial recursions over the sampled structure.

Mechanism	(i)	(ii)	(iii)	(iv)	(v)
Restarts with a cutoff [LSZ93, AGM$^+$96]	no	no	no	–	no
Markov cutoff [Zam23]	no	no	no	–	no
Restart strategies via $\mathbb{E}[1/\mathrm{rt}]$ [Sca23]	no	no	no	–	no
Monotone Local Search, product form [FGLS19, GL17]	yes	yes	no	yes	no
Theorem 3.5 with Lemma 3.9	yes	yes	yes	yes	yes

Table 5: Features (i)–(v) of the paragraph above. A dash means that the mechanism has no sample size.

C Linear savings, product-form running times and Max Leaf Spanning Tree

This appendix explains why Max Leaf Spanning Tree needs sample-dependent running times: its natural extension problem is W[1]-hard (Appendix C.1), and product-form running-time exponents certify nothing below a single unsampled call when $k/n \le d/(bc)$, nor, when $c = 1$, for the worst solution size. The product-form analyses we tried had their numerical optimum at $c = 1$ (Appendix C.2). The proofs of Corollary 4.3 and of the converse part of Proposition 4.5 are in Appendices D.2.1 and D.2.2. We keep the notation of Section 4.

C.1 MaxLeaf-Extension is W[1]-hard

The applications of the paper concern Max Leaf Spanning Tree (MaxLeaf), here in its decision version: given a connected graph G and an integer ℓ, decide whether G has a spanning tree with at least ℓ leaves. A *leaf* of a tree is a vertex of degree 1, and $\mathrm{leaves}(T)$ denotes the set of leaves of a tree T. The most direct way to apply Monotone Local Search to MaxLeaf is to sample vertices that are to become leaves and to complete the sample with an algorithm for the following extension problem.

Definition C.1 (MaxLeaf-Extension)**.** MaxLeaf-Extension: given a graph G, a set $X \subseteq V(G)$ and an integer $\ell' \ge 0$, decide whether G has a spanning tree T with $X \subseteq \mathrm{leaves}(T)$ and $|\,\mathrm{leaves}(T) \setminus X| \ge \ell'$. The parameter is ℓ'.

The words "at least" matter: if exactly ℓ' further leaves were required, the forward direction of the reduction below could fail, because vertices that receive no pendant subdivision vertex become additional leaves. MaxLeaf parameterized by ℓ is fixed-parameter tractable, with an $O^*(3.188^\ell)$-time algorithm by Zehavi [Zeh18]. We show that its extension problem parameterized by ℓ' is not, unless FPT = W[1]; in particular it has no algorithm with running time $c^{\ell'} N^{O(1)}$, which is what the original Monotone Local Search [FGLS19] would need.

In the framework of Definition 3.1, MaxLeaf is the implicit set system Φ_{ML} whose instances are pairs (G, ℓ) of a graph G and an integer $\ell \ge 0$, with universe $V(G)$ and with solutions the sets $L \subseteq V(G)$ with $|L| = \ell$ and $L \subseteq \mathrm{leaves}(T)$ for some spanning tree T of G; on connected graphs, Φ_{ML}-Subset is the decision version above. Membership is decided in polynomial time: graphs with at most two vertices are checked directly (a one-vertex tree has no leaf, and both vertices of a two-vertex tree are leaves), and if $|V(G)| \ge 3$, such a tree T exists exactly when $G - L$ is connected and every vertex of L has a neighbor outside L. On an input $((G, \ell), X, k)$, Φ_{ML}-Extension asks for $Y \subseteq V(G) \setminus X$ with $|Y| \le k - |X|$ and $|X \cup Y| = \ell$ such that $X \cup Y \subseteq \mathrm{leaves}(T)$ for some

spanning tree T. So the input is a NO-instance if $\ell < |X|$ or $\ell > k$, and otherwise it is equivalent to the instance $(G, X, \ell - |X|)$ of MaxLeaf-Extension, since a spanning tree with at least $\ell - |X|$ leaves outside X has a set of exactly $\ell - |X|$ of them, and every such set is a valid Y.

Theorem C.2 (MaxLeaf-Extension is W[1]-hard)**.** MaxLeaf-Extension *parameterized by ℓ' is W[1]-hard. Consequently it has no algorithm with running time $f(\ell')N^{O(1)}$, for any computable function f, unless* FPT = W[1].

Proof. We give a parameterized reduction from Independent Set parameterized by the solution size, which is W[1]-hard [CFK+15]. The idea is to subdivide every edge and to prescribe every subdivision vertex as a leaf: a leaf's neighbor in a tree with at least three vertices is not a leaf, so the further leaves lie in $V(G)$ and are pairwise non-adjacent in G. Let (G, k) be an instance, asking whether G has an independent set of size k. If $k \leq 0$ the answer is YES; if $k \geq 1$ and $V(G) = \emptyset$ it is NO. In these cases we output a single edge with $X = \emptyset$ and $\ell' = 0$ (a yes-instance) or $\ell' = 3$ (a no-instance). So let $k \geq 1$ and $|V(G)| \geq 1$; the case $k > |V(G)|$ needs no separate treatment.

Construction. Let G' be obtained from G by subdividing every edge $e = uv \in E(G)$ with a new vertex x_e (adjacent to u and v only), adding a vertex r adjacent to every vertex of $V(G)$, and adding a vertex z adjacent only to r. Let $X = \{z\} \cup \{x_e : e \in E(G)\}$, and output (G', X, ℓ') with $\ell' = k$. This takes polynomial time, G' is connected, and the parameter is unchanged. Note that $V(G') \setminus X = V(G) \cup \{r\}$.

Forward direction. Let S be an independent set of G with $|S| = k$. Let T consist of the edge rz, the edges rv for all $v \in V(G)$, and, for each $e = uv \in E(G)$, one edge $x_e y_e$ with $y_e \in \{u, v\} \setminus S$; such a y_e exists because S is independent, so u and v are not both in S. Then T is connected, since every vertex is joined to r directly or through a vertex of $V(G)$, and it has $1+|V(G)|+|E(G)| = |V(G')|-1$ edges; hence T is a spanning tree of G'. In T the vertex z and every x_e have degree 1, and every $v \in S$ has degree 1, because its only edge in T is rv (no x_e is attached to a vertex of S). So $X \cup S \subseteq \text{leaves}(T)$, and $|\text{leaves}(T) \setminus X| \geq |S| = \ell'$ since $S \cap X = \emptyset$.

Backward direction. Let T be a spanning tree of G' with $X \subseteq \text{leaves}(T)$ and $|\text{leaves}(T) \setminus X| \geq \ell'$, and put $S' = \text{leaves}(T) \setminus X$. Since $|V(G)| \geq 1$, the tree T has at least three vertices, so no two leaves of T are adjacent in T (two adjacent leaves would form a component with two vertices). The only neighbor of z in G' is r, so $rz \in E(T)$, and r is not a leaf because its neighbor $z \in X$ is a leaf. Hence $S' \subseteq V(G') \setminus (X \cup \{r\}) = V(G)$. Let $e = uv \in E(G)$. The vertex x_e is a leaf of T, and its neighbor in T is u or v; this neighbor is not a leaf, so u and v are not both in S'. Thus S' is an independent set of G with $|S'| \geq \ell' = k$, and any k of its vertices form an independent set of size k.

Since parameterized reductions preserve membership in FPT, an algorithm with running time $f(\ell')N^{O(1)}$ for MaxLeaf-Extension would give one with running time $f(k)N^{O(1)}$ for Independent Set, and hence FPT = W[1]. □

The map $(G', X, \ell') \mapsto ((G', |X| + \ell'), X, |X| + \ell')$ takes polynomial time, keeps the parameter (the number $k - |X|$ of further elements equals ℓ'), and by the description of Φ_{ML}-Extension after Definition C.1 maps yes-instances of MaxLeaf-Extension to yes-instances of Φ_{ML}-Extension and no-instances to no-instances. So Φ_{ML}-Extension on inputs (I, X, k), parameterized by $k - |X|$, is W[1]-hard as well, which rules out the case $b = d = 1$ of Corollary 4.6 for Φ_{ML}, under FPT $\neq$ W[1]. It does not rule out extension algorithms with $b > 1$ or $d > 1$, which the multivariate version allows, and an earlier unpublished attempt pursued exactly these. The next subsection records what is known about this route.

C.2 Limits of product-form running times

The product form is the case of Corollary 4.3 with the following parameters, which the discussion after Proposition C.4 uses.

Remark C.3 (Product form through linear savings). Let $1 \le d \le bc$ and $0 < k < n$, and put $\sigma = k/n$. Let $S \in \mathcal{F}_I$ with $|S| = k$ be such that g has the product form (10) on all inputs (I, X, k) with $X \subseteq S$. Then every $X \in \binom{S}{t}$ has $g(I, X, k) = n \log b + k \log c - t \log(bc/d)$, so (7) holds with equality for every t, with

$$\beta = \log b + \sigma \log c, \qquad \rho = \log \frac{bc}{d} \ge 0, \qquad C = 0, \qquad p = 2^{-\rho} = \frac{d}{bc}.$$

Since g is constant on each $\binom{S}{t}$, Jensen's inequality in the proof of Corollary 4.3(1) is an equality, so the upper bound computed there is exact: $G_k = 2^{nF^*(\sigma)}$. The criterion $\rho > \log(1/\sigma)$ of Corollary 4.3(2) reads $\sigma > d/(bc)$.

Two exact statements limit what product-form running-time exponents can certify.

Proposition C.4 (Two limits of the product form)**.** *Let $b, c, d \ge 1$, $0 \le k \le n$, and let M_k be as in Proposition 4.5. Then:*

(a) $\mathrm{base}(b, c, d) \ge b$*, with equality if and only if $c = 1$. If $c = 1$, then* $\max_{0 \le k \le n} M_k = M_0 = b^n$*.*

(b) Let $1 \le k \le n$. If $k/n \le d/(bc)$, then $M_k = b^n c^k$; if $k/n > d/(bc)$, then $M_k < b^n c^k$.

By (a), if $c = 1$, the main theorem and multivariate Monotone Local Search certify no running-time bound below the bound $b^n N^{O(1)}$ of a single call of the extension algorithm with $X = \emptyset$. Part (b) is a sampling threshold: $b^n c^k$ is the value of $2^{g(I,\emptyset,k)}$ for the unsampled call.

Proof. (a) Since $\min(d, bc) \ge 1 > 0$, $\mathrm{base}(b, c, d) - b = \min(d, bc)(1 - 1/c)$ vanishes exactly when $c = 1$. If $c = 1$, then $M_0 = b^n$ by (11), and $M_k \le \mathrm{base}(b, 1, d)^n = b^n$ for all k by Proposition 4.5(b).

(b) Write $a_t = \binom{n}{t}\binom{k}{t}^{-1}(d/(bc))^t$, so that $M_k = b^n c^k \min_{0 \le t \le k} a_t$ and $a_0 = 1$. For $0 \le t < k$,

$$\frac{a_{t+1}}{a_t} = \frac{n - t}{k - t} \cdot \frac{d}{bc},$$

because $\binom{n}{t+1}/\binom{n}{t} = (n-t)/(t+1)$ and $\binom{k}{t+1}/\binom{k}{t} = (k-t)/(t+1)$. As $k \le n$, the factor $(n-t)/(k-t)$ is at least n/k and is nondecreasing in t. If $k/n \le d/(bc)$, every ratio is at least $(n/k)(d/(bc)) \ge 1$, so the sequence (a_t) is nondecreasing and $\min_t a_t = a_0 = 1$. If $k/n > d/(bc)$, then $a_1 = (n/k)(d/(bc)) < 1 = a_0$, so $M_k \le b^n c^k a_1 < b^n c^k$. □

Part (b) is the exact, finite-n form of the following asymptotic observation: the limit of $\frac{1}{n}\log$ of the term $\binom{n}{t}\binom{k}{t}^{-1} b^{n-t} c^{k-t} d^t$ of the classical expression, for fixed k/n and t/n, is a convex function of the sampling fraction t/n and is minimized at $t = 0$ exactly when $k/n \le d/(bc)$. Both are the product-form case of Corollary 4.3(2): with $\rho = \log(bc/d)$ and $\sigma = k/n$ (Remark C.3), the condition $\rho > \log(1/\sigma)$ reads $k/n > d/(bc)$.

We complement Proposition C.4 with numerical evidence from a local numerical optimization (Appendix F): with the measure-and-conquer branching algorithm of Binkele-Raible and Fernau [BRF14] as a MaxLeaf-Extension algorithm, the prescribed leaves X as initial floating leaves, and a general linear measure optimized for the base $b + d - d/c$, every variant we tried had its optimum at $c = 1$, where by Proposition C.4(a) multivariate Monotone Local Search certifies nothing below a single call of the extension algorithm with $X = \emptyset$. A first-order sensitivity analysis

indicates the reason: each branching that determines the base has a branch that creates no leaves and does not decrease ℓ'. This is evidence about the analyses we tried, not a theorem, and it is not used elsewhere; the resulting bound, $O^*(1.8957^n)$ with a numerically computed base, is also weaker than $O^*(1.8619^n)$ [AKML11].

D Deferred proofs

This appendix proves the statements of the main text whose proofs were deferred. Each statement is restated, with its number, before its proof, together with the lemmas that only these proofs use. The subsections follow the order of the main text: optimality in its class (Appendix D.1), linear savings and product-form running times (Appendix D.2), the Steiner computations (Appendix D.3), Algorithm RS-θ (Appendix D.4), the closure bound of Algorithm RSC (Appendix D.5), the black-box barrier (Appendix D.6), balanced set-inclusion families (Appendix D.7), the deterministic algorithms (Appendix D.8), Connected Feedback Vertex Set (Appendix D.9), the weighted finishing steps (Appendix D.10) and the further applications (Appendix D.11).

D.1 Optimality in its class

Theorem 3.12 (Optimality in its class)**.** *Let I, k, $\mathcal{A}$ and g be as fixed above, and $\mathcal{E}_t$, $q_\theta(t)$, $\mu_t(\mathcal{E})$ and π^* as in Definition 3.11. Then:*

(a) *every algorithm in $\mathcal{C}_{\mathrm{thr}}$ whose total charge is at most B on every run answers* YES *with probability at most $\mathsf{B}\pi^*$;*

(b) *if $\pi^* \geq 2^{-\psi n}/\Gamma$, then Algorithm $\mathsf{MLS}\text{-}\theta(\psi, \Gamma, \kappa)$ answers* YES *with probability at least $1 - e^{-\kappa}$, and it runs in time $2^{\psi n} N^{O(1)}$ by Theorem 3.5(i);*

(c) *for every t and every integer θ, $q_\theta(t)2^{-\theta} \leq \mu_t(\mathcal{E})$. If, for an integer $g_{\max} \geq 0$, g takes integer values in $\{0, \ldots, g_{\max}\}$ on $\mathcal{E}_t$, then $\max_{0 \leq \theta \leq g_{\max}} q_\theta(t)2^{-\theta} \geq \mu_t(\mathcal{E})/(g_{\max}+1)$.*

Proof. (a) Let $\mathcal{G}_{i-1}$ be everything the algorithm has observed before trial i, including the coins it uses to choose (t_i, θ_i), and let $i_{\max}$ be the number of trials. Every trial is charged at least $2^0 = 1$, so $i_{\max} \leq \mathsf{B}$. The event $\{i \leq i_{\max}\}$ is decided before trial i, so it is determined by $\mathcal{G}_{i-1}$. Because $\mathcal{A}$ is correct, trial i succeeds exactly when $X_i \in \mathcal{E}_{t_i}$ and $g(X_i) \leq \theta_i$. As X_i is uniform and independent of $\mathcal{G}_{i-1}$,

$$\Pr\big[\text{trial } i \text{ succeeds} \mid \mathcal{G}_{i-1}\big] = \frac{q_{\theta_i}(t_i)}{\binom{n}{t_i}} \leq \pi^* 2^{\theta_i}.$$

By the union bound and the tower property,

$$\Pr[\textsc{yes}] \leq \sum_{i \geq 1} \mathbb{E}\Big[\mathbf{1}[i \leq i_{\max}] \Pr\big[\text{trial } i \text{ succeeds} \mid \mathcal{G}_{i-1}\big]\Big] \leq \pi^* \, \mathbb{E}\Big[\sum_{i \leq i_{\max}} 2^{\theta_i}\Big] \leq \pi^* \mathsf{B}.$$

(b) Let (t, θ) attain π^* and put $p = q_\theta(t)/\binom{n}{t} = \pi^* 2^\theta \geq 2^{\theta - \psi n}/\Gamma$. Since $2^{\theta-\psi n}/\Gamma \leq p \leq 1$, we have $2^\theta \leq \Gamma 2^{\psi n}$, so $\theta \leq \psi n + \log \Gamma$, which is below the cap $\Theta \geq \psi n + \log \Gamma + 1$. So the algorithm runs trials of the pair (t, θ), and their number is $m_{t,\theta} \geq M_{t,\theta} \geq \kappa \Gamma 2^{\psi n - \theta} \geq \kappa/p$. Each such trial succeeds with probability p, because $\mathcal{A}$ accepts every $X \in \mathcal{E}_t$. So all of them fail with probability at most $(1-p)^{m_{t,\theta}} \leq e^{-p m_{t,\theta}} \leq e^{-\kappa}$.

(c) For every θ,

$$q_\theta(t)2^{-\theta} = \sum_{X \in \mathcal{E}_t,\ g(X) \leq \theta} 2^{-\theta} \leq \sum_{X \in \mathcal{E}_t,\ g(X) \leq \theta} 2^{-g(X)} \leq \mu_t(\mathcal{E}).$$

In the integer case, with $q_{-1}(t) = 0$, exactly $q_\theta(t) - q_{\theta-1}(t)$ sets in $\mathcal{E}_t$ have running-time exponent θ, so $\mu_t(\mathcal{E}) = \sum_{\theta=0}^{g_{\max}} (q_\theta(t) - q_{\theta-1}(t)) 2^{-\theta} \le \sum_{\theta=0}^{g_{\max}} q_\theta(t) 2^{-\theta}$, and the largest of these $g_{\max}+1$ terms is at least $\mu_t(\mathcal{E})/(g_{\max}+1)$. □

D.2 Linear savings and product-form running times

This subsection proves Corollary 4.3 and the converse part of Proposition 4.5; the rest of what is known about product-form running times is in Appendix C.

D.2.1 Proof of the linear-savings corollary

We prove Corollary 4.3 here, with the notation of Section 4.2.

Corollary 4.3 (Linear mean savings)**.** *Let $\beta = \beta(I,k)$, $\rho = \rho(I,k)$ and $C = C(I,k)$ be nonnegative reals computable from (I,k) in time $N^{O(1)}$, with $\beta n + C \le N^{O(1)}$. Let I be an instance and k an integer with $\sigma = k/n \in (0,1)$ such that for every $t \in \{0,\dots,k\}$ and every $S \in \mathcal{F}_I$ with $|S| = k$,*

$$\mathbb{E}_X\big[g(I,X,k)\big] \le \beta n - \rho t + C, \qquad X \text{ uniform in } \binom{S}{t}. \tag{7}$$

Let $F^*(\sigma)$ be as in (6), and let $\kappa > 0$ be as in Theorem 3.5. Then:

(1) Every such S satisfies $G(S) \le 2^{C+nF^*(\sigma)}$. Algorithm MLS-$\theta$ with $\Gamma = 1$, confidence κ and a rate computable from (I,k) in time $N^{O(1)}$ gives a randomized algorithm that, given I and k, runs in time $2^{\max(0,\,C+nF^*(\sigma))}N^{O(1)}$, answers NO *if $\mathcal{F}_I$ contains no set of size at most k, and answers* YES *with probability at least $1-e^{-\kappa}$ if $\mathcal{F}_I$ contains a set of size k.*

(2) $F^*(\sigma) < \beta$ if and only if $\rho > \log(1/\sigma)$.

Proof. (1) Let $S \in \mathcal{F}_I$ with $|S| = k$, and put $u = 2^\rho = 1/p$. As in the proof of Theorem 4.1, Jensen's inequality and (7) give $\mu_t(S) \ge \binom{k}{t} 2^{-(\beta n - \rho t + C)}$ for every $t \in \{0,\dots,k\}$. Hence, by the binomial theorem, for every $y \ge 0$

$$W_S(y) \ge 2^{-\beta n - C} \sum_{t=0}^{k} \binom{k}{t} (uy)^t = 2^{-\beta n - C}(1+uy)^k, \qquad \text{so} \qquad \frac{(1+y)^n}{W_S(y)} \le 2^{\beta n + C}\, 2^{E(y)},$$

where $E(y) = n\log(1+y) - k\log(1+uy)$. The derivative $E'(y) = \frac{1}{\ln 2}\left(\frac{n}{1+y} - \frac{ku}{1+uy}\right)$ has the sign of

$$n(1+uy) - ku(1+y) = n(1-\sigma u) + uy(n-k),$$

which is affine in y with nonnegative slope $u(n-k)$.

If $\sigma \le p$, then $\sigma u \le 1$, so $E' \ge 0$ on $[0,\infty)$ and $\min_{y\ge 0} E(y) = E(0) = 0$. At $y = 0$ the upper bound is therefore $2^{\beta n + C} = 2^{C + nF^*(\sigma)}$.

If $\sigma > p$, then $\sigma u > 1$, so E decreases on $[0, y_0]$ and increases on $[y_0, \infty)$, where $y_0 = (\sigma u - 1)/(u(1-\sigma)) > 0$ is the root of the affine function above. From $1 + y_0 = (u-1)/(u(1-\sigma))$ and $1 + uy_0 = \sigma(u-1)/(1-\sigma)$,

$$\begin{aligned} \frac{E(y_0)}{n} &= \log(1+y_0) - \sigma\log(1+uy_0) = H(\sigma) + (1-\sigma)\log(u-1) - \log u \\ &= H(\sigma) + (1-\sigma)\log(1-p) - \sigma\rho, \end{aligned}$$

where the last step uses $\log(u-1) = \log(1-p) + \rho$ and $\log u = \rho$. By (6) this is $F^*(\sigma) - \beta$, so $(1+y_0)^n / W_S(y_0) \le 2^{C+nF^*(\sigma)}$.

The algorithm of (1) computes a rational number ψ with $C + nF^*(\sigma) \le \psi n \le C + nF^*(\sigma) + 1$, in time $N^{O(1)}$ from approximations of β, ρ, C and the logarithms in $F^*(\sigma)$, and runs Algorithm MLS-θ with rate $\psi^+ = \max(0, \psi)$, $\Gamma = 1$ and confidence κ. In both cases above, the generating-function condition holds at $y = 0$ if $\sigma \le p$ and at $y = y_0$ if $\sigma > p$, with $\Gamma = 1$ and $2^{C+nF^*(\sigma)}$ in place of $2^{\psi n}$, hence also with the larger value $2^{\psi^+ n}$, so $(\star)$ holds with rate ψ^+ by the first inequality of Lemma 3.9. The rate ψ^+ is rational, computable and nonnegative, and $\psi^+ \le \max(0, (C+1)/n + \beta) \le N^{O(1)}$ because $F^*(\sigma) \le \beta$ and $n \ge 1$. So Theorem 3.5 applies, and its time bound $2^{\psi^+ n} N^{O(1)} \le 2^{\max(0, C+nF^*(\sigma)+1)} N^{O(1)} \le 2 \cdot 2^{\max(0, C+nF^*(\sigma))} N^{O(1)}$ gives the claims.

(2) By the remark after (6), $F^*(\sigma) < \beta$ holds exactly when $\sigma > p = 2^{-\rho}$, which is equivalent to $\rho > \log(1/\sigma)$. □

D.2.2 Proof of the converse for product-form running times

We prove here part (c) of Proposition 4.5 and the final inequalities stated after it; parts (a) and (b) are proved in Section 4.3. We keep the notation of that proposition.

Proposition 4.5 (Product form). *Let $b, c, d \ge 1$, let $0 \le k \le n$, and let μ_t, M_k and G_k be given by* (11). *Then:*

(a) The generating function is $W(y) = \sum_{t=0}^{k} \mu_t y^t = b^{-n} c^{-k} (1 + ybc/d)^k$, and

$$M_k = \min_{0 \le t \le k} \frac{\binom{n}{t}}{\binom{k}{t}} b^{n-t} c^{k-t} d^t.$$

(b) $M_k \le G_k \le \mathrm{base}(b, c, d)^n$. More precisely, $y^* = (c-1)d/(bc)$ gives $(1+y^*)^n / W(y^*) = (b + d - d/c)^n$, and $y = 0$ gives $(1+0)^n / W(0) = b^n c^k \le (bc)^n$.

(c) Conversely, the following hold:

- *if $c = 1$, then $G_0 = M_0 = b^n = \mathrm{base}(b, c, d)^n$;*
- *if $c > 1$ and $d < bc$, let $\sigma^* = cd/(bc + cd - d)$; then $\sigma^* \in (0, 1)$; if $\sigma^* n$ is an integer, then $G_{\sigma^* n} = (b + d - d/c)^n$; and for every $n \ge 1$,*

$$\max_{0 \le k \le n} G_k \;\ge\; (b + d - d/c)^n \Big(\frac{b}{b + d - d/c} \Big)^{1/\sigma^*};$$

- *if $c > 1$ and $d \ge bc$, then $G_n = M_n = (bc)^n = \mathrm{base}(b, c, d)^n$.*

Consequently, with $\mathsf{K} = \big((b + d - d/c)/b\big)^{1/\sigma^*}$ in the second case and $\mathsf{K} = 1$ otherwise,

$$\frac{\mathrm{base}(b, c, d)^n}{(n+1)\mathsf{K}} \;\le\; \max_{0 \le k \le n} M_k \;\le\; \mathrm{base}(b, c, d)^n.$$

Proof of Proposition 4.5(c) and of the final inequalities. (c) Put $u = bc/d$ and, for real $x \in [0, n]$ and $y \ge 0$, $E_x(y) = n \log(1+y) - x \log(1+uy)$, so that by (a) $(1+y)^n / W(y) = b^n c^k 2^{E_k(y)}$ for integers k. As in the proof of Corollary 4.3, the derivative $E_x'(y)$ has the sign of $n(1 - (x/n)u) + uy(n - x)$.

The case $c = 1$. For $k = 0$ only $X = \emptyset$ occurs: $W(y) = b^{-n}$, so $G_0 = \inf_{y \ge 0} b^n (1+y)^n = b^n$, attained at $y = 0$, and $M_0 = b^n$. Also $\mathrm{base}(b, 1, d) = b$.

The case $c > 1$ and $d < bc$. Here $\sigma^* > 0$, and $\sigma^* < 1$ because $cd < bc + cd - d$ is equivalent to $d < bc$. Moreover $\sigma^* u = bc^2/(bc + cd - d) > 1$, because this is equivalent to $bc^2 - bc > cd - d$, that is $bc(c-1) > d(c-1)$, which holds as $c > 1$ and $d < bc$. Take $x = \sigma^* n$. Then $E'_x(0) < 0$, and since $x < n$ the affine function that carries the sign of E'_x has positive slope, so E_x has a unique minimizer on $[0, \infty)$, the root of $n(1+uy) = xu(1+y)$. Since $1+uy^* = c$ and $\sigma^* u(1+y^*) = \frac{bc^2}{bc+cd-d} \cdot \frac{bc+cd-d}{bc} = c$, the root is y^*. Therefore

$$\inf_{y \ge 0} \Big[n \log\big(b(1+y)\big) + x \log \frac{c}{1+uy} \Big] = n \log\big(b(1+y^*)\big) + x \log \frac{c}{c} = n \log\Big(b + d - \frac{d}{c}\Big), \tag{24}$$

and if $\sigma^* n$ is an integer k, the left-hand side is $\log G_k$, so $G_k = (b + d - d/c)^n$.

For general $n \ge 1$, define for real $x \in [0, n]$

$$\mathsf{f}(x) = \inf_{y \ge 0} \Big[n \log\big(b(1+y)\big) + x \log \frac{c}{1+uy} \Big],$$

so that $\mathsf{f}(k) = \log G_k$ for integers $0 \le k \le n$, by (a). The function f is finite: since $u > 1$ we have $1 + uy \le u(1+y)$, so the bracket is at least $n \log b + (n-x)\log(1+y) + x \log(c/u) \ge n \log b - n |\log(c/u)|$. As an infimum of functions that are affine in x, f is concave on $[0, n]$. We have $\mathsf{f}(0) = n \log b$ (the bracket is $n \log(b(1+y))$, minimized at $y = 0$), and $\mathsf{f}(\sigma^* n) = n \log(b + d - d/c)$ by (24); note that $b + d - d/c \ge b$, because $c \ge 1$. Let $k = \lfloor \sigma^* n \rfloor$, so $\sigma^* n - 1 < k \le \sigma^* n$. By concavity on $[0, \sigma^* n]$,

$$\begin{aligned} \log G_k = \mathsf{f}(k) &\ge \Big(1 - \frac{k}{\sigma^* n}\Big)\mathsf{f}(0) + \frac{k}{\sigma^* n}\,\mathsf{f}(\sigma^* n) = n \log b + \frac{k}{\sigma^*} \log \frac{b + d - d/c}{b} \\ &\ge n \log\Big(b + d - \frac{d}{c}\Big) - \frac{1}{\sigma^*} \log \frac{b + d - d/c}{b}, \end{aligned}$$

which is the claimed bound.

The case $c > 1$ and $d \ge bc$. Take $k = n$. Now $u \le 1$, so $1 + uy \le 1 + y$ and

$$\frac{(1+y)^n}{W(y)} = (bc)^n \Big(\frac{1+y}{1+uy}\Big)^n \ge (bc)^n,$$

with equality at $y = 0$; so $G_n = (bc)^n$. Also $M_n = \min_{0 \le t \le n} (bc)^{n-t} d^t = (bc)^n$, attained at $t = 0$, since $d \ge bc$.

The final inequalities. The upper bound $\max_k M_k \le \mathrm{base}(b, c, d)^n$ follows from (b). For the lower bound, Lemma 3.9 gives $M_k \ge G_k/(n+1)$ for every k, and the three cases give some k with $G_k \ge \mathrm{base}(b, c, d)^n / \mathsf{K}$: in the second case $\mathrm{base}(b, c, d) = b + d - d/c$ because $d < bc$. □

D.3 The Steiner computations

This subsection proves the statements of Sections 5.1 and 5.2, with the notation of those subsections.

Lemma 5.3 (Vertex-weighted directed Steiner tree)**.** *For $\vec{G}$, n, r, K and φ as above, $\mathrm{OPT}_\varphi(\vec{G}, r, K)$ can be computed deterministically in time $2^{|K|}(n + \varphi_{\max})^{O(1)}$ and space $(n + \varphi_{\max})^{O(1)}$.*

Proof. If $K = \emptyset$, then $D = \emptyset$ is feasible and the answer is 0; let $K \ne \emptyset$. Let φ' agree with φ outside $\{r\} \cup K$ and vanish on $\{r\} \cup K$. Give every arc uv the length $\mathrm{len}(uv) = n\,\varphi'(v) + 1$, a positive integer.

Branching walks. A *branching walk from* v is a finite sequence $((u_1,\mathcal{W}_1),\dots,(u_q,\mathcal{W}_q))$, $q\ge 0$, in which $vu_i\in A(\vec G)$ and $\mathcal{W}_i$ is a branching walk from u_i, for every i; this is an inductive definition, and the empty sequence is the branching walk with $q=0$. Its *image* is $\{v\}$ together with the images of the $\mathcal{W}_i$, its *arcs* are the arcs vu_i together with the arcs of the $\mathcal{W}_i$ (counted with multiplicity), and its *length* is the sum of the lengths of its arcs. Equivalently, a branching walk is a rooted ordered tree mapped into $\vec G$ so that the root goes to v and every tree edge, directed away from the root, goes to an arc of $\vec G$. By induction, every vertex of the image other than v is the head of one of its arcs, and every vertex of the image is reachable from v in the subdigraph induced by the image.

For $Y\subseteq V(\vec G)$, $v\in V(\vec G)$ and an integer $\mathsf{L}\ge 0$ let $b_Y(v,\mathsf{L})$ be the number of branching walks from v of length L whose image is disjoint from Y. Then $b_Y(v,\mathsf{L})=0$ if $v\in Y$, and for $v\notin Y$

$$b_Y(v,0)=1,\qquad b_Y(v,\mathsf{L})=\sum_{\substack{vu\in A(\vec G)\\ u\notin Y}}\ \sum_{\mathsf{L}_1=0}^{\mathsf{L}-\mathrm{len}(vu)} b_Y(u,\mathsf{L}_1)\,b_Y(v,\mathsf{L}-\mathrm{len}(vu)-\mathsf{L}_1)\quad(\mathsf{L}\ge 1).$$

Indeed, the only branching walk of length 0 is the empty one, since every arc has positive length. A nonempty branching walk from v splits uniquely into its first pair $(u_1,\mathcal{W}_1)$ and the branching walk formed by the remaining pairs, which again starts at v; the length is $\mathrm{len}(vu_1)$ plus the two lengths, and the image avoids Y exactly when $u_1\notin Y$ and both pieces avoid Y (the image of the remainder contains v). The right-hand side refers only to lengths below L, so the table can be filled in order of increasing L.

Inclusion–exclusion. For $\mathsf{L}\ge 0$ put $b(\mathsf{L})=\sum_{Y\subseteq K}(-1)^{|Y|}b_Y(r,\mathsf{L})$. A branching walk from r with image $\mathcal{I}$ is counted in $b_Y(r,\mathsf{L})$ exactly for the sets $Y\subseteq K\setminus\mathcal{I}$, and $\sum_{Y\subseteq K\setminus\mathcal{I}}(-1)^{|Y|}$ is 1 if $K\subseteq\mathcal{I}$ and 0 otherwise. So $b(\mathsf{L})$ is the number of branching walks from r of length L whose image contains K.

Least length. We show that $b(\mathsf{L})>0$ for some L if and only if $\mathrm{OPT}=\mathrm{OPT}_\varphi(\vec G,r,K)<\infty$, and that the least such L, say L^*, satisfies $\lfloor \mathsf{L}^*/n\rfloor=\mathrm{OPT}$. First let a branching walk from r have image $\mathcal{I}\supseteq K$ and length L. Every vertex of $\mathcal{I}\setminus\{r\}$ is the head of one of its arcs, and the arcs have lengths at least $n\,\varphi'$ of their heads, so, as $\varphi'\ge 0$ and $\varphi'(r)=0$,

$$\mathsf{L}\ \ge\ n\sum_{v\in\mathcal{I}}\varphi'(v)\ =\ n\,\varphi(D_{\mathcal{I}}),\qquad D_{\mathcal{I}}=\mathcal{I}\setminus(\{r\}\cup K).$$

Every terminal is reachable from r in $\vec G[\mathcal{I}]=\vec G[\{r\}\cup K\cup D_{\mathcal{I}}]$, so $D_{\mathcal{I}}$ is feasible and $\lfloor \mathsf{L}/n\rfloor\ge\varphi(D_{\mathcal{I}})\ge\mathrm{OPT}$. Conversely, let D be feasible with $\varphi(D)=\mathrm{OPT}$, and let T be an out-tree of $\vec G[\{r\}\cup K\cup D]$ rooted at r that spans the vertices reachable from r there, for instance a breadth-first search tree. Ordering the children of every vertex arbitrarily turns T into a branching walk from r whose image contains K. It has $|A(T)|\le n-1$ arcs, each non-root vertex of T is the head of exactly one of them, and the non-root vertices of T outside K lie in D; so its length is at most $n\,\varphi(D)+|A(T)|\le n\,\mathrm{OPT}+n-1$. Hence L^* exists and $\lfloor \mathsf{L}^*/n\rfloor\le\mathrm{OPT}$, and with the first part $\lfloor \mathsf{L}^*/n\rfloor=\mathrm{OPT}$. As $\mathrm{OPT}\le\varphi(V(\vec G))\le n\,\varphi_{\max}$, it suffices to compute $b(\mathsf{L})$ for $0\le\mathsf{L}\le\mathsf{L}_{\max}:=n^2\varphi_{\max}+n-1$; if all these values are 0, then $\mathrm{OPT}=\infty$.

Running time and space. A branching walk of length at most L has at most L arcs; it is determined by a rooted ordered tree with $e\le\mathsf{L}$ edges, of which there are at most 4^e, and by the images of the e non-root vertices. So $b_Y(v,\mathsf{L})\le(\mathsf{L}+1)(4n)^{\mathsf{L}}$, and all numbers involved, including the partial sums of $b(\mathsf{L})$, have $O(\mathsf{L}_{\max}\log n+n)$ bits, which is polynomial in $n+\varphi_{\max}$. For one set Y the table $\big(b_Y(v,\mathsf{L})\big)_{v,\,\mathsf{L}\le\mathsf{L}_{\max}}$ has $n(\mathsf{L}_{\max}+1)$ entries, each computed with $O(n\mathsf{L}_{\max})$ arithmetic operations. The algorithm runs through the $2^{|K|}$ sets $Y\subseteq K$ one at a time, keeps one table and the

$\mathsf{L}_{\max}+1$ partial sums of $b(\mathsf{L})$, and finally returns $\lfloor \mathsf{L}^*/n \rfloor$ or ∞. This takes time $2^{|K|}(n+\varphi_{\max})^{O(1)}$ and space $(n+\varphi_{\max})^{O(1)}$. □

Lemma 5.6 (Directed Steiner encoding)**.** *Let $J=(\mathcal{D},B,R)$ be a state of an n-vertex graph G and $\varphi\colon B\to\{0,1,\dots,\varphi_{\max}\}$. If $C_1\in\mathcal{D}$, then $\mathrm{OPT}_\varphi(J)=\mathrm{OPT}_\varphi(\vec{G}_J,\hat{C}_1,K_1)$ for $K_1=\{\hat{C}:C\in\mathcal{D}\setminus\{C_1\}\}\cup\check{R}$. Consequently, for every state J of G, $\mathrm{OPT}_\varphi(J)$ can be computed deterministically in time $2^{|\mathrm{T}(J)|}(n+\varphi_{\max})^{O(1)}$ and space $(n+\varphi_{\max})^{O(1)}$.*

Proof. Let $C_1\in\mathcal{D}$ and put $\rho=\hat{C}_1$ and $K=K_1$. The non-root, non-terminal vertices of $\vec{G}_J$ are the vertices $\hat{b}$ with $b\in B$, so a feasible set for $(\vec{G}_J,\rho,K)$ in the sense of Section 5.1 has the form $\hat{Z}=\{\hat{b}:b\in Z\}$ with $Z\subseteq B$, and its weight is $\varphi(Z)$. We show that

$$\mathrm{OPT}_\varphi(\vec{G}_J,\rho,K)=\min\{\varphi(Y):Y\text{ a solution of }J\},\tag{25}$$

which is the claim.

From solutions to feasible sets. Let Y be a solution of J. Let H be the graph obtained from the connected graph $G[V(\mathcal{D})\cup Y]$ by contracting each member of $\mathcal{D}$ into one vertex; it is connected, and its vertices correspond to the vertices $\hat{C}$ ($C\in\mathcal{D}$) and $\hat{y}$ ($y\in Y$) of $\vec{G}_J$, among them ρ. Every edge of H joins two vertices $y,y'\in Y\subseteq B$ or a vertex $C\in\mathcal{D}$ to a vertex $y\in N(C)\cap Y$, since distinct members of $\mathcal{D}$ are non-adjacent; in both cases $\vec{G}_J$ has the two opposite arcs between the corresponding vertices. So every $\hat{C}$ ($C\in\mathcal{D}$) and every $\hat{y}$ ($y\in Y$) is reachable from ρ in $\vec{G}_J[\{\rho\}\cup K\cup\hat{Y}]$. Every $r\in R$ lies in $N[Y]$, so some $y\in Y\cap N[r]\subseteq N[r]\cap B$ exists, $\hat{y}$ is reachable from ρ, and $\check{r}$ is reached through the arc $\hat{y}\check{r}$. Hence $\hat{Y}$ is feasible, of weight $\varphi(Y)$, and the left-hand side of (25) is at most the right-hand side.

From feasible sets to solutions. Let $\hat{Z}$ be feasible, let Q be the set of vertices reachable from ρ in $\vec{G}_J[\{\rho\}\cup K\cup\hat{Z}]$, which contains ρ and K, and let $Y'=\{b\in Z:\hat{b}\in Q\}$. Then $Y'\subseteq B$ and $\varphi(Y')\le\varphi(Z)$. The vertices of $\check{R}$ have no outgoing arcs, so every vertex of $Q\setminus\check{R}$ is reached from ρ by a path all of whose vertices lie in $Q\setminus\check{R}$, that is, are of the form $\hat{C}$ ($C\in\mathcal{D}$) or $\hat{b}$ ($b\in Y'$). Each arc of such a path joins two vertices whose underlying sets (C or $\{b\}$) are joined by an edge of G, and every $C\in\mathcal{D}$ induces a connected subgraph of G; as $Q\setminus\check{R}$ consists of all $\hat{C}$ ($C\in\mathcal{D}$) and all $\hat{b}$ ($b\in Y'$), the graph $G[V(\mathcal{D})\cup Y']$ is connected. It is nonempty, because it contains C_1. Every $\check{r}$ with $r\in R$ lies in Q, and its predecessor on a path from ρ is some $\hat{b}\in Q$ with $b\in N[r]\cap B$, so $b\in Y'$; hence $R\subseteq N[Y']$. So Y' is a solution of J, and the right-hand side of (25) is at most $\varphi(Y')\le\varphi(Z)$. This proves (25).

For the running time, let first $\mathcal{D}\neq\emptyset$. The digraph $\vec{G}_J$ is built in polynomial time, has at most $2n$ vertices and weights at most $\varphi_{\max}$, and Lemma 5.3 is applied once, with $|K_1|=|\mathcal{D}|-1+|R|=|\mathrm{T}(J)|-1$ terminals; this takes time $2^{|\mathrm{T}(J)|}(n+\varphi_{\max})^{O(1)}$ and space $(n+\varphi_{\max})^{O(1)}$, deterministically. If $\mathcal{D}=\emptyset$, every solution Y of J is nonempty, because $V(\mathcal{D})\cup Y\neq\emptyset$. For $b\in B$ put $J^{(b)}=(\{\{b\}\},B\setminus\{b\},R\setminus N[b])$. This is a state ($\{b\}$ is connected, $B\setminus\{b\}$ avoids it and $R\setminus N[b]$ avoids $N[b]$), and a set $Y\subseteq B$ with $b\in Y$ is a solution of J if and only if $Y\setminus\{b\}$ is a solution of $J^{(b)}$: $G[Y]=G[\{b\}\cup(Y\setminus\{b\})]$, and $R\subseteq N[Y]=N[b]\cup N[Y\setminus\{b\}]$ holds if and only if $R\setminus N[b]\subseteq N[Y\setminus\{b\}]$. Hence $\mathrm{OPT}_\varphi(J)=\min_{b\in B}(\varphi(b)+\mathrm{OPT}_\varphi(J^{(b)}))$, with $\min\emptyset=\infty$, and $|\mathrm{T}(J^{(b)})|=|R\setminus N[b]|+1\le|\mathrm{T}(J)|+1$. Applying the first part to the at most n states $J^{(b)}$ gives the same bounds. □

Corollary 5.8 (Steiner extension)**.** *For every state J of an n-vertex graph, $\mathrm{OPT}(J)$ can be computed deterministically in time $2^{|\mathrm{T}(J)|}n^{O(1)}$ and polynomial space. Consequently, the algorithm $\mathcal{A}_{\mathrm{St}}$ of Definition 5.7 solves Φ_{CDS}-Extension in polynomial space, and*

$$g(G,X,k)=|\mathrm{T}(J_X)|=n-|N[X]|+\mathrm{comp}(G[X])$$

is a running-time exponent for it in the sense of Definition 3.2, with values in $\{0, \dots, n\}$.

Proof. The first statement is Lemma 5.6 with $\varphi \equiv 1$, so that $\varphi_{\max} = 1$ and the bounds become $2^{|\mathrm{T}(J)|} n^{O(1)}$ and $n^{O(1)}$.

For the second statement, Φ_{CDS} is an implicit set system (Definition 3.1): $V(G)$ is listed and connected domination is tested in polynomial time, and strings that do not encode a graph get an empty universe. By the description of the solutions of J_X after (12), $\mathcal{A}_{\mathrm{St}}$ accepts (G, X, k) exactly when some $Y \subseteq V \setminus X$ with $|Y| \le k - |X|$ makes $X \cup Y$ a connected dominating set. Its running time on every input, whether or not X lies in a solution, is $2^{|\mathrm{T}(J_X)|} n^{O(1)}$ by the first statement, and $|\mathrm{T}(J_X)|$ is computed in polynomial time by (13), which also gives the range. □

D.4 Algorithm RS-θ

This subsection proves the statements of Sections 7.2 to 7.4, with the notation of Section 7, and the lower bound of Section 7.6 on uniform lobsters (Appendix D.4.3).

D.4.1 The tree reduction and the three-state recursion

Lemma 7.3 (Tree reduction)**.** *Let G be a connected graph, let S be a connected dominating set of G, let F be a spanning tree of $G[S]$ and let o be an owner map with counts ν. Then for every $X \subseteq S$,*

$$|\mathrm{T}(J_X)| \;\le\; \Pi_{F,\nu}(X).$$

Equality holds for every $X \subseteq S$ if $G[S] = F$ and every vertex of $V \setminus S$ has exactly one neighbor in S. Consequently $W_S(y) \ge W_{F,\nu}(y)$ for every $y \ge 0$.

Proof. By (13), $|\mathrm{T}(J_X)| = |(V \setminus S) \setminus N[X]| + |S \setminus N[X]| + \mathrm{comp}(G[X])$, and we bound its three parts by the three terms of $\Pi_{F,\nu}(X)$ in turn. (1) If $u \in V \setminus S$ is not in $N[X]$, then its neighbor $o(u)$ is not in X; so $|(V \setminus S) \setminus N[X]| \le \sum_{v \in S \setminus X} |o^{-1}(v)| = \sum_{v \in S \setminus X} \nu_v$. (2) If $v \in S$ is not in $N[X]$, then $N_G[v] \cap X = \emptyset$, so $N_F[v] \cap X = \emptyset$ because $N_F[v] \subseteq N_G[v]$. (3) $F[X]$ is a spanning subgraph of $G[X]$, as $X \subseteq S = V(F)$; so $\mathrm{comp}(G[X]) \le \mathrm{comp}(F[X])$.

For the equality case, suppose that $o(u)$ is the only neighbor of u in S for every $u \notin S$. Then, as $X \subseteq S$, $u \notin N[X]$ if and only if $o(u) \notin X$, so (1) is an equality. If moreover $G[S] = F$, then $N_G[v] \cap S = N_F[v]$ for $v \in S$ and $G[X] = F[X]$, so (2) and (3) are equalities. The last claim holds termwise, as $2^{-|\mathrm{T}(J_X)|} \ge 2^{-\Pi_{F,\nu}(X)}$. □

Lemma 7.4 (Tree recursion)**.** *Let $y \ge 0$, and let the messages msg_v be defined by (18) for any order of the children. Then every msg_v has nonnegative entries and $W_{F,\nu}(y) = \mathrm{fin}(\mathrm{msg}_r)$.*

Proof. The proof is a double induction whose core is a decomposition of the potential of a partial subtree into the potentials of its two pieces, with one correction for the edge joining v to the new child and one for the domination of that child; the three states carry the information that these corrections need.

Let F' be the subtree consisting of a vertex v and the subtrees $F_{w_1}, \dots, F_{w_i}$ of its first i children ($0 \le i \le d$). For $X \subseteq V(F')$ let

$$\Pi^{\mathrm{in}}_{F'}(X) = \sum_{u \in V(F') \setminus X} \nu_u + \big|\{u \in V(F') \setminus \{v\} : N_F[u] \cap X = \emptyset\}\big| + \mathrm{comp}(F'[X]);$$

for $u \ne v$ in F' the set $N_F[u]$ lies inside $V(F')$, since the parent of u is v or lies in the same subtree F_{w_k} for some $k \le i$. Let $P' = (P'_1, P'_2, P'_3)$ be the sums of $y^{|X|} 2^{-\Pi^{\mathrm{in}}_{F'}(X)}$ over the sets $X \subseteq V(F')$ in

states 1, 2 and 3 of v, where "child" means one of $w_1, \dots, w_i$. We show, by induction on the height of v and then on i, that $P' = (\cdots(\mathrm{Init}_y(\nu_v) \circ \mathrm{msg}_{w_1}) \circ \cdots) \circ \mathrm{msg}_{w_i}$; for $i = d$ this identifies msg_v as the triple of F_v, whose entries are nonnegative. Write $\Pi^{\mathrm{in}}_{F_v}$ for $\Pi^{\mathrm{in}}_{F'}$ when $F' = F_v$.

For $i = 0$, F' consists of v alone. The set $X = \{v\}$ has $\Pi^{\mathrm{in}}_{F'}(X) = 1$ (one component, no owner term) and $X = \emptyset$ has $\Pi^{\mathrm{in}}_{F'}(X) = \nu_v$, and state 2 is empty because v has no children yet; so the triple is $(y/2, 0, 2^{-\nu_v}) = \mathrm{Init}_y(\nu_v)$. For the step from $i-1$ to i, let F'' be the subtree for $i-1$, with triple P'' and potential $\Pi^{\mathrm{in}}_{F''}$, let $w = w_i$, and let $Q = \mathrm{msg}_w$ be the triple of F_w, by the induction hypothesis. Every $X \subseteq V(F')$ splits uniquely as $X = X'' \cup X_w$ with $X'' \subseteq V(F'')$ and $X_w \subseteq V(F_w)$, and

$$\begin{aligned}\Pi^{\mathrm{in}}_{F'}(X) = \Pi^{\mathrm{in}}_{F''}(X'') + \Pi^{\mathrm{in}}_{F_w}(X_w) - [v \in X \text{ and } w \in X] \\ + [v \notin X,\ w \notin X \text{ and no child of } w \text{ is in } X].\end{aligned}$$

Indeed, the owner terms are additive over the split. A vertex $u \notin \{v, w\}$ has its closed neighborhood inside F'' or inside F_w, so it is counted in $\Pi^{\mathrm{in}}_{F'}$ exactly when it is counted in $\Pi^{\mathrm{in}}_{F''}$ or $\Pi^{\mathrm{in}}_{F_w}$. The vertex w is not counted in $\Pi^{\mathrm{in}}_{F_w}$, and it is counted in $\Pi^{\mathrm{in}}_{F'}$ exactly when $N_F[w] \cap X = \emptyset$, that is, when v, w and the children of w are not in X; the vertex v is counted in neither. Finally, $F'[X]$ is the disjoint union of $F''[X'']$ and $F_w[X_w]$, together with the edge vw if $v, w \in X$, which joins two components into one.

Now we sum the weights over the states of v and w. If $v \in X$, then w contributes the factor $2Q_1$ if $w \in X$ (one component fewer) and $Q_2 + Q_3$ otherwise (w is dominated by v); so $P'_1 = P''_1(2Q_1 + Q_2 + Q_3)$. If v is in state 2 for F'', then v stays in state 2, and w contributes Q_1, Q_2 or $Q_3/2$ according to its state (in state 3, w is not dominated, which adds 1 to the potential and so multiplies the weight by $\frac{1}{2}$). If v is in state 3 for F'', then $w \in X$ moves v to state 2 (term $P''_3 Q_1$), and otherwise v stays in state 3, with the factor $Q_2 + Q_3/2$. So $P' = P'' \circ Q$.

At the root, $N_F[r]$ consists of r and its children, so $\Pi_{F,\nu}(X) = \Pi^{\mathrm{in}}_{F_r}(X) + [r \text{ is in state } 3]$, and $W_{F,\nu}(y) = \mathrm{msg}_{r,1} + \mathrm{msg}_{r,2} + \mathrm{msg}_{r,3}/2 = \mathrm{fin}(\mathrm{msg}_r)$. □

D.4.2 Antinorm certificates in the cumulative cone

We collect the facts about $\mathcal{K}_{\mathrm{cum}}$ that the certificate argument uses, and then the abstract mechanism, which Section 8 reuses with another cone.

Lemma D.4.1 (Closure facts). *(a) If $Q \in \mathbb{R}^3_{\geq 0}$ and $\Delta \in \mathcal{K}_{\mathrm{cum}}$, then $\Delta \circ Q \in \mathcal{K}_{\mathrm{cum}}$ and $Q \circ \Delta \in \mathcal{K}_{\mathrm{cum}}$.*

(b) $\mathrm{fin}(\Delta) \geq 0$ for every $\Delta \in \mathcal{K}_{\mathrm{cum}}$.

Proof. (a) The operation $\circ$ is bilinear and $\mathcal{K}_{\mathrm{cum}}$ is a convex cone, so it suffices to check the three generators of $\mathcal{K}_{\mathrm{cum}}$. For $Q \geq 0$ we list each product and its cumulative coordinates: $(1,-1,0)\circ Q = (2Q_1+Q_2+Q_3,\ -Q_1-Q_2-\frac{Q_3}{2},\ 0)$, with cumulative coordinates $(2Q_1+Q_2+Q_3,\ Q_1+\frac{Q_3}{2},\ Q_1+\frac{Q_3}{2})$; $(0,1,-1)\circ Q = (0,\ Q_2+\frac{Q_3}{2},\ -Q_2-\frac{Q_3}{2})$, with $(0,\ Q_2+\frac{Q_3}{2},\ 0)$; and $(0,0,1)\circ Q = (0,\ Q_1,\ Q_2+\frac{Q_3}{2}) \geq 0$. In the other order, $Q \circ (1,-1,0) = (Q_1,\ Q_3,\ -Q_3)$, with cumulative coordinates $(Q_1,\ Q_1+Q_3,\ Q_1)$; $Q \circ (0,1,-1) = (0,\ \frac{Q_2}{2},\ \frac{Q_3}{2}) \geq 0$; and $Q \circ (0,0,1) = (Q_1,\ \frac{Q_2}{2},\ \frac{Q_3}{2}) \geq 0$. All cumulative coordinates are nonnegative. (b) $\mathrm{fin}(\Delta) = \frac{1}{2}(\Delta_1 + \Delta_2) + \frac{1}{2}(\Delta_1 + \Delta_2 + \Delta_3)$, and both brackets are nonnegative for $\Delta \in \mathcal{K}_{\mathrm{cum}}$. □

The nonnegativity of the other factor in (a) is needed: $(1,-1,0)\circ(0,1,-1) = (0,-\frac{1}{2},0) \notin \mathcal{K}_{\mathrm{cum}}$.

The next lemma isolates the mechanism: a convex region that the merge operation maps into itself up to a known scale, for any bilinear operation and any cone with the closure property of Lemma D.4.1(a).

Lemma D.4.2 (Antinorm certificates for a bilinear merge)**.** *Let $\circ$ be a bilinear operation on $\mathbb{R}^m$, let $\mathcal{K} \subseteq \mathbb{R}^m$ be a convex cone containing $\mathbb{R}^m_{\geq 0}$ such that $\Delta \circ Q \in \mathcal{K}$ and $Q \circ \Delta \in \mathcal{K}$ for every $\Delta \in \mathcal{K}$ and every $Q \in \mathbb{R}^m_{\geq 0}$, and for a finite nonempty set $Z \subseteq \mathbb{R}^m_{\geq 0}$ put $\mathcal{P}_Z = \operatorname{conv}(Z) + \mathcal{K}$. Then:*

(a) $\mathcal{P}_Z + \mathcal{K} = \mathcal{P}_Z$, and $\eta \mathcal{P}_Z \subseteq \eta' \mathcal{P}_Z$ whenever $0 < \eta' \leq \eta$;

(b) let $Y, Y', Z \subseteq \mathbb{R}^m_{\geq 0}$ be finite and nonempty with $y \circ z \in \mathcal{P}_{Y'}$ for all $y \in Y$ and $z \in Z$; if $P \in \eta \mathcal{P}_Y$ and $Q \in \eta' \mathcal{P}_Z$ with $\eta, \eta' > 0$ and $Q \geq 0$, then $P \circ Q \in \eta\eta' \mathcal{P}_{Y'}$.

Proof. (a) The first claim holds because $\mathcal{K} + \mathcal{K} = \mathcal{K}$. For the second, let $P = \eta(z + \Delta)$ with $z \in \operatorname{conv}(Z)$ and $\Delta \in \mathcal{K}$. Then $P = \eta'(z + \Delta'')$ with $\Delta'' = \frac{\eta - \eta'}{\eta'} z + \frac{\eta}{\eta'} \Delta$, and $\Delta'' \in \mathcal{K}$ because $z \geq 0$ and $\mathcal{K} \supseteq \mathbb{R}^m_{\geq 0}$ is a convex cone.

(b) Write $P = \eta(y + \Delta)$ and $Q = \eta' \hat{Q}$ with $\hat{Q} = z + \Delta' \geq 0$, where $y \in \operatorname{conv}(Y)$, $z \in \operatorname{conv}(Z)$ and $\Delta, \Delta' \in \mathcal{K}$. By bilinearity,

$$\frac{P \circ Q}{\eta\eta'} = y \circ z + y \circ \Delta' + \Delta \circ \hat{Q}.$$

With convex combinations $y = \sum_i \alpha_i y_i$ and $z = \sum_j \beta_j z_j$ of points of Y and of Z we get $y \circ z = \sum_{i,j} \alpha_i \beta_j \, y_i \circ z_j \in \mathcal{P}_{Y'}$, as $\mathcal{P}_{Y'}$ is convex. Since $y \geq 0$ and $\hat{Q} \geq 0$, the closure property gives $y \circ \Delta' \in \mathcal{K}$ and $\Delta \circ \hat{Q} \in \mathcal{K}$, and (a) gives $P \circ Q \in \eta\eta' \mathcal{P}_{Y'}$. □

Lemma 7.5 (Antinorm certificate)**.** *Let $y \geq 0$ and $e_1, e_2 > 0$, and let $Z \subseteq \mathbb{R}^3_{\geq 0}$ be finite and nonempty with*

(a) $z \circ z' \in \mathcal{P}_Z$ for all $z, z' \in Z$, and

(b) $\operatorname{Init}_y(j) \in e_1 e_2^j \, \mathcal{P}_Z$ for every integer $j \geq 0$.

Then for every tree F with s vertices and every $\nu : V(F) \to \mathbb{Z}_{\geq 0}$, writing $\nu(F) = \sum_{v \in V(F)} \nu_v$, we have $W_{F,\nu}(y) \geq c_Z \, e_1^s e_2^{\nu(F)}$. If $e_2 \leq 1$, then (b) holds for all j as soon as it holds for $j \leq j_0$ and $(y/2, 0, 0) \in e_1 e_2^{j_0+1} \mathcal{P}_Z$.

Proof. Root F anywhere and let msg_v be the messages of Lemma 7.4, which are nonnegative. By induction from the leaves, $\operatorname{msg}_v \in e_1^{|V(F_v)|} e_2^{\nu(F_v)} \mathcal{P}_Z$, where $\nu(F_v) = \sum_{u \in V(F_v)} \nu_u$: the start $\operatorname{Init}_y(\nu_v)$ lies in $e_1 e_2^{\nu_v} \mathcal{P}_Z$ by (b), and each merge with a child w multiplies the scale by $e_1^{|V(F_w)|} e_2^{\nu(F_w)}$, by Lemma D.4.2(b) with $Y = Y' = Z$, applied with $Q = \operatorname{msg}_w \geq 0$, whose closure hypothesis is Lemma D.4.1(a) and whose hypothesis on Z is (a). At the root, $\operatorname{msg}_r = \eta(z + \Delta)$ with $\eta = e_1^s e_2^{\nu(F)}$, $z \in \operatorname{conv}(Z)$ and $\Delta \in \mathcal{K}_{\mathrm{cum}}$. Since fin is linear, Lemma 7.4 and Lemma D.4.1(b) give

$$W_{F,\nu}(y) = \operatorname{fin}(\operatorname{msg}_r) = \eta\big(\operatorname{fin}(z) + \operatorname{fin}(\Delta)\big) \geq \eta \operatorname{fin}(z) \geq \eta \, c_Z.$$

For the last claim let $j > j_0$. Then $\operatorname{Init}_y(j) - (y/2, 0, 0) = (0, 0, 2^{-j}) \in \mathcal{K}_{\mathrm{cum}}$, so by Lemma D.4.2(a) $\operatorname{Init}_y(j) \in e_1 e_2^{j_0+1} \mathcal{P}_Z + \mathcal{K}_{\mathrm{cum}} = e_1 e_2^{j_0+1} \mathcal{P}_Z \subseteq e_1 e_2^j \mathcal{P}_Z$, because $e_1 e_2^j \leq e_1 e_2^{j_0+1}$. □

Corollary 7.6 (From certificates to exponential moments)**.** *Suppose that (y, e_1, e_2, Z) satisfies the hypotheses of Lemma 7.5. Then for every connected graph G and every connected dominating set S of G with $|S| = s$, writing $\sigma = s/n$,*

$$W_S(y) \geq c_Z \, e_1^s e_2^{n-s} \qquad \text{and} \qquad \max_{0 \leq t \leq s} \frac{\mu_t(S)}{\binom{n}{t}} \;\geq\; c_Z \, 2^{-n \, \psi_{\mathrm{cert}}(\sigma)},$$

where $\psi_{\mathrm{cert}}(\sigma) = \log(1 + y) - \sigma \log e_1 - (1 - \sigma) \log e_2$.

Proof. Since $G[S]$ is connected, it has a spanning tree F; since S is dominating, an owner map exists, and its counts satisfy $\sum_v \nu_v = n - s$. Lemmas 7.3 and 7.5 give the first inequality. The second is the first inequality of Lemma 3.9. □

Remark D.4.3 (Why the cumulative cone). By Lemma D.4.1(a) and (b), moving weight from state 2 to state 1, or from state 3 to state 2, never decreases the final value $\mathrm{fin}(\mathrm{msg}_r)$. The nonnegative orthant is also preserved by $\circ$, so Lemma 7.5 holds verbatim with $\mathbb{R}^3_{\geq 0}$ in place of $\mathcal{K}_{\mathrm{cum}}$. The cumulative cone is larger: in its order a deficit in state 3 can be compensated by weight in states 1 or 2 (an attempt with the orthant is reported in Appendix F.2).

D.4.3 A lower bound on uniform lobsters

We prove here the following proposition of Section 7.6, with the notation of that section.

Proposition 7.9 (Uniform $(2,2)$-lobsters)**.** *Let $L \geq 1$, let $n = 7L$ be the number of vertices of Λ_L, and let S^* be the set of its spine vertices and legs. Then:*

(a) S^* is the unique minimum connected dominating set of Λ_L; so $\gamma_{\mathrm{c}}(\Lambda_L) = 3L$ and $\sigma = 3/7$.

(b) Run Algorithm RS-θ with any rate $\psi \geq 0$ in place of ψ^* (and the corresponding cap and trial counts). For $s = \gamma_{\mathrm{c}}(\Lambda_L)$, the probability that s is accepted is at most $P_{\mathrm{nec}}\, 2^{\psi n} \max_t \mu_t(S^*)/\binom{n}{t}$, where $P_{\mathrm{nec}} \leq n^{O(1)}$ is the polynomial factor of Proposition 3.10 (with $\Gamma = 2$).

(c) For all $L \geq 1$ and $0 \leq t \leq n$, $\mu_t(S^*)/\binom{n}{t} \leq (n+1)\, 1.71927^{-n}$. Conversely, $\min_t \binom{n}{t}/\mu_t(S^*) \leq C \cdot 1.71928^n$ for a constant C and all $L \geq 1$. So $\left(\min_t \binom{n}{t}/\mu_t(S^*)\right)^{1/n}$ lies in $[1.71927, 1.71928]$ up to factors $(n+1)^{1/n}$ and $C^{1/n}$; numerically its limit is $1.7192725\ldots$, attained near $y^* = 0.3453$ in the generating-function form.

Consequently, RS-θ run with a rate ψ in place of ψ^* outputs $\gamma_{\mathrm{c}}(\Lambda_L)$ with probability bounded away from 0 on all Λ_L only if $2^\psi \geq 1.71927$; so the certified base 1.7204 exceeds the least base with which this algorithm succeeds on all Λ_L by at most 0.0012.

Proof. (a) Let S be a connected dominating set of Λ_L, where $L \geq 1$, so that $n \geq 7$. If a leg u is not in S, then its two claws, whose only neighbor is u, must lie in S to be dominated; they are not adjacent and every path between them passes through u, contradicting the connectivity of $G[S]$. So all legs lie in S. The neighbors of a leg of x_i are x_i and its two claws, which have no other neighbors, and Λ_L has at least two legs; so a path in $G[S]$ from a leg of x_i to another leg passes through x_i, and $x_i \in S$. Hence $S \supseteq S^*$. Conversely, S^* induces a tree (a caterpillar: Λ_L without its claws) and dominates every claw through its leg, so it is a connected dominating set.

(b) By (a), the only connected dominating set of size at most $s = \gamma_{\mathrm{c}}(\Lambda_L)$ is S^*. A trial accepts only if $\mathcal{A}_{\mathrm{St}}$ is run on a sample X for which some Y with $|Y| \leq s - t$ makes $X \cup Y$ a connected dominating set, so only if $X \cup Y = S^*$ and in particular $X \subseteq S^*$. Hence the probability that s is accepted is at most the expected number of trials that run $\mathcal{A}_{\mathrm{St}}$ on a subset of S^*, which Proposition 3.10 with $\Gamma = 2$ bounds by the displayed quantity.

(c) $G[S^*]$ is a tree and every vertex outside S^* is a claw, so Lemma 7.3 holds with equality, with $\nu = 2$ at the legs and $\nu = 0$ on the spine, and $W_{S^*} = W_{F,\nu}$ for $F = \Lambda_L[S^*]$. Root F at x_L, with x_{i-1} a child of x_i. By Lemma 7.4, the message of x_1 is

$$\mathbf{u}_y = \mathrm{Init}_y(0) \circ \mathrm{Init}_y(2) \circ \mathrm{Init}_y(2) = \left(\tfrac{y}{2}\left(y + \tfrac{1}{4}\right)^2,\ \tfrac{y^2}{4} + \tfrac{y}{8},\ \tfrac{1}{64}\right),$$

and the message of x_i is $\mathbf{u}_y \circ \mathrm{msg}_{x_{i-1}}$ for $i \geq 2$, which is linear in $\mathrm{msg}_{x_{i-1}}$ with matrix

$$\mathbf{M}_y = \begin{pmatrix} 2u_1 & u_1 & u_1 \\ u_2 + u_3 & u_2 & u_2/2 \\ 0 & u_3 & u_3/2 \end{pmatrix}, \qquad (u_1, u_2, u_3) = \mathbf{u}_y.$$

So $W_{S^*}(y) = \mathrm{fin}(\mathbf{M}_y^{L-1}\mathbf{u}_y)$. The entries of $\mathbf{u}_y$ and $\mathbf{M}_y$ are polynomials in y with nonnegative coefficients, so they are nonnegative and nondecreasing in $y \geq 0$. Write $f = (1, 1, \frac{1}{2})$, so that $\mathrm{fin}(P) = f^{\mathsf{T}}P$.

For the first bound, $\mu_t(S^*) = 0$ for $t > 3L$. For $0 \leq t \leq 3L$ put $y_t = t/(n-t) \in [0, \frac{3}{4}]$. As in the proof of Lemma 3.9, $\mu_t(S^*)\, y_t^t \leq W_{S^*}(y_t)$ and $\binom{n}{t} y_t^t \geq (1+y_t)^n/(n+1)$, so (also for $t = 0$, where $y_t = 0$)

$$\mu_t(S^*) \Big/ \binom{n}{t} \leq (n+1)\, W_{S^*}(y_t)/(1+y_t)^n.$$

The certificate of Appendix F.3 covers $[0, \frac{3}{4}]$ by finitely many intervals $[a, b]$ and gives for each a rational vector $w > 0$ and a rational $\lambda > 0$ with $w^{\mathsf{T}}\mathbf{M}_b \leq \lambda w^{\mathsf{T}}$ componentwise, $c\, w^{\mathsf{T}}\mathbf{u}_b \leq \lambda$ for $c = \max_i f_i/w_i$, and $1.71927^7\lambda \leq (1+a)^7$. For $y \in [a, b]$ this gives

$$\begin{aligned} W_{S^*}(y) &\leq f^{\mathsf{T}}\mathbf{M}_b^{L-1}\mathbf{u}_b \leq c\, w^{\mathsf{T}}\mathbf{M}_b^{L-1}\mathbf{u}_b \leq c\,\lambda^{L-1} w^{\mathsf{T}}\mathbf{u}_b \leq \lambda^L \\ &\leq (1+a)^{7L}\, 1.71927^{-n} \leq (1+y)^n\, 1.71927^{-n}, \end{aligned}$$

and with $y = y_t$ the first bound of (c) follows. For the converse, at $y^* = 0.3453$ the same certificate gives a rational vector $v > 0$ and a rational $\lambda' > 0$ with $\mathbf{M}_{y^*} v \geq \lambda' v$ componentwise and $(1+y^*)^7 \leq 1.71928^7\lambda'$. With $c' = \min_i u_i/v_i > 0$ for $\mathbf{u}_{y^*} = (u_1, u_2, u_3)$, we get $W_{S^*}(y^*) \geq c' f^{\mathsf{T}}\mathbf{M}_{y^*}^{L-1} v \geq c'\lambda'^{L-1} f^{\mathsf{T}} v$, and by Lemma 3.9 $\min_t \binom{n}{t}/\mu_t(S^*) \leq (1+y^*)^n/W_{S^*}(y^*) \leq C \cdot 1.71928^n$ with $C = \lambda'/(c' f^{\mathsf{T}} v)$. The numerical value in (c) and values for finite L are reported in Appendix F.3; they are not used in the proof.

For the consequence, suppose $2^\psi < 1.71927$. By (c), $\max_t \mu_t(S^*)/\binom{n}{t} \leq (n+1)\, 1.71927^{-n}$ for all L, so by (b) the probability that γ_{c} is accepted is at most $n^{O(1)}(2^\psi/1.71927)^n$, which tends to 0. As RS-θ never accepts a value below γ_{c}, it then outputs a value above γ_{c} with probability tending to 1. □

The bound does not rely on Λ_L being a tree, on which Max Leaf Spanning Tree is easy: Proposition 7.9 holds for every graph G, with $L \geq 2$, whose unique maximum-leaf spanning tree is Λ_L: part (a) by the argument below, which replaces its proof, and parts (b) and (c) with their proofs verbatim. Indeed, a minimum connected dominating set D of G yields a spanning tree with at least $n - |D|$ leaves (Lemma 5.1), which must be Λ_L, so $D \supseteq S^*$ and $D = S^*$. Next, every vertex of S^* has degree at least 3 in Λ_L (a leg has its spine vertex and two claws; a spine vertex has two legs and, as $L \geq 2$, a spine neighbor), and the leaves of Λ_L are exactly the claws. Let e be an edge of G that is not in Λ_L. If both ends of e lie in S^*, the cycle of $\Lambda_L + e$ runs through the subtree $\Lambda_L[S^*]$; delete an edge $f \neq e$ of it. Each end of f lies in S^* and loses at most one unit of degree, so keeps degree at least 2, and no other degree decreases, so the new spanning tree has exactly the claws as leaves. If $e = cv$ joins a claw c with leg ℓ to a vertex $v \in S^* \setminus \{\ell\}$, delete the edge $c\ell$, which lies on the cycle of $\Lambda_L + e$: then c is still a leaf, ℓ keeps degree 2, and again the leaves are exactly the claws. In both cases we obtain a spanning tree other than Λ_L with as many leaves, contrary to uniqueness. The remaining edges of G outside Λ_L join two claws; they change neither $G[S^*]$ nor the neighbors in S^* of any claw. So $G[S^*] = \Lambda_L[S^*]$ and each claw has exactly one neighbor in S^*, which is what parts (b) and (c) use.

The extremal structure is neither the star of stars of Proposition 5.12, which is tight for Lemma 5.11 but has a smaller base than Λ_L under exponential moments, nor a path. With at most 8 pendant vertices per solution vertex, the base given by the generating function, computed as in Proposition 7.9(c), is at most 1.5878 for stars of stars and at most 1.6972 for paths (numerical; further families in Appendix F.3).

D.5 The closure bound of Algorithm RSC

This subsection proves Proposition 8.6, the lower bound on the closure generating function W_S^{cl} of a minimum connected dominating set S, in the three steps described in Section 8: the tree model (Appendix D.5.1), the ten-state recursion and the light owners (Appendices D.5.2 and D.5.3), and the antinorm certificate (Appendix D.5.4). The notation is that of Section 8.

Proposition 8.6 (Closure generating function of a minimum solution)**.** *Let $p_2 = 0.07$, $p_3 = 0.077$, $y = 31/100$ and $\hat{e} = 145729958/189014575$. For every connected graph G with n vertices and every connected dominating set S of G with $|S| = \gamma_{\mathrm{c}}(G)$,*

$$W_S^{\mathrm{cl}}(y) \;\geq\; 0.6060\,\hat{e}^{\,n}.$$

D.5.1 From the graph to a tree model

From now on G is a connected graph, S a connected dominating set of G with $|S| = s$, F a spanning tree of $G[S]$, and $O = V \setminus S$ the set of *outside* vertices. For $u \in O$ let $N_S(u) = N(u) \cap S$, which is nonempty because S is dominating. An outside vertex u is *private* if $|N_S(u)| = 1$ and *shared* otherwise; a shared vertex has *type* $\tau(u) = 2$ if $|N_S(u)| = 2$ and $\tau(u) = 3$ if $|N_S(u)| \geq 3$. For $X' \subseteq S$ and $v \in S$ let $k_F(v) = |N_F(v) \cap X'|$ be the number of sampled F-neighbors of v (the dependence on X' is suppressed), and let

$$C_F(X') = \{v \in S \setminus X' :\; k_F(v) \geq 2\}$$

be the candidates that the tree F certifies. For $\Upsilon \subseteq C_F(X')$ let

$$\pi_{X'}(\Upsilon) = \prod_{v \in \Upsilon} p_{k_F(v)} \prod_{v \in C_F(X') \setminus \Upsilon} q_{k_F(v)},$$

the probability that a random subset of $C_F(X')$ which contains each v independently with probability $p_{k_F(v)}$ equals Υ. For $X \subseteq S$ put

$$\Psi_F(X) = \big|\{v \in S :\; N_F[v] \cap X = \emptyset\}\big| + \mathrm{comp}(F[X]),$$

and for $u \in O$ and $X' \subseteq X \subseteq S$ put, with $k(u) = |N(u) \cap X'|$,

$$h_u(X', X) = 2^{-\mathbf{1}[N(u) \cap X = \emptyset]}\, q_{k(u)}^{\mathbf{1}[k(u) \geq 2]}.$$

The factor h_u combines the factor $\frac{1}{2}$ that an outside vertex not dominated by the closed sample X contributes to $2^{-|\mathrm{T}(J_X)|}$ with the probability $q_{k(u)}$ that a candidate outside the solution with $k(u)$ sampled neighbors is not put into Υ. Note that $k(u) \leq |N_S(u)|$, so $k(u) \leq 2$ for a vertex of type 2.

We first record that more sampled vertices never mean more terminals. This is why adding candidates inside the solution never decreases $2^{-|\mathrm{T}(J_X)|}$.

Lemma D.5.1 (Monotonicity of the number of terminals)**.** *For $X \subseteq Y \subseteq V$ we have $|\mathrm{T}(J_Y)| \leq |\mathrm{T}(J_X)|$.*

Proof. It suffices to add one vertex $v \notin X$, and we use (13). If $v \notin N[X]$, then $|N[X \cup \{v\}]| \geq |N[X]|+1$, because v is new in the closed neighborhood, and $\mathrm{comp}(G[X \cup \{v\}]) \leq \mathrm{comp}(G[X])+1$. If $v \in N(X)$, then v is adjacent to at least one component of $G[X]$, so $\mathrm{comp}(G[X \cup \{v\}]) \leq \mathrm{comp}(G[X])$, and $N[X \cup \{v\}] \supseteq N[X]$. In both cases $n - |N[\cdot]| + \mathrm{comp}(G[\cdot])$ does not increase. □

We shall use the following consequence of monotonicity several times. If f is a nonnegative function on the subsets of a finite set C that is nondecreasing under inclusion, and $\mu_{\mathbf{p}}$ is the product distribution on the subsets of C that contains each $v \in C$ independently with probability $\mathbf{p}_v$, then $\sum_{\Upsilon} \mu_{\mathbf{p}}(\Upsilon) f(\Upsilon)$ is nondecreasing in each $\mathbf{p}_v$: writing the sum as $\mathbf{p}_v\, \mathbb{E}[f(\Upsilon' \cup \{v\})] + (1-\mathbf{p}_v)\mathbb{E}[f(\Upsilon')]$ with Υ' the random subset of $C \setminus \{v\}$, its derivative in $\mathbf{p}_v$ is $\mathbb{E}[f(\Upsilon' \cup \{v\}) - f(\Upsilon')] \geq 0$. We call this the *monotone coupling*.

The next lemma replaces the graph by the tree F. It keeps the candidates that F certifies, with the (smaller) closure probabilities that F certifies, drops the other candidates inside S, and turns every candidate outside S into a factor $q_{k(u)}$.

Lemma D.5.2 (Closure tree model)**.** *For $0 \leq p_2 \leq p_3 \leq 1$ and $y \geq 0$,*

$$W_S^{\mathrm{cl}}(y) \;\geq\; \sum_{X' \subseteq S} \; \sum_{\Upsilon \subseteq C_F(X')} y^{|X'|}\, \pi_{X'}(\Upsilon)\, 2^{-\Psi_F(X' \cup \Upsilon)} \prod_{u \in O} h_u(X', X' \cup \Upsilon).$$

Proof. Fix $X' \subseteq S$ and let Υ be the random closure of Definition 8.3. Split the candidate set $C(X')$ into three disjoint parts: $C^{\mathrm{out}} = C(X') \cap O$, the set $C_F(X')$, and $C^{\mathrm{rest}} = (C(X') \cap S) \setminus C_F(X')$. These parts cover $C(X')$ because $C_F(X') \subseteq C(X') \cap S$: a vertex $v \in S \setminus X'$ with two F-neighbors in X' has two G-neighbors in X', as $F \subseteq G$. The sets $\Upsilon \cap C^{\mathrm{out}}$, $\Upsilon' = \Upsilon \cap C_F(X')$ and $\Upsilon'' = \Upsilon \cap C^{\mathrm{rest}}$ are independent, and the event $\Upsilon \subseteq S$ is the event $\Upsilon \cap C^{\mathrm{out}} = \emptyset$, which has probability $\prod_{u \in C^{\mathrm{out}}} q_{|N(u) \cap X'|}$. On this event $X' \cup \Upsilon \supseteq X' \cup \Upsilon'$, so $2^{-|\mathrm{T}(J_{X' \cup \Upsilon})|} \geq 2^{-|\mathrm{T}(J_{X' \cup \Upsilon'})|}$ by Lemma D.5.1. The set Υ' contains each $v \in C_F(X')$ independently with probability $p_{|N(v) \cap X'|} \geq p_{k_F(v)}$, because $|N(v) \cap X'| \geq k_F(v)$ and $p_2 \leq p_3$, and $\Upsilon' \mapsto 2^{-|\mathrm{T}(J_{X' \cup \Upsilon'})|}$ is nondecreasing by Lemma D.5.1; so the monotone coupling gives

$$\mathbb{E}_{\Upsilon}\Big[\mathbf{1}[\Upsilon \subseteq S]\, 2^{-|\mathrm{T}(J_{X' \cup \Upsilon})|}\Big] \;\geq\; \prod_{u \in C^{\mathrm{out}}} q_{|N(u) \cap X'|} \sum_{\Upsilon' \subseteq C_F(X')} \pi_{X'}(\Upsilon')\, 2^{-|\mathrm{T}(J_{X' \cup \Upsilon'})|}.$$

A vertex $u \in O$ is not in $X' \subseteq S$, so $u \in C^{\mathrm{out}}$ if and only if $k(u) \geq 2$, and the product over C^{out} equals $\prod_{u \in O} q_{k(u)}^{\mathbf{1}[k(u) \geq 2]}$.

Now fix $X \subseteq S$ (we shall take $X = X' \cup \Upsilon'$). By (13), $|\mathrm{T}(J_X)| = |S \setminus N[X]| + |O \setminus N[X]| + \mathrm{comp}(G[X])$. First, if $v \in S$ is not in $N[X]$, then $N_G[v] \cap X = \emptyset$ and hence $N_F[v] \cap X = \emptyset$, since $N_F[v] \subseteq N_G[v]$. Second, a vertex $u \in O$ is not in X, so $u \notin N[X]$ if and only if $N(u) \cap X = \emptyset$. Third, $F[X]$ is a spanning subgraph of $G[X]$, because $X \subseteq S = V(F)$, so $\mathrm{comp}(G[X]) \leq \mathrm{comp}(F[X])$; for $X = \emptyset$ both sides are 0. Together,

$$|\mathrm{T}(J_X)| \;\leq\; \Psi_F(X) + \sum_{u \in O} \mathbf{1}[N(u) \cap X = \emptyset].$$

Inserting this with $X = X' \cup \Upsilon'$ into the previous display, multiplying by $y^{|X'|}$ and summing over $X' \subseteq S$ gives the claim. □

The factors h_u couple the outside vertices with all of their neighbors in S. To obtain a recursion along F we assign every outside vertex to one of its neighbors in S, its owner.

Definition D.5.3 (Owners)**.** An *owner map* is a map $o : O \to S$ with $o(u) \in N_S(u)$ for all $u \in O$. For $v \in S$ let $\nu_v = |o^{-1}(v)|$ be the number of vertices that v owns, so $\sum_{v \in S} \nu_v = n - s$. For $u \in O$ and $X \subseteq S$ let $\bar{h}_u(X) = 2^{-\mathbf{1}[o(u) \notin X]}$.

The following pointwise inequality compares h_u with the owner factor $\bar{h}_u$; it is the only place where the values of h_u are examined. Put

$$\varepsilon_2 = -\ln q_2, \qquad \varepsilon_3 = -\ln q_3, \qquad \delta_2 = \ln 2, \qquad \delta_3 = \ln(2q_3),$$

so $0 \le \varepsilon_2 \le \varepsilon_3$ and $0 \le \delta_3 \le \delta_2$ (recall $q_2 \ge q_3 > \frac{1}{2}$). For $u \in O$ let $N'_u = N_S(u) \setminus \{o(u)\}$, and for $X' \subseteq S$ let $\mathbf{1}_u = \mathbf{1}[N'_u \cap X' \neq \emptyset]$ indicate that some other neighbor of u in S is sampled.

Lemma D.5.4 (Pointwise bound)**.** *Let $0 < p_2 \le p_3 < \frac{1}{2}$, let o be an owner map, let $u \in O$ have the owner $o = o(u)$, and let $X' \subseteq X \subseteq S$. Then $h_u(X', X) \ge \bar{h}_u(X)$ whenever $o \notin X$, and:*

(i) if u is private, then $h_u(X', X) \ge \bar{h}_u(X)$;

(ii) if u is shared of type τ, then $h_u(X', X) \ge \bar{h}_u(X)\, q_\tau^{\mathbf{1}[o \in X]}$;

(iii) if u is shared of type τ, $o \notin X$ and $N'_u \cap X \neq \emptyset$, then $h_u(X', X) \ge e^{\delta_\tau}\, \bar{h}_u(X)$;

(iv) if u is shared of type 3, then $\ln\big(h_u(X', X)/\bar{h}_u(X)\big) \ge \mathbf{1}_u\big(\delta_3\, \mathbf{1}[o \notin X] - \varepsilon_3\, \mathbf{1}[o \in X]\big)$;

(v) if u is shared of type 2, then $\ln\big(h_u(X', X)/\bar{h}_u(X)\big) \ge \mathbf{1}_u\big(\delta_2\, \mathbf{1}[o \notin X] - \varepsilon_2\, \mathbf{1}[o \in X']\big)$.

Proof. Suppose first that $o \notin X$, so $\bar{h}_u(X) = \frac{1}{2}$ and $o \notin X'$. If $N'_u \cap X \neq \emptyset$, then u is dominated and $h_u(X', X) = q_{k(u)}^{\mathbf{1}[k(u) \ge 2]} \ge q_3$; so $h_u/\bar{h}_u \ge 2q_3 = e^{\delta_3}$, and if moreover $\tau(u) = 2$, then $k(u) \le |N'_u| = 1$ gives $h_u = 1$ and $h_u/\bar{h}_u = 2 = e^{\delta_2}$. This proves (iii), and it covers the case $\mathbf{1}_u = 1$ of (iv) and (v), since $X' \subseteq X$. If $N'_u \cap X = \emptyset$, then $N(u) \cap X' = \emptyset$, so $h_u(X', X) = 2^{-\mathbf{1}[N(u) \cap X = \emptyset]} \ge \frac{1}{2} = \bar{h}_u(X)$. In all cases $h_u \ge \bar{h}_u$ when $o \notin X$.

Suppose now that $o \in X$, so $\bar{h}_u(X) = 1$ and u is dominated; then $h_u(X', X) = q_{k(u)}^{\mathbf{1}[k(u) \ge 2]}$. If $\mathbf{1}_u = 0$, then $N(u) \cap X' \subseteq \{o\}$, so $h_u = 1$. If $\mathbf{1}_u = 1$, then $h_u \ge q_3$, which gives (iv); if moreover $\tau(u) = 2$, then $k(u) \le 2$ and $k(u) = 2$ only if $o \in X'$, so $h_u \ge q_2^{\mathbf{1}[o \in X']}$, which gives (v); and in both cases $h_u \ge q_{\tau(u)}$, which gives (ii), as $q_\tau \le 1$. For (i), a private u has $N'_u = \emptyset$, so $k(u) \le 1$ and $h_u = 1$ when $o \in X$. □

Replacing h_u by $\bar{h}_u q_\tau^{\mathbf{1}[o(u) \in X]}$ for every shared vertex would give a model of product form along the tree. We call the factor $q_\tau^{\mathbf{1}[o(u) \in X]}$ an *exposure* of the owner of u, of type $\tau = \tau(u)$. Exposing every shared vertex loses too much (for $p_2 = p_3 = 1/20$, about $2^{0.012n}$ in the running time on the uniform $(2,2)$-lobster Λ_L of Proposition 7.9, numerically), because the exposure applies the factor q_τ to a shared vertex whenever its owner is in X, although $h_u < 1$ requires at least two sampled neighbors. For some shared vertices we therefore keep the exact factor h_u and bound its effect by a correction term, using Jensen's inequality. For the others, (iii) recovers part of the loss: when the owner is outside the closed sample, an exposed vertex is usually dominated by one of its other neighbors, and then h_u is e^{δ_τ} times the owner factor $\frac{1}{2}$. The tree model records this as a *credit* $\varpi_\tau \ge 1$ on the factor of an owner outside the closed sample.

Definition D.5.5 (Tree model)**.** Let o be an owner map, let the shared vertices be partitioned into a set O^{co} of *corrected* vertices and a set O^{ex} of *exposed* vertices, and let $\varpi_2, \varpi_3 \ge 1$ be real numbers, the *credits*. For $v \in S$ let ν_v be as in Definition D.5.3, let $c_{v,\tau} = |\{u \in o^{-1}(v) \cap O^{\mathrm{ex}} : \tau(u) = \tau\}|$ for

$\tau \in \{2,3\}$ be the numbers of exposed vertices of each type that v owns, and $c_v = c_{v,2} + c_{v,3} \le \nu_v$. For $X \subseteq S$ let

$$\hat{h}_v(X) = \begin{cases} q_2^{c_{v,2}} q_3^{c_{v,3}} & \text{if } v \in X, \\ 2^{-\nu_v} \varpi_2^{c_{v,2}} \varpi_3^{c_{v,3}} & \text{if } v \notin X, \end{cases}$$

and for $X' \subseteq S$ and $\Upsilon \subseteq C_F(X')$ let

$$\mathrm{wt}(X', \Upsilon) = y^{|X'|}\, \pi_{X'}(\Upsilon)\, 2^{-\Psi_F(X' \cup \Upsilon)} \prod_{v \in S} \hat{h}_v(X' \cup \Upsilon), \qquad W^{\mathrm{tm}} = \sum_{X' \subseteq S} \sum_{\Upsilon \subseteq C_F(X')} \mathrm{wt}(X', \Upsilon).$$

Let $\Pr_{\mathrm{tm}}$ be the probability distribution on the pairs (X', Υ) with $\Pr_{\mathrm{tm}}(X', \Upsilon) = \mathrm{wt}(X', \Upsilon)/W^{\mathrm{tm}}$, and write $X = X' \cup \Upsilon$ for the closed sample under $\Pr_{\mathrm{tm}}$; we write $\mathbb{E}_{\mathrm{tm}}$ for the expectation under $\Pr_{\mathrm{tm}}$.

The value W^{tm} depends only on F, y, p_2, p_3, ϖ_2, ϖ_3 and the numbers ν_v, $c_{v,2}$ and $c_{v,3}$; we write $W^{\mathrm{tm}}(F, \nu, c)$ when this matters. It is positive: the pair $(\emptyset, \emptyset)$ has weight $2^{-s} \prod_v 2^{-\nu_v} \varpi_2^{c_{v,2}} \varpi_3^{c_{v,3}} > 0$, since $C_F(\emptyset) = \emptyset$. For $u \in O$ put $\tilde{h}_u = h_u(X', X)/\bar{h}_u(X)$, a random variable under $\Pr_{\mathrm{tm}}$; it is positive when $p_3 < 1$. For an exposed u of type τ with owner o put also

$$\tilde{h}_u^{\mathrm{ex}} = \frac{h_u(X', X)}{\bar{h}_u(X)\, q_\tau^{\mathbf{1}[o \in X]}\, \varpi_\tau^{\mathbf{1}[o \notin X]}}.$$

Lemma D.5.6 (Exposures, credits and the Jensen correction)**.** *Let $0 < p_2 \le p_3 < \frac{1}{2}$, $y \ge 0$ and $\varpi_2, \varpi_3 \ge 1$. In the setting of Definition D.5.5,*

$$W_S^{\mathrm{cl}}(y) \;\ge\; W^{\mathrm{tm}} \exp\Big(\sum_{u \in O^{\mathrm{co}}} \mathbb{E}_{\mathrm{tm}}[\ln \tilde{h}_u] + \sum_{u \in O^{\mathrm{ex}}} \mathbb{E}_{\mathrm{tm}}[\ln \tilde{h}_u^{\mathrm{ex}}] \Big).$$

In particular, if all these expectations are nonnegative, then $W_S^{\mathrm{cl}}(y) \ge W^{\mathrm{tm}}$.

Proof. Let $X' \subseteq S$, $\Upsilon \subseteq C_F(X')$ and $X = X' \cup \Upsilon$. Grouping the outside vertices by their owners,

$$\prod_{u \in O} \bar{h}_u(X) \prod_{u \in O^{\mathrm{ex}}} q_{\tau(u)}^{\mathbf{1}[o(u) \in X]} \varpi_{\tau(u)}^{\mathbf{1}[o(u) \notin X]} = \prod_{v \in S} \hat{h}_v(X),$$

because the vertices owned by $v \in X$ contribute 1 each, except the exposed ones, which contribute $q_{\tau(u)}$, and the ν_v vertices owned by $v \notin X$ contribute $\frac{1}{2}$ each, the exposed ones with the extra factor $\varpi_{\tau(u)}$. Hence the summand for (X', Υ) on the right-hand side of Lemma D.5.2 equals

$$\mathrm{wt}(X', \Upsilon) \prod_{u \in O^{\mathrm{ex}}} \tilde{h}_u^{\mathrm{ex}} \prod_{u \in O^{\mathrm{co}}} \tilde{h}_u \prod_{u \text{ private}} \frac{h_u(X', X)}{\bar{h}_u(X)} \;\ge\; \mathrm{wt}(X', \Upsilon) \prod_{u \in O^{\mathrm{ex}}} \tilde{h}_u^{\mathrm{ex}} \prod_{u \in O^{\mathrm{co}}} \tilde{h}_u,$$

because the private factors are at least 1 by Lemma D.5.4(i). Summing over (X', Υ) and using Lemma D.5.2,

$$W_S^{\mathrm{cl}}(y) \;\ge\; W^{\mathrm{tm}}\, \mathbb{E}_{\mathrm{tm}}\Big[\exp\Big(\sum_{u \in O^{\mathrm{co}}} \ln \tilde{h}_u + \sum_{u \in O^{\mathrm{ex}}} \ln \tilde{h}_u^{\mathrm{ex}} \Big)\Big] \;\ge\; W^{\mathrm{tm}} \exp\Big(\mathbb{E}_{\mathrm{tm}}\Big[\sum_{u \in O^{\mathrm{co}}} \ln \tilde{h}_u + \sum_{u \in O^{\mathrm{ex}}} \ln \tilde{h}_u^{\mathrm{ex}} \Big] \Big),$$

where the last step is Jensen's inequality for the convex function exp; all the random variables are positive because $q_3 > 0$. □

Taking expectations in Lemma D.5.4(iv) and (v), for every corrected u,

$$\mathbb{E}_{\mathrm{tm}}[\ln \tilde{h}_u] \;\geq\; \delta_3 \Pr_{\mathrm{tm}}(\mathbf{1}_u = 1,\ o(u) \notin X) - \varepsilon_3 \Pr_{\mathrm{tm}}(\mathbf{1}_u = 1,\ o(u) \in X) \qquad \text{if } \tau(u) = 3, \tag{26}$$

$$\mathbb{E}_{\mathrm{tm}}[\ln \tilde{h}_u] \;\geq\; \delta_2 \Pr_{\mathrm{tm}}(\mathbf{1}_u = 1,\ o(u) \notin X) - \varepsilon_2 \Pr_{\mathrm{tm}}(\mathbf{1}_u = 1,\ o(u) \in X') \qquad \text{if } \tau(u) = 2. \tag{27}$$

For an exposed u of type τ with owner o, Lemma D.5.4(ii) gives $\ln \tilde{h}^{\mathrm{ex}}_u \geq 0$ on the event $o \in X$, and on the event $o \notin X$ we have $\ln \tilde{h}^{\mathrm{ex}}_u = \ln \tilde{h}_u - \ln \varpi_\tau$ with $\ln \tilde{h}_u \geq \delta_\tau \mathbf{1}[N'_u \cap X \neq \emptyset]$ by (iii) and the first assertion of the lemma; hence

$$\mathbb{E}_{\mathrm{tm}}[\ln \tilde{h}^{\mathrm{ex}}_u] \;\geq\; \Pr_{\mathrm{tm}}(o \notin X)\Big(\delta_\tau \Pr_{\mathrm{tm}}(N'_u \cap X \neq \emptyset \mid o \notin X) - \ln \varpi_\tau\Big) \tag{28}$$

Every event of the form $U \cap X = \emptyset$ has positive probability under $\Pr_{\mathrm{tm}}$, because the pair $(\emptyset, \emptyset)$ has positive weight; in particular $\Pr_{\mathrm{tm}}(o \notin X) > 0$, so the conditional probability is defined. Appendix D.5.3 shows that the right-hand sides of (26) and (27) are nonnegative when the owner $o(u)$ is *light* for the type of u (Definition D.5.10): it owns few vertices and has small degree in F. It also bounds the conditional probability in (28) from below, which yields the admissible credits. To compute the probabilities we need a recursion for the tree model.

D.5.2 The ten-state recursion

As in Lemma 7.4, the tree model is a sum over configurations of a product of local factors, which can be evaluated from the leaves of F upwards. The weight $\mathrm{wt}(X', \Upsilon)$ factorizes over the vertices and edges of F. Since $F[X]$ is a forest, $\mathrm{comp}(F[X]) = |X| - |E(F[X])|$, so

$$2^{-\Psi_F(X)} = \prod_{v \in X} \tfrac{1}{2} \prod_{vw \in E(F[X])} 2 \prod_{v \in S:\, N_F[v] \cap X = \emptyset} \tfrac{1}{2};$$

and $\pi_{X'}(\Upsilon)$ is the product, over the vertices $v \in S \setminus X'$, of $p_{k_F(v)}$ if $v \in \Upsilon$, of $q_{k_F(v)}$ if $v \in C_F(X') \setminus \Upsilon$, and of 1 otherwise. The factor of a vertex depends on its own status and on how many of its F-neighbors lie in X' and in X; since p_k and q_k depend on k only through $\min(k, 3)$, a vertex has to count its sampled neighbors up to three. Write

$$\lambda = p_3/p_2 \;\geq 1.$$

A message records the status of a vertex together with the information about its children that is still needed once the status of its parent is known.

States. Root F at a vertex r. Fix a vertex v and a configuration on the subtree formed by v and some of its children's subtrees. The *state* of v is one of the following ten; "child" refers to the children included so far, and "sampled" means "in X'."

- A: $v \in X'$.
- C_0, C_1, C_2, C_3: $v \in \Upsilon$, and v has 0, 1, 2, or at least 3 sampled children. When the configuration is completed, $v \in \Upsilon$ is allowed only if $v \in C_F(X')$, that is, if the number of sampled children plus $\mathbf{1}[\text{parent} \in X']$ is at least 2.
- O_{00}: $v \notin X$, and no child of v is in X.
- O_{01}: $v \notin X$, no child of v is sampled, and some child is in Υ.

- O_1, O_2, O_3: $v \notin X$, and exactly one, exactly two, or at least three children of v are sampled.

A *message* is a vector $P \in \mathbb{R}^{10}$ indexed by these states, in this order; we write its coordinates as $P = (A, C_0, C_1, C_2, C_3, O_{00}, O_{01}, O_1, O_2, O_3)$ for a parent and $Q = (a, c_0, c_1, c_2, c_3, o_{00}, o_{01}, o_1, o_2, o_3)$ for a child. (These letters denote numbers; the states C_j are those of the vertices in the closure Υ.) The weight of a vertex in Υ is carried as $\frac{p_2}{2}$, and the factor λ that turns p_2 into p_3 is applied when the vertex turns out to have at least three sampled neighbors.

Merge. For a message Q of a child let

$$\chi_{\mathsf{S}}(Q) = 2a + 2c_1 + 2\lambda(c_2 + c_3) + o_{00} + o_{01} + q_2 o_1 + q_3(o_2 + o_3),$$
$$\chi^{\mathrm{s}}_{\mathsf{C}}(Q) = 2a, \qquad \chi^{\mathrm{u}}_{\mathsf{C}}(Q) = 2c_2 + 2\lambda c_3 + o_{00} + o_{01} + o_1 + q_2 o_2 + q_3 o_3,$$
$$\chi^{\mathrm{s}}_{\mathsf{O}}(Q) = a, \qquad \chi^{\mathrm{x}}_{\mathsf{O}}(Q) = c_2 + \lambda c_3, \qquad \chi^{\mathrm{u}}_{\mathsf{O}}(Q) = \tfrac{1}{2}o_{00} + o_{01} + o_1 + q_2 o_2 + q_3 o_3,$$

and $\chi_{\mathsf{O}}(Q) = \chi^{\mathrm{s}}_{\mathsf{O}}(Q) + \chi^{\mathrm{x}}_{\mathsf{O}}(Q) + \chi^{\mathrm{u}}_{\mathsf{O}}(Q) = a + c_2 + \lambda c_3 + \frac{1}{2}o_{00} + o_{01} + o_1 + q_2 o_2 + q_3 o_3$. These are the total weights that the child contributes when the parent is sampled (χ_{S}); when the parent is in Υ, split by whether the child is sampled ($\chi^{\mathrm{s}}_{\mathsf{C}}$) or not ($\chi^{\mathrm{u}}_{\mathsf{C}}$); and when the parent is not in X, split by whether the child is sampled ($\chi^{\mathrm{s}}_{\mathsf{O}}$), is in Υ ($\chi^{\mathrm{x}}_{\mathsf{O}}$), or neither ($\chi^{\mathrm{u}}_{\mathsf{O}}$). Writing $\chi^{\mathrm{s}}_{\mathsf{C}}$ for $\chi^{\mathrm{s}}_{\mathsf{C}}(Q)$ and so on, define

$$P \circ Q = \begin{pmatrix} A\,\chi_{\mathsf{S}}(Q) \\ C_0\chi^{\mathrm{u}}_{\mathsf{C}} \\ C_1\chi^{\mathrm{u}}_{\mathsf{C}} + C_0\chi^{\mathrm{s}}_{\mathsf{C}} \\ C_2\chi^{\mathrm{u}}_{\mathsf{C}} + C_1\chi^{\mathrm{s}}_{\mathsf{C}} \\ C_3(\chi^{\mathrm{u}}_{\mathsf{C}} + \chi^{\mathrm{s}}_{\mathsf{C}}) + C_2\chi^{\mathrm{s}}_{\mathsf{C}} \\ O_{00}\chi^{\mathrm{u}}_{\mathsf{O}} \\ O_{01}(\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}}) + O_{00}\chi^{\mathrm{x}}_{\mathsf{O}} \\ O_1(\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}}) + (O_{00} + O_{01})\chi^{\mathrm{s}}_{\mathsf{O}} \\ O_2(\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}}) + O_1\chi^{\mathrm{s}}_{\mathsf{O}} \\ O_3(\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}} + \chi^{\mathrm{s}}_{\mathsf{O}}) + O_2\chi^{\mathrm{s}}_{\mathsf{O}} \end{pmatrix}^{\top},$$

and

$$\mathrm{fin}(P) = A + C_2 + \lambda C_3 + \tfrac{1}{2}O_{00} + O_{01} + O_1 + q_2 O_2 + q_3 O_3.$$

The operation $\circ$ is bilinear and maps nonnegative messages to nonnegative messages.

Initial messages and forcing. For integers $\nu \geq 0$ and $c_2, c_3 \geq 0$ let

$$\mathrm{Init}(\nu; c_2, c_3) = \big(q_2^{c_2} q_3^{c_3}\tfrac{y}{2},\ q_2^{c_2} q_3^{c_3}\tfrac{p_2}{2},\ 0,\ 0,\ 0,\ 2^{-\nu}\varpi_2^{c_2}\varpi_3^{c_3},\ 0,\ 0,\ 0,\ 0\big).$$

A *forcing* is a map f from a subset of S to $\{\mathsf{in}, \mathsf{out}, \mathsf{excl}\}$; it restricts the configurations to those with $v \in X'$ if $f(v) = \mathsf{in}$, with $v \notin X'$ if $f(v) = \mathsf{out}$, and with $v \notin X$ if $f(v) = \mathsf{excl}$ (the vertex is *excluded*; it may be a candidate, but it is not put into Υ). For a forced vertex we use the *forced initial message*: $\mathrm{Init}(\nu; c_2, c_3)$ with all coordinates except the first set to 0 if $f(v) = \mathsf{in}$, with the first coordinate set to 0 if $f(v) = \mathsf{out}$, and with the first two coordinates set to 0 if $f(v) = \mathsf{excl}$. Note that $f(v) = \mathsf{out}$ still allows $v \in \Upsilon$.

Definition D.5.7 (Messages)**.** Let F be rooted at r, let $\nu, c_2, c_3 : S \to \mathbb{Z}_{\geq 0}$, let $0 < p_2 \leq p_3 < 1$, $y \geq 0$, $\varpi_2, \varpi_3 \geq 1$, and let f be a forcing. For every $v \in S$ fix an order $w_1, \ldots, w_d$ of its children. The *messages* M_v, $v \in S$, are defined from the leaves upwards by

$$M_v = \Big(\cdots\big(\mathrm{Init}_f(\nu_v; c_{v,2}, c_{v,3}) \circ M_{w_1}\big) \circ \cdots\Big) \circ M_{w_d},$$

where Init_f is the forced initial message if v is forced and Init otherwise.

Lemma D.5.8 (Ten-state recursion)**.** *Let M_v, $v \in S$, be the messages of Definition D.5.7, for any orders of the children. Then every M_v is nonnegative, and with the weights* wt *of Definition D.5.5*

$$\begin{aligned} A_r &= \sum_{(X',\Upsilon)\models f,\ r\in X'} \mathrm{wt}(X',\Upsilon), \\ (C_2)_r + \lambda(C_3)_r &= \sum_{(X',\Upsilon)\models f,\ r\in \Upsilon} \mathrm{wt}(X',\Upsilon), \\ \tfrac{1}{2}(O_{00})_r + (O_{01})_r + (O_1)_r + q_2(O_2)_r + q_3(O_3)_r &= \sum_{(X',\Upsilon)\models f,\ r\notin X} \mathrm{wt}(X',\Upsilon), \end{aligned}$$

where $M_r = (A_r, \dots, (O_3)_r)$ and $(X',\Upsilon) \models f$ ranges over the pairs with $X' \subseteq S$ and $\Upsilon \subseteq C_F(X')$ that respect the forcing. In particular $\mathrm{fin}(M_r)$ *is the sum of* wt *over these pairs, and* $\mathrm{fin}(M_r) = W^{\mathrm{tm}}$ *for the empty forcing.*

Proof. The proof assigns to every partial configuration on a subtree a weight that collects those factors of $\mathrm{wt}(X',\Upsilon)$ which are already determined inside the subtree, and it shows by induction along the merges that each message sums these weights state by state. At the root nothing remains undetermined, and the displayed sums follow.

For a vertex v and $0 \le i \le d$ let F_v^i be the subtree formed by v and the subtrees of $w_1, \dots, w_i$; write F_v for F_v^d. A *partial configuration* on a vertex set U is a pair (X'_U, Υ_U) of disjoint subsets of U respecting the forcing. For a partial configuration on $V(F_v^i)$, define its *partial weight* as the product of the following factors, where k denotes the number of sampled F-neighbors of the vertex in question:

- for every vertex $x \neq v$ of F_v^i (whose F-neighbors all lie in F_v^i, since the parent of x is in F_v^i): the *complete vertex factor*, namely $\frac{y}{2} q_2^{c_{x,2}} q_3^{c_{x,3}}$ if $x \in X'$; $\frac{p_k}{2} q_2^{c_{x,2}} q_3^{c_{x,3}}$ if $x \in \Upsilon$ and $k \ge 2$, and 0 if $x \in \Upsilon$ and $k \le 1$; if $x \notin X' \cup \Upsilon$, the product of $2^{-\nu_x} \varpi_2^{c_{x,2}} \varpi_3^{c_{x,3}}$, of q_k if $k \ge 2$, and of $\frac{1}{2}$ if no F-neighbor of x is in $X' \cup \Upsilon$;
- for v itself, the *partial vertex factor*: $\frac{y}{2} q_2^{c_{v,2}} q_3^{c_{v,3}}$ if $v \in X'$, $\frac{p_2}{2} q_2^{c_{v,2}} q_3^{c_{v,3}}$ if $v \in \Upsilon$, and $2^{-\nu_v} \varpi_2^{c_{v,2}} \varpi_3^{c_{v,3}}$ if $v \notin X' \cup \Upsilon$;
- a factor 2 for every edge of F_v^i with both ends in $X' \cup \Upsilon$.

By the factorization displayed before the definition of the states, the product of the complete vertex factors of all vertices of S and of the edge factors is $\mathrm{wt}(X',\Upsilon)$ for $\Upsilon \subseteq C_F(X')$, and 0 if Υ contains a vertex with fewer than two sampled F-neighbors. (For the vertex factor of $x \in \Upsilon$, the requirement $x \in C_F(X')$ is exactly that x has at least two sampled F-neighbors; the factor $\frac{1}{2}$ per vertex of X, the factor p_k or q_k of $\pi_{X'}$, and the factor $\hat{h}_x$ are all included.) Let $M_v^i \in \mathbb{R}^{10}$ be the vector whose coordinate for a state is the sum of the partial weights of the partial configurations on $V(F_v^i)$ in which v is in that state. We show by induction that M_v^i is the message obtained after merging $w_1, \dots, w_i$; for $i = d$ this is M_v, which is then nonnegative.

For $i = 0$, $F_v^0 = \{v\}$; the configurations are $v \in X'$ (state A, weight $\frac{y}{2} q_2^{c_{v,2}} q_3^{c_{v,3}}$), $v \in \Upsilon$ (state C_0, weight $\frac{p_2}{2} q_2^{c_{v,2}} q_3^{c_{v,3}}$) and $v \notin X' \cup \Upsilon$ (state O_{00}, weight $2^{-\nu_v} \varpi_2^{c_{v,2}} \varpi_3^{c_{v,3}}$), restricted by the forcing. This is $\mathrm{Init}_f(\nu_v; c_{v,2}, c_{v,3})$.

For the step, let $w = w_i$ and $Q = M_w$, which by induction collects the partial configurations of F_w with w as the special vertex. A partial configuration on $V(F_v^i)$ is a pair of partial configurations on $V(F_v^{i-1})$ and $V(F_w)$. Its partial weight is the product of their partial weights, of the factor that completes the vertex factor of w (now that the status of its parent v is known), and of the factor 2 if $v, w \in X$; the partial vertex factor of v does not change. We list, for each status of v and each

state of w, the completing factor, the edge factor, and the new state of v. If w is in state C_j or O_j, it has j sampled children (counted up to 3, with $j = 0$ for O_{00} and O_{01}).

- $v \in X'$. If w is in state A, the edge gives 2. If w is in state C_j, it has $k = j + 1$ sampled neighbors, so it is a valid candidate if and only if $j \geq 1$, with the completing factor p_k/p_2, which is 1 for $j = 1$ and λ for $j \geq 2$; the edge gives 2. If w is in an O-state, it is dominated by v, and it has $k = j + 1$ sampled neighbors, so it is a candidate not put into Υ, with factor q_k, if and only if $j \geq 1$: q_2 in O_1 and q_3 in O_2 and O_3. The state of v remains A. The total is $A\,\chi_{\mathsf{S}}(Q)$.
- $v \in \Upsilon$. A sampled child contributes $2a = \chi^{\mathrm{s}}_{\mathsf{C}}$ (edge factor 2) and raises the number of sampled children of v by one. A child in Υ has an unsampled parent, so it has $k = j$ sampled neighbors and is valid only in the states C_2 (factor 1) and C_3 (factor λ), with edge factor 2; a child not in X is dominated by v, has $k = j$ sampled neighbors, and so gets the factor q_2 in state O_2 and q_3 in state O_3. These cases contribute $\chi^{\mathrm{u}}_{\mathsf{C}}$ and leave the count of v unchanged. This gives the second to fifth coordinates of $P \circ Q$, with the count capped at 3.
- $v \notin X$. A sampled child contributes $a = \chi^{\mathrm{s}}_{\mathsf{O}}$ and moves v from O_{00} or O_{01} to O_1, from O_1 to O_2, and from O_2 or O_3 to O_3. A child in Υ is valid only in the states C_2 and C_3, contributes $c_2+\lambda c_3 = \chi^{\mathrm{x}}_{\mathsf{O}}$, dominates v, and moves v from O_{00} to O_{01}. A child not in X has no dominating parent: it gets $\frac{1}{2}$ in state O_{00}, q_2 in state O_2 and q_3 in state O_3; this contributes $\chi^{\mathrm{u}}_{\mathsf{O}}$ and leaves the state of v unchanged. This gives the last five coordinates of $P \circ Q$.

In all cases no other vertex factor changes, because only v and w gain a neighbor with known status. So $M_v^i = M_v^{i-1} \circ M_w$.

At the root, the configurations on $V(F_r) = S$ are complete once the partial vertex factor of r is completed, and r has no parent. In state A nothing remains. In state C_j, r is valid if and only if $j \geq 2$, with the completing factor 1 for $j = 2$ and λ for $j = 3$. In state O_{00}, r is undominated (factor $\frac{1}{2}$); in O_{01} and O_1 it is dominated and has at most one sampled neighbor; in O_2 and O_3 it is a candidate not put into Υ (factor q_2, respectively q_3). This gives the displayed sums, whose total is $\mathrm{fin}(M_r)$. □

For $p_2 = p_3$ the factor λ is 1 and the pairs of states $\mathsf{C}_2, \mathsf{C}_3$ and $\mathsf{O}_2, \mathsf{O}_3$ can be merged, which gives an eight-state recursion; in the limit $p_2 = p_3 \to 0$ the C-coordinates vanish, the states O_{01}, O_1, O_2 and O_3 play the role of one state, and the recursion reduces to the three-state recursion of Lemma 7.4.

D.5.3 Light owners and credits

The probabilities in (26)–(28) are ratios of messages at the root, once F is rooted at the owner or at a neighbor of the exposed vertex. The next lemma compares coordinates of messages; it is the core of the argument. Put

$$\mathsf{b} = 2 + \frac{2\max(p_2,\, p_3 - p_2)}{y}.$$

Throughout this subsection $0 < p_2 \leq p_3 < \frac{1}{2}$, $y > 0$, $1 \leq \varpi_\tau \leq 2q_\tau$ for $\tau \in \{2, 3\}$, and $\nu, c_2, c_3 : S \to \mathbb{Z}_{\geq 0}$ with $c_{v,2} + c_{v,3} \leq \nu_v$ for all v; these are the standing assumptions on the tree model. Note that $\varpi_\tau \leq 2q_\tau$ and $c_v \leq \nu_v$ give

$$2^{-\nu_v}\varpi_2^{c_{v,2}}\varpi_3^{c_{v,3}} \;\leq\; 2^{-\nu_v}(2q_2)^{c_{v,2}}(2q_3)^{c_{v,3}} \;\leq\; q_2^{c_{v,2}}q_3^{c_{v,3}}, \tag{29}$$

that is, the factor $\hat{h}_v$ of Definition D.5.5 is not larger for $v \notin X$ than for $v \in X$.

Lemma D.5.9 (Message inequalities)**.** *Let F be rooted anywhere and let f be a forcing. Then the messages $M_v = (A, C_0, \dots, O_3)$ of Definition D.5.7 satisfy:*

(a) $C_0 \geq \frac{p_2}{2}(O_{00} + O_{01})$ *and* $C_j \geq \frac{p_2}{2} O_j$ *for* $j = 1, 2, 3$*, for every v with $f(v) \neq \mathsf{excl}$;*

(b) $C_0 + C_1 + C_2 + C_3 \leq \frac{p_2}{y} A$*, for every v with $f(v) \neq \mathsf{out}$ none of whose children is excluded;*

(c) $\chi_{\mathsf{S}}(M_v) \leq \mathsf{b}\, \chi_{\mathsf{O}}(M_v)$*, for every v with $f(v) \neq \mathsf{out}$ none of whose children is excluded;*

(d) $\chi_{\mathsf{S}}(M_v) \geq \chi_{\mathsf{O}}(M_v)$ *for every v with $f(v) \neq \mathsf{excl}$, and* $\chi_{\mathsf{S}}(M_v) \geq q_3\, \chi_{\mathsf{O}}(M_v)$ *for every v with $f(v) = \mathsf{excl}$.*

Proof. All messages are nonnegative by Lemma D.5.8. We prove (a) and (b) by induction along the merges that produce M_v, and then derive (c) and (d).

(a) For an initial message with $f(v) = \mathsf{in}$ all coordinates involved are 0. Otherwise $C_1 = C_2 = C_3 = O_1 = O_2 = O_3 = O_{01} = 0$, $C_0 = \frac{p_2}{2} q_2^{c_{v,2}} q_3^{c_{v,3}}$ and $O_{00} = 2^{-\nu_v} \varpi_2^{c_{v,2}} \varpi_3^{c_{v,3}}$, so $C_0 \geq \frac{p_2}{2} O_{00}$ by (29). For a merge with a child message $Q \geq 0$, the merge formulas give

$$\begin{aligned}
C_0' &= C_0\, \chi^{\mathrm{u}}_{\mathsf{C}}, \qquad & O_{00}' + O_{01}' &= (O_{00} + O_{01})(\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}}),\\
C_1' &= C_1 \chi^{\mathrm{u}}_{\mathsf{C}} + C_0 \chi^{\mathrm{s}}_{\mathsf{C}}, \qquad & O_1' &= O_1(\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}}) + (O_{00} + O_{01})\chi^{\mathrm{s}}_{\mathsf{O}},\\
C_2' &= C_2 \chi^{\mathrm{u}}_{\mathsf{C}} + C_1 \chi^{\mathrm{s}}_{\mathsf{C}}, \qquad & O_2' &= O_2(\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}}) + O_1 \chi^{\mathrm{s}}_{\mathsf{O}},\\
C_3' &= C_3(\chi^{\mathrm{u}}_{\mathsf{C}} + \chi^{\mathrm{s}}_{\mathsf{C}}) + C_2 \chi^{\mathrm{s}}_{\mathsf{C}}, \qquad & O_3' &= O_3(\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}} + \chi^{\mathrm{s}}_{\mathsf{O}}) + O_2 \chi^{\mathrm{s}}_{\mathsf{O}},
\end{aligned}$$

and $\chi^{\mathrm{u}}_{\mathsf{C}} - (\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}}) = c_2 + \lambda c_3 + \frac{1}{2} o_{00} \geq 0$, $\chi^{\mathrm{s}}_{\mathsf{C}} = 2\chi^{\mathrm{s}}_{\mathsf{O}} \geq \chi^{\mathrm{s}}_{\mathsf{O}}$. Hence each primed C-coordinate is at least $\frac{p_2}{2}$ times the corresponding primed O-coordinate, term by term; for instance $C_1' \geq \frac{p_2}{2} O_1(\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}}) + \frac{p_2}{2}(O_{00} + O_{01}) \cdot 2\chi^{\mathrm{s}}_{\mathsf{O}} \geq \frac{p_2}{2} O_1'$.

(b) If $f(v) = \mathsf{in}$, the C-coordinates of the initial message are 0, and they stay 0 under merging; if $f(v) = \mathsf{excl}$, both sides are 0. Otherwise the initial message has $C_0 = \frac{p_2}{2} q_2^{c_{v,2}} q_3^{c_{v,3}} = \frac{p_2}{y} A$. For a merge with a child message Q, summing the second to fifth coordinates of $P \circ Q$ gives

$$C_0' + C_1' + C_2' + C_3' = (C_0 + C_1 + C_2 + C_3)(\chi^{\mathrm{u}}_{\mathsf{C}} + \chi^{\mathrm{s}}_{\mathsf{C}}), \qquad A' = A\, \chi_{\mathsf{S}}(Q),$$

and

$$\chi_{\mathsf{S}}(Q) - (\chi^{\mathrm{u}}_{\mathsf{C}} + \chi^{\mathrm{s}}_{\mathsf{C}}) = (2c_1 - p_2\, o_1) + \big(2(\lambda - 1)c_2 - (p_3 - p_2)\, o_2\big) \;\geq\; 0$$

by (a) for the child, which is not excluded: $2c_1 \geq p_2 o_1$, and $2(\lambda - 1)c_2 \geq (\lambda - 1)p_2 o_2 = (p_3 - p_2)o_2$. So the inequality is preserved.

(c) For $M_v = (a, c_0, c_1, c_2, c_3, o_{00}, o_{01}, o_1, o_2, o_3)$,

$$\chi_{\mathsf{S}}(M_v) - 2\chi_{\mathsf{O}}(M_v) = 2c_1 + 2(\lambda - 1)c_2 - o_{01} - (2 - q_2)o_1 - (2q_2 - q_3)o_2 - q_3 o_3 \;\leq\; 2\max(1, \lambda - 1)(c_1 + c_2),$$

since $2q_2 \geq q_3$. By (b), $c_1 + c_2 \leq \frac{p_2}{y} a \leq \frac{p_2}{y} \chi_{\mathsf{O}}(M_v)$, since all terms of χ_{O} are nonnegative, and $2\max(1, \lambda - 1)p_2 = 2\max(p_2, p_3 - p_2)$. Hence $\chi_{\mathsf{S}}(M_v) \leq \mathsf{b}\, \chi_{\mathsf{O}}(M_v)$.

(d) In the same notation,

$$\chi_{\mathsf{S}}(M_v) - \chi_{\mathsf{O}}(M_v) = a + \tfrac{1}{2} o_{00} + \lambda c_3 + (2c_1 - p_2 o_1) + \big((2\lambda - 1)c_2 - (p_3 - p_2)o_2\big),$$

which is nonnegative by (a) if $f(v) \neq \mathsf{excl}$, because $(2\lambda - 1)c_2 \geq (2\lambda - 1)\frac{p_2}{2} o_2 = (p_3 - \frac{p_2}{2})o_2 \geq (p_3 - p_2)o_2$. If $f(v) = \mathsf{excl}$, then $a = c_0 = \cdots = c_3 = 0$ (the initial message has these coordinates 0, and they stay 0), so

$$\chi_{\mathsf{S}}(M_v) - q_3 \chi_{\mathsf{O}}(M_v) = \big(1 - \tfrac{q_3}{2}\big)o_{00} + (1 - q_3)o_{01} + (q_2 - q_3)o_1 + q_3(1 - q_2)o_2 + q_3(1 - q_3)o_3 \;\geq\; 0. \;\square$$

Definition D.5.10 (Light vertices)**.** A pair (ν, d) of nonnegative integers is 3-*light* if

$$2^{-\nu}\mathsf{b}^{-d} \;\geq\; \Big(1+\frac{p_3}{y}\Big)\frac{\varepsilon_3\, y}{\delta_3},$$

and 2-*light* if

$$2^{-\nu}\mathsf{b}^{-d} \;\geq\; \frac{\varepsilon_2\, y}{\delta_2};$$

otherwise it is 3-*failing*, respectively 2-*failing*. Given an owner map, a vertex $v \in S$ is τ-light if $(\nu_v, d_F(v))$ is τ-light, and τ-failing otherwise.

Since $\varepsilon_3 \geq \varepsilon_2$ and $\delta_3 \leq \delta_2$, a 3-light pair is 2-light. The right-hand sides are positive because $q_3 > \frac{1}{2}$; both are small when p_3 is small, because $\varepsilon_\tau \approx p_\tau$.

The following lemma is the place where the shape of the tree enters. Let T_u be the smallest subtree of F that contains $N_S(u)$. If u is shared, T_u has at least two vertices, and its leaves lie in $N_S(u)$; otherwise a leaf outside $N_S(u)$ could be removed.

Lemma D.5.11 (Nonnegative correction at light owners)**.** *Under the standing assumptions of Appendix D.5.3 (in particular $y > 0$), in the setting of Definition D.5.5, let u be a shared vertex of type τ whose owner $o = o(u)$ is τ-light and is a leaf of T_u. Then $\mathbb{E}_{\mathrm{tm}}[\ln \tilde{h}_u] \geq 0$.*

Proof. The proof decomposes the event $\mathbf{1}_u = 1$ into disjoint events, each given by a forcing that uses only the values in and out and forces neither o nor any F-neighbor of o to be out. For each event it bounds the two probabilities in (26), respectively (27), by the same product of child messages at o, using Lemma D.5.9; the lightness of o then gives the sign.

The other neighbors lie in one branch. Since o is a leaf of T_u, it has exactly one neighbor w in T_u, and $T_u - o$ is a tree that contains w and all of $N'_u = N_S(u) \setminus \{o\}$. So N'_u lies in the component of $F - o$ that contains w, and w is the only F-neighbor of o in that component. Hence $N'_u \cap N_F(o) \subseteq \{w\}$.

Decomposition of the event $\mathbf{1}_u = 1$. Let $N^\circ_u = N'_u \setminus \{w\} = \{v_1, \ldots, v_h\}$ in any order, so no v_i is an F-neighbor of o. For $1 \leq i \leq h$ let f_i be the forcing with $f_i(v_i) = \mathsf{in}$ and $f_i(v_j) = \mathsf{out}$ for $j < i$; if $w \in N'_u$, let f_{h+1} be the forcing with $f_{h+1}(w) = \mathsf{in}$ and $f_{h+1}(v_j) = \mathsf{out}$ for all $j \leq h$. The events "(X', Υ) respects f_i" are pairwise disjoint, and their union is the event $N'_u \cap X' \neq \emptyset$, that is, $\mathbf{1}_u = 1$: a sampled vertex of N°_u is matched by the first such v_i, and otherwise w is the sampled vertex of N'_u. No forcing f_i forces an F-neighbor of o to be out, none forces o, and none excludes a vertex. If $\tau = 2$, then N'_u consists of one vertex and there is a single forcing f_1.

One forced event. Fix one of the forcings $f = f_i$, and root F at o; its children are the $d = d_F(o)$ neighbors of o. Let $M_o = (A, C_0, \ldots, O_3)$ be the message of o under f. By Lemma D.5.8,

$$W^{\mathrm{tm}} \Pr_{\mathrm{tm}}(f,\ o \in X') = A, \qquad W^{\mathrm{tm}} \Pr_{\mathrm{tm}}(f,\ o \in X) = A + C_2 + \lambda C_3,$$
$$W^{\mathrm{tm}} \Pr_{\mathrm{tm}}(f,\ o \notin X) = \tfrac{1}{2}O_{00} + O_{01} + O_1 + q_2 O_2 + q_3 O_3,$$

where "f" denotes the event that (X', Υ) respects f. We bound all three by the product $\mathrm{pr}_o = \prod_{w'} \chi_{\mathsf{O}}(M_{w'})$ over the children w' of o. Since o is not forced, the merge formulas give

$$A = q_2^{c_{o,2}} q_3^{c_{o,3}}\, \frac{y}{2} \prod_{w'} \chi_{\mathsf{S}}(M_{w'}) \;\leq\; \frac{y}{2} \prod_{w'} \chi_{\mathsf{S}}(M_{w'}),$$
$$O_{00} + O_{01} + O_1 + O_2 + O_3 = 2^{-\nu_o} \varpi_2^{c_{o,2}} \varpi_3^{c_{o,3}}\, \mathrm{pr}_o \;\geq\; 2^{-\nu_o}\, \mathrm{pr}_o;$$

the first inequality holds because $q_\tau \leq 1$, the second because $\varpi_\tau \geq 1$, and the identity holds because the sum of the last five coordinates of $P \circ Q$ is $(O_{00} + O_{01} + O_1 + O_2 + O_3)\, \chi_{\mathsf{O}}(Q)$. No child w'

of o is forced out and no vertex is excluded, so Lemma D.5.9(c) gives $\chi_{\mathsf{S}}(M_{w'}) \le \mathsf{b}\,\chi_{\mathsf{O}}(M_{w'})$, and Lemma D.5.9(b) at o gives $C_2+\lambda C_3 \le \lambda(C_0+C_1+C_2+C_3) \le \frac{\lambda p_2}{y}A = \frac{p_3}{y}A$. Since $\frac12, 1, q_2, q_3 \ge \frac12$,

$$A \le \frac{y}{2}\,\mathsf{b}^d\,\mathrm{pr}_o, \qquad A+C_2+\lambda C_3 \le \Big(1+\frac{p_3}{y}\Big)\frac{y}{2}\,\mathsf{b}^d\,\mathrm{pr}_o, \qquad \tfrac12 O_{00}+O_{01}+O_1+q_2O_2+q_3O_3 \ge \frac12\,2^{-\nu_o}\,\mathrm{pr}_o.$$

If $\tau = 3$, then, as (ν_o, d) is 3-light,

$$\delta_3\big(\tfrac12 O_{00}+O_{01}+O_1+q_2O_2+q_3O_3\big) - \varepsilon_3(A+C_2+\lambda C_3) \;\ge\; \frac{\mathrm{pr}_o\,\mathsf{b}^d}{2}\Big(\delta_3\,2^{-\nu_o}\mathsf{b}^{-d} - \varepsilon_3\Big(1+\frac{p_3}{y}\Big)y\Big) \;\ge\; 0;$$

if $\tau = 2$, then, as (ν_o, d) is 2-light,

$$\delta_2\big(\tfrac12 O_{00}+O_{01}+O_1+q_2O_2+q_3O_3\big) - \varepsilon_2 A \;\ge\; \frac{\mathrm{pr}_o\,\mathsf{b}^d}{2}\Big(\delta_2\,2^{-\nu_o}\mathsf{b}^{-d} - \varepsilon_2\,y\Big) \;\ge\; 0.$$

Conclusion. Dividing by W^{tm} and summing over the disjoint events f_i gives $\delta_3 \Pr_{\mathrm{tm}}(\mathbf{1}_u = 1,\ o \notin X) \ge \varepsilon_3 \Pr_{\mathrm{tm}}(\mathbf{1}_u = 1,\ o \in X)$ if $\tau = 3$, and $\delta_2 \Pr_{\mathrm{tm}}(\mathbf{1}_u = 1,\ o \notin X) \ge \varepsilon_2 \Pr_{\mathrm{tm}}(\mathbf{1}_u = 1,\ o \in X')$ if $\tau = 2$; then (26), respectively (27), gives $\mathbb{E}_{\mathrm{tm}}[\ln \tilde h_u] \ge 0$. □

The credits rest on a lower bound for the odds that a vertex of S is sampled, given that some other vertices are outside the closed sample.

Lemma D.5.12 (Odds of being sampled)**.** *In the setting of Definition D.5.5, let $U \subseteq S$ and $w \in S \setminus U$ with $|U \cap N_F(w)| \le 1$, and let E be the event that $U \cap X = \emptyset$. Then*

$$\Pr_{\mathrm{tm}}(E,\ w \in X') \;\ge\; \frac{y}{2}\,q_3^{|U\cap N_F(w)|}\,\Pr_{\mathrm{tm}}(E,\ w \notin X).$$

Proof. The event E is the event that (X', Υ) respects the forcing f with $f(v) = \mathsf{excl}$ for $v \in U$. Root F at w and let $M_w = (A, C_0, \dots, O_3)$ be its message under f; w itself is not forced. By Lemma D.5.8 and the identities used in the proof of Lemma D.5.11,

$$W^{\mathrm{tm}}\Pr_{\mathrm{tm}}(E,\ w \in X') = A = \frac{y}{2}\,q_2^{c_{w,2}}q_3^{c_{w,3}}\prod_x \chi_{\mathsf{S}}(M_x),$$
$$W^{\mathrm{tm}}\Pr_{\mathrm{tm}}(E,\ w \notin X) \le O_{00}+\cdots+O_3 = 2^{-\nu_w}\varpi_2^{c_{w,2}}\varpi_3^{c_{w,3}}\prod_x \chi_{\mathsf{O}}(M_x),$$

the products being over the children x of w, that is, over $N_F(w)$. By Lemma D.5.9(d), $\chi_{\mathsf{S}}(M_x) \ge \chi_{\mathsf{O}}(M_x)$ for $x \notin U$ and $\chi_{\mathsf{S}}(M_x) \ge q_3\chi_{\mathsf{O}}(M_x)$ for $x \in U$, and by (29) $q_2^{c_{w,2}}q_3^{c_{w,3}} \ge 2^{-\nu_w}\varpi_2^{c_{w,2}}\varpi_3^{c_{w,3}}$. The claim follows. □

Lemma D.5.13 (Credits)**.** *Put $\vartheta = \frac{y}{2}q_3$ and $\rho_1 = (1+\vartheta)^{-1}$. Under the standing assumptions of Appendix D.5.3, in the setting of Definition D.5.5, let u be an exposed vertex of type τ with owner o. Then*

$$\Pr_{\mathrm{tm}}\big(N'_u \cap X \ne \emptyset \;\big|\; o \notin X\big) \;\ge\; \begin{cases} 1-\rho_1 & \text{if } \tau = 2,\\ 1-\rho_1^2 & \text{if } \tau = 3. \end{cases}$$

Consequently, if

$$\ln \varpi_2 \le (1-\rho_1)\,\delta_2 \qquad \text{and} \qquad \ln \varpi_3 \le (1-\rho_1^2)\,\delta_3, \tag{30}$$

then $\mathbb{E}_{\mathrm{tm}}[\ln \tilde h_u^{\mathrm{ex}}] \ge 0$ for every exposed vertex u.

Proof. Let E_0 be the event $o \notin X$. For $w \in N'_u$, Lemma D.5.12 with $U = \{o\}$ gives $\Pr(E_0, w \in X') \geq \vartheta \Pr(E_0, w \notin X)$, where we write Pr for $\Pr_{\mathrm{tm}}$. Since $\Pr(E_0, w \in X') \leq \Pr(E_0) - \Pr(E_0, w \notin X)$, this gives $(1+\vartheta)\Pr(E_0, w \notin X) \leq \Pr(E_0)$, that is,

$$\Pr(w \notin X \mid E_0) \leq \rho_1. \tag{31}$$

If $\tau = 2$, then $N'_u = \{w\}$ and the claim is (31). If $\tau = 3$, choose $w_1, w_2 \in N'_u$ with $\mathrm{dist}_F(o, w_1) \leq \mathrm{dist}_F(o, w_2)$, and let E_1 be the event $o, w_1 \notin X$, which has positive probability. We apply Lemma D.5.12 to $w = w_2$ and $U = \{o, w_1\}$: both o and w_1 cannot be F-neighbors of w_2, for then the path o, w_2, w_1 would give $\mathrm{dist}_F(o, w_1) = 2 > 1 = \mathrm{dist}_F(o, w_2)$. So $|U \cap N_F(w_2)| \leq 1$, and as above $\Pr(w_2 \notin X \mid E_1) \leq \rho_1$. Hence

$$\Pr(w_1 \notin X,\ w_2 \notin X \mid E_0) = \Pr(w_1 \notin X \mid E_0)\Pr(w_2 \notin X \mid E_1) \leq \rho_1^2,$$

and $\Pr(N'_u \cap X \neq \emptyset \mid E_0) \geq 1 - \rho_1^2$. The last assertion follows from (28): its right-hand side is at least $\Pr(E_0)(\delta_\tau(1 - \rho_1^{\tau-1}) - \ln \varpi_\tau) \geq 0$ under (30). □

The certificate needs one restriction at the leaves of F, which follows from the minimality of S.

Lemma D.5.14 (Private neighbors of leaves)**.** *If S is a minimum connected dominating set with $|S| \geq 2$, then every leaf v of F has a private neighbor, that is, some $u \in O$ with $N_S(u) = \{v\}$.*

Proof. Since $F - v$ is a spanning tree of $S \setminus \{v\} \neq \emptyset$, the set $S \setminus \{v\}$ induces a connected subgraph, and it dominates v, which is adjacent to its F-neighbor. If every outside neighbor of v had a second neighbor in S, then $S \setminus \{v\}$ would also dominate O, so it would be a connected dominating set of size $|S| - 1$, contradicting minimality. □

The certificate of Appendix D.5.4 bounds W^{tm} for the following class of quadruples.

Definition D.5.15 (Admissible quadruples)**.** A quadruple (F, ν, c_2, c_3) of a tree F and maps $\nu, c_2, c_3 : V(F) \to \mathbb{Z}_{\geq 0}$ is *admissible* if, for every $v \in V(F)$ with $d = d_F(v)$ and $c_v = c_{v,2} + c_{v,3}$: $c_v \leq \nu_v$; $c_{v,\tau} = 0$ whenever (ν_v, d) is τ-light or $d = 0$, for $\tau \in \{2, 3\}$; and $c_v \leq \nu_v - 1$ whenever $d = 1$ and $c_v \geq 1$.

The analysis assigns owners and classifies them as follows. A leaf of T_u always exists in $N_S(u)$, so the rule is well defined.

Definition D.5.16 (Owner rule)**.** Choose an owner map o such that $o(u)$ is a leaf of T_u for every shared u (for a private u the owner is its unique neighbor in S). Let ν_v be the resulting numbers of owned vertices, let O^{co} be the set of shared vertices u whose owner is $\tau(u)$-light and O^{ex} the set of shared vertices u whose owner is $\tau(u)$-failing. So $c_{v,\tau} = 0$ if v is τ-light, and $c_{v,\tau}$ is the number of shared vertices of type τ owned by v if v is τ-failing.

Corollary D.5.17 (Reduction to the tree model)**.** *Assume the standing assumptions of Appendix D.5.3 (in particular $y > 0$). Let ϖ_2, ϖ_3 satisfy $1 \leq \varpi_\tau \leq 2q_\tau$ and (30), let S be a minimum connected dominating set with $|S| = s$, and let o, O^{co}, O^{ex}, ν, c_2, c_3 be given by Definition D.5.16. Then (F, ν, c_2, c_3) is admissible, $s + \sum_{v \in S} \nu_v = n$, and $W_S^{\mathrm{cl}}(y) \geq W^{\mathrm{tm}}(F, \nu, c)$.*

Proof. For $u \in O^{\mathrm{co}}$, the owner $o(u)$ is $\tau(u)$-light and is a leaf of T_u, so Lemma D.5.11 gives $\mathbb{E}_{\mathrm{tm}}[\ln \tilde{h}_u] \geq 0$; for $u \in O^{\mathrm{ex}}$, Lemma D.5.13 gives $\mathbb{E}_{\mathrm{tm}}[\ln \tilde{h}_u^{\mathrm{ex}}] \geq 0$. So Lemma D.5.6 gives the inequality. The identity $\sum_{v \in S} \nu_v = n - s$ is from Definition D.5.3. For admissibility, $c_v \leq \nu_v$ because the exposed vertices owned by v are among the vertices it owns, and $c_{v,\tau} = 0$ for τ-light v by Definition D.5.16. Now let $c_v \geq 1$. Then v owns a shared vertex u, so $|N_S(u)| \geq 2$, hence $s \geq 2$ and $d_F(v) \geq 1$, as F is a tree with at least two vertices. If $d_F(v) = 1$, then v is a leaf of F, and Lemma D.5.14 gives a private neighbor u' of v; by Definition D.5.16 v owns u', and u' is not shared, so at most $\nu_v - 1$ of the vertices owned by v are exposed. □

D.5.4 The antinorm certificate

The last step bounds $W^{\mathrm{tm}}(F,\nu,c)$ from below for every admissible quadruple (Definition D.5.15), by the method of Lemma 7.5: a convex region that the merge maps into itself up to a known scale. The certificate works with eight states rather than ten: it forgets whether a vertex has more than two sampled children, except where a sampled parent makes the count matter. This is a relaxation of the tree model, which never overestimates it.

The relaxed model. Root F at r; the relaxed model depends on this choice, and everything below holds for every root. For $X' \subseteq S$ and a candidate $v \in C_F(X')$ put $\tilde p(v) = p_3$ if $v \neq r$, the parent of v is in X' and v has at least two sampled children, and $\tilde p(v) = p_2$ otherwise; so $\tilde p(v) \le p_{k_F(v)}$, since $\tilde p(v) = p_3$ only if $k_F(v) \ge 3$. Let $\tilde\pi_{X'}(\Upsilon) = \prod_{v\in\Upsilon}\tilde p(v)\prod_{v\in C_F(X')\setminus\Upsilon}(1-\tilde p(v))$, let $\widetilde{\mathrm{wt}}(X',\Upsilon)$ be $\mathrm{wt}(X',\Upsilon)$ of Definition D.5.5 with $\tilde\pi_{X'}$ in place of $\pi_{X'}$, and let $\widetilde W(F,\nu,c) = \sum_{X'}\sum_{\Upsilon\subseteq C_F(X')}\widetilde{\mathrm{wt}}(X',\Upsilon)$. For a message $P \in \mathbb{R}^8$ indexed by $\mathsf{A}, \mathsf{C}_0, \mathsf{C}_1, \mathsf{C}_2, \mathsf{O}_{00}, \mathsf{O}_{01}, \mathsf{O}_1, \mathsf{O}_2$, where now C_2 and O_2 mean "at least two sampled children," define the *eight-state merge* $P \mathbin{\tilde\circ} Q$ by the formula for $P\circ Q$ of Appendix D.5.2 with the pairs of coordinates (C_2, C_3) and (O_2, O_3) of a parent merged into C_2 and O_2, that is, by the merge formula of Appendix D.5.2 with the functionals

$$\begin{aligned}
\tilde\chi_{\mathsf{S}}(Q) &= 2a + 2c_1 + 2\lambda c_2 + o_{00} + o_{01} + q_2 o_1 + q_3 o_2,\\
\tilde\chi^{\mathrm{s}}_{\mathsf{C}}(Q) &= 2a, \qquad \tilde\chi^{\mathrm{u}}_{\mathsf{C}}(Q) = 2c_2 + o_{00} + o_{01} + o_1 + q_2 o_2,\\
\tilde\chi^{\mathrm{s}}_{\mathsf{O}}(Q) &= a, \qquad \tilde\chi^{\mathrm{x}}_{\mathsf{O}}(Q) = c_2, \qquad \tilde\chi^{\mathrm{u}}_{\mathsf{O}}(Q) = \tfrac12 o_{00} + o_{01} + o_1 + q_2 o_2,
\end{aligned}$$

the fourth coordinate $C_2(\tilde\chi^{\mathrm{u}}_{\mathsf{C}} + \tilde\chi^{\mathrm{s}}_{\mathsf{C}}) + C_1\tilde\chi^{\mathrm{s}}_{\mathsf{C}}$, the eighth coordinate $O_2(\tilde\chi^{\mathrm{u}}_{\mathsf{O}} + \tilde\chi^{\mathrm{x}}_{\mathsf{O}} + \tilde\chi^{\mathrm{s}}_{\mathsf{O}}) + O_1\tilde\chi^{\mathrm{s}}_{\mathsf{O}}$, and

$$\begin{aligned}
\widetilde{\mathrm{fin}}(P) &= A + C_2 + \tfrac12 O_{00} + O_{01} + O_1 + q_2 O_2,\\
\widetilde{\mathrm{Init}}(\nu; c_2, c_3) &= \big(q_2^{c_2} q_3^{c_3}\tfrac{y}{2},\ q_2^{c_2} q_3^{c_3}\tfrac{p_2}{2},\ 0,\ 0,\ 2^{-\nu}\varpi_2^{c_2}\varpi_3^{c_3},\ 0,\ 0,\ 0\big).
\end{aligned}$$

The *relaxed messages* are $\widetilde M_v = \widetilde{\mathrm{Init}}(\nu_v; c_{v,2}, c_{v,3}) \mathbin{\tilde\circ} \widetilde M_{w_1} \mathbin{\tilde\circ} \cdots \mathbin{\tilde\circ} \widetilde M_{w_d}$, for any orders of the children, merged from left to right as in Definition D.5.7; no forcing is used. In the merge, the closure probability p_3 enters only through the terms $2\lambda c_2$, with $\lambda = p_3/p_2$, and $q_3 o_2$ of $\tilde\chi_{\mathsf{S}}$: a child with at least two sampled children and a sampled parent is closed with probability p_3, and everywhere else the probability is p_2.

Lemma D.5.18 (The relaxed recursion)**.** *Under the standing assumptions of Appendix D.5.3, for every tree F and every choice of the root r, $\widetilde W(F,\nu,c) \le W^{\mathrm{tm}}(F,\nu,c)$ and $\widetilde W(F,\nu,c) = \widetilde{\mathrm{fin}}(\widetilde M_r)$.*

Proof. For the inequality, fix $X' \subseteq S$ and put $g(\Upsilon) = y^{|X'|}2^{-\Psi_F(X'\cup\Upsilon)}\prod_{v\in S}\hat h_v(X'\cup\Upsilon)$ for $\Upsilon \subseteq C_F(X')$, so that $\sum_\Upsilon \pi_{X'}(\Upsilon)g(\Upsilon)$ and $\sum_\Upsilon \tilde\pi_{X'}(\Upsilon)g(\Upsilon)$ are the contributions of X' to W^{tm} and to $\widetilde W$. The function g is nondecreasing under inclusion: let $v \in C_F(X')\setminus\Upsilon$ and $X = X'\cup\Upsilon$. Adding v to X does not increase the number of vertices of S with $N_F[\cdot]\cap X = \emptyset$, and it decreases $\mathrm{comp}(F[X])$, because the at least two F-neighbors of v in $X' \subseteq X$ lie in distinct components of $F[X]$ (a path between two of them in $F[X]$ together with v would be a cycle of F); so $2^{-\Psi_F}$ does not decrease. The factor $\hat h_v$ changes from $2^{-\nu_v}\varpi_2^{c_{v,2}}\varpi_3^{c_{v,3}}$ to $q_2^{c_{v,2}}q_3^{c_{v,3}}$, which is not smaller by (29), and no other factor changes. Since $\pi_{X'}$ and $\tilde\pi_{X'}$ are product distributions on the subsets of $C_F(X')$ with $\tilde p(v) \le p_{k_F(v)}$ for every v, the monotone coupling of Appendix D.5.1 gives $\sum_\Upsilon \tilde\pi_{X'}(\Upsilon)g(\Upsilon) \le \sum_\Upsilon \pi_{X'}(\Upsilon)g(\Upsilon)$; summing over X' gives $\widetilde W \le W^{\mathrm{tm}}$.

For the identity, the proof of Lemma D.5.8 applies with $\tilde p(v)$ in place of $p_{k_F(v)}$ and with the empty forcing: the factor of a vertex w in Υ or in $C_F(X')\setminus\Upsilon$ is now $\tilde p(w)$ or $1-\tilde p(w)$, which is p_3, respectively q_3, exactly when w has a sampled parent and at least two sampled children,

and p_2, respectively q_2, otherwise; so a vertex has to count its sampled children only up to two, the states $\mathsf{C}_2, \mathsf{C}_3$ and $\mathsf{O}_2, \mathsf{O}_3$ can be merged, and in the case analysis of the merge the completing factors become those of $\tilde{\chi}_{\mathsf{S}}$, $\tilde{\chi}^{\mathrm{u}}_{\mathsf{C}}$ and $\tilde{\chi}^{\mathrm{u}}_{\mathsf{O}}$, $\tilde{\chi}^{\mathrm{x}}_{\mathsf{O}}$ above (a child in C_2 or O_2 of a sampled parent gets λ, respectively q_3; of an unsampled parent it gets 1, respectively q_2). The root has no parent, so $\tilde{p}(r) = p_2$, which gives $\widetilde{\mathrm{fin}}$. □

In the rest of this subsection all messages are eight-dimensional, and we write $\circ$, fin and Init for $\tilde{\circ}$, $\widetilde{\mathrm{fin}}$ and $\widetilde{\mathrm{Init}}$. The region of the certificate must be closed upwards in an order of the states that the merge respects. The order is the reflexive and transitive closure of

$$\mathsf{C}_2 \succ \mathsf{C}_1 \succ \mathsf{C}_0, \qquad \mathsf{C}_2 \succ \mathsf{O}_{01} \succ \mathsf{O}_{00}, \qquad \mathsf{O}_{01} \succ \mathsf{O}_1 \succ \mathsf{O}_2,$$

with A incomparable to the other states. Informally, a higher state is never more restrictive: for instance, a vertex in C_1 is valid in every situation in which one in C_0 is, and a vertex in O_{01} is dominated and has no sampled child, so it imposes nothing on its parent that a vertex in O_{00}, O_1 or O_2 does not. A set U of states is an *up-set* if $\mathsf{x} \in U$ and $\mathsf{x}' \succ \mathsf{x}$ imply $\mathsf{x}' \in U$. Let

$$\mathcal{K}_{\mathrm{pos}} = \Big\{ P \in \mathbb{R}^8 : \sum_{\mathsf{x} \in U} P_{\mathsf{x}} \geq 0 \text{ for every up-set } U \Big\},$$

and write $P \succeq P'$ if $P - P' \in \mathcal{K}_{\mathrm{pos}}$. Let e_{x} denote the unit vector of the state x. The next lemma collects the properties of $\mathcal{K}_{\mathrm{pos}}$ that the certificate uses: a merge with a nonnegative message, on either side, maps $\mathcal{K}_{\mathrm{pos}}$ into itself, and fin is nonnegative on $\mathcal{K}_{\mathrm{pos}}$. As for the cumulative cone of Lemma D.4.1, both properties are checked on a finite set of generators.

Lemma D.5.19 (The poset cone)**.** *Let $0 < p_2 \leq p_3 < 1$. The cone $\mathcal{K}_{\mathrm{pos}}$ defined above, together with the eight-state merge $\circ$ and the functional* fin*, has the following properties.*

(a) $\mathcal{K}_{\mathrm{pos}}$ is a closed convex cone containing $\mathbb{R}^8_{\geq 0}$, and it is the convex cone generated by the eight unit vectors e_{x} and the six vectors $e_{\mathsf{x}'} - e_{\mathsf{x}}$ for the covering pairs $\mathsf{x}' \succ \mathsf{x}$ displayed above.

(b) If $Q \in \mathbb{R}^8_{\geq 0}$ and $\Delta \in \mathcal{K}_{\mathrm{pos}}$, then $\Delta \circ Q \in \mathcal{K}_{\mathrm{pos}}$ and $Q \circ \Delta \in \mathcal{K}_{\mathrm{pos}}$.

(c) $\mathrm{fin}(\Delta) \geq 0$ for every $\Delta \in \mathcal{K}_{\mathrm{pos}}$.

Proof. (a) $\mathcal{K}_{\mathrm{pos}}$ is an intersection of closed half-spaces through 0, hence a closed convex cone, and it contains the orthant. Each of the fourteen vectors lies in $\mathcal{K}_{\mathrm{pos}}$: the up-set sums of e_{x} are 0 or 1, and so are those of $e_{\mathsf{x}'} - e_{\mathsf{x}}$ with $\mathsf{x}' \succ \mathsf{x}$, because an up-set containing x contains x'. So the cone they generate is contained in $\mathcal{K}_{\mathrm{pos}}$.

For the converse, note that the seven states other than A form a rooted tree with root C_2, in which C_1 and O_{01} are the children of C_2, C_0 is the child of C_1, O_{00} and O_1 are the children of O_{01}, and O_2 is the child of O_1; and $\mathsf{x} \prec \mathsf{x}'$ holds exactly when x' is a proper ancestor of x. The up-sets are the sets closed under taking parents, with A added or not. Let $P \in \mathcal{K}_{\mathrm{pos}}$. The up-set $\{\mathsf{A}\}$ gives $P_{\mathsf{A}} \geq 0$, and $P - P_{\mathsf{A}} e_{\mathsf{A}}$ again lies in $\mathcal{K}_{\mathrm{pos}}$, so we may assume $P_{\mathsf{A}} = 0$. We show by induction on $|T|$ that a vector P supported on a subtree T containing the root, whose sums over the subsets of T closed under taking parents are nonnegative, is a nonnegative combination of the vectors e_{x} with $\mathsf{x} \in T$ and $e_{\mathsf{y}} - e_{\mathsf{x}}$ with $\mathsf{x} \in T$ and parent y. If $T = \{\mathsf{C}_2\}$, then $P = P_{\mathsf{C}_2} e_{\mathsf{C}_2}$ with $P_{\mathsf{C}_2} \geq 0$. Otherwise let x be a leaf of T with parent y, and put $P' = P - P_{\mathsf{x}} e_{\mathsf{x}}$ if $P_{\mathsf{x}} \geq 0$ and $P' = P - (-P_{\mathsf{x}})(e_{\mathsf{y}} - e_{\mathsf{x}})$ otherwise. In both cases P' is supported on $T - \mathsf{x}$, and $P - P'$ is a nonnegative multiple of a generator. Let $U' \subseteq T - \mathsf{x}$ be closed under taking parents; it is also closed under taking parents in T, as x is a leaf. The sum of P' over U' equals the sum of P over U', except in the second case with

$\mathsf{y} \in U'$, where it equals the sum of P over $U' \cup \{\mathsf{x}\}$, which is again closed under taking parents. So all these sums are nonnegative, and the induction hypothesis applies to P' and $T - \mathsf{x}$.

(b) The operation $\circ$ is bilinear and $\mathcal{K}_{\mathrm{pos}}$ is a convex cone, so by (a) it suffices to check the fourteen generators Δ. For a unit vector $\Delta = e_{\mathsf{x}}$ both products are nonnegative, since $\circ$ maps nonnegative messages to nonnegative messages. For the six generators $e_{\mathsf{x}'} - e_{\mathsf{x}}$ we read both products off the merge formula, with $Q = (a, c_0, c_1, c_2, o_{00}, o_{01}, o_1, o_2)$ and with the functionals $\tilde{\chi}_{\mathsf{S}}, \tilde{\chi}^{\mathrm{s}}_{\mathsf{C}}, \dots$ evaluated at Q, where they are nonnegative (we drop the tildes):

$$\begin{aligned}
(e_{\mathsf{C}_2} - e_{\mathsf{C}_1}) \circ Q &= \chi^{\mathrm{u}}_{\mathsf{C}}\,(e_{\mathsf{C}_2} - e_{\mathsf{C}_1}),\\
(e_{\mathsf{C}_1} - e_{\mathsf{C}_0}) \circ Q &= \chi^{\mathrm{u}}_{\mathsf{C}}\,(e_{\mathsf{C}_1} - e_{\mathsf{C}_0}) + \chi^{\mathrm{s}}_{\mathsf{C}}\,(e_{\mathsf{C}_2} - e_{\mathsf{C}_1}),\\
(e_{\mathsf{C}_2} - e_{\mathsf{O}_{01}}) \circ Q &= (\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}})(e_{\mathsf{C}_2} - e_{\mathsf{O}_{01}}) + \chi^{\mathrm{s}}_{\mathsf{O}}\big[(e_{\mathsf{C}_2} - e_{\mathsf{O}_{01}}) + (e_{\mathsf{O}_{01}} - e_{\mathsf{O}_1})\big]\\
&\qquad + (a + c_2 + \tfrac{1}{2}o_{00})e_{\mathsf{C}_2},\\
(e_{\mathsf{O}_{01}} - e_{\mathsf{O}_{00}}) \circ Q &= \chi^{\mathrm{u}}_{\mathsf{O}}\,(e_{\mathsf{O}_{01}} - e_{\mathsf{O}_{00}}),\\
(e_{\mathsf{O}_{01}} - e_{\mathsf{O}_1}) \circ Q &= (\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}})(e_{\mathsf{O}_{01}} - e_{\mathsf{O}_1}) + \chi^{\mathrm{s}}_{\mathsf{O}}\,(e_{\mathsf{O}_1} - e_{\mathsf{O}_2}),\\
(e_{\mathsf{O}_1} - e_{\mathsf{O}_2}) \circ Q &= (\chi^{\mathrm{u}}_{\mathsf{O}} + \chi^{\mathrm{x}}_{\mathsf{O}})(e_{\mathsf{O}_1} - e_{\mathsf{O}_2}),
\end{aligned}$$

and

$$\begin{aligned}
Q \circ (e_{\mathsf{C}_2} - e_{\mathsf{C}_1}) &= (2(\lambda-1)a,\, 2c_0,\, 2c_1,\, 2c_2,\, 0,\, o_{00} + o_{01},\, o_1,\, o_2),\\
Q \circ (e_{\mathsf{C}_1} - e_{\mathsf{C}_0}) &= 2a\, e_{\mathsf{A}},\\
Q \circ (e_{\mathsf{C}_2} - e_{\mathsf{O}_{01}}) &= ((2\lambda-1)a,\, c_0,\, c_1,\, c_2,\, 0,\, 0,\, 0,\, 0) + o_{00}\,(e_{\mathsf{O}_{01}} - e_{\mathsf{O}_{00}}),\\
Q \circ (e_{\mathsf{O}_{01}} - e_{\mathsf{O}_{00}}) &= \tfrac{1}{2}\,(0,\, 0,\, 0,\, 0,\, o_{00},\, o_{01},\, o_1,\, o_2),\\
Q \circ (e_{\mathsf{O}_{01}} - e_{\mathsf{O}_1}) &= p_2\, a\, e_{\mathsf{A}},\\
Q \circ (e_{\mathsf{O}_1} - e_{\mathsf{O}_2}) &= \big((p_3 - p_2)a,\, p_2c_0,\, p_2c_1,\, p_2c_2,\, p_2o_{00},\, p_2o_{01},\, p_2o_1,\, p_2o_2\big).
\end{aligned}$$

In the third line of the first list, the coefficient of e_{C_2} in $(e_{\mathsf{C}_2} - e_{\mathsf{O}_{01}}) \circ Q$ is $\chi^{\mathrm{u}}_{\mathsf{C}} + \chi^{\mathrm{s}}_{\mathsf{C}}$, and $\chi^{\mathrm{u}}_{\mathsf{C}} + \chi^{\mathrm{s}}_{\mathsf{C}} - \chi_{\mathsf{O}} = a + c_2 + \frac{1}{2}o_{00}$. In the second list, the entries involving λ and p_3 come from the terms $2\lambda c_2$ and $q_3 o_2$ of $\tilde{\chi}_{\mathsf{S}}$; they are nonnegative because $\lambda \geq 1$ and $p_3 \geq p_2$. So every product is a nonnegative combination of generators, and it lies in $\mathcal{K}_{\mathrm{pos}}$.

(c) fin is linear with the coefficients $(1, 0, 0, 1, \frac{1}{2}, 1, 1, q_2)$, so by (a) it suffices that it is nonnegative on the generators: its values on the unit vectors are these coefficients, and its values on the six vectors $e_{\mathsf{x}'} - e_{\mathsf{x}}$ are 1, 0, 0, $\frac{1}{2}$, 0 and p_2, in the order of the covering pairs displayed above. □

For a finite nonempty set $Z \subseteq \mathbb{R}^8_{\geq 0}$ let $\mathcal{P}_Z = \mathrm{conv}(Z) + \mathcal{K}_{\mathrm{pos}}$ and $c_Z = \min_{z \in Z} \mathrm{fin}(z)$. For $\mathcal{Y} \subseteq \mathbb{R}^8$ and $P \in \mathbb{R}^8$ we write $P \in \mathcal{Y} + \mathcal{K}_{\mathrm{pos}}$ also as "P dominates a point of $\mathcal{Y}$." Write $\mathrm{Init}(\nu) = \mathrm{Init}(\nu; 0, 0)$ and $\mathrm{Init}(\infty) = (\frac{y}{2}, \frac{p_2}{2}, 0, 0, 0, 0, 0, 0)$, the limit of $\mathrm{Init}(\nu)$ as $\nu \to \infty$, and let

$$L(\nu) = \{(c_2, c_3) \in \mathbb{Z}^2_{\geq 0} :\ 1 \leq c_2 + c_3 \leq \nu\}$$

be the *exposure lattice*. The initial message is linear in the pair $(q_2^{c_2} q_3^{c_3},\ \varpi_2^{c_2} \varpi_3^{c_3})$:

$$\mathrm{Init}(\nu; c_2, c_3) = q_2^{c_2} q_3^{c_3}\, \mathrm{Init}(\infty) + \varpi_2^{c_2} \varpi_3^{c_3}\, 2^{-\nu} e_{\mathsf{O}_{00}}. \tag{32}$$

The next lemma states the conditions that the certificate verifies. The first two conditions are the analogues of the two conditions of the antinorm certificate for RS-θ (Lemma 7.5), with one scale $\hat{e}$ for solution vertices and owned vertices in place of e_1, e_2, so that the bound $c_Z \hat{e}^n$ does not depend on the solution fraction s/n and no case distinction on s/n is needed; the remaining four control the initial messages of vertices with exposures, which the proof places in a scaled copy of $\mathcal{P}_Z$ either directly or after their first few merges.

Lemma D.5.20 (Certificate conditions)**.** *Let $0 < p_2 \le p_3 < \frac{1}{2}$, $y > 0$, $1 \le \varpi_\tau \le 2q_\tau$ and $0 < \hat{e} < q_3$. Let $Z \subseteq \mathbb{R}^8_{\ge 0}$ be finite and nonempty, let $\nu_0 \ge 0$ and $\nu_E \ge 1$ be integers, and let $d^{(3)}_{\min}, d^{(2)}_{\min} : \{1, \dots, \nu_E\} \to \mathbb{Z}_{\ge 1}$ with $d^{(3)}_{\min} \le d^{(2)}_{\min}$; put $j_\tau(\nu) = \max(1, d^{(\tau)}_{\min}(\nu) - 1)$. Suppose that:*

(C1) $z \circ z' \in \mathcal{P}_Z$ *for all* $z, z' \in Z$*;*

(C2) $\mathrm{Init}(\nu) \in \hat{e}^{1+\nu}\mathcal{P}_Z$ *for* $0 \le \nu \le \nu_0$*, and* $\mathrm{Init}(\infty) \in \hat{e}^{\nu_0+2}\mathcal{P}_Z$*;*

(C3) for $1 \le \nu \le \nu_E$ *and* $\tau \in \{2, 3\}$*, every pair* (ν, d) *with* $1 \le d < d^{(\tau)}_{\min}(\nu)$ *is* τ*-light;*

(C4) for $2 \le \nu \le \nu_E$*: if* $d^{(3)}_{\min}(\nu) \le 1$*, then* $\mathrm{Init}(\nu; 0, c_3) \in \hat{e}^{1+\nu}\mathcal{P}_Z$ *for* $1 \le c_3 \le \nu - 1$*, and if moreover* $d^{(2)}_{\min}(\nu) \le 1$*, then* $\mathrm{Init}(\nu; c_2, c_3) \in \hat{e}^{1+\nu}\mathcal{P}_Z$ *for all* $(c_2, c_3) \in L(\nu - 1)$ *with* $c_2 \ge 1$*;*

(C5) for $1 \le \nu \le \nu_E$ *and* $(c_2, c_3) \in L(\nu)$*, with* $j = j_3(\nu)$ *if* $c_2 = 0$ *and* $j = j_2(\nu)$ *if* $c_2 \ge 1$*:* $\hat{e}^{-1-\nu}\mathrm{Init}(\nu; c_2, c_3) \circ Q_1 \circ \cdots \circ Q_j \in \mathcal{P}_Z$ *(merged from left to right) for all* $Q_1, \dots, Q_j \in \mathcal{P}_Z \cap \mathbb{R}^8_{\ge 0}$*;*

(C6) $(q_3/\hat{e})^{\nu_E+1} \ge \hat{e}^{-(\nu_0+1)}$.

Then every admissible quadruple (F, ν, c_2, c_3) in which F has s vertices satisfies, with $\nu(F) = \sum_{v \in V(F)} \nu_v$,

$$W^{\mathrm{tm}}(F, \nu, c) \;\ge\; \widetilde{W}(F, \nu, c) \;\ge\; c_Z\, \hat{e}^{\,s+\nu(F)}.$$

Proof. The first inequality is Lemma D.5.18, for any root. Step 1 shows that membership in scaled copies of $\mathcal{P}_Z$ is supermultiplicative under the merge. Step 2 places the initial message $\mathrm{Init}(\nu_v; c_{v,2}, c_{v,3})$ of most vertices v in $\hat{e}^{1+\nu_v}\mathcal{P}_Z$, and Step 3 treats the remaining vertices, whose message reaches the right scaled copy only after its first few merges. Step 4 propagates this up the tree and reads off the bound at the root.

Step 1: supermultiplicativity. The merge $\circ$, the cone $\mathcal{K}_{\mathrm{pos}}$ and the set Z satisfy the hypotheses of Lemma D.4.2: $\circ$ is bilinear; by Lemma D.5.19(a), $\mathcal{K}_{\mathrm{pos}}$ is a convex cone containing $\mathbb{R}^8_{\ge 0}$; by Lemma D.5.19(b), $\Delta \circ Q$ and $Q \circ \Delta$ lie in $\mathcal{K}_{\mathrm{pos}}$ for $\Delta \in \mathcal{K}_{\mathrm{pos}}$ and $Q \in \mathbb{R}^8_{\ge 0}$; and $z \circ z' \in \mathcal{P}_Z$ for $z, z' \in Z$ by (C1). Hence Lemma D.4.2 gives $\mathcal{P}_Z + \mathcal{K}_{\mathrm{pos}} = \mathcal{P}_Z$, $\eta\mathcal{P}_Z \subseteq \eta'\mathcal{P}_Z$ for $0 < \eta' \le \eta$, and (its part (b) with $Y = Y' = Z$) $P \circ Q \in \eta\eta'\mathcal{P}_Z$ whenever $P \in \eta\mathcal{P}_Z$, $Q \in \eta'\mathcal{P}_Z$ and $Q \ge 0$. In particular, if $P \ge 0$ and $P \in \eta\mathcal{P}_Z$, then $\eta'' P \in \eta''\eta\,\mathcal{P}_Z \subseteq \eta\mathcal{P}_Z$ for every $\eta'' \ge 1$.

Step 2: initial messages. Let (F, ν, c_2, c_3) be admissible and $v \in V(F)$, with $\nu = \nu_v$, $(c_2, c_3) = (c_{v,2}, c_{v,3})$, $c = c_2 + c_3$ and $d = d_F(v)$. We claim that $\mathrm{Init}(\nu; c_2, c_3) \in \hat{e}^{1+\nu}\mathcal{P}_Z$, unless $c \ge 1$, $\nu \le \nu_E$ and $d \ge 2$; that remaining case is handled in Step 3.

- *Case $c = 0$ and $\nu \le \nu_0$*: this is (C2).
- *Case $c = 0$ and $\nu > \nu_0$*: $\mathrm{Init}(\nu) - \mathrm{Init}(\infty) = 2^{-\nu} e_{\mathsf{O}_{00}} \in \mathcal{K}_{\mathrm{pos}}$, so $\mathrm{Init}(\nu) \in \hat{e}^{\nu_0+2}\mathcal{P}_Z + \mathcal{K}_{\mathrm{pos}} = \hat{e}^{\nu_0+2}\mathcal{P}_Z \subseteq \hat{e}^{1+\nu}\mathcal{P}_Z$ by Step 1, as $\hat{e}^{1+\nu} \le \hat{e}^{\nu_0+2}$ (since $\hat{e} < 1$).
- *Case $c \ge 1$ and $\nu > \nu_E$*: by (32), $\mathrm{Init}(\nu; c_2, c_3) - q_3^\nu \mathrm{Init}(\infty) = (q_2^{c_2} q_3^{c_3} - q_3^\nu)\mathrm{Init}(\infty) + \varpi_2^{c_2}\varpi_3^{c_3} 2^{-\nu} e_{\mathsf{O}_{00}} \ge 0$, because $q_2^{c_2} q_3^{c_3} \ge q_3^c \ge q_3^\nu$; so $\mathrm{Init}(\nu; c_2, c_3) \succeq q_3^\nu \mathrm{Init}(\infty)$. Moreover $\hat{e}^{-1-\nu} q_3^\nu = \hat{e}^{-1}(q_3/\hat{e})^\nu \ge \hat{e}^{-1}(q_3/\hat{e})^{\nu_E+1} \ge \hat{e}^{-\nu_0-2}$ by (C6), as $q_3/\hat{e} > 1$. So $\hat{e}^{-1-\nu}\mathrm{Init}(\nu; c_2, c_3)$ dominates a multiple $\eta''\,\hat{e}^{-\nu_0-2}\mathrm{Init}(\infty)$ with $\eta'' \ge 1$, which lies in $\mathcal{P}_Z$ by (C2) and Step 1.
- *Case $c \ge 1$, $\nu \le \nu_E$ and $d \le 1$*: admissibility excludes $d = 0$ and gives $c \le \nu - 1$ for $d = 1$, so $\nu \ge 2$; it also says that $(\nu, 1)$ is τ-failing for every τ with $c_\tau \ge 1$, so by (C3) $d^{(\tau)}_{\min}(\nu) \le 1$ for these τ. If $c_2 = 0$, then $c_3 \ge 1$, $d^{(3)}_{\min}(\nu) \le 1$ and the first part of (C4) applies. If $c_2 \ge 1$, then $d^{(2)}_{\min}(\nu) \le 1$, hence also $d^{(3)}_{\min}(\nu) \le 1$, and $(c_2, c_3) \in L(\nu - 1)$, so the second part of (C4) applies.

Step 3: failing internal vertices with few owned vertices. Let $c \geq 1$, $\nu \leq \nu_E$ and $d \geq 2$. If $c_2 \geq 1$, then (ν, d) is 2-failing by admissibility, so $d \geq d^{(2)}_{\min}(\nu)$ by (C3) and $d - 1 \geq \max(1, d^{(2)}_{\min}(\nu) - 1) = j_2(\nu)$; if $c_2 = 0$, then (ν, d) is 3-failing, and likewise $d-1 \geq j_3(\nu)$. Let j be $j_2(\nu)$ or $j_3(\nu)$ accordingly. With F rooted as in Step 4, v has at least $d-1 \geq j$ children $w_1, \dots, w_j, \dots$, in the order of merging. Suppose that $\widetilde{M}_{w_i} \in \eta_i \mathcal{P}_Z$ with $\eta_i > 0$ for $i \leq j$ (this will be the induction hypothesis of Step 4). The messages $\widetilde{M}_{w_i}/\eta_i$ are nonnegative and lie in $\mathcal{P}_Z$, so (C5) gives, after j merges and by bilinearity,

$$\mathrm{Init}(\nu; c_2, c_3) \circ \widetilde{M}_{w_1} \circ \cdots \circ \widetilde{M}_{w_j} \;\in\; \hat{e}^{1+\nu} \eta_1 \cdots \eta_j \, \mathcal{P}_Z.$$

Step 4: induction over the tree. Root F at any vertex r, and let $\eta(F_v) = \hat{e}^{|V(F_v)|+\nu(F_v)}$ with $\nu(F_v) = \sum_{x \in V(F_v)} \nu_x$. We show $\widetilde{M}_v \in \eta(F_v)\mathcal{P}_Z$ by induction from the leaves. If Step 2 applies to v, start from $\mathrm{Init}(\nu_v; c_{v,2}, c_{v,3}) \in \hat{e}^{1+\nu_v}\mathcal{P}_Z$ and merge the children one by one; each merge with a child w multiplies the scale by $\eta(F_w)$, by Step 1, as $\widetilde{M}_w \geq 0$. Otherwise Step 3 handles the first j children and Step 1 the remaining ones. At the root, $\widetilde{M}_r = \hat{e}^{s+\nu(F)}(z + \Delta)$ with $z \in \mathrm{conv}(Z)$ and $\Delta \in \mathcal{K}_{\mathrm{pos}}$, so by Lemma D.5.18 and Lemma D.5.19(c)

$$\widetilde{W}(F, \nu, c) = \mathrm{fin}(\widetilde{M}_r) = \hat{e}^{s+\nu(F)}\big(\mathrm{fin}(z) + \mathrm{fin}(\Delta)\big) \;\geq\; \hat{e}^{s+\nu(F)} c_Z,$$

since fin is linear and $\mathrm{fin}(z) \geq c_Z$ for $z \in \mathrm{conv}(Z)$. □

Condition (C5) is established by the certificate in two steps. First, for a point $P \geq 0$ and finite sets $Y_1, \dots, Y_{j-1} \subseteq \mathbb{R}^8_{\geq 0}$ (the *stages*), if every product $x \circ z$ with $x \in Y_{i-1}$ and $z \in Z$ lies in $\mathrm{conv}(Y_i) + \mathcal{K}_{\mathrm{pos}}$ for $i = 1, \dots, j$, where $Y_0 = \{P\}$ and $Y_j = Z$, then $P \circ Q_1 \circ \cdots \circ Q_j \in \mathcal{P}_Z$ for all $Q_i \in \mathcal{P}_Z \cap \mathbb{R}^8_{\geq 0}$: by induction on i, Lemma D.4.2(b) applied to the sets (Y_{i-1}, Z, Y_i) in place of (Y, Z, Y'), with $\eta = \eta' = 1$, gives $P \circ Q_1 \circ \cdots \circ Q_i \in \mathrm{conv}(Y_i) + \mathcal{K}_{\mathrm{pos}}$ (the closure hypothesis of that lemma was checked in Step 1 above). Second, the property in (C5) is preserved under convex combinations of P, because $P \mapsto P \circ Q_1 \circ \cdots \circ Q_j$ is linear and $\mathcal{P}_Z$ is convex; it is preserved when j is increased, by Step 1; and by (32), $\mathrm{Init}(\nu; c_2, c_3)$ for $(c_2, c_3) \in L(\nu)$ is a convex combination of the points $\mathrm{Init}(\nu; c_2', c_3')$ over the vertices (c_2', c_3') of the convex hull of $\big\{(q_2^{c_2} q_3^{c_3}, \varpi_2^{c_2} \varpi_3^{c_3}) : 0 \leq c_2 + c_3 \leq \nu\big\}$, with the same coefficients that express the pair of (c_2, c_3) through the hull vertices; the vertex $(0, 0)$, if used, contributes $\mathrm{Init}(\nu)$, which lies in $\hat{e}^{1+\nu}\mathcal{P}_Z$ by the first two cases of Step 2. Since $j_3(\nu) \leq j_2(\nu)$, it therefore suffices to provide stages for $\hat{e}^{-1-\nu}\mathrm{Init}(\nu; 0, c_3)$ with $j = j_3(\nu)$ for $1 \leq c_3 \leq \nu$ and for $\hat{e}^{-1-\nu}\mathrm{Init}(\nu; c_2, c_3)$ with $j = j_2(\nu)$ for the hull vertices with $c_2 \geq 1$. The same convexity argument reduces the second part of (C4) to the hull vertices of the lattice for $\nu - 1$.

The certificate verifies the conditions of Lemma D.5.20 and the hypotheses of Corollary D.5.17 for the parameters of RSC. Condition (C3) does not involve the certificate: it follows from Lemma D.5.21, which can be checked by hand and is also checked in Lean (Appendix F.9).

Lemma D.5.21 (Light pairs for RSC)**.** *For $p_2 = 0.07$, $p_3 = 0.077$ and $y = 31/100$ the pairs $(1,1)$, $(1,2)$, $(2,1)$ and $(3,1)$ are 3-light, and the pairs $(1,3)$ and $(2,2)$ are 2-light.*

Proof. Here $q_2 = \frac{93}{100}$, $q_3 = \frac{923}{1000}$, $\max(p_2, p_3 - p_2) = p_2$ and $\mathsf{b} = 2 + \frac{14}{31} = \frac{76}{31}$. Since $-\ln(1-p) = \sum_{k \geq 1} p^k/k \leq p + \frac{p^2}{2} + \frac{p^3}{3(1-p)}$ for $0 < p < 1$, we have $\varepsilon_3 \leq \frac{443756467}{5538000000} < 0.08013$ and $\varepsilon_2 \leq \frac{404957}{5580000} < 0.07258$. Since $\ln \frac{1+z}{1-z} = 2\sum_{k \geq 0} \frac{z^{2k+1}}{2k+1} \geq 2z + \frac{2}{3}z^3$ for $0 \leq z < 1$, with $z = \frac{423}{1423}$ we have $\delta_3 = \ln \frac{1846}{1000} > 0.61202$, and with $z = \frac{1}{3}$ and four terms, $\delta_2 = \ln 2 > 0.69313$. Hence, with $1 + \frac{p_3}{y} = \frac{387}{310}$,

$$\Big(1 + \frac{p_3}{y}\Big)\frac{\varepsilon_3\, y}{\delta_3} < \frac{387}{310} \cdot \frac{0.08013 \cdot 0.31}{0.61202} < 0.0507, \qquad \frac{\varepsilon_2\, y}{\delta_2} < \frac{0.07258 \cdot 0.31}{0.69313} < 0.0325.$$

The left-hand side $2^{-\nu}\mathsf{b}^{-d} = 2^{-\nu}(\frac{31}{76})^d$ is decreasing in ν and d, and its values are $\frac{961}{11552} > 0.0831$ at $(1,2)$, $\frac{31}{304} > 0.1019$ at $(2,1)$, $\frac{31}{608} > 0.0509$ at $(3,1)$, which gives the 3-light pairs, and $\frac{29791}{877952} > 0.0339$ at $(1,3)$ and $\frac{961}{23104} > 0.0415$ at $(2,2)$, which gives the 2-light pairs. □

Lemma D.5.22 (Credits for RSC)**.** *For $p_2 = 0.07$, $p_3 = 0.077$, $y = 31/100$, $\varpi_2 = \frac{272657}{250000} = 1.090628$ and $\varpi_3 = \frac{288677}{250000} = 1.154708$, we have $1 \le \varpi_\tau \le 2q_\tau$ and (30) holds.*

Proof. $2q_2 = 1.86$ and $2q_3 = 1.846$. Here $\vartheta = \frac{31}{200} \cdot \frac{923}{1000}$ and $\rho_1 = \frac{200000}{228613}$, so $1 - \rho_1 = \frac{28613}{228613}$ and $1 - \rho_1^2 = \frac{12263903769}{52263903769}$. The two inequalities of (30) read $\ln \varpi_2 \le \frac{28613}{228613} \ln 2$ and $\ln \varpi_3 \le \frac{12263903769}{52263903769}(\ln 2 + \ln q_3)$; numerically the two sides are 0.086753678 versus 0.086753685 and 0.143847499 versus 0.143847506 (left-hand sides rounded up, right-hand sides rounded down), so the credits were chosen as large as the inequalities allow, to six decimals. They follow from the truncated-series bounds $\ln \varpi_\tau \le \sum_{k=1}^{15} (-1)^{k+1}(\varpi_\tau - 1)^k/k$, an alternating series with decreasing terms cut after a positive term; $\ln 2 \ge 2\sum_{k=0}^{6} 3^{-2k-1}/(2k+1)$; and $-\ln q_3 \le \sum_{k=1}^{6} p_3^k/k + p_3^7/(7q_3)$: with these bounds, evaluated exactly in rational arithmetic (Appendix F.4), the right-hand sides exceed the left-hand sides by more than $6 \cdot 10^{-9}$ and $5 \cdot 10^{-9}$. □

What the certificate proves. The certificate is a finite set Z of 1024 points of $\mathbb{R}^8$, and $\mathcal{P}_Z = \mathrm{conv}(Z) + \mathcal{K}_{\mathrm{pos}}$ is the region that they span together with the poset cone. Each of (C1), (C2), (C4) and (C5) is a finite list of statements of the form "this rational vector lies in this region". Together they say two things: the message of a single vertex lies in a scaled copy of $\mathcal{P}_Z$ ((C2), (C4) and (C5), the last after a few merges), and the merge of two messages in scaled copies of $\mathcal{P}_Z$ lies in the product scaled copy ((C1), extended from the points of Z to the whole region by bilinearity and Step 1 of the proof of Lemma D.5.20). By induction from the leaves, the message of every tree, of any size and shape, then lies in $\hat{e}^{s+\nu(F)}\mathcal{P}_Z$, and since $\mathrm{fin} \ge c_Z$ on $\mathcal{P}_Z$, the generating function is at least $c_Z \hat{e}^{s+\nu(F)}$. Finitely many exact checks thus prove a bound for arbitrarily large trees, in the same way as an invariant region of a map bounds all of its iterates.

The next proposition records the parameters of the certificate for the remaining conditions and the sizes of its point sets.

Proposition D.5.23 (The certificate)**.** *Let $p_2 = 0.07$, $p_3 = 0.077$, $y = 31/100$, $\varpi_2 = 272657/250000$, $\varpi_3 = 288677/250000$, $\hat{e} = 145729958/189014575$, $\nu_0 = 24$, $\nu_E = 36$, $d_{\min}^{(3)}(1) = 3$, $d_{\min}^{(3)}(2) = d_{\min}^{(3)}(3) = 2$, $d_{\min}^{(2)}(1) = 4$, $d_{\min}^{(2)}(2) = 3$, $d_{\min}^{(2)}(3) = 2$ and $d_{\min}^{(3)}(\nu) = d_{\min}^{(2)}(\nu) = 1$ for $4 \le \nu \le 36$. There is a set Z of 1024 nonnegative rational points with $c_Z > 0.6060$, and there are nonnegative rational stages for (C5), namely for $\nu = 1$ a set Y_1 of 62 points for $(c_2, c_3) = (0,1)$ and sets Y_1, Y_2 of 62 and 304 points for $(1,0)$, and for $\nu = 2$ sets Y_1 of 61, 62 and 62 points for $(1,0)$, $(1,1)$ and $(2,0)$, such that conditions (C1)–(C6) of Lemma D.5.20 hold.*

For all other pairs $(\nu, (c_2, c_3))$ we have $j = 1$ in (C5), and no intermediate stages are needed.

Proof. Condition (C3) is Lemma D.5.21, together with the fact, noted after Definition D.5.10, that a 3-light pair is 2-light, and the hypotheses on the credits are Lemma D.5.22. Condition (C6) is an inequality between explicit rationals, $(q_3/\hat{e})^{37} \ge \hat{e}^{-25}$; numerically the two sides are about 779 and 666. The values of $d_{\min}^{(3)}$ and $d_{\min}^{(2)}$ satisfy $d_{\min}^{(3)} \le d_{\min}^{(2)}$. The certificate file of Appendix F.4 stores the values stated above for $1 \le \nu \le 4$ and the value 0 in place of 1 for $5 \le \nu \le 36$, and its check uses these stored values; they give the same conditions: for $d_{\min}^{(\tau)}(\nu) \in \{0,1\}$ no integer d satisfies $1 \le d < d_{\min}^{(\tau)}(\nu)$, so (C3) requires nothing at ν; the hypotheses $d_{\min}^{(\tau)}(\nu) \le 1$ in (C4) hold; and $j_\tau(\nu) = \max(1, d_{\min}^{(\tau)}(\nu) - 1) = 1$ in (C5); these are the only places where the hypotheses of Lemma D.5.20 use the values besides $d_{\min}^{(3)} \le d_{\min}^{(2)}$.

Conditions (C1) and (C2) are finite lists of relations of the form $P \in \mathrm{conv}(\mathcal{Y}) + \mathcal{K}_{\mathrm{pos}}$, with $\mathcal{Y} = Z$. By the convexity argument after Lemma D.5.20, (C4) follows from (C1), (C2), the first part of (C4) and the second part at the hull vertices with $c_2 \geq 1$ of the lattice for $\nu - 1$, which are again such relations. By the argument on stages and the convexity argument after Lemma D.5.20, (C5) follows from (C1), (C2) and the relations $x \circ z \in \mathrm{conv}(Y_i) + \mathcal{K}_{\mathrm{pos}}$ for $x \in Y_{i-1}$, $z \in Z$ and $1 \leq i \leq j$, for the chains $Y_0 = \{P\}, Y_1, \ldots, Y_j = Z$ that start at the scaled initial messages P listed there and use the stages of the proposition when $j \geq 2$. So (C1), (C2), (C4) and (C5) follow from finitely many relations, each asserting that a given rational vector P (a product $z \circ z'$, a scaled initial message, or a staged product $x \circ z$) lies in $\mathrm{conv}(\mathcal{Y}) + \mathcal{K}_{\mathrm{pos}}$ for a given finite set $\mathcal{Y} \subseteq \mathbb{R}^8_{\geq 0}$. For each such P the certificate gives nonnegative rational coefficients α_i with $\sum_i \alpha_i \geq 1$ and points $z_i \in \mathcal{Y}$ such that $P \succeq \sum_i \alpha_i z_i$; this implies $P \in \mathrm{conv}(\mathcal{Y}) + \mathcal{K}_{\mathrm{pos}}$, because $\sum_i \alpha_i z_i - \sum_i \frac{\alpha_i}{\sum_j \alpha_j} z_i$ is a nonnegative vector. All these relations, the nonnegativity of the points of Z and of the stages, and the value $c_Z = 0.60605\ldots$ have been verified in exact rational arithmetic (Appendix F.4). □

Proof of Proposition 8.6. Let S be a minimum connected dominating set and $s = |S|$. By Lemma D.5.22 and Corollary D.5.17, the owner rule produces an admissible quadruple (F, ν, c_2, c_3) with $s + \sum_v \nu_v = n$ and $W^{\mathrm{cl}}_S(y) \geq W^{\mathrm{tm}}(F, \nu, c)$. By Proposition D.5.23 the hypotheses of Lemma D.5.20 hold (note $0 < \hat{e} < q_3$, as $\hat{e} < 0.771 < 0.923$), so $W^{\mathrm{cl}}_S(y) \geq c_Z\, \hat{e}^n \geq 0.6060\, \hat{e}^n$, since $c_Z > 0.6060$. □

D.6 The black-box barrier

We work in the black-box model of Definition 9.2, with the notation of Section 9.2, and assume $n \geq 1$. Appendix D.6.1 states the lower bounds for mean exponents, which Corollary 9.3 evaluates for Connected Dominating Set, and Appendix D.6.2 proves them together with Corollary 9.3 and Theorem 9.4.

D.6.1 Lower bounds for mean exponents

One sample size. The first lower bound concerns the hypothesis (5) of Theorem 4.1 at a single sample size t, that is, the class $\mathcal{C}_{\mathrm{mean}}(t, \lambda)$. For $h > 0$ put $\mathsf{N}(h) = \lfloor \lambda \binom{k}{t} / h \rfloor + 1$ and

$$\Sigma(h) = \frac{\binom{n}{t}}{\binom{k}{t}} \cdot \begin{cases} 2^h & \text{if } 0 < h \leq \lambda, \\ \min(2^h, \mathsf{N}(h)) & \text{if } h > \lambda. \end{cases}$$

The idea is that the adversary gives touched t-sets the running-time exponent h and plants the running-time exponent 0 on the untouched ones: as long as fewer than $\mathsf{N}(h)$ subsets of the planted solution are touched, the planted instance lies in $\mathcal{C}_{\mathrm{mean}}(t, \lambda)$. Since $\binom{k}{t}$ is a positive integer, $\mathsf{N}(h) > \binom{k}{t}$ holds exactly when $h \leq \lambda$; so in the first case the planted instance lies in the class whatever the algorithm touches, and only the EXT charges count. In particular $\Sigma(\lambda) = \mathsf{R}_t$ when $\lambda > 0$.

Theorem D.6.1 (Black-box lower bound, one sample size)**.** *Let $0 \leq t \leq k \leq n$ and $\lambda \geq 0$, and let $\mathcal{H}$ be a deterministic algorithm in the black-box model (Definition 9.2) that is correct for $\mathcal{C}_{\mathrm{mean}}(t, \lambda)$. For every $h > 0$, the cost of $\mathcal{H}$ on some no-instance is at least $\frac{1}{3} \min\{\binom{n}{k}, \Sigma(h)\}$. In particular, with $h^* = \max\left(1, \lceil \log((\lambda + 1)\binom{k}{t}) \rceil\right)$,*

$$\mathrm{cost}(\mathcal{H}) \;\geq\; \frac{1}{3} \min\Big\{ \binom{n}{k},\; \max\Big(\mathsf{R}_t,\; \binom{n}{t} \frac{\lambda}{h^*}\Big) \Big\}.$$

So, up to the factor λ/h^* and the cap $\binom{n}{k}$, the cost of a deterministic black-box algorithm is at least $\binom{n}{t}$, whereas the randomized algorithm of the mean-exponent theorem has cost R_t up to a polynomial factor; the ratio is exponential in n whenever $\log\binom{k}{t}$ exceeds λ by a linear amount (Corollary D.6.2). Theorem D.6.1 follows from Theorem D.6.3 below applied to the subclass in which the mean bound at every level other than t is n, and the proof below proceeds in this way.

The adversary can use integer running-time exponents in $\{0,\dots,n\}$, as for the number of Steiner terminals in Section 6: whenever $(\lambda+1)\binom{k}{t} \le 2^n$, the second claim of Theorem D.6.1 holds with R_t replaced by $\mathsf{R}_t/2$, and by R_t itself for integer λ, against algorithms that are correct for the instances of $\mathcal{C}_{\mathrm{mean}}(t,\lambda)$ with integer running-time exponents (see the paragraph on integer running-time exponents in Appendix D.6.2). We record the gap in quantitative form.

Corollary D.6.2 (Gap)**.** *Let $\lambda \ge 1$ and $\varepsilon > 0$ with $\log\binom{k}{t} \ge \lambda + \varepsilon n$. Every deterministic black-box algorithm that is correct for $\mathcal{C}_{\mathrm{mean}}(t,\lambda)$ has cost at least $\min\{\binom{n}{k}, 2^{\varepsilon n}\mathsf{R}_t\}/(3(n+2))$ on some no-instance, while the Markov-cutoff algorithm of Remark 4.2, with the exact threshold $\lambda+1$, decides the class with one-sided error at cost at most $6(\lambda+1)\mathsf{R}_t$.*

In applications the mean bound usually holds for every sample size, and the lower bound persists under this stronger hypothesis.

All sample sizes. For $0 \le j \le k$ and $h > 0$ let $\mathsf{N}_j(h)$, $\Sigma_j(h)$ and h_j^* be $\mathsf{N}(h)$, $\Sigma(h)$ and h^* with (t,λ) replaced by (j,λ_j), and put $\mathsf{R}_j = \binom{n}{j}\binom{k}{j}^{-1}2^{\lambda_j}$. The planting argument applies at all levels at once, because the condition of $\mathcal{C}^{\mathrm{all}}_{\mathrm{mean}}$ at level j involves only the j-subsets of the solution.

Theorem D.6.3 (Black-box lower bound, all sample sizes)**.** *Let $\lambda_0,\dots,\lambda_k \ge 0$, and let $\mathcal{H}$ be a deterministic black-box algorithm that is correct for $\mathcal{C}^{\mathrm{all}}_{\mathrm{mean}}(\lambda_0,\dots,\lambda_k)$. For all $h_0,\dots,h_k > 0$, the cost of $\mathcal{H}$ on some no-instance is at least $\frac{1}{3}\min\{\binom{n}{k}, \min_{0\le j\le k}\Sigma_j(h_j)\}$. In particular,*

$$\mathrm{cost}(\mathcal{H}) \;\ge\; \frac{1}{3}\min\Big\{\binom{n}{k},\; \min_{0\le j\le k}\max\Big(\mathsf{R}_j,\; \binom{n}{j}\frac{\lambda_j}{h_j^*}\Big)\Big\},$$

and $\lambda_j/h_j^* \ge 1/(n+2)$ for every j with $\lambda_j \ge 1$.

D.6.2 Proofs

The planting step. All lower bounds use the same adversary. Fix a deterministic algorithm $\mathcal{H}$ and run it on the *no-instance* $\mathcal{F} = \emptyset$, $g(X,\ell) = h_{|X|}$ for every ℓ, where the reals $h_0,\dots,h_n \ge 0$ are chosen later. Since $\mathcal{H}$ is deterministic, its run is a fixed sequence of calls. Call a set X *touched* if COST or EXT was called on (X,ℓ) for some ℓ. Let T_j be the number of touched j-sets, γ_j and e_j the numbers of COST and EXT calls on j-sets, and z the number of MEM calls. Then $\mathsf{T}_j \le \gamma_j + e_j$, and the cost of the run is

$$\mathrm{cost} = \sum_j \gamma_j + \sum_j e_j 2^{h_j} + z. \tag{33}$$

For a k-set S let $q_j(S)$ be the number of touched j-subsets of S, and call S *clean* if MEM(S) was not called and no EXT call was made on a subset of S.

Lemma D.6.4 (Planting)**.** *Let S be a clean k-set and let $g' \colon 2^U \times \mathbb{N} \to \mathbb{R}_{\ge 0}$ agree with g on (X,ℓ) for every touched X and every ℓ. Then $\mathcal{H}$ answers* NO *on the instance $(\{S\}, g')$. Moreover, for a*

uniformly random $S \in \binom{U}{k}$,

$$\Pr[\mathrm{MEM}(S) \textit{ called}] \le \frac{z}{\binom{n}{k}}, \qquad \Pr[\textit{some EXT call on a subset of } S] \le \sum_j e_j \frac{\binom{k}{j}}{\binom{n}{j}},$$

$$\textit{and} \quad \mathbb{E}\, q_j(S) = \mathsf{T}_j \frac{\binom{k}{j}}{\binom{n}{j}} \quad \textit{for every } j.$$

Proof. We show, by induction along the run, that the instance $(\{S\}, g')$ answers every call exactly as the no-instance does. Since $\mathcal{H}$ is deterministic, it then makes the same calls on both instances and returns the same answer, and on the no-instance that answer is NO by correctness. Consider a call of the run. A COST call concerns a touched set, on which g' and g agree. A call $\mathrm{MEM}(X)$ answers YES on $(\{S\}, g')$ only for $X = S$, which was not queried, so it answers NO on both instances. A call $\mathrm{EXT}(X, \ell)$ answers NO on $(\{S\}, g')$ if $X \not\subseteq S$, which cleanness ensures, so it too answers NO on both instances. For the probabilities, the z calls of MEM query at most z of the $\binom{n}{k}$ equally likely sets; a fixed j-set lies in S with probability $\binom{n-j}{k-j}/\binom{n}{k} = \binom{k}{j}/\binom{n}{j}$, which gives the second bound by the union bound and the expectation by linearity. □

The three lower bounds differ only in which planted instances lie in the class. The next lemma turns this into one counting inequality.

Lemma D.6.5 (Counting)**.** *Let $\mathcal{H}$ be correct for a class $\mathcal{C}$, and let $M_0, \dots, M_k \ge 1$ be integers such that for every clean k-set S with $q_j(S) < M_j$ for all $j \in \{0, \dots, k\}$ there is a g' as in Lemma D.6.4 with $(\{S\}, g') \in \mathcal{C}$. Then*

$$1 \le \frac{z}{\binom{n}{k}} + \sum_{j=0}^{k} \Big(e_j \frac{\binom{k}{j}}{\binom{n}{j}} + \mathbf{1}\big[M_j \le \tbinom{k}{j}\big] \, \frac{\mathsf{T}_j \binom{k}{j}}{M_j \binom{n}{j}} \Big). \tag{34}$$

Proof. Let S be a clean k-set with $q_j(S) < M_j$ for all $j \le k$. The instance $(\{S\}, g')$ lies in $\mathcal{C}$ and its family $\{S\}$ contains a set of size exactly k, so correctness forces $\mathcal{H}$ to answer YES on it, which contradicts Lemma D.6.4. Hence, for uniformly random $S \in \binom{U}{k}$, the event that S is clean and $q_j(S) < M_j$ for all $j \le k$ has probability zero, and the union bound gives

$$1 \le \Pr[\mathrm{MEM}(S) \text{ called}] + \Pr[\text{some EXT call on a subset of } S] + \sum_{j=0}^{k} \Pr[q_j(S) \ge M_j].$$

Lemma D.6.4 bounds the first two terms. For the third, $q_j(S) \le \binom{k}{j}$, so $\Pr[q_j(S) \ge M_j] = 0$ if $M_j > \binom{k}{j}$, and otherwise Markov's inequality and Lemma D.6.4 give $\Pr[q_j(S) \ge M_j] \le \mathbb{E}\, q_j(S)/M_j = \mathsf{T}_j \binom{k}{j}/(M_j \binom{n}{j})$. □

We use the following facts about Σ.

Lemma D.6.6 (The function Σ)**.** *Let $0 \le t \le k \le n$, $\lambda \ge 0$, and let N, Σ, R_t and h^* be as in Section 9.2 and Appendix D.6.1.*

(a) $\Sigma(h) \le \binom{n}{t} 2^h / \binom{k}{t}$ *for every $h > 0$, and* $\Sigma(h) \le \binom{n}{t} \mathsf{N}(h) / \binom{k}{t}$ *for every $h > \lambda$.*

(b) *If $\lambda > 0$ then $\Sigma(\lambda) = \mathsf{R}_t$; if $\lambda = 0$ then $\Sigma(h) = \mathsf{R}_t$ for every $h > 0$. In both cases some $h > 0$ has $\Sigma(h) = \mathsf{R}_t$.*

(c) $\Sigma(h^*) \geq \binom{n}{t}\lambda/h^*$.

(d) *If* $\lambda \geq 1$ *then* $\lambda/h^* \geq 1/(n+2)$.

Proof. (a) Both cases of the definition of Σ are at most $\binom{n}{t}2^h/\binom{k}{t}$, and the second case is at most $\binom{n}{t}\mathsf{N}(h)/\binom{k}{t}$.

(b) If $\lambda > 0$, the first case applies at $h = \lambda$ and gives $\Sigma(\lambda) = \binom{n}{t}2^\lambda/\binom{k}{t} = \mathsf{R}_t$. If $\lambda = 0$, every $h > 0$ falls in the second case with $\mathsf{N}(h) = 1$, so $\Sigma(h) = \binom{n}{t}\binom{k}{t}^{-1}\min(2^h, 1) = \binom{n}{t}/\binom{k}{t} = \mathsf{R}_t$.

(c) By definition, $h^* \geq 1$ and $2^{h^*} \geq (\lambda+1)\binom{k}{t}$. If $h^* \leq \lambda$, then $\Sigma(h^*) = \binom{n}{t}2^{h^*}/\binom{k}{t} \geq \binom{n}{t}(\lambda+1) \geq \binom{n}{t}\lambda/h^*$. If $h^* > \lambda$, then $\mathsf{N}(h^*) \leq \lambda\binom{k}{t}/h^* + 1 \leq (\lambda+1)\binom{k}{t} \leq 2^{h^*}$, so $\Sigma(h^*) = \binom{n}{t}\mathsf{N}(h^*)/\binom{k}{t}$, and $\mathsf{N}(h^*) > \lambda\binom{k}{t}/h^*$ gives the claim.

(d) Since $\lceil x \rceil \leq x + 1$ and $\log\binom{k}{t} \leq n$, $h^* \leq \max(1, \log(\lambda+1) + n + 1) = \log(\lambda+1) + n + 1$. For $\lambda \geq 1$ we have $2^\lambda \geq 1 + \lambda$ (equality at $\lambda = 1$, and the left side grows faster), that is $\log(\lambda+1) \leq \lambda$, so $h^* \leq \lambda + n + 1 \leq (n+2)\lambda$. □

Proof of Theorem D.6.3. Run the planting step with the given values $h_0, \ldots, h_k$ at levels $0, \ldots, k$ and, say, $h_j = 0$ at levels $j > k$, which play no role, as a set of size $j > k$ is not a subset of a k-set. We apply Lemma D.6.5 with $M_j = \mathsf{N}_j(h_j)$. Let S be a clean k-set with $q_j(S) < \mathsf{N}_j(h_j)$ for all $j \leq k$, and define g' to be 0 on the untouched subsets of S of every size, with every second argument, and g elsewhere. The hypothesis of $\mathcal{C}^{\text{all}}_{\text{mean}}$ at level j involves only $g'(X, k)$ for $X \in \binom{S}{j}$; every touched j-subset of S has $g' = h_j$ and every untouched one has $g' = 0$, so the mean of $g'(\cdot, k)$ over $\binom{S}{j}$ is $q_j(S)h_j/\binom{k}{j} \leq (\mathsf{N}_j(h_j) - 1)h_j/\binom{k}{j} \leq \lambda_j$. Hence $(\{S\}, g') \in \mathcal{C}^{\text{all}}_{\text{mean}}(\lambda_0, \ldots, \lambda_k)$, and (34) holds.

We bound the term of (34) at level j. By Lemma D.6.6(a), $e_j\binom{k}{j}/\binom{n}{j} \leq e_j 2^{h_j}/\Sigma_j(h_j)$. The indicator is 1 exactly when $h_j > \lambda_j$, since $\mathsf{N}_j(h_j) > \binom{k}{j}$ holds exactly when $h_j \leq \lambda_j$; then $\Sigma_j(h_j) \leq \binom{n}{j}\mathsf{N}_j(h_j)/\binom{k}{j}$ by Lemma D.6.6(a), so the Markov term is at most $\mathsf{T}_j/\Sigma_j(h_j)$. As $h_j > 0$, $\mathsf{T}_j + e_j 2^{h_j} \leq \gamma_j + e_j + e_j 2^{h_j} \leq 2(\gamma_j + e_j 2^{h_j})$. So the level-$j$ term is at most $2\,\text{cost}_j/\Sigma_j(h_j)$ with $\text{cost}_j = \gamma_j + e_j 2^{h_j}$, and by (33) the right-hand side of (34) is at most $\text{cost}/\binom{n}{k} + 2\,\text{cost}/\min_j \Sigma_j(h_j) \leq 3\,\text{cost}/\min\{\binom{n}{k}, \min_j \Sigma_j(h_j)\}$, where cost is the cost of the run on the no-instance. This proves the first claim.

For the second claim, choose at each level j the better of two values of h_j: a value with $\Sigma_j(h_j) = \mathsf{R}_j$, which exists by Lemma D.6.6(b), and $h_j = h_j^*$, for which $\Sigma_j(h_j^*) \geq \binom{n}{j}\lambda_j/h_j^*$ by Lemma D.6.6(c). The first claim with these values gives the displayed bound, and Lemma D.6.6(d) gives $\lambda_j/h_j^* \geq 1/(n+2)$ for $\lambda_j \geq 1$. □

Proof of Theorem D.6.1. Let $\mathcal{H}$ be correct for $\mathcal{C}_{\text{mean}}(t, \lambda)$, and let $\lambda_t = \lambda$ and $\lambda_j = n$ for $j \neq t$. Every instance of $\mathcal{C}^{\text{all}}_{\text{mean}}(\lambda_0, \ldots, \lambda_k)$ satisfies the level-t condition, so it lies in $\mathcal{C}_{\text{mean}}(t, \lambda)$, and $\mathcal{H}$ is correct for the subclass. Apply Theorem D.6.3 with $h_t = h$ and $h_j = n$ for $j \neq t$. At a level $j \neq t$ we have $h_j = n \leq \lambda_j$, so $\Sigma_j(n) = \binom{n}{j}2^n/\binom{k}{j} \geq 2^n \geq \binom{n}{k}$, and likewise $\mathsf{R}_j = \binom{n}{j}2^n/\binom{k}{j} \geq \binom{n}{k}$. Hence in both bounds of Theorem D.6.3 the levels $j \neq t$ do not lower the minimum below $\binom{n}{k}$, and the bounds read $\frac{1}{3}\min\{\binom{n}{k}, \Sigma(h)\}$ and $\frac{1}{3}\min\{\binom{n}{k}, \max(\mathsf{R}_t, \binom{n}{t}\lambda/h^*)\}$. □

Integer running-time exponents. In the proof of Theorem D.6.1 the adversary uses the running-time exponents n at the levels $j \neq t$ and h at level t. The second claim uses $h = h^*$, an integer, which is at most n exactly when $(\lambda+1)\binom{k}{t} \leq 2^n$, and a value h with $\Sigma(h) = \mathsf{R}_t$, which is $h = \lambda$ when $\lambda > 0$ and need not be an integer. An integer replacement in $\{1, \ldots, n\}$ loses at most a factor 2: if $\lambda \geq n$, take $h = n \leq \lambda$, for which $\Sigma(n) = \binom{n}{t}2^n/\binom{k}{t} \geq 2^n \geq \binom{n}{k}$, so the capped bound

$\frac{1}{3}\binom{n}{k}$ is attained in full; if $1 \le \lambda < n$, take $h = \lfloor\lambda\rfloor \le \lambda$, for which $\Sigma(h) = \binom{n}{t}2^{\lfloor\lambda\rfloor}/\binom{k}{t} \ge \mathsf{R}_t/2$; and if $0 \le \lambda < 1$, take $h = 1 > \lambda$, for which $\mathsf{N}(1) \ge 1$ and $\Sigma(1) = \binom{n}{t}\binom{k}{t}^{-1}\min(2, \mathsf{N}(1)) \ge \binom{n}{t}/\binom{k}{t} \ge \mathsf{R}_t/2$, as $2^\lambda < 2$. So, whenever $(\lambda+1)\binom{k}{t} \le 2^n$, the second claim of Theorem D.6.1 holds with R_t replaced by $\mathsf{R}_t/2$ against algorithms that are correct for the instances of $\mathcal{C}_{\text{mean}}(t, \lambda)$ with integer running-time exponents in $\{0, \dots, n\}$; the planted running-time exponents are 0, so the planted instances have integer running-time exponents as well. If λ is an integer, the three choices give $\Sigma(h) \ge \mathsf{R}_t$ or the full cap, so R_t itself remains.

Proof of Corollary D.6.2. First, $\binom{n}{t} = \mathsf{R}_t\binom{k}{t}2^{-\lambda} \ge 2^{\varepsilon n}\mathsf{R}_t$, and $\lambda/h^* \ge 1/(n+2)$ by Lemma D.6.6(d), so $\max(\mathsf{R}_t, \binom{n}{t}\lambda/h^*) \ge 2^{\varepsilon n}\mathsf{R}_t/(n+2)$, and Theorem D.6.1 gives the lower bound. For the upper bound, computation is free in the model, so the algorithm of Remark 4.2 can use the exact threshold $\lambda + 1$. Each trial then makes one COST call and at most one EXT call on a set with $g \le \lambda + 1$, so it costs at most $1 + 2^{\lambda+1} \le 3 \cdot 2^\lambda$. The number of trials is $\lceil(\lambda+1)\binom{n}{t}\binom{k}{t}^{-1}\rceil \le 2(\lambda+1)\binom{n}{t}\binom{k}{t}^{-1}$, since the argument of the ceiling is at least 1. The total cost is therefore at most $6(\lambda+1)\mathsf{R}_t$, and Remark 4.2 gives the one-sided error. □

Corollary 9.3 (The mean bounds of Connected Dominating Set)**.** *Let n be even, $k = n/2$, $\lambda_0 = n$ and $\lambda_j = n - 1 - j(n-2)/k$ for $1 \le j \le k$. Every deterministic black-box algorithm that is correct for $\mathcal{C}^{\text{all}}_{\text{mean}}(\lambda_0, \dots, \lambda_k)$ has cost at least*

$$\frac{2^{H(1/4)n}}{6(n+1)(n+2)} = \frac{(4 \cdot 3^{-3/4})^n}{6(n+1)(n+2)} \ge \frac{1.7547^n}{6(n+1)(n+2)}$$

on some no-instance, while Algorithm MLS-θ *with the parameters of Corollary 4.3(1), which calls only* COST *and* EXT *and needs only the mean bounds $\lambda_0, \dots, \lambda_k$ as parameters, decides the same class with one-sided error at cost $\sqrt{3}^n n^{O(1)}$.*

For its proof we use the function f_σ of (9) with $\beta = 1$, $\rho = 2$ and $\sigma = \frac{1}{2}$, that is,

$$f_{1/2}(\tau) = H(\tau) - \tfrac{1}{2}H(2\tau) + 1 - 2\tau \qquad (0 \le \tau \le \tfrac{1}{2}).$$

Proof. The lower bound comes from the second bound of Theorem D.6.3: we estimate R_j and $\binom{n}{j}$ by entropy and show that, at every level, the larger of the two exponents is at least $H(\frac{1}{4})$. The upper bound is the linear-savings corollary.

For $1 \le j \le k$, λ_j is nonincreasing in j and $\lambda_k = 1$, so every $\lambda_j \ge 1$ (also $\lambda_0 = n \ge 1$), and Theorem D.6.3 gives cost at least $\frac{1}{3}\min\{\binom{n}{k}, \min_j \max(\mathsf{R}_j, \binom{n}{j}/(n+2))\}$. We bound its terms. First, $\binom{n}{n/2} \ge 2^n/(n+1)$ by (1), and $\mathsf{R}_0 = 2^n$. For $1 \le j \le k$ put $\tau = j/n \in (0, \frac{1}{2}]$. Since $\lambda_j = n - 1 - 2j + 4j/n \ge n(1-2\tau) - 1$, (1) gives

$$\mathsf{R}_j \ge \frac{2^{nH(\tau)}}{n+1} \cdot 2^{-kH(j/k)} \cdot 2^{n(1-2\tau)-1} = \frac{2^{nf_{1/2}(\tau)}}{2(n+1)}, \qquad \frac{\binom{n}{j}}{n+2} \ge \frac{2^{nH(\tau)}}{(n+1)(n+2)}.$$

So

$$\max\Big(\mathsf{R}_j, \frac{\binom{n}{j}}{n+2}\Big) \ge \frac{2^{n\max(f_{1/2}(\tau), H(\tau))}}{2(n+1)(n+2)}.$$

We show that $\max(f_{1/2}(\tau), H(\tau)) \ge H(\frac{1}{4})$ for every $\tau \in (0, \frac{1}{2}]$. For $\tau \ge \frac{1}{4}$ this holds because H increases on $[0, \frac{1}{2}]$. For $0 < \tau \le \frac{1}{4}$, Remark 4.4 with $(\beta, \rho, \sigma) = (1, 2, \frac{1}{2})$ and $p = \frac{1}{4}$ shows that $f_{1/2}$ is convex with minimizer $\tau^* = \frac{1}{3}$, so it is nonincreasing on $[0, \frac{1}{3}]$ and $f_{1/2}(\tau) \ge f_{1/2}(\frac{1}{4}) = H(\frac{1}{4}) - \frac{1}{2}H(\frac{1}{2}) + 1 - \frac{1}{2} = H(\frac{1}{4})$. (In terms of binomial coefficients: the ratio of consecutive terms

of $\binom{n}{j}\binom{k}{j}^{-1}2^{n-2j}$ is $(n-j)/(4(k-j))$, which is at most 1 for $j \le n/3$.) Since $H(\frac{1}{4}) < 1$, all terms are at least $2^{nH(1/4)}/(2(n+1)(n+2))$, and Theorem D.6.3 gives the lower bound. Finally $H(\frac{1}{4}) = \frac{1}{4}\log 4 + \frac{3}{4}\log\frac{4}{3} = 2 - \frac{3}{4}\log 3$, so $2^{H(1/4)} = 4 \cdot 3^{-3/4} = 1.7547\ldots$.

For the upper bound, the class satisfies (7) with $(\beta, \rho, C) = (1, 2, 1)$ at every sample size: $\lambda_0 = n \le n+1$, and for $1 \le j \le k$, $\lambda_j = n - 1 - 2j + 4j/n \le n - 2j + 1$ because $j \le n/2$. By Corollary 4.3(1), Algorithm MLS-θ decides the class with one-sided error in time $2^{C+nF^*(1/2)}N^{O(1)}$, where $F^*(\frac{1}{2}) = 1 + \frac{1}{2}\log\frac{1}{2} + \frac{1}{2}\log\frac{3}{2} = \frac{1}{2}\log 3$ by (6), since $p = \frac{1}{4} < \frac{1}{2}$. In the model, a trial of (t, θ) in Algorithm MLS-θ makes one COST call and at most one EXT call, on a set with $g \le \theta$, so it costs at most $2^{\theta+1}$. The algorithm of Corollary 4.3(1) uses $\Gamma = 1$, confidence $\kappa = 1$ and a rate ψ with $\psi n \le C + nF^*(\frac{1}{2}) + 1 \le n + 2$. By (2), $\Theta \le \psi n + 3$ and $m_{t,\theta}2^{\theta+1} \le (2M_{t,\theta} + 1)2^{\theta+1} \le 16\kappa(\Theta+1)2^{\psi n} + 2^{\Theta+1} \le 16(\kappa(\Theta+1)+1)2^{\psi n}$. Summing over the $(k+1)(\Theta+1)$ pairs (t, θ), the cost is at most $16(k+1)(\Theta+1)(\kappa(\Theta+1)+1)2^{\psi n}$, and $2^{\psi n} \le 4 \cdot \sqrt{3}^{\,n}$, so the cost is $\sqrt{3}^{\,n}n^{O(1)}$. □

Theorem 9.4 (Black-box lower bound for exponential moments)**.** *Let $\psi \ge 0$ and $\Gamma \ge 1$, and let $\mathcal{H}$ be a deterministic black-box algorithm that is correct for $\mathcal{C}_{\exp}(\psi, \Gamma)$. Call a level $t \in \{0, \ldots, k\}$* feasible *if $J_t := \lceil \binom{n}{t}2^{-\psi n}/\Gamma \rceil \le \binom{k}{t}/2$. Then for every feasible t, the cost of $\mathcal{H}$ on the no-instance $\mathcal{F} = \emptyset$, $g \equiv n$, is at least $\frac{1}{3}\min\{\binom{n}{k}, \binom{n}{t}\}$.*

Proof. Take $h_j = n$ for all j, so that $g \equiv n$. Fix a feasible t and apply Lemma D.6.5 with $M_t = \binom{k}{t} - J_t + 1$ and $M_j = \binom{k}{j} + 1$ for $j \ne t$; these are integers, and $M_t \ge 1$ since $J_t \le \binom{k}{t}/2$. Let S be a clean k-set with $q_j(S) < M_j$ for all $j \le k$; the conditions at levels $j \ne t$ hold for every S, and the condition at level t says that at least J_t sets of $\binom{S}{t}$ are untouched. Let g' be 0 on J_t of them, with every second argument, and g elsewhere. Then $\mu_t(S) \ge J_t \ge \binom{n}{t}2^{-\psi n}/\Gamma$ under g', so $(\{S\}, g') \in \mathcal{C}_{\exp}(\psi, \Gamma)$. In (34) the indicator vanishes at the levels $j \ne t$, and $M_t > \binom{k}{t}/2$ at level t, so, using $\binom{k}{j}/\binom{n}{j} \le 1 \le 2^n/\binom{n}{k}$,

$$1 \le \frac{z}{\binom{n}{k}} + \sum_j e_j \frac{\binom{k}{j}}{\binom{n}{j}} + \frac{\mathsf{T}_t\binom{k}{t}}{M_t\binom{n}{t}} \le \frac{z + \sum_j e_j 2^n}{\binom{n}{k}} + \frac{2\mathsf{T}_t}{\binom{n}{t}}.$$

By (33), $z + \sum_j e_j 2^n \le$ cost and $\mathsf{T}_t \le \gamma_t + e_t \le$ cost, so the right-hand side is at most $3\,\text{cost}/\min\{\binom{n}{k}, \binom{n}{t}\}$. □

D.7 Balanced set-inclusion families

This subsection constructs the balanced families of Definition 9.5 and proves the following theorem of Section 9.3, whose notation we keep.

Theorem 9.6 (Balanced families in polynomial space)**.** *Fix $\mathsf{r} \ge 1$. Let $n \ge 16$ and $\iota = \max(2, \lfloor \log n / \log\log n \rfloor)$. A deterministic algorithm, given $k_1, \ldots, k_\mathsf{r}$ and $t_1, \ldots, t_\mathsf{r}$ as in Definition 9.5, lists (possibly with repetitions) the members of an $(n; k; t)$-balanced family $\mathcal{B}$ with*

$$|\mathcal{B}| \;\le\; \iota^{12\mathsf{r}(n/\iota+1)}\,\frac{\binom{n}{t}}{\prod_j \binom{k_j}{t_j}} \;=\; \frac{\binom{n}{t}}{\prod_j \binom{k_j}{t_j}}\,2^{O(\mathsf{r}n(\log\log n)^2/\log n)}.$$

It runs in time $n^{O(1)}$ times this bound, plus $n^{O(1)}$, and in space $n^{O(1)}$.

The construction achieves exact balance. Its building block is a covering of *ordered* structures on small blocks by subsets that are good on every prefix: when a set is ordered by non-increasing weight, a prefix-good subset captures its proportional share of the weight (Lemma D.7.2), and at

least a $1/p$ fraction of all $\tilde{t}$-subsets of a p-set is prefix-good (Lemma D.7.1). Formally, for a finite set S with a linear order $u_1, \dots, u_p$ and an integer $0 \le \tilde{t} \le p$, call $Y \subseteq S$ with $|Y| = \tilde{t}$ *prefix-good* if $p\,|Y \cap \{u_1, \dots, u_i\}| \ge \tilde{t}\,i$ for $i = 1, \dots, p$.

Lemma D.7.1 (Rotation)**.** *Let $p \ge 1$. For every $\xi \in \{0,1\}^p$ with $\sum_i \xi_i = \tilde{t}$, some cyclic rotation of ξ has i-th prefix sum at least $(\tilde{t}/p)i$ for every i. Consequently, for a fixed order of S, a uniformly random $\tilde{t}$-subset of S is prefix-good with probability at least $1/p$.*

Proof. Let $\pi_0 = 0$ and $\pi_i = \sum_{l \le i}(\xi_l - \tilde{t}/p)$, so $\pi_p = 0$. Pick $m \in \{0, \dots, p-1\}$ minimizing π_m. The rotation that starts at ξ_{m+1} has i-th prefix sum minus $(\tilde{t}/p)i$ equal to $\pi_{m+i} - \pi_m \ge 0$ if $m + i \le p$, and to $\pi_p + \pi_{m+i-p} - \pi_m = \pi_{m+i-p} - \pi_m \ge 0$ if $m + i > p$. For the second claim, identify $\tilde{t}$-subsets of S with their indicator vectors. The cyclic group of rotations acts on these vectors and the uniform distribution is invariant under it. Each orbit has size dividing p and contains a prefix-good member, so at least a $1/p$ fraction of every orbit, and hence of all $\tilde{t}$-subsets, is prefix-good. □

The second ingredient is the weight guarantee of prefix-good sets.

Lemma D.7.2 (Abel summation)**.** *If $w \ge 0$ is non-increasing along the order $u_1, \dots, u_p$ of S and $Y \subseteq S$ is prefix-good with $|Y| = \tilde{t}$, then $w(Y) \ge (\tilde{t}/p)w(S)$.*

Proof. Put $c_i = |Y \cap \{u_1, \dots, u_i\}|$, $c_0 = 0$ and $w(u_{p+1}) = 0$. Then $w(Y) = \sum_{i=1}^{p} w(u_i)(c_i - c_{i-1}) = \sum_{i=1}^{p} c_i(w(u_i) - w(u_{i+1}))$. Each difference is nonnegative and $c_i \ge (\tilde{t}/p)i$, so $w(Y) \ge (\tilde{t}/p)\sum_{i=1}^{p} i(w(u_i) - w(u_{i+1})) = (\tilde{t}/p)\sum_{i=1}^{p} w(u_i) = (\tilde{t}/p)w(S)$. □

The block families are found by greedy Set Cover, whose guarantee we use in the following form: on an instance whose sets each cover at most Δ elements, greedy returns a cover of size at most $1 + \ln \Delta$ times the optimum [Joh74, Lov75].

Lemma D.7.3 (Block families)**.** *Let $\iota \ge 1$, let $q_1, \dots, q_{\mathsf{r}} \ge 0$ be integers with $\sum_j q_j \le \iota$, and let $\tilde{t}_j$ be integers with $0 \le \tilde{t}_j \le q_j$ and $\tilde{t} = \sum_j \tilde{t}_j$. A deterministic algorithm computes, in time polynomial in $\iota!\,2^{\iota}$, an $(\iota; q; \tilde{t})$-balanced family $\mathcal{B}$ over $[\iota]$ with*

$$|\mathcal{B}| \le N_{\mathrm{cov}}\,(1 + \ln \iota!)^2, \qquad N_{\mathrm{cov}} = \binom{\iota}{\tilde{t}} \prod_j \frac{\max(q_j, 1)}{\binom{q_j}{\tilde{t}_j}}.$$

Proof. An *ordered structure* is a tuple of pairwise disjoint sets $S_j \subseteq [\iota]$ with $|S_j| = q_j$, each with a linear order; let $\mathcal{O}$ be the set of ordered structures, so that $|\mathcal{O}| = \iota!/(\iota - \sum_j q_j)! \le \iota!$. A candidate $X \in \binom{[\iota]}{\tilde{t}}$ *covers* a structure if $|X \cap S_j| = \tilde{t}_j$ for every j (so $X \subseteq \bigcup_j S_j$, as $\sum_j \tilde{t}_j = \tilde{t} = |X|$) and $X \cap S_j$ is prefix-good in S_j for every j with $q_j \ge 1$.

A random candidate covers a fixed structure with probability at least $1/N_{\mathrm{cov}}$. A uniform $X \in \binom{[\iota]}{\tilde{t}}$ has the right class counts with probability $\prod_j \binom{q_j}{\tilde{t}_j}/\binom{\iota}{\tilde{t}}$, because the candidates with these counts are exactly the unions of a $\tilde{t}_j$-subset of each S_j. Conditioned on that event, the sets $X \cap S_j$ are independent and uniform among the $\tilde{t}_j$-subsets of S_j, so by Lemma D.7.1 each is prefix-good with probability at least $1/q_j$ (a class with $q_j = 0$ contributes the factor 1). In particular $N_{\mathrm{cov}} \ge \binom{\iota}{\tilde{t}}/\prod_j \binom{q_j}{\tilde{t}_j} \ge 1$.

Size. Take $N_{\mathrm{cand}} = \lfloor N_{\mathrm{cov}} \ln |\mathcal{O}| \rfloor + 1$ independent uniform candidates. A fixed structure is uncovered by all of them with probability at most $(1 - 1/N_{\mathrm{cov}})^{N_{\mathrm{cand}}} < e^{-N_{\mathrm{cand}}/N_{\mathrm{cov}}} < 1/|\mathcal{O}|$, so by the union bound some choice covers all structures, and the minimum cover has at most $N_{\mathrm{cand}} \le N_{\mathrm{cov}}(1 + \ln |\mathcal{O}|)$ members. A candidate covers at most $|\mathcal{O}|$ structures, so greedy Set Cover returns a cover $\mathcal{B}$ with $|\mathcal{B}| \le (1 + \ln |\mathcal{O}|)N_{\mathrm{cov}}(1 + \ln |\mathcal{O}|) \le N_{\mathrm{cov}}(1 + \ln \iota!)^2$.

Balance. Given disjoint $S_j \subseteq [\iota]$ with $|S_j| = q_j$ and $w \geq 0$, order each S_j by non-increasing w (ties broken arbitrarily), take a member of $\mathcal{B}$ covering this ordered structure, and apply Lemma D.7.2 to each class j with $q_j \geq 1$: it gives $w(X \cap S_j) \geq (\tilde{t}_j/q_j)w(S_j)$, simultaneously for all classes.

Time. The covering relation has $|\mathcal{O}| \leq \iota!$ elements and at most 2^ι candidate sets, and both it and the greedy computation take time polynomial in $\iota!\, 2^\iota$. □

The block targets are distributed over the blocks by systematic rounding, which keeps every block target within one of its proportional share and has mean equal to that share. Formally, let $p_1, \ldots, p_b \geq 0$ be integers with $\sum_i p_i = k \geq 1$, let $t \in \{0, \ldots, k\}$ and $\delta = t/k$, and put $p_{\leq i} = p_1 + \cdots + p_i$ ($p_{\leq 0} = 0$). For an offset $\upsilon \in [0, 1)$, the *systematic rounding* of (p_i) with target t is

$$t_i(\upsilon) = \lfloor \delta p_{\leq i} + \upsilon \rfloor - \lfloor \delta p_{\leq i-1} + \upsilon \rfloor \qquad (1 \leq i \leq b).$$

Lemma D.7.4 (Systematic rounding)**.** *In the setting above, for every $\upsilon \in [0, 1)$: $t_i(\upsilon) \in \{\lfloor \delta p_i \rfloor, \lceil \delta p_i \rceil\}$, $0 \leq t_i(\upsilon) \leq p_i$ and $\sum_i t_i(\upsilon) = t$. For υ uniform in $[0, 1)$, $\mathbb{E}\, t_i(\upsilon) = \delta p_i$.*

Proof. For real x and $x' \geq 0$, $\lfloor x + x' \rfloor - \lfloor x \rfloor \in \{\lfloor x' \rfloor, \lceil x' \rceil\}$; and $0 \leq \lfloor \delta p_i \rfloor \leq \lceil \delta p_i \rceil \leq p_i$ since $0 \leq \delta \leq 1$. The sum telescopes to $\lfloor t + \upsilon \rfloor - \lfloor \upsilon \rfloor = t$. Finally, writing $\{x\}$ for the fractional part, $\mathbb{E}\lfloor x + \upsilon \rfloor = \lfloor x \rfloor + \Pr[\upsilon \geq 1 - \{x\}] = \lfloor x \rfloor + \{x\} = x$, so $\mathbb{E}\, t_i(\upsilon) = \delta p_{\leq i} - \delta p_{\leq i-1} = \delta p_i$. □

Two entropy facts control the size of the product construction.

Lemma D.7.5 (Entropy facts)**.** *The binary entropy function H has the following properties.*

(E1) $|H(x) - H(x')| \leq H(|x - x'|)$ *for* $x, x' \in [0, 1]$.

(E2) For an integer $p \geq 1$, a real $\delta \in [0, 1]$ and an integer q with $|q - \delta p| < 1$ and $0 \leq q \leq p$: $pH(q/p) \geq pH(\delta) - \log(ep)$.

Proof. (E1) We may assume $x \leq x'$ and put $h = x' - x$. Since H is concave, $y \mapsto H(y+h) - H(y)$ is non-increasing on $[0, 1-h]$, so $H(x') - H(x) \leq H(h) - H(0) = H(h)$. Applying this to $1 - x' \leq 1 - x$ and using $H(y) = H(1 - y)$ gives $H(x) - H(x') \leq H(h)$.

(E2) For $p = 1$, $|H(q) - H(\delta)| \leq 1 \leq \log e$. Let $p \geq 2$. By (E1), $H(\delta) - H(q/p) \leq H(|q/p - \delta|)$, and $|q/p - \delta| < 1/p \leq \frac{1}{2}$, so by monotonicity of H on $[0, \frac{1}{2}]$ this is at most $H(1/p)$. Finally $pH(1/p) = \log p + (p-1)\log \frac{p}{p-1} \leq \log p + (p-1)\frac{\log e}{p-1} = \log(ep)$, using $\ln(1 + x) \leq x$. □

The construction. Let $\iota \geq 2$ be a block length. Partition $U = [n]$ into $b = \lceil n/\iota \rceil$ consecutive blocks $U_1, \ldots, U_b$, all of size $n_i = \iota$ except possibly the last, and put $\delta_j = t_j/k_j$ ($\delta_j = 0$ if $k_j = 0$). The family $\mathcal{B}$ consists of all sets $X = \bigcup_i \mathrm{ord}_i(Y_i)$ where:

(1) (p_{ij}) is a nonnegative integer $b \times \mathsf{r}$ matrix with $\sum_j p_{ij} \leq n_i$ for every i and $\sum_i p_{ij} = k_j$ for every j;

(2) (t_{ij}) has $t_{ij} \in \{\lfloor \delta_j p_{ij} \rfloor, \lceil \delta_j p_{ij} \rceil\}$ and $\sum_i t_{ij} = t_j$ for every j;

(3) Y_i ranges over the block family $\mathcal{B}_i = \mathcal{B}(n_i; (p_{ij})_j; (t_{ij})_j)$ of Lemma D.7.3 on $[n_i]$;

(4) $\mathrm{ord}_i \colon [n_i] \to U_i$ is the order-preserving bijection.

The matrix (p_{ij}) is a guess of how the unknown solution classes meet the blocks, and (t_{ij}) distributes the targets proportionally.

Proof of Theorem 9.6. Balance passes from the blocks to the whole universe through the rounding lemma; the size bound compares the product of the block bounds with the global binomial quotient, through an injection for the numerator and the entropy facts for the denominator; time and space follow from the order of enumeration.

Balance. Let (S_j) and w be given, and put $p_{ij} = |S_j \cap U_i|$; this matrix satisfies (1). Fix a class j with $k_j \geq 1$, and let $t_{ij}(\upsilon_j)$ be the systematic rounding of $(p_{ij})_i$ with target t_j (Lemma D.7.4). The quantity

$$\sum_{i:\, p_{ij} \geq 1} \frac{t_{ij}(\upsilon_j)}{p_{ij}}\, w(S_j \cap U_i)$$

has expectation $\delta_j w(S_j)$ for uniform υ_j (blocks with $p_{ij} = 0$ contain no element of S_j), so some υ_j makes it at least $\delta_j w(S_j)$. For a class with $k_j = 0$ all $t_{ij} = 0$. By Lemma D.7.4 the resulting matrix (t_{ij}) satisfies (2); the rounding parameters of different classes are chosen independently, and every combination occurs in (2). By Lemma D.7.3, applied in block i to the sets $\mathrm{ord}_i^{-1}(S_j \cap U_i)$ and the weight $w \circ \mathrm{ord}_i$, the family $\mathcal{B}_i$ contains Y_i with $|Y_i \cap \mathrm{ord}_i^{-1}(S_j)| = t_{ij}$, $Y_i \subseteq \bigcup_j \mathrm{ord}_i^{-1}(S_j)$ and class-j weight at least $(t_{ij}/p_{ij})w(S_j \cap U_i)$, simultaneously for all j with $p_{ij} \geq 1$. The union $X = \bigcup_i \mathrm{ord}_i(Y_i)$ lies in $\mathcal{B}$, satisfies $X \subseteq \bigcup_j S_j$ and $|X \cap S_j| = \sum_i t_{ij} = t_j$, and $w(X \cap S_j) \geq \delta_j w(S_j)$ for every j with $k_j \geq 1$.

Size. Write $t_i = \sum_j t_{ij}$, so that $\sum_i t_i = t$. The number of listed tuples is at most $\#(p_{ij}) \cdot \#(t_{ij}) \cdot \max \prod_i |\mathcal{B}_i|$, where $\#(p_{ij}) \leq (\iota+1)^{\mathsf{r}b}$ (each entry lies in $\{0, \dots, \iota\}$) and $\#(t_{ij}) \leq 2^{\mathsf{r}b}$. By Lemma D.7.3, $|\mathcal{B}_i| \leq \iota^4 N_{\mathrm{cov},i}$, because $1 + \ln n_i! \leq 1 + \iota \ln \iota \leq 1 + \iota(\iota-1) \leq \iota^2$. Also $N_{\mathrm{cov},i} \leq \iota^{\mathsf{r}} \binom{n_i}{t_i} / \prod_j \binom{p_{ij}}{t_{ij}}$. We bound the two products over i.

- *Numerator.* $\prod_i \binom{n_i}{t_i} \leq \binom{n}{t}$, because choosing t_i elements in each block U_i gives distinct t-subsets of U.
- *Denominator.* For $p_{ij} \geq 1$ we have $|t_{ij} - \delta_j p_{ij}| < 1$, so (1) and (E2) give

  $$\log \binom{p_{ij}}{t_{ij}} \geq p_{ij} H(\delta_j) - \log(e p_{ij}) - \log(p_{ij}+1) \geq p_{ij} H(\delta_j) - \log\bigl(e\iota(\iota+1)\bigr),$$

  and the inequality between the outer terms also holds for $p_{ij} = 0$. Summing over i and using $\sum_i p_{ij} H(\delta_j) = k_j H(\delta_j) \geq \log \binom{k_j}{t_j}$ (by (1)) gives $\prod_{i,j} \binom{p_{ij}}{t_{ij}} \geq \prod_j \binom{k_j}{t_j} \cdot (e\iota(\iota+1))^{-\mathsf{r}b}$.

Altogether

$$|\mathcal{B}| \leq \frac{\binom{n}{t}}{\prod_j \binom{k_j}{t_j}} \bigl[2(\iota+1)\iota \cdot e\iota(\iota+1)\bigr]^{\mathsf{r}b} \iota^{4b} \leq \frac{\binom{n}{t}}{\prod_j \binom{k_j}{t_j}}\, \iota^{12\mathsf{r}b},$$

because $2e\iota^2(\iota+1)^2 \leq \iota^8$ for $\iota \geq 2$ (at $\iota = 2$ the left side is $72e < 256$, and the ratio of the two sides decreases in ι). Finally $b \leq n/\iota + 1$. For $n \geq 16$ we have $\iota \geq 2$, and $\iota \geq \log n / \log\log n - 1 \geq \frac{1}{2} \log n / \log\log n$ once $\log n / \log\log n \geq 2$, which holds for $n \geq 16$; hence $\log \iota^{12\mathsf{r}(n/\iota+1)} \leq 12\mathsf{r}(n/\iota + 1) \log\log n = O(\mathsf{r} n (\log\log n)^2 / \log n)$.

Time and space. Only two block sizes occur, so there are at most $2(\iota+1)^{\mathsf{r}} 2^{\mathsf{r}}$ distinct block parameters $(n_i; (p_{ij})_j; (t_{ij})_j)$, which is polynomial in n for fixed r. Each block family takes time polynomial in $\iota!\, 2^\iota \leq n^2$, because $\iota! \leq \iota^\iota = 2^{\iota \log \iota} \leq 2^{(\log n / \log\log n) \log\log n} = n$ and $2^\iota \leq n$; all block families are computed once and stored in polynomial space. The matrices (p_{ij}) are enumerated by depth-first search over the blocks, extending a partial choice only if the remaining column sums fit into the remaining blocks, that is, $\sum_j (\text{remaining } k_j) \leq \sum_{i' > i} n_{i'}$ after block i; this is the exact feasibility condition, because any distribution of the remaining demand that respects the block

capacities is admissible. The matrices (t_{ij}) are enumerated by depth-first search, keeping each remaining column sum between the sums of the remaining floors and ceilings, which is again exact. The product over blocks is enumerated by an odometer; every block family is nonempty, because for its parameters at least one ordered structure exists. So every leaf of the search yields a listed set, the time is $n^{O(1)}$ per listed set plus $n^{O(1)}$, and only the search state is kept. □

Remark D.7.6 (Comparison with the construction of Fomin, Gaspers, Lokshtanov and Saurabh). Compared with [FGLS19], the construction of Theorem 9.6 uses no hashing: a fixed partition into blocks suffices because the block targets follow the block counts: the numerator of the size bound is at most $\binom{n}{t}$ because distinct choices in the blocks give distinct t-sets, and the denominator loses only a factor $(e\iota(\iota+1))^{\mathsf{r}} = 2^{O(\mathsf{r}\log\iota)}$ per block. Small blocks make the greedy step take polynomial time and space, and hence the whole construction runs in polynomial space. In exchange, the size bound has a larger subexponential factor, $2^{O(n(\log\log n)^2/\log n)}$.

The subexponential factor. With the explicit constants of Theorem 9.6, the factor by which the balanced families multiply the running-time bound of Theorem 10.4 is, for the two-class families of Algorithm 9.9 ($\mathsf{r} = 2$), at most $\iota^{24(n/\iota+1)}$ with $\iota = \lfloor \log n / \log\log n \rfloor$, about 2^{12718} at $n = 1000$ against $1.7361^{1000} \approx 2^{796}$. The product of this factor and 1.7361^n is below the trivial 2^n if and only if $\iota \geq 1203$, that is, if and only if $\log n \geq 16895.338\ldots$. These are upper bounds, not measured family sizes; for $n \leq 20$ and block lengths 4 to 6 the actual product families contain 80–90% of all t-sets (Appendix F).

D.7.1 Trees across block partitions

The product construction of Theorem 9.6 handles conditions that are separable over a fixed partition of the universe into blocks. The analyses of RS-θ and RSC rest on tests along an unknown spanning tree of the solution, and the following count shows that no family of $2^{O(n)}$ fixed partitions into blocks of size $n^{o(1)}$ avoids cutting almost all edges of some such tree. This is an obstacle for one construction strategy, not an impossibility result.

Proposition D.7.7 (Trees across block partitions)**.** *Let S be a set of k elements, let $c \geq 0$ and $B \geq 1$, and let $\mathcal{P}$ be a family of at most 2^{ck} partitions of S into blocks of size at most B. If $\varepsilon \in (0,1]$ satisfies $\varepsilon \log k > c + \log(B+1) + 2 + 3(\log k)/k$, then some tree T with vertex set S has, for every partition in $\mathcal{P}$, at least $(1-\varepsilon)k$ edges whose ends lie in different blocks. In particular, if $B = k^{o(1)}$, then for every fixed $\varepsilon > 0$ and all large k some tree has $(1-\varepsilon)k$ such edges in every partition of the family.*

Proof. Fix a partition in $\mathcal{P}$. A tree T on S is determined by its restrictions to the blocks, which are forests, and by the set of its edges between blocks. A forest on b labeled vertices becomes a tree on $b+1$ vertices when a new vertex is joined to one vertex of each component, and the forest is recovered by deleting the new vertex; so by Cayley's formula there are at most $(b+1)^{b-1} \leq (B+1)^b$ forests on a block of size b, and at most $(B+1)^k$ choices of the restrictions in all. If m edges lie between blocks, they form an m-subset of the fewer than k^2 pairs of elements in different blocks, which gives at most $\binom{k^2}{m} \leq (ek^2/m)^m$ choices for $m \geq 1$, a bound that is nondecreasing in m for $m \leq k^2$. For $\varepsilon = 1$ there is nothing to prove, so let $\varepsilon < 1$ and $\alpha = 1-\varepsilon \in (0,1)$. Then at most $k(B+1)^k(ek/\alpha)^{\alpha k}$ trees have fewer than αk edges between blocks of this partition (there are at most k values of $m < \alpha k$), and since $\alpha\log(e/\alpha) \leq \log e + \max_{0<\alpha\leq 1} \alpha\log(1/\alpha) < 2$, this number is at most $k(B+1)^k 4^k k^{\alpha k}$. Over the family, at most $2^{ck}k(B+1)^k 4^k k^{(1-\varepsilon)k} = k^{k-2} \cdot 2^{k(c+\log(B+1)+2)+3\log k-\varepsilon k\log k}$ trees have fewer than $(1-\varepsilon)k$ edges between blocks of some partition, and this is less than k^{k-2} because the

exponent of 2 is negative by the hypothesis on ε. By Cayley's formula there are k^{k-2} trees on S, so some tree has at least $(1-\varepsilon)k$ edges between blocks of every partition in $\mathcal{P}$. The last sentence holds because $c+\log(B+1)+2+3(\log k)/k = o(\log k)$ when $B = k^{o(1)}$. □

For a solution S of linear size $k = \Theta(n)$, a family of $2^{O(n)}$ partitions of the universe into blocks of size $n^{o(1)}$ induces on S a family as in Proposition D.7.7, so some tree T on S has $(1-o(1))k$ edges between blocks of every partition of the family; on an instance in which S induces exactly the tree T, these are edges of the only spanning tree of the solution.

D.8 The deterministic algorithms

This subsection proves the statements of Sections 9.4 and 10, with the notation of those sections.

D.8.1 Exponential moments of integer affine certificates

Lemma 9.7 (Chord lemma)**.** *Let $k \geq 1$, let $\Xi \geq 0$ be an integer, $a = \lfloor \Xi/k \rfloor$, $k_{a+1} = \Xi - ak$ and $k_a = k - k_{a+1}$. Let $v \in \mathbb{R}^k$ have $\sum_u v_u = \Xi$ and no entry in the open interval $(a, a+1)$. Then for every $y > 0$,*

$$\prod_u (1 + y2^{v_u}) \;\geq\; \mathsf{P}^*(y) := \left(1 + y2^{a}\right)^{k_a} \left(1 + y2^{a+1}\right)^{k_{a+1}}.$$

Without the restriction on the entries, $\prod_u (1 + y2^{v_u}) \geq (1 + y2^{\Xi/k})^k$.

Proof. The function $\mathrm{cv}(x) = \ln(1 + y2^x)$ is convex. Let chord be the affine function that agrees with cv at a and $a+1$. A convex function lies on or above its chord outside the chord's interval, so $\mathrm{cv}(v_u) \geq \mathrm{chord}(v_u)$ for every u. Let w^* have k_a entries a and k_{a+1} entries $a+1$; its total is $ak + k_{a+1} = \Xi$. Since cv = chord at a and $a+1$, and chord is affine with $\sum_u v_u = \sum_u w^*_u$,

$$\sum_u \mathrm{cv}(v_u) \geq \sum_u \mathrm{chord}(v_u) = \sum_u \mathrm{chord}(w^*_u) = \sum_u \mathrm{cv}(w^*_u) = \ln \mathsf{P}^*(y).$$

The second claim is Jensen's inequality for cv. □

Proposition 9.8 (Real weights give only the mean-exponent bound)**.** *Let S be a k-subset of a universe of size n, with $k \geq 1$, and let α be real and $\Xi \geq 0$. For $w\colon S \to \mathbb{R}_{\geq 0}$ with $w(S) = \Xi$ let $\mu^w_t(S) = \sum_{X \in \binom{S}{t}} 2^{-\alpha + w(X)}$ for $0 \leq t \leq k$, the exponential moments of the surrogate exponent $\alpha - w(X)$, and put $\mathsf{R}(w) = \min_{0 \leq t \leq k} \binom{n}{t} / \mu^w_t(S)$. Let $\bar{w} \equiv \Xi/k$ be the uniform profile. Then $\mathsf{R}(\bar{w}) = \min_{0 \leq t \leq k} \binom{n}{t}\binom{k}{t}^{-1} 2^{\alpha - t\Xi/k}$, and $\mathsf{R}(w) \leq (n+1)\,\mathsf{R}(\bar{w})$ for every such w. If Ξ is an integer and w^* is the profile of Lemma 9.7, then $\mathsf{R}(w) \leq (n+1)\,\mathsf{R}(w^*)$ for every such w with integer values.*

Proof. For a profile w let $W^w_S(y) = \sum_t \mu^w_t(S) y^t = 2^{-\alpha} \prod_{u \in S} (1 + y2^{w(u)})$ be the generating function of the surrogate exponent. Lemma 3.9, whose proof uses only $\mu_t(S) > 0$ for $0 \leq t \leq k$ and $W_S(y) = \sum_t \mu_t(S) y^t$, gives $\mathsf{R}(w) \leq \inf_{y \geq 0} (1+y)^n / W^w_S(y) \leq (n+1)\mathsf{R}(w)$. By the second claim of Lemma 9.7, $W^w_S(y) \geq W^{\bar{w}}_S(y)$ for every $y > 0$, so $\mathsf{R}(w) \leq \inf_{y \geq 0} (1+y)^n / W^w_S(y) \leq \inf_{y \geq 0} (1+y)^n / W^{\bar{w}}_S(y) \leq (n+1)\mathsf{R}(\bar{w})$. If w has integer values, it has no entry in $(a, a+1)$, and the first claim of Lemma 9.7 gives $\mathsf{R}(w) \leq (n+1)\mathsf{R}(w^*)$ in the same way. Finally, for the uniform profile $\mu^{\bar{w}}_t(S) = \binom{k}{t} 2^{-\alpha + t\Xi/k}$, which gives the formula for $\mathsf{R}(\bar{w})$. □

Theorem 9.10 (Deterministic exponential moments of integer affine certificates)**.** *Let $k \geq 1$, a rational $\alpha = \alpha(I,k)$ and an integer $\Xi = \Xi(I,k)$ with $0 \leq \Xi \leq N^{O(1)}$ be computable in time $N^{O(1)}$. Say that $S \in \mathcal{F}_I$ with $|S| = k$ has an* integer certificate *if some $w\colon S \to \mathbb{Z}_{\geq 0}$ with $w(S) = \Xi$ satisfies*

$$g(I, X, k) \leq \alpha - w(X) \qquad \textit{for every nonempty } X \subseteq S.$$

Put $a = \lfloor \Xi/k \rfloor$, $k_{a+1} = \Xi - ak$, $k_a = k - k_{a+1}$ and $\mathsf{P}^*(y) = (1+y2^a)^{k_a}(1+y2^{a+1})^{k_{a+1}}$. Algorithm 9.9 is deterministic, answers NO *if $\mathcal{F}_I$ has no set of size at most k, and answers* YES *if some $S \in \mathcal{F}_I$ with $|S| = k$ has an integer certificate. It runs in time*

$$N^{O(1)}\, 2^{O(n(\log\log n)^2/\log n)}\, \max\Bigl(1,\ \inf_{y>0} \frac{((1+y)^n - 1)\, 2^\alpha}{\mathsf{P}^*(y) - 1}\Bigr)$$

and in space $N^{O(1)}$ plus the space of $\mathcal{A}$.

Proof. The early answer. If $\alpha < \Xi$, no solution has an integer certificate: for $X = S$ it would give $0 \le g(I, S, |S|) \le \alpha - \Xi < 0$. So answering NO is correct. From now on $\alpha \ge \Xi$.

Number of guesses. There are at most $k(\Xi + 1) \le N^{O(1)}$ guesses, and for each the choice in step (2) ranges over $O(k^2)$ pairs and takes polynomial time: the logarithm of each candidate is a rational plus logarithms of integers, so approximations to within $\frac{1}{4}$ are computable in time $N^{O(1)}$, and a minimizer of the approximations is within a factor $2^{1/2} \le 2$ of the minimum.

Soundness. The algorithm answers YES only when $\mathcal{A}$ accepts some (I, X, k), which means that $X \cup Y \in \mathcal{F}_I$ for some Y with $|Y| \le k - t$, a solution of size at most k.

Completeness. Let w be an integer certificate for S. Put $S^- = \{u \in S : w(u) \le a\}$ and $S^+ = S \setminus S^-$; as the weights are integers, every $u \in S^+$ has $w(u) \ge a + 1$. Then $S^- \ne \emptyset$, since otherwise $\Xi \ge (a+1)k > \Xi$. The pair $(k^-, \Xi^-) = (|S^-|, w(S^-))$ satisfies $1 \le k^- \le k$ and $\Xi^- \le ak^-$; moreover $\Xi^+ = w(S^+) \ge (a+1)k^+$, and if $k^+ = 0$ then $S^+ = \emptyset$ and $\Xi^+ = 0$. So this guess is tried. For it, the balanced family of step (3), applied to the classes S^-, S^+ and the weight w, contains some $X \subseteq S$ with $|X \cap S^-| = t^-$, $|X \cap S^+| = t^+$ and $w(X) \ge (t^-/k^-)\Xi^- + (t^+/k^+)\Xi^+ = t^-\bar{w}^- + t^+\bar{w}^+$ (the second term is absent if $k^+ = 0$). As $t \ge 1$, X is nonempty, so $g(I, X, k) \le \alpha - w(X) \le \theta$, and $\mathcal{A}$ is run on (I, X, k) and accepts, with witness $S \setminus X$.

The threshold is nonnegative. For every guess, $\bar{w}^-, \bar{w}^+ \ge 0$, so $t^-\bar{w}^- + t^+\bar{w}^+ \le k^-\bar{w}^- + k^+\bar{w}^+ = \Xi^- + \Xi^+ = \Xi \le \alpha$. (For $k^+ = 0$ this uses $\Xi^- = \Xi$, which is the condition $\Xi^+ = 0$.) Hence $\theta \ge 0$, and each listed set takes time at most $N^{O(1)} + 2^\theta N^{O(1)} \le 2^\theta N^{O(1)}$.

Time. Fix a guess, and let $v \in \mathbb{R}^k$ have k^- entries $\bar{w}^-$ and k^+ entries $\bar{w}^+$. Its total is $\Xi^- + \Xi^+ = \Xi$, and it has no entry in $(a, a+1)$, because $\bar{w}^- \le a$ and $\bar{w}^+ \ge a+1$. For $1 \le t \le k$ put

$$\mathsf{P}_t = \sum_{t^- + t^+ = t} \binom{k^-}{t^-} \binom{k^+}{t^+} 2^{t^-\bar{w}^- + t^+\bar{w}^+},$$

a positive sum of at most $t + 1$ terms. Its largest term is at least $\mathsf{P}_t/(t+1)$, so by the choice in step (2), which is within a factor 2 of the minimum, and by Theorem 9.6, the running time of the guess is at most

$$2^{O(n(\log\log n)^2/\log n)} N^{O(1)} \min_{1 \le t \le k} (t+1) \binom{n}{t} \frac{2^\alpha}{\mathsf{P}_t}.$$

Expanding the product over the entries of v by subsets and grouping the subsets by their numbers of light and heavy entries gives, for every $y > 0$,

$$\sum_{t=1}^{k} \mathsf{P}_t y^t = \prod_u (1 + y2^{v_u}) - 1 \le \max_{1 \le t \le k} \frac{\mathsf{P}_t}{\binom{n}{t}} \sum_{t=1}^{k} \binom{n}{t} y^t \le \max_{1 \le t \le k} \frac{\mathsf{P}_t}{\binom{n}{t}} ((1+y)^n - 1),$$

so $\min_{1 \le t \le k} \binom{n}{t}/\mathsf{P}_t \le ((1+y)^n - 1)/(\prod_u (1 + y2^{v_u}) - 1)$. Lemma 9.7 bounds the product from below by $\mathsf{P}^*(y)$, and $\mathsf{P}^*(y) > 1$ for $y > 0$. As this holds for every $y > 0$, and $t+1 \le n+1$, the running time of the guess is at most $2^{O(n(\log\log n)^2/\log n)} N^{O(1)} \inf_{y>0}((1+y)^n - 1)2^\alpha/(\mathsf{P}^*(y) - 1)$. Multiplying by

the polynomial number of guesses, and adding the polynomial time of the early answer and of the choices in steps (1) and (2), which the maximum with 1 absorbs, gives the time bound.

Space. The guesses are run one after another, each in the space of Theorem 9.6 plus that of $\mathcal{A}$. □

D.8.2 Deterministic Connected Dominating Set

Lemma 10.1 (Rate certificate)**.** *For every $u \in [0.15, 0.85]$ there is $y \in \{\frac{1+\sqrt{5}}{8}, \frac{1}{2}, \frac{5}{16}, \frac{1}{4}, \frac{27}{128}, \frac{29}{160}\}$ with $\chi_y(u) \le \log(\sqrt{5} - \frac{1}{2})$.*

Proof. For fixed y the function χ_y is affine in u on each interval $[\frac{1}{a+1}, \frac{1}{a}]$, so it suffices to check endpoints. The proof has an algebraic part, on one interval, where a single y gives equality at both endpoints, and a rational part, on five intervals, where the endpoint inequalities are inequalities between rational numbers.

Reduction to endpoints. On each interval $u \in [\frac{1}{a+1}, \frac{1}{a}]$ we use the formula with this value of a. For u in the interior or at the right endpoint this is $\lfloor 1/u \rfloor$; at the left endpoint $u = \frac{1}{a+1}$ we have $\lfloor 1/u \rfloor = a+1$, but both formulas give the same value there, namely $\log(1+y)+1-\frac{1}{a+1}\log(1+y2^{a+1})$, since the coefficients are $(0, \frac{1}{a+1})$ for a and $(\frac{1}{a+1}, 0)$ for $a+1$. For fixed y and a the two coefficients $(a+1)u - 1$ and $1 - au$ are affine in u, so χ_y is affine on the interval and attains its maximum at an endpoint. The intervals that meet $[0.15, 0.85]$ are those with $a = 1, \ldots, 6$.

The interval $a = 2$, $u \in [\frac{1}{3}, \frac{1}{2}]$ (algebraic). Take $y_0 = (1 + \sqrt{5})/8$. Then $1 + 4y_0 = \frac{3+\sqrt{5}}{2} = \left(\frac{1+\sqrt{5}}{2}\right)^2$, $1+8y_0 = 2+\sqrt{5} = \left(\frac{1+\sqrt{5}}{2}\right)^3$ and $2(1+y_0) = (9+\sqrt{5})/4$. At $u = \frac{1}{2}$ the coefficients are $\frac{1}{2}$ and 0, so $2^{\chi_{y_0}(1/2)} = 2(1+y_0)/(1+4y_0)^{1/2}$. At $u = \frac{1}{3}$ they are 0 and $\frac{1}{3}$, so $2^{\chi_{y_0}(1/3)} = 2(1+y_0)/(1+8y_0)^{1/3}$. Both equal

$$\frac{9+\sqrt{5}}{4} \cdot \frac{2}{1+\sqrt{5}} = \frac{9+\sqrt{5}}{2(1+\sqrt{5})} = \frac{(9+\sqrt{5})(\sqrt{5}-1)}{8} = \frac{8\sqrt{5}-4}{8} = \sqrt{5} - \frac{1}{2}.$$

Since χ_{y_0} is affine on the interval, $\chi_{y_0}(u) = \log(\sqrt{5} - \frac{1}{2})$ for every $u \in [\frac{1}{3}, \frac{1}{2}]$.

The other intervals (rational). Use $y = \frac{1}{2}$ for $a = 1$ on $[\frac{1}{2}, \frac{17}{20}]$, $y = \frac{5}{16}$ for $a = 3$ on $[\frac{1}{4}, \frac{1}{3}]$, $y = \frac{1}{4}$ for $a = 4$ on $[\frac{1}{5}, \frac{1}{4}]$, $y = \frac{27}{128}$ for $a = 5$ on $[\frac{1}{6}, \frac{1}{5}]$, and $y = \frac{29}{160}$ for $a = 6$ on $[\frac{3}{20}, \frac{1}{6}]$. At each of the ten endpoints u, both coefficients are nonnegative rationals; with h their least common denominator, the inequality $\chi_y(u) \le \log \frac{1736}{1000}$ is equivalent to

$$(2(1+y))^h \;\le\; \left(\frac{1736}{1000}\right)^h (1+y2^a)^{h((a+1)u-1)} (1+y2^{a+1})^{h(1-au)},$$

an inequality between rational numbers with nonnegative integer exponents. It holds at all ten endpoints, by an exact evaluation (Appendix F.5). The largest endpoint value of $2^{\chi_y(u)}$ on these intervals is $\sqrt{3} = 1.7320\ldots$, at $u = \frac{1}{2}$ for $a = 1$. Since $\frac{1736}{1000} < \sqrt{5} - \frac{1}{2}$, the lemma holds on these intervals. As the five intervals and $[\frac{1}{3}, \frac{1}{2}]$ cover $[0.15, 0.85]$, the lemma follows. □

Lemma 10.2 (Running time of the integer-affine algorithm for tree-type certificates)**.** *Let $n \ge 34$, $0.15n < s < 0.8n$, $\Xi \in \{n-2, n-1\}$ and $\alpha = \Xi + 1$. For every implicit set system, extension algorithm and running-time exponent with values in $\{0, \ldots, n\}$, Algorithm 9.9 on $(I, k = s, \alpha, \Xi)$ runs in time*

$$(\sqrt{5} - \tfrac{1}{2})^n \, 2^{O(n(\log\log n)^2/\log n)} \, N^{O(1)}.$$

Proof. Write $u = s/\Xi$ and $a = \lfloor 1/u \rfloor = \lfloor \Xi/k \rfloor$. In the notation of Theorem 9.10, $k_{a+1} = \Xi - as = \Xi(1-au)$ and $k_a = s - k_{a+1} = \Xi((a+1)u - 1)$, so

$$\log \mathsf{P}^*(y) = \Xi\, \mathsf{p}_y, \qquad \mathsf{p}_y = \big((a+1)u-1\big)\log(1+y2^a) + (1-au)\log(1+y2^{a+1}).$$

The two coefficients of p_y are nonnegative and sum to $u \le 1$, so $0 \le \mathsf{p}_y \le \log(1+y2^{a+1})$. If $\mathsf{P}^*(y) \ge 2$, then $\mathsf{P}^*(y) - 1 \ge \mathsf{P}^*(y)/2$, and, as $(1+y)^n - 1 \le (1+y)^n$ and $\alpha + 1 \le n+1$,

$$\frac{1}{n}\log\frac{\big((1+y)^n-1\big)\,2^\alpha}{\mathsf{P}^*(y)-1} \le \frac{1}{n}\big(n\log(1+y) + \alpha + 1 - \Xi\mathsf{p}_y\big) \le \chi_y(u) + \frac{1+2\mathsf{p}_y}{n},$$

using $\chi_y(u) = \log(1+y) + 1 - \mathsf{p}_y$ in the notation of Lemma 10.1, $\alpha + 1 = \Xi + 2 \le n+1$ and $\Xi\mathsf{p}_y \ge (n-2)\mathsf{p}_y$.

We check the range of u. Since $\Xi \le n-1 < n$ and $n \ge 34$, $u > 0.15n/\Xi > 0.15$, and $u < 0.8n/\Xi \le 0.8n/(n-2) \le 0.85$, where the last inequality is equivalent to $n \ge 34$. So $u \in [0.15, 0.85]$, and $1 \le a \le 6$ because $1/u \in [1/0.85, 1/0.15]$. Lemma 10.1 gives, for this u, some y from its list with $\chi_y(u) \le \log(\sqrt{5} - \frac{1}{2})$. Every listed y is at least $\frac{29}{160}$, so

$$\mathsf{P}^*(y) \ge 1 + (k_a + k_{a+1})\, y2^a = 1 + sy2^a \ge 1 + 0.15 \cdot 34 \cdot 2 \cdot \tfrac{29}{160} > 2.$$

Moreover $y \le \frac{1}{2}$ and $a \le 6$, so $\mathsf{p}_y \le \log(1+2^6)$ is bounded by a constant. Hence Theorem 9.10, evaluated at this y, bounds the time of the run by

$$N^{O(1)}\, 2^{O(n(\log\log n)^2/\log n)} \max\big(1, 2^{n\chi_y(u)+1+2\mathsf{p}_y}\big) \le N^{O(1)}\, 2^{O(n(\log\log n)^2/\log n)}\, \big(\sqrt{5} - \tfrac{1}{2}\big)^n,$$

where the last step uses $\chi_y(u) \le \log(\sqrt{5} - \frac{1}{2})$, $\mathsf{p}_y \ge 0$ and $\sqrt{5} - \frac{1}{2} > 1$. □

Lemma 10.3 (Deterministic decision for one size)**.** *Let Φ be an implicit set system, let $\mathcal{A}$ be a deterministic algorithm that solves Φ-Extension with a running-time exponent g with values in $\{0, \ldots, n\}$, and let $(\alpha, \Xi) \in \{(n-1, n-2), (n, n-1)\}$. For an instance I with a universe of n elements and $1 \le s \le n$, let* Det(I, s) *be the following procedure: if $n < 34$, or $s \le 0.15n$, or $s \ge 0.8n$, test every s-subset of U_I for membership in $\mathcal{F}_I$ and answer* YES *if and only if one of them is a solution; otherwise run Algorithm 9.9 on $(I, k = s, \alpha, \Xi)$ with $\mathcal{A}$ and g. Then* Det *is deterministic, runs in time $(\sqrt{5} - \frac{1}{2})^n 2^{O(n(\log\log n)^2/\log n)} N^{O(1)}$ and in space $N^{O(1)}$ plus the space of $\mathcal{A}$, answers* NO *if $\mathcal{F}_I$ has no solution of size at most s, and answers* YES *whenever $\mathcal{F}_I$ has a solution S of size exactly s which, in the case $n \ge 34$ and $0.15n < s < 0.8n$, has an integer certificate for (α, Ξ) in the sense of Theorem 9.10.*

Proof. The enumeration is deterministic, uses polynomial space, since membership in $\mathcal{F}_I$ is tested in polynomial time (Definition 3.1), and answers YES exactly when $\mathcal{F}_I$ has a solution of size s. It takes time $\binom{n}{s} N^{O(1)}$, which is $N^{O(1)}$ for $n < 34$ and otherwise at most $2^{H(0.15)n} N^{O(1)} = O(1.5261^n) N^{O(1)}$ for $s \le 0.15n$ and $2^{H(0.2)n} N^{O(1)} = O(1.6494^n) N^{O(1)}$ for $s \ge 0.8n$, by (1), since the binary entropy function H is increasing on $[0, \frac{1}{2}]$ and symmetric about $\frac{1}{2}$; both bases are below $\sqrt{5} - \frac{1}{2}$. In the other case, α is a rational and Ξ a nonnegative integer at most n, both computable in polynomial time, and $\alpha \ge \Xi$, so Algorithm 9.9 does not stop early. Theorem 9.10 gives that the run is deterministic, uses space $N^{O(1)}$ plus the space of $\mathcal{A}$, answers NO if $\mathcal{F}_I$ has no solution of size at most s, and answers YES if a solution of size s has an integer certificate; Lemma 10.2 gives the time bound. □

Theorem 10.4 (Deterministic Connected Dominating Set and MaxLeaf)**.** Connected Dominating Set*, and hence* Max Leaf Spanning Tree *(by Lemma 5.1, and directly when $n \le 2$), can be solved by a deterministic algorithm in time*

$$O\Big(\big(\sqrt{5} - \tfrac{1}{2}\big)^n\, 2^{O(n(\log\log n)^2/\log n)}\Big) = O(1.7361^n)$$

and polynomial space, where $\sqrt{5} - \frac{1}{2} = 1.7360\ldots$.

Proof. For $s = 1, \ldots, n$ run the procedure $\mathrm{Det}(G, s)$ of Lemma 10.3 for the implicit set system Φ_{CDS}, the extension algorithm $\mathcal{A}_{\mathrm{St}}$ with its running-time exponent $g(G, X, k) = |\mathrm{T}(J_X)|$, which is deterministic, uses polynomial space and has values in $\{0, \ldots, n\}$ by Corollary 5.8, and $(\alpha, \Xi) = (n-1, n-2)$; output the least s at which it answers YES. By Lemma 10.3, $\mathrm{Det}(G, s)$ answers NO if $\gamma_{\mathrm{c}}(G) > s$. At $s = \gamma_{\mathrm{c}}(G)$ a minimum connected dominating set S has size exactly s, and, for $n \geq 2$, the weights $w = d_F - 1$ of (19) are an integer certificate for S: they are nonnegative integers with $w(S) = n - 2 = \Xi$ and $g(G, X, s) = |\mathrm{T}(J_X)| \leq \alpha - w(X)$ for every nonempty $X \subseteq S$. So $\mathrm{Det}(G, \gamma_{\mathrm{c}}(G))$ answers YES, and the deterministic form of Lemma 5.2 shows that the output is $\gamma_{\mathrm{c}}(G)$, and that the same algorithm solves Max Leaf Spanning Tree. The time is n times the bound of Lemma 10.3, which is $O(1.7361^n)$ because $2^{O(n(\log\log n)^2/\log n)} = 2^{o(n)}$ and $\sqrt{5} - \frac{1}{2} < 1.7361$, and the space is polynomial. □

D.9 Connected Feedback Vertex Set

This subsection proves the two hardness facts of Section 11.3.1 and constructs the extension algorithm of Lemma 11.7. We keep the notation of Section 11.3.

D.9.1 Hardness of the extension problem

Lemma D.9.1 (The extension problem is W[2]-hard)**.** Connected Feedback Vertex Set Extension*, parameterized by the number* $k - |X|$ *of further vertices, is W[2]-hard.*

Proof. We reduce from Set Cover parameterized by the number of sets, which is W[2]-hard [CFK+15]. Let $(\mathcal{U}, \mathcal{S}, k)$ be an instance with ground set $\mathcal{U}$, a family $\mathcal{S}$ of subsets of $\mathcal{U}$ and an integer k. We may assume that $\mathcal{U} \neq \emptyset$, that $\mathcal{S} \neq \emptyset$ and that every element lies in some set, since otherwise the instance is trivial. Build a graph G with a vertex for every element, a vertex for every set and a root r; join r to every set vertex, and join each set vertex to the elements it contains. Then G is connected. Put $X = \{r\} \cup \mathcal{U}$ and take the size bound $|X| + k$, so that k further vertices may be added.

The vertices outside X are the set vertices. Let Y be a set of set vertices. The graph $G - (X \cup Y)$ consists of set vertices only, which are pairwise non-adjacent, so it is a forest. In $G[X \cup Y]$ the vertices of X are pairwise non-adjacent and every vertex of Y is adjacent to r. Hence $G[X \cup Y]$ is connected if and only if every element vertex has a neighbor in Y, that is, if and only if Y covers $\mathcal{U}$. So $(G, X, |X| + k)$ is a YES-instance if and only if $\mathcal{U}$ has a cover by at most k sets. The reduction is polynomial and keeps the parameter. □

The next statement uses the Set Cover Conjecture of [CDL+16]: for every $\varepsilon > 0$ there is an integer h such that Set Cover with sets of size at most h, on a universe $\mathcal{U}$ of $n_{\mathcal{U}}$ elements and with $|\mathcal{S}|$ sets, cannot be solved in time $(2 - \varepsilon)^{n_{\mathcal{U}}}(n_{\mathcal{U}} + |\mathcal{S}|)^{O(1)}$.

Proposition D.9.2 (Product-form running times cannot help)**.** *Assume the Set Cover Conjecture. Let* $c, d \geq 1$ *be constants such that* Connected Feedback Vertex Set Extension *can be solved in time* $c^{k-|X|}d^{|X|}N^{O(1)}$ *on inputs* (G, X, k)*. Then* $c \geq 2$ *and* $d \geq 2$*, and the running-time bound of Corollary 4.6 with* $(b, c, d) = (1, c, d)$ *has a base at least* 2*.*

Proof. Suppose that $d < 2$. Put $\varepsilon = (2 - d)/2 > 0$ and fix an integer $j \geq 1$ with $c^{1/j}d \leq 2 - \varepsilon$; it exists because $c^{1/j} \to 1$ and $d < 2 - \varepsilon$. Neither ε nor j depends on h. Let h be arbitrary and let $(\mathcal{U}, \mathcal{S}, k)$ be an instance of Set Cover with sets of size at most h and $n_{\mathcal{U}} = |\mathcal{U}|$; we may assume $k \leq n_{\mathcal{U}}$, since otherwise the answer is YES if and only if every element lies in some set.

1. *Padding.* Let $\hat{k} = \lceil k/j \rceil$ and $p = j\hat{k} - k < j$. Add p new elements, each with its own singleton set. Every cover of the new instance contains the p singletons, because they are the only sets containing the new elements. Hence the new instance has a cover with at most $j\hat{k}$ sets if and only if the old one has a cover with at most k sets.

2. *Unions.* Replace the sets by all unions of at most j of them. There are at most $(|\mathcal{S}| + j + 1)^j$ such unions, a polynomial for fixed j. The padded instance has a cover with at most $j\hat{k}$ sets if and only if the union instance has a cover with at most $\hat{k}$ sets: group the sets of a cover into $\hat{k}$ groups of at most j, and conversely split each union.

3. *Reduction.* Apply the reduction of Lemma D.9.1 to the union instance. It produces a forced set with $|X| \leq n_{\mathcal{U}} + j + 1$ and $\hat{k} \leq n_{\mathcal{U}}/j + 1$ further vertices, so the assumed algorithm decides it in time

$$c^{\hat{k}} d^{n_{\mathcal{U}}+j+1} N^{O(1)} \leq c\, d^{j+1} \big(c^{1/j} d\big)^{n_{\mathcal{U}}} N^{O(1)} \leq (2-\varepsilon)^{n_{\mathcal{U}}} (n_{\mathcal{U}} + |\mathcal{S}|)^{O(1)},$$

since N is polynomial in $n_{\mathcal{U}} + |\mathcal{S}|$ for fixed j.

This holds for every h with the same ε, contradicting the conjecture. Hence $d \geq 2$.

Now suppose that $c < 2$. Put $\varepsilon = (2-c)/2 > 0$ and fix an integer $j \geq 1$ with $c^{1+1/j} \leq 2 - \varepsilon$; it exists because $c^{1/j} \to 1$ and $c < 2 - \varepsilon$. Let h be arbitrary, let $(\mathcal{U}, \mathcal{S}, k)$ be an instance of Set Cover with sets of size at most h, $n_{\mathcal{U}} = |\mathcal{U}| \geq 1$ and $k \leq n_{\mathcal{U}}$, in which every element lies in some set (otherwise the answer is no), and apply the padding and union steps above. They yield an instance $(\mathcal{U}', \mathcal{S}', \hat{k})$ with $n_{\mathcal{U}} \leq |\mathcal{U}'| < n_{\mathcal{U}} + j$ and $\hat{k} \leq n_{\mathcal{U}}/j + 1$, in which every element lies in some set, and which has a cover with at most $\hat{k}$ sets if and only if the original instance has a cover with at most k sets. Build the graph of the proof of Lemma D.9.1 for $(\mathcal{U}', \mathcal{S}')$, with a root r, one vertex for every set of $\mathcal{S}'$ and one vertex for every element of $\mathcal{U}'$, and attach to r and to every element vertex v a *private triangle*: two new vertices v', v'', adjacent to v and to each other. Call r and the element vertices *anchors*; as $\mathcal{U}' \neq \emptyset$, there are at least two of them. Let G'' be the resulting connected graph, put $X = \emptyset$ and $s = |\mathcal{U}'| + 1 + \hat{k}$. We claim that G'' has a connected feedback vertex set of size at most s if and only if $\mathcal{U}'$ has a cover by at most $\hat{k}$ sets of $\mathcal{S}'$.

If Y is such a cover, then $A = \{r\} \cup \mathcal{U}' \cup Y$, viewed as a vertex set, is a connected feedback vertex set of size at most s: $G''[A]$ is connected, because every set vertex is adjacent to r and every element vertex is adjacent to a set vertex in Y; and $G'' - A$ consists of isolated set vertices and the edges $v'v''$, so it is a forest. Conversely, let A be a connected feedback vertex set of G'' with $|A| \leq s$. Every anchor lies in A: if an anchor v is not in A, then v' or v'' is, because A meets the triangle at v; the neighbors of v' and of v'' are v and each other, so the connectivity of $G''[A]$ forces $A \subseteq \{v', v''\}$, and then $G'' - A$ contains the triangle of another anchor, a contradiction. Removing all triangle vertices from A keeps it a connected feedback vertex set of no larger size: a triangle vertex is adjacent only to its anchor, which lies in A, and to its twin, so it is an interior vertex of no path of $G''[A]$ between other vertices, and returning v' and v'' to $G'' - A$ adds at most the isolated edge $v'v''$. So we may assume that $A = \{r\} \cup \mathcal{U}' \cup Y$ for a set Y of at most $s - |\mathcal{U}'| - 1 = \hat{k}$ set vertices, and the connectivity of $G''[A]$ means that every element vertex has a neighbor in Y, that is, that Y covers $\mathcal{U}'$. This proves the claim.

The assumed algorithm decides $(G'', \emptyset, s)$ in time

$$c^s N^{O(1)} \leq c^{n_{\mathcal{U}}+j+n_{\mathcal{U}}/j+2} N^{O(1)} = c^{j+2} \big(c^{1+1/j}\big)^{n_{\mathcal{U}}} N^{O(1)} \leq (2-\varepsilon)^{n_{\mathcal{U}}} (n_{\mathcal{U}} + |\mathcal{S}|)^{O(1)},$$

again for every h with the same ε, contradicting the conjecture. Hence $c \geq 2$.

With $b = 1$ the base of Corollary 4.6 is $1 + \min(d, c)(1 - 1/c)$. If $d \geq c$, it equals $c \geq 2$. If $d < c$, it equals $1 + d - d/c > 1 + d - 1 = d \geq 2$. □

Proposition D.9.2 concerns extension algorithms whose running-time bound depends only on $k - |X|$ and $|X|$ (the case $b = 1$); it says nothing about bounds with a further factor $b^{n-|X|}$, $b > 1$.

D.9.2 Contractions

The extension algorithm contracts the components of the sample, and its iterative compression runs along a sequence of edge contractions. This subsection collects the facts about contractions that the proofs use. For a graph $\tilde{G}$ and an edge $uv \in E(\tilde{G})$, the graph $\tilde{G}/uv$ is obtained by replacing u and v by a new vertex w adjacent to $N_{\tilde{G}}(u) \cup N_{\tilde{G}}(v) \setminus \{u, v\}$; it is simple. For a set $C \subseteq V(\tilde{G})$ with $\tilde{G}[C]$ connected, contracting C means replacing C by one vertex adjacent to $N_{\tilde{G}}(C)$; it is the result of $|C| - 1$ edge contractions inside C. We use the standard count: a graph with p vertices, e edges and q components satisfies $e \geq p - q$, with equality if and only if it is a forest, since a connected graph on p_i vertices has at least $p_i - 1$ edges, with equality if and only if it is a tree. The lemma below records that an edge contraction preserves the number of components and the property of being a forest, and that contracting the sample does not change the extension problem.

Let G be connected, $X \subseteq V(G)$ nonempty, and let $X_1, \ldots, X_q$ be the vertex sets of the components of $G[X]$, so $q = \mathrm{comp}(G[X])$. The *contracted graph* G_X is obtained from G by contracting each X_j to a new vertex x_j. Its vertex set is $(V(G) \setminus X) \cup \hat{X}$ with $\hat{X} = \{x_1, \ldots, x_q\}$; we call the vertices of $\hat{X}$ *forced*. Distinct components of $G[X]$ are not adjacent, so $\hat{X}$ is an independent set of G_X.

Lemma D.9.3 (Contractions)**.** *(1) Let $\tilde{G}$ be a graph and $uv \in E(\tilde{G})$. Then $\mathrm{comp}(\tilde{G}/uv) = \mathrm{comp}(\tilde{G})$; in particular, $\tilde{G}$ is connected if and only if $\tilde{G}/uv$ is. If $\tilde{G}$ is a forest, then $\tilde{G}/uv$ is a forest.*

(2) With G, X, G_X and $\hat{X}$ as above, G_X is connected, and for every $Y \subseteq V(G) \setminus X$, the set $X \cup Y$ is a connected feedback vertex set of G if and only if $\hat{X} \cup Y$ is a connected feedback vertex set of G_X.

Proof. (1) The map h that sends u and v to w and fixes the other vertices sends every edge of $\tilde{G}$ other than uv to an edge of $\tilde{G}/uv$, and every edge of $\tilde{G}/uv$ is the image of an edge of $\tilde{G}$; since $h^{-1}(w) = \{u, v\}$ is connected in $\tilde{G}$, h induces a bijection between the components. If $\tilde{G}$ is a forest, then $\tilde{G}/uv$ has one vertex fewer, at least one edge fewer and the same number of components; as $|E(\tilde{G})| = |V(\tilde{G})| - \mathrm{comp}(\tilde{G})$, the count $e \geq p - q$ holds with equality for $\tilde{G}/uv$, so it is a forest. (2) G_X arises from the connected graph G by edge contractions, so it is connected by (1). The graphs $G - (X \cup Y)$ and $G_X - (\hat{X} \cup Y)$ are identical: both have vertex set $V(G) \setminus (X \cup Y)$, and contracting the sets X_j does not change the edges between vertices outside X. The graph $G_X[\hat{X} \cup Y]$ is obtained from $G[X \cup Y]$ by contracting the connected sets $X_1, \ldots, X_q$, so by (1) one is connected if and only if the other is. □

For the iterative compression we need contraction closure in the presence of forced vertices, which adapts the observation in the proof of Theorem B.3 of [CNP+11]. Let $\tilde{G}$ be a graph with a forced set $\hat{X}$, let $uv \in E(\tilde{G})$, let $\tilde{G}_{uv} = \tilde{G}/uv$ with new vertex w, and let the forced set $\hat{X}_{uv}$ of $\tilde{G}_{uv}$ be the image of $\hat{X}$, so $w \in \hat{X}_{uv}$ if and only if $u \in \hat{X}$ or $v \in \hat{X}$. The *excess* of a set A in a graph with forced set $\hat{X}$ is $|A \setminus \hat{X}|$.

Lemma D.9.4 (Contraction closure with forced vertices)**.** *Let $\tilde{G}$, $\hat{X}$, uv, $\tilde{G}_{uv}$, w and $\hat{X}_{uv}$ be as above, and let A be a connected feedback vertex set of $\tilde{G}$ with $\hat{X} \subseteq A$. Put $A' = A$ if $u, v \notin A$, and $A' = (A \setminus \{u, v\}) \cup \{w\}$ otherwise. Then A' is a connected feedback vertex set of $\tilde{G}_{uv}$, $\hat{X}_{uv} \subseteq A'$, and $|A' \setminus \hat{X}_{uv}| \leq |A \setminus \hat{X}|$.*

Proof. Suppose first that $u, v \notin A$. Then $A' = A$, $\tilde{G}_{uv}[A] = \tilde{G}[A]$, which is connected, and $\tilde{G}_{uv} - A = (\tilde{G} - A)/uv$, which is a forest by Lemma D.9.3(1). Since $w \notin A'$ and $\hat{X} \subseteq A$ contains neither u nor v, we have $\hat{X}_{uv} = \hat{X}$, so $\hat{X}_{uv} \subseteq A'$ and $A' \setminus \hat{X}_{uv} = A \setminus \hat{X}$.

Suppose now that $u \in A$ or $v \in A$. Then $\tilde{G}_{uv} - A' = \tilde{G} - (A \cup \{u, v\})$ is an induced subgraph of the forest $\tilde{G} - A$. The set $A \cup \{u, v\}$ induces a connected subgraph of $\tilde{G}$, because $\tilde{G}[A]$ is connected and uv is an edge with an end in A; and $\tilde{G}_{uv}[A'] = \tilde{G}[A \cup \{u, v\}]/uv$ is connected by Lemma D.9.3(1). Every forced vertex of $\tilde{G}_{uv}$ other than w lies in $\hat{X} \setminus \{u, v\} \subseteq A \setminus \{u, v\} \subseteq A'$, and $w \in A'$; so $\hat{X}_{uv} \subseteq A'$. Finally, $A' \setminus \hat{X}_{uv} \subseteq (A \setminus \hat{X} \setminus \{u, v\}) \cup (\{w\} \setminus \hat{X}_{uv})$. If $w \notin \hat{X}_{uv}$, then $u, v \notin \hat{X}$, and at least one of them lies in $A \setminus \hat{X}$, so the excess does not increase. If $w \in \hat{X}_{uv}$, the excess is at most $|A \setminus \hat{X}|$ directly. □

D.9.3 Cut & Count with forced vertices

The counting step of the extension algorithm is the Cut & Count algorithm of [CNP+11, CNP+22] for the constrained problem, run on a tree decomposition of the form "forest plus a fixed modulator," as in their proof of the bound $3^k n^{O(1)}$ for Connected Feedback Vertex Set. The journal version [CNP+22] states these bounds but leaves their proofs to the extended arXiv version [CNP+11] (see [CNP+22, Section 1.5]), so the numbers below that concern Connected Feedback Vertex Set refer to [CNP+11]; the corresponding results for Feedback Vertex Set are [CNP+22, Theorems 4.6 and 5.1]. Constrained Connected Feedback Vertex Set asks, given a graph $\tilde{G}$ with $\tilde{n}$ vertices, a nonempty set $Y_{\text{in}} \subseteq V(\tilde{G})$ and an integer η, whether $\tilde{G}$ has a connected feedback vertex set $Y \supseteq Y_{\text{in}}$ with $|Y| = \eta$. We use the following facts from these papers.

(CC1) ([CNP+11, Theorem A.13].) Given a tree decomposition of $\tilde{G}$ of width w, a Monte Carlo algorithm decides Constrained Connected Feedback Vertex Set in time $4^w \tilde{n}^{O(1)}$. It never accepts a no-instance, and it rejects a yes-instance with probability at most 1/2.

(CC2) (Proof of [CNP+11, Theorem A.13].) The algorithm fixes a vertex $r \in Y_{\text{in}}$ and computes, modulo 2, the number of candidate solutions with a marked candidate forest (markers on side 1 of its cut), together with a consistent cut of the candidate forest and a consistent cut of the candidate solution that puts r on side 1. Its dynamic program gives every vertex of a bag one of four states, which we write $\mathsf{y}_1, \mathsf{y}_2$ (in the solution, on side 1 or 2 of its cut) and $\mathsf{z}_1, \mathsf{z}_2$ (in the forest, on side 1 or 2 of its cut; these are the states $0_1, 0_2, 1_1, 1_2$ of [CNP+11]). A counted object gives a state y_1 or y_2 to every vertex of Y_{in} and the state y_1 to r, and no edge has the pair of states $\{\mathsf{y}_1, \mathsf{y}_2\}$ or $\{\mathsf{z}_1, \mathsf{z}_2\}$, because both cuts are consistent.

(CC3) (Proof of [CNP+22, Theorem 5.1] for Feedback Vertex Set, which the proof of Theorem B.3 of [CNP+11] repeats for the connected problem.) Let $\text{Mod} \subseteq V(\tilde{G})$ be such that $\tilde{G} - \text{Mod}$ is a forest, and use the tree decomposition obtained from a decomposition of width 1 of $\tilde{G} - \text{Mod}$ by adding Mod to every bag. Below the bags that forget the vertices of Mod, each table entry for a coloring ς of Mod refers only to entries with the same coloring of Mod. So the count is the sum over the colorings $\varsigma\colon \text{Mod} \to \{\mathsf{y}_1, \mathsf{y}_2, \mathsf{z}_1, \mathsf{z}_2\}$ of the counts with Mod colored by ς (the "core evaluations"), and each of these is computed in time and space polynomial in $\tilde{n}$.

By (CC2), only the following colorings of Mod can have a nonzero count. A coloring $\varsigma\colon \text{Mod} \to \{\mathsf{y}_1, \mathsf{y}_2, \mathsf{z}_1, \mathsf{z}_2\}$ is *reasonable* if $\varsigma(v) \in \{\mathsf{y}_1, \mathsf{y}_2\}$ for $v \in Y_{\text{in}} \cap \text{Mod}$, $\varsigma(r) = \mathsf{y}_1$ when $r \in \text{Mod}$, and no edge uv of $\tilde{G}[\text{Mod}]$ has $\{\varsigma(u), \varsigma(v)\} = \{\mathsf{y}_1, \mathsf{y}_2\}$ or $\{\varsigma(u), \varsigma(v)\} = \{\mathsf{z}_1, \mathsf{z}_2\}$. Summing the core evaluations over the reasonable colorings therefore gives the same parity as the algorithm of (CC1). The following lemma bounds the number of reasonable colorings for the modulators produced by

iterative compression. It is the forced-vertex version of the count $4^3 3^{k-1}$ in the proof of Theorem B.3 of [CNP+11], in which only the vertex r and the at most two vertices outside the connected part of the modulator have four possible states.

Lemma D.9.5 (Number of reasonable colorings)**.** *Let $\tilde{G}$, Y_{in} and r be as above. Let* Mod $= C \cup C'$ *with $C \cap C' = \emptyset$, $\tilde{G}[C]$ connected and $|C'| \leq 2$, and let $\hat{X} \subseteq Y_{\text{in}} \cap C$ with $r \in \hat{X}$. Then* Mod *has at most $4^{|C'|}3^{|C\setminus\hat{X}|}2^{|\hat{X}|-1}$ reasonable colorings, and they can be listed in time $4^{|C'|}3^{|C\setminus\hat{X}|}2^{|\hat{X}|-1}\tilde{n}^{O(1)}$ and polynomial space. In particular, if $|\text{Mod} \setminus \hat{X}| \leq k'+2$, there are at most $8 \cdot 3^{k'}2^{|\hat{X}|}$ of them.*

Proof. Fix a spanning tree of $\tilde{G}[C]$ rooted at r, and order C so that every vertex comes after its parent. Let ς be reasonable. Then $\varsigma(r) = \mathsf{y}_1$. Let $v \in C \setminus \{r\}$ have parent p. If $\varsigma(p) = \mathsf{y}_j$, then $\varsigma(v) \in \{\mathsf{y}_j, \mathsf{z}_1, \mathsf{z}_2\}$, because $\{\varsigma(p), \varsigma(v)\} \neq \{\mathsf{y}_1, \mathsf{y}_2\}$; if $\varsigma(p) = \mathsf{z}_j$, then $\varsigma(v) \in \{\mathsf{z}_j, \mathsf{y}_1, \mathsf{y}_2\}$. If moreover $v \in \hat{X} \subseteq Y_{\text{in}}$, then $\varsigma(v) \in \{\mathsf{y}_1, \mathsf{y}_2\}$, which leaves one option when $\varsigma(p)$ is a y-state and two when it is a z-state. So, given the values on the earlier vertices, $\varsigma(v)$ lies in an explicit set of at most 3 values, and at most 2 if $v \in \hat{X}$. The vertices of C' take at most 4 values. Enumerating all choices in this order lists a superset of the reasonable colorings, of size at most $4^{|C'|}3^{|C\setminus\hat{X}|}2^{|\hat{X}|-1}$; testing each candidate takes polynomial time, and only the current candidate is stored.

For the last claim, put $n_1 = |C \setminus \hat{X}|$ and $n_2 = |C'|$. Then $n_1 + n_2 = |\text{Mod} \setminus \hat{X}| \leq k'+2$ and $n_2 \leq 2$, so $4^{n_2}3^{n_1} = (4/3)^{n_2}3^{n_1+n_2} \leq (16/9)3^{k'+2} = 16 \cdot 3^{k'}$, and $16 \cdot 3^{k'}2^{|\hat{X}|-1} = 8 \cdot 3^{k'}2^{|\hat{X}|}$. □

A *structured modulator* for $(\tilde{G}, \hat{X}, k')$, where $\hat{X} \subseteq V(\tilde{G})$ is nonempty and $k' \geq 0$, is a set Mod $= C \cup C'$ such that $\tilde{G} - \text{Mod}$ is a forest, $\tilde{G}[C]$ is connected, $\hat{X} \subseteq C$, $|C'| \leq 2$, $C \cap C' = \emptyset$ and $|\text{Mod} \setminus \hat{X}| \leq k'+2$.

Lemma D.9.6 (The test)**.** *Let $\tilde{G}$ be a graph with $\tilde{n}$ vertices, let $\emptyset \neq \hat{X} \subseteq Y_{\text{in}} \subseteq V(\tilde{G})$, let $k' \geq 0$, and let* Mod *be a structured modulator for $(\tilde{G}, \hat{X}, k')$. One can decide in time $3^{k'}2^{|\hat{X}|}\tilde{n}^{O(1)}$ and polynomial space, with one-sided error, whether $\tilde{G}$ has a connected feedback vertex set $Y \supseteq Y_{\text{in}}$ with $|Y| \leq |\hat{X}| + k'$: the algorithm never accepts if there is no such set, and it accepts with probability at least $1/2$ if there is one.*

Proof. Fix $r \in \hat{X}$. For every η with $|Y_{\text{in}}| \leq \eta \leq \min(|\hat{X}| + k', \tilde{n})$, run the algorithm of (CC1) on $(\tilde{G}, Y_{\text{in}}, \eta)$ with the root r and the tree decomposition of (CC3), computing the count as the sum of the core evaluations over the reasonable colorings of Mod, listed by Lemma D.9.5; accept if one of these runs accepts. By the remark before Lemma D.9.5, each run decides as the algorithm of (CC1) does. Hence no run accepts if there is no set Y as in the lemma, and if there is one, the run with $\eta = |Y|$, which lies in the range tried, accepts with probability at least $1/2$. There are at most $\tilde{n}+1$ runs, each sums at most $8 \cdot 3^{k'}2^{|\hat{X}|}$ core evaluations by Lemma D.9.5, and each core evaluation takes polynomial time and space by (CC3); the sum modulo 2 is accumulated coloring by coloring. □

D.9.4 The extension algorithm

Lemma 11.7 (The extension algorithm)**.** *There is a randomized algorithm $\mathcal{A}$ for* Connected Feedback Vertex Set Extension *that, given a connected graph G with n vertices, a set $X \subseteq V(G)$ and an integer $k \geq |X|$, runs in polynomial space and, for every outcome of its coins, in time $3^{k-|X|}2^{\text{comp}(G[X])}n^{O(1)}$. If it accepts, it outputs a set $Y \subseteq V(G) \setminus X$ with $|Y| \leq k - |X|$ such that $X \cup Y$ is a connected feedback vertex set of G. On a* yes*-instance it accepts with probability at least $3/4$. In particular, the running-time exponent* (22) *bounds its running time.*

The extension algorithm contracts the sample and then follows the iterative compression of the proof of Theorem B.3 of [CNP+11] along a contraction sequence, keeping track of the images of the forced vertices. Lemma D.9.4 guarantees that every intermediate graph has a solution of no larger excess, and the previous solution, expanded, is a structured modulator for Lemma D.9.6. Running times are stated for every outcome of the coins, as Corollary 3.15 requires. The extension algorithm solves a *forced problem*, in which the sample has been contracted: given a connected graph $\tilde{G}$ with $\tilde{n}$ vertices, a nonempty set $\hat{X} \subseteq V(\tilde{G})$, weights $\tilde{w}\colon V(\tilde{G}) \to \{0, \dots, w_{\max}\}$ with $\tilde{w} = 0$ on $\hat{X}$, and integers $k' \geq 0$ (the number of further vertices) and θ', find a connected feedback vertex set $A \supseteq \hat{X}$ of $\tilde{G}$ with $|A \setminus \hat{X}| \leq k'$ and $\tilde{w}(A) \leq \theta'$; we call such an A a *solution* of the forced instance $(\tilde{G}, \hat{X}, \tilde{w}, k', \theta')$. The weights are carried so that Appendix D.10 can reuse the following wrapper, which turns a procedure for the forced problem into an extension algorithm. For $X = \emptyset$ it guesses one solution vertex, as in Remark A.9 of [CNP+11].

Lemma D.9.7 (From the forced problem to the extension problem)**.** *Let G be a connected graph with n vertices, $w\colon V(G) \to \{0, \dots, w_{\max}\}$, θ an integer, $X \subseteq V(G)$ and $k \geq |X|$, and put $k' = k - |X|$. Let $p \in (0, 1]$, and let* Forced *be a randomized procedure that, given a forced instance $(\tilde{G}, \hat{X}, \tilde{w}, k'', \theta')$ with $\tilde{n} \leq n$, runs in polynomial space and, for every outcome of its coins, in time $3^{k''} 2^{|\hat{X}|} (\tilde{n} + w_{\max})^{O(1)}$, outputs only solutions of the instance, and outputs one with probability at least p whenever one exists. Consider the following algorithm.*

1. *If $X \neq \emptyset$: compute the contracted graph G_X with forced set $\hat{X}$, put $\tilde{w} = w$ on $V(G) \setminus X$ and $\tilde{w} = 0$ on $\hat{X}$, and run* Forced *on $(G_X, \hat{X}, \tilde{w}, k', \theta - w(X))$. If it outputs A, accept with $Y = A \setminus \hat{X}$; otherwise reject.*

2. *If $X = \emptyset$: if G is a forest, accept with $Y = \emptyset$ when $\theta \geq 0$ and reject otherwise. If G is not a forest and $k' = 0$, reject. Otherwise, for every $v \in V(G)$, let $\tilde{w}_v$ agree with w except that $\tilde{w}_v(v) = 0$, and run* Forced *on $(G, \{v\}, \tilde{w}_v, k' - 1, \theta - w(v))$; if one run outputs A, accept with $Y = A$. Otherwise reject.*

The algorithm runs in polynomial space and, for every outcome of its coins, in time $3^{k'} 2^{\mathrm{comp}(G[X])} (n + w_{\max})^{O(1)}$. If it accepts, then $Y \subseteq V(G) \setminus X$, $|Y| \leq k'$, and $X \cup Y$ is a connected feedback vertex set of G with $w(X \cup Y) \leq \theta$; if such a set Y exists, it accepts with probability at least p.

Proof. Case $X \neq \emptyset$. The graph G_X is connected with at most n vertices, and $\tilde{w} = 0$ on $\hat{X}$. For $Y \subseteq V(G) \setminus X$, by Lemma D.9.3(2) the set $X \cup Y$ is a connected feedback vertex set of G if and only if $\hat{X} \cup Y$ is one of G_X, and $w(X \cup Y) = w(X) + \tilde{w}(\hat{X} \cup Y)$. So the sets Y in question are exactly the sets $A \setminus \hat{X}$ for the sets A that Forced may output on $(G_X, \hat{X}, \tilde{w}, k', \theta - w(X))$, and the claims follow from the properties of Forced, with $|\hat{X}| = \mathrm{comp}(G[X])$.

Case $X = \emptyset$. If G is a forest, then $\emptyset$ is a connected feedback vertex set, and no set has smaller weight than $w(\emptyset) = 0$, so the answer is correct. Otherwise every connected feedback vertex set is nonempty. If a run for v outputs A, then $A \ni v$ is a connected feedback vertex set with $|A| \leq 1 + (k' - 1)$ and $w(A) = \tilde{w}_v(A) + w(v) \leq \theta$. Conversely, let Y^* be a connected feedback vertex set with $|Y^*| \leq k'$ and $w(Y^*) \leq \theta$. Then $Y^* \neq \emptyset$, so $k' \geq 1$; for $v \in Y^*$, the set Y^* contains $\{v\}$, has $|Y^* \setminus \{v\}| \leq k' - 1$ and $\tilde{w}_v(Y^*) = w(Y^*) - w(v) \leq \theta - w(v)$, so the run for v outputs a set with probability at least p. The time is $n \cdot 3^{k'-1} 2^1 (n + w_{\max})^{O(1)} \leq 3^{k'} (n + w_{\max})^{O(1)}$, with $\mathrm{comp}(G[\emptyset]) = 0$. □

Proof of Lemma 11.7. The forced problem. We first solve the forced problem without weights: given a connected graph $\tilde{G}$ with $\tilde{n} \leq n$ vertices, a nonempty set $\hat{X} \subseteq V(\tilde{G})$ and $k' \geq 0$, find a connected

feedback vertex set $A \supseteq \hat{X}$ of $\tilde{G}$ with $|A \setminus \hat{X}| \le k'$. Compute graphs $\tilde{G}_{\tilde{n}} = \tilde{G}, \tilde{G}_{\tilde{n}-1}, \dots, \tilde{G}_1$ with $\tilde{G}_i = \tilde{G}_{i+1}/u_iv_i$ for an edge u_iv_i of $\tilde{G}_{i+1}$ and new vertex w_i; all of them are connected by Lemma D.9.3(1). Let $\hat{X}_i$ be the image of $\hat{X}$ in $\tilde{G}_i$; then $|\hat{X}_i| \le |\hat{X}|$, $\tilde{G}_1$ has one vertex, $\hat{X}_1 = V(\tilde{G}_1)$, and we set $A_1 = V(\tilde{G}_1)$. For $i = 1, \dots, \tilde{n}-1$, given a connected feedback vertex set $A_i \supseteq \hat{X}_i$ of $\tilde{G}_i$ with $|A_i \setminus \hat{X}_i| \le k'$, let $\mathrm{Mod} = (A_i \setminus \{w_i\}) \cup \{u_i, v_i\}$ in $\tilde{G}_{i+1}$, and let $C = \mathrm{Mod}$, $C' = \emptyset$ if $w_i \in A_i$, and $C = A_i$, $C' = \{u_i, v_i\}$ otherwise. Start with $Y_{\mathrm{in}} = \hat{X}_{i+1}$ and go through the vertices $v \notin \hat{X}_{i+1}$ in a fixed order: run the test of Lemma D.9.6 on $(\tilde{G}_{i+1}, Y_{\mathrm{in}} \cup \{v\}, \hat{X}_{i+1}, k', \mathrm{Mod})$ independently $\lceil \log(4\tilde{n}^2) \rceil$ times, and if one run accepts, add v to Y_{in}. Afterward, if Y_{in} is a connected feedback vertex set of $\tilde{G}_{i+1}$ with $|Y_{\mathrm{in}} \setminus \hat{X}_{i+1}| \le k'$, set $A_{i+1} = Y_{\mathrm{in}}$; otherwise stop without a solution. Output $A_{\tilde{n}}$. This is the self-reduction of the proof of [CNP+22, Theorem 5.1] and of Theorem B.3 of [CNP+11], with the forced vertices kept in Y_{in} throughout.

The modulator. Write u, v, w for u_i, v_i, w_i. The graphs $\tilde{G}_{i+1} - \{u, v\}$ and $\tilde{G}_i - w$ are identical, so $\tilde{G}_{i+1} - \mathrm{Mod} = \tilde{G}_i - (A_i \cup \{w\})$ is an induced subgraph of the forest $\tilde{G}_i - A_i$. Every forced vertex of $\tilde{G}_{i+1}$ other than u, v is a forced vertex of $\tilde{G}_i$ other than w, hence lies in $A_i \setminus \{w\}$. If $w \in A_i$, then $\tilde{G}_{i+1}[\mathrm{Mod}]$ is connected by Lemma D.9.3(1), because $\tilde{G}_{i+1}[\mathrm{Mod}]/uv = \tilde{G}_i[A_i]$, and $\hat{X}_{i+1} \subseteq \mathrm{Mod} = C$. If $w \notin A_i$, then $w \notin \hat{X}_i$, so $u, v \notin \hat{X}_{i+1}$, $\hat{X}_{i+1} \subseteq A_i = C$, and $\tilde{G}_{i+1}[A_i] = \tilde{G}_i[A_i]$ is connected. Finally $|\mathrm{Mod} \setminus \hat{X}_{i+1}| \le |A_i \setminus \hat{X}_i| + 2 \le k' + 2$. So Mod is a structured modulator for $(\tilde{G}_{i+1}, \hat{X}_{i+1}, k')$, and $\hat{X}_{i+1} \subseteq Y_{\mathrm{in}} \cup \{v\}$ in every test.

Correctness. Every set A_i is checked before it is kept, so an output is a connected feedback vertex set $A \supseteq \hat{X}$ of $\tilde{G}$ with $|A \setminus \hat{X}| \le k'$. Suppose such a set exists. By Lemma D.9.4, applied along the contraction sequence, so does every $\tilde{G}_i$ for $\hat{X}_i$. Step i is the self-reduction of Lemma D.10.1 with $\Omega = V(\tilde{G}_{i+1})$, $Z_0 = \hat{X}_{i+1}$, the family $\mathcal{P}$ of the connected feedback vertex sets $Y \supseteq \hat{X}_{i+1}$ of $\tilde{G}_{i+1}$ with $|Y \setminus \hat{X}_{i+1}| \le k'$, which is nonempty, the test $\mathsf{T}(Y_{\mathrm{in}})$ of Lemma D.9.6 on $(\tilde{G}_{i+1}, Y_{\mathrm{in}}, \hat{X}_{i+1}, k', \mathrm{Mod})$, $\delta = 1/(4\tilde{n})$ and $\rho = \lceil \log(4\tilde{n}^2) \rceil \ge \log(|\Omega|/\delta)$, since $|\Omega| \le \tilde{n}$. By Lemma D.9.6, the test rejects Y_{in} if no member of $\mathcal{P}$ contains it (for $Y \supseteq \hat{X}_{i+1}$ the conditions $|Y| \le |\hat{X}_{i+1}| + k'$ and $|Y \setminus \hat{X}_{i+1}| \le k'$ agree), and with probability at least $1/2$ if one does. So step i fails to produce a member of $\mathcal{P}$ with probability at most $1/(4\tilde{n})$; there are fewer than $\tilde{n}$ steps, so by the union bound a solution is found with probability at least $3/4$. Each step runs at most $\tilde{n}\lceil \log(4\tilde{n}^2) \rceil$ tests, each in time $3^{k'} 2^{|\hat{X}_{i+1}|} \tilde{n}^{O(1)} \le 3^{k'} 2^{|\hat{X}|} \tilde{n}^{O(1)}$ and polynomial space by Lemma D.9.6; only the current A_i is stored between steps. So the forced problem is solved in time $3^{k'} 2^{|\hat{X}|} \tilde{n}^{O(1)}$ and polynomial space.

The algorithm $\mathcal{A}$. Let $\mathcal{A}$ be the algorithm of Lemma D.9.7 with $w \equiv 0$, $\theta = 0$ and, as Forced, the procedure above, which ignores $\tilde{w}$ and θ'; every call then has $\tilde{w} \equiv 0$ and $\theta' = 0$, so every output satisfies $\tilde{w}(A) \le \theta'$, and the hypotheses of Lemma D.9.7 hold with $w_{\max} = 0$ and $p = 3/4$. The lemma gives the time, space and correctness claims. The running-time exponent (22) is nonnegative and computable in polynomial time; comparisons of $g(G, X, k)$ with numbers of the form $i \log 3 + j$ with integers i, j, which are the only ones the algorithms below make, reduce to integer comparisons. □

D.9.5 Tightness of the base 1.8

This subsection proves Propositions 11.9 and 11.10. The proof of Proposition 11.9 uses the following evaluation of the infimum over y of the function $\phi_{3,\sigma}$ of (21).

Lemma D.9.8 (The critical fraction for $c = 3$)**.** *For $\sigma^* = 26/45$, $\inf_{y \ge 0} \phi_{3,\sigma^*}(y) = \log 1.8$, attained at $y = 4/5$.*

Proof. The map $y \mapsto u = \lambda_{\max}(M_{3y/2})$ is an increasing bijection from $[0, \infty)$ onto $[1, \infty)$. Its inverse is $\zeta = 3y/2 = (u^2 - u)/(2u - 1)$, because u is a root of $x^2 - (1 + 2\zeta)x + \zeta$ (and the larger one, as the

other root $\zeta/u = (u-1)/(2u-1)$ is less than 1). Substituting, $1+y = (2u^2+4u-3)/(3(2u-1))$, and $\phi_{3,\sigma^*}(y)\ln 2 = \hat{\phi}(u)$ with

$$\hat{\phi}(u) = \ln(2u^2+4u-3) - \ln(2u-1) - \ln 3 - \sigma^* \ln \frac{u}{3} \qquad (u \geq 1).$$

Differentiating and bringing the three terms to a common denominator gives

$$u\,\hat{\phi}'(u) = \frac{4u^2+4u}{2u^2+4u-3} - \frac{2u}{2u-1} - \frac{26}{45} = \frac{(u-3)(76u^2-108u+26)}{45\,(2u^2+4u-3)(2u-1)}.$$

The denominator is positive for $u \geq 1$. The quadratic $76u^2-108u+26$ is negative at $u=1$ and has exactly one root $u_1 = (54+\sqrt{940})/76 \in (1,3)$ in $[1,\infty)$. So $\hat{\phi}$ increases on $[1,u_1]$, decreases on $[u_1,3]$ and increases on $[3,\infty)$, and its minimum on $[1,\infty)$ is $\min(\hat{\phi}(1),\hat{\phi}(3))$. Now $\hat{\phi}(3) = \ln(27/15) = \ln 1.8$, which is the value at $y = 4/5$, and $\hat{\phi}(1) = \sigma^* \ln 3$. Finally $\sigma^* \ln 3 > \ln 1.8$ is equivalent to $5^{45} > 3^{64}$, which holds: $5^{45} = 125^{15} > (\frac{3}{2}\cdot 81)^{15} = (\frac{3}{2})^{15}3^{60}$, and $(\frac{3}{2})^{15} > (\frac{3}{2})^{12} = (\frac{81}{16})^3 > 5^3 > 3^4$. □

Proposition 11.9 (Tightness of 1.8)**.** *Let I be an instance and $S \in \mathcal{F}_I$ with $|S|/n = 26/45$. Suppose that $g(I,X,k)$ is nondecreasing in k for every $X \subseteq S$, and that, for a path F with vertex set S and a constant $\gamma' \geq 0$,*

$$g(I,X,|S|) \;\geq\; (|S|-|X|)\log 3 + |X| - e(F[X]) - \gamma' \qquad \text{for every } X \subseteq S.$$

(a) *The exponential moments of S (Definition 3.3, with $k = |S|$) satisfy*

$$\max_{0\leq t\leq |S|} \frac{\mu_t(S)}{\binom{n}{t}} \leq 2^{\gamma'+1}(n+1)(5/9)^n.$$

(b) *For every integer k with $|S| \leq k \leq n$, a run of Algorithm* MLS-$\theta(\psi,\Gamma,\kappa)$ *on (I,k) runs $\mathcal{A}$ on a subset of S with probability at most $P_{\text{nec}} \cdot 2^{\gamma'+1}(n+1)\big(2^{\psi}\cdot 5/9\big)^n$, with $P_{\text{nec}} \leq N^{O(1)}$ as in Proposition 3.10.*

(c) *Suppose that $\mathcal{A}$ never accepts a* NO*-instance and that S is the only set in $\mathcal{F}_I$ of size at most $|S|$. Then a run on (I,k) with $k \leq |S|$ answers* YES*, and a run on (I,k) with any $k \leq n$ returns a solution of size at most $|S|$, only if it runs $\mathcal{A}$ on a subset of S.*

(d) *Under the hypotheses of (c), let $\varepsilon > 0$ and $\psi \leq \log 1.8 - \varepsilon$, and run Algorithm* MLS-$\theta(\psi,\Gamma,\kappa)$ *on (I,k) for $k = 1,\ldots,n$, with any Γ and κ as in Theorem 3.5 (for a one-sided-error $\mathcal{A}$, with 2Γ as in Corollary 3.15). The probability that some run answers* YES *at a $k \leq |S|$, or that some run returns a set of size at most $|S|$, is at most $nP_{\text{nec}}2^{\gamma'+1}(n+1)2^{-\varepsilon n}$, where $P_{\text{nec}} \leq N^{O(1)}$ is the largest of the factors of Proposition 3.10 over these runs.*

Proof. (a) By the assumed lower bound on g, $W_S(y) \leq 2^{\gamma'}3^{-|S|}\mathcal{Z}_F(3y/2)$ for $y \geq 0$, by the computation in the proof of Corollary 11.5(a) with the inequality reversed. By Lemma 11.3 with $\zeta = 3y/2$, $3^{-|S|}\mathcal{Z}_F(\zeta) \leq 2(\lambda_{\max}(M_\zeta)/3)^{|S|}$, so, with $\sigma^* = 26/45$, $W_S(y)/(1+y)^n \leq 2^{\gamma'+1}2^{-n\phi_{3,\sigma^*}(y)} \leq 2^{\gamma'+1}(5/9)^n$ by Lemma D.9.8. By Lemma 3.9, $\max_t \mu_t(S)/\binom{n}{t} \leq (n+1)\sup_{y\geq 0} W_S(y)/(1+y)^n$, which proves (a).

(b) Let $|S| \leq k \leq n$. By Proposition 3.10 for S and k, the expected number of trials of the run on (I,k) that run $\mathcal{A}$ on a subset of S is at most $P_{\text{nec}}\cdot 2^{\psi n}\max_{t\leq|S|}\mu_t^{(k)}(S)/\binom{n}{t}$. As g is nondecreasing in its budget, (4) gives $\mu_t^{(k)}(S) \leq \mu_t(S)$, and (a) bounds the expectation by $P_{\text{nec}}\cdot 2^{\gamma'+1}(n+1)(2^{\psi}\cdot 5/9)^n$. By Markov's inequality, this also bounds the probability that at least one such trial occurs.

(c) A run answers YES, or returns a set, only after a trial with some sample X of size t in which $\mathcal{A}$ accepts (I, X, k). Since $\mathcal{A}$ never accepts a NO-instance, some $Y \subseteq U_I \setminus X$ with $|Y| \le k-t$ satisfies $X \cup Y \in \mathcal{F}_I$, and a returned set is such a set $X \cup Y$. If $k \le |S|$, then $|X \cup Y| \le k \le |S|$; if the run returns a set of size at most $|S|$, then this set is an $X \cup Y$ of size at most $|S|$. In both cases $X \cup Y$ is a set in $\mathcal{F}_I$ of size at most $|S|$, so $X \cup Y = S$ and $X \subseteq S$.

(d) A run with $k < |S|$ never answers YES, by item (ii) of Theorem 3.5, since S is the only set in $\mathcal{F}_I$ of size at most $|S|$ and $|S| > k$. By (c), a YES at $k = |S|$ or a returned set of size at most $|S|$ requires a run that runs $\mathcal{A}$ on a subset of S, and this run has $k \ge |S|$, because a returned set $X \cup Y$ has size at most $|X| + (k - |X|) = k$ and equals S by the proof of (c). By (b) and the union bound over the at most n runs with $k \ge |S|$, the probability is at most $nP_{\mathrm{nec}}2^{\gamma'+1}(n+1)(2^{\psi} \cdot 5/9)^n$, and $2^{\psi} \cdot 5/9 \le 2^{-\varepsilon} \cdot 1.8 \cdot 5/9 = 2^{-\varepsilon}$. □

Proposition 11.10 (A tight family for Connected Feedback Vertex Set)**.** *Let $\varepsilon > 0$ be a constant. There is an infinite family of graphs on which the algorithm of the proof of Theorem 11.8, run with rates $\psi \le \log 1.8 - \varepsilon$ in place of $\log 1.8$ and with any Γ and κ as in Theorem 3.5, outputs a minimum connected feedback vertex set with probability $o(1)$ as the number n of vertices grows.*

Proof. Let $L \ge 2$ and let $T \subseteq \{0, \dots, L-1\}$ contain 0 and $L-1$. Let $G_{L,T}$ consist of a path $v_0 v_1 \dots v_L$ and, for each $i \in T$, a vertex w_i adjacent to v_i and v_{i+1}, and put $S = \{v_1, \dots, v_{L-1}\}$. We show that S is the unique minimum connected feedback vertex set of $G_{L,T}$, that $G_{L,T}[S]$ is a path, and that $g(G_{L,T}, X, |S|) = (|S| - |X|)\log 3 + |X| - e(G_{L,T}[X])$ for every $X \subseteq S$. The family consists of the graphs with $L = 26j + 1$ and $|T| = 19j - 2$, $j \ge 1$.

Write $G = G_{L,T}$. The set S is a connected feedback vertex set: $G[S]$ is the path $v_1 \dots v_{L-1}$, and $G - S$ consists of the edges $v_0 w_0$ and $v_L w_{L-1}$ and isolated vertices. Let A be any connected feedback vertex set. It meets the triangle $v_0 v_1 w_0$ in a vertex x and the triangle $v_{L-1} v_L w_{L-1}$ in a vertex z. For $1 \le h \le L-1$, deleting v_h separates $\{v_0, \dots, v_{h-1}\} \cup \{w_i : i < h\}$ from $\{v_{h+1}, \dots, v_L\} \cup \{w_i : i \ge h\}$. Either $v_h \in \{x, z\}$, or x lies in the first part and z in the second; in the latter case $v_h \in A$ because $G[A]$ is connected. Hence $S \subseteq A$, so S is the unique minimum solution. For $X \subseteq S$, $G[X]$ is a subgraph of a path, so $\mathrm{comp}(G[X]) = |X| - e(G[X])$, and the formula for g is (22). With $L = 26j+1$ and $|T| = 19j - 2$ (possible, since $2 \le 19j - 2 \le L$) we get $|S| = 26j$ and $n = L + 1 + |T| = 45j$, so $|S|/n = 26/45$.

We check the hypotheses of Proposition 11.9 for $I = G$, the path $F = G[S]$ and $\gamma' = 0$. The running-time exponent $g(G, X, k) = (k - |X|)\log 3 + \mathrm{comp}(G[X])$ is nondecreasing in k, and the formula above gives the lower bound on $g(G, X, |S|)$ with equality. By Lemma 11.7, the extension algorithm never accepts a NO-instance and returns sets Y such that $X \cup Y$ is a connected feedback vertex set. Every connected feedback vertex set contains S, so S is the only one of size at most $|S|$. The algorithm of Theorem 11.8 outputs S only if some run returns a set of size at most $|S|$ (its default output $V(G)$ has size $n > |S|$), so by Proposition 11.9(d) with $\gamma' = 0$ this has probability at most $2nP_{\mathrm{nec}}(n+1)2^{-\varepsilon n}$, which tends to 0 as $j \to \infty$, because the input length N is polynomial in n for the graphs $G_{L,T}$. □

D.10 Weighted finishing steps

This subsection proves Theorems 11.16 and 11.17. In both cases the sampling analysis is that of Section 11.5, and only the extension algorithm is new. Both extension algorithms turn a randomized decision procedure into one that outputs a witness, by the following self-reduction, which is also the one used in the proof of Lemma 11.7.

Lemma D.10.1 (Witness by self-reduction)**.** *Let Ω be a set with $m \geq 1$ elements, let $\mathcal{P}$ be a nonempty family of subsets of Ω, let Z_0 be a subset of some member of $\mathcal{P}$, let $0 < \delta < 1$ and let $\rho \geq \log(m/\delta)$ be an integer. Let T be a randomized test that, on input $Z \subseteq \Omega$, never accepts if no member of $\mathcal{P}$ contains Z, and accepts with probability at least $1/2$ otherwise, with independent coins in every run. Starting from $Z = Z_0$, go through the elements $v \in \Omega \setminus Z_0$ in a fixed order and add v to Z if one of ρ independent runs of T on $Z \cup \{v\}$ accepts. Then T is run at most $m\rho$ times, and the final set Z is a member of $\mathcal{P}$ with probability at least $1 - \delta$.*

Proof. Throughout, Z is contained in a member of $\mathcal{P}$: this holds for Z_0, and v is added only after an accepting run on $Z \cup \{v\}$, which by the soundness of T is contained in a member. For $v \in \Omega \setminus Z_0$ let E_v be the event that, when v is processed with the current set Z, the set $Z \cup \{v\}$ is contained in a member of $\mathcal{P}$ and all ρ runs reject. Conditioned on the history before v is processed, each run then rejects with probability at most $1/2$, independently, so $\Pr[E_v] \leq 2^{-\rho} \leq \delta/m$, and by the union bound some E_v occurs with probability at most δ. Suppose that no E_v occurs, and let $P \in \mathcal{P}$ contain the final set Z. If some $v \in P \setminus Z$ existed, then $v \notin Z_0$, and when v was processed the current set was contained in Z, so the tested set was contained in P and all runs rejected it, that is, E_v occurred. Hence $Z = P$. There are at most m elements to process, each with ρ runs. □

D.10.1 Maximum-Weight Connected k-Subgraph

Theorem 11.16 (Maximum-Weight Connected k-Subgraph)**.** *Let $1 \leq w_{\max} \leq n^{O(1)}$. There is a randomized algorithm that, given a graph G with n vertices, weights $w \colon V(G) \to \{-w_{\max}, \ldots, w_{\max}\}$ and an integer $1 \leq k \leq n$, either reports correctly that G has no connected k-set or outputs a connected k-set, in time $O(1.6444^n)$ and polynomial space. The output has maximum weight with probability at least $1 - 1/e$. In particular, the decision version ("is there a connected k-set of weight at least θ?") is solved with one-sided error.*

The finishing step is the algebraic sieve of Björklund, Kaski and Kowalik [BKK16] for Graph Motif, with one more indeterminate that records the weight, as they do for the edit costs of Closest Graph Motif. Their cost-constrained lemma [BKK16, Lemma 3] lets the coloring vary, whereas we need the coloring fixed and a cost attached to each vertex; so we derive the variant we need from the proofs of their Lemmas 1 and 2. As in the proof of Lemma 11.11, the components of the sample are contracted and given fresh colors, which the motif demands exactly once.

Notation from [BKK16]. Let H be a graph with vertex set $\{1, \ldots, n_H\}$, let $D \subseteq V(H)$, let $w' \colon V(H) \to \{-w_{\max}, \ldots, w_{\max}\}$ and let p be an integer with $\max(1, |D|) \leq p \leq n_H$. A *branching walk* in H is a pair (T, φ) of an ordered rooted tree T, whose vertices are numbered in preorder, and a homomorphism $\varphi \colon V(T) \to V(H)$; it is *simple* if φ is injective, *properly ordered* if siblings $a < b$ satisfy $\varphi(a) < \varphi(b)$, it has *size* $|V(T)|$, and it *spans* $\varphi(V(T))$. With a variable x_u for every vertex u and variables $y_{(u,u')}, y_{(u',u)}$ for every edge uu', the *fingerprint* of a walk is $\prod_{ab \in E(T),\, a<b} y_{(\varphi(a),\varphi(b))} x_{\varphi(b)}$, and

$$P_p(\vec{x}, \vec{y}) = \sum_{s \in V(H)} x_s \sum_{(T,\varphi)} \prod_{ab \in E(T),\, a<b} y_{(\varphi(a),\varphi(b))} x_{\varphi(b)},$$

the inner sum over the properly ordered branching walks of size p that start at s ([BKK16, Section 3.2]). Every monomial of P_p has degree p in $\vec{x}$ and degree $p - 1$ in $\vec{y}$. By [BKK16, Lemma 7], a monomial of P_p is multilinear in $\vec{x}$ if and only if it comes from a simple walk, and a simple walk can be reconstructed from its monomial; so the monomials of P_p that are multilinear in $\vec{x}$ are in

bijection with the simple properly ordered branching walks of size p, each has coefficient 1, and its $\vec{x}$-part is $\prod_{u\in K} x_u$ for the set K that the walk spans, which induces a connected subgraph. Conversely, every set K with $|K| = p$ and $H[K]$ connected is spanned by such a walk: take a spanning tree of $H[K]$, root it, order siblings by their vertex numbers and number the vertices in preorder.

Color the vertices of H as follows: each $u \in D$ gets its own color c_u with multiplicity $m(c_u) = 1$, and every other vertex gets the color $\star$ with multiplicity $m(\star) = p - |D|$. Give each color c a set Σ_c of $m(c)$ *shades*, these sets pairwise disjoint, put $\Sigma = \bigcup_c \Sigma_c$, so $|\Sigma| = p$, and let Λ be a set of p *labels*. A set $K \subseteq V(H)$ is *properly colored* if every color c occurs on at most $m(c)$ vertices of K. A set K with $|K| = p$ is properly colored if and only if $D \subseteq K$: if $D \subseteq K$, each c_u occurs once and $\star$ occurs $p - |D|$ times; conversely, at most $p - |D|$ vertices of K have color $\star$, so at least $|D|$ vertices of K lie in D, that is, $D \subseteq K$. The *cost* of a vertex is $\mathrm{cost}(u) = w_{\max} - w'(u) \in \{0, \dots, 2w_{\max}\}$, and $\mathrm{cost}(K) = \sum_{u\in K} \mathrm{cost}(u) = p\, w_{\max} - w'(K)$ for $|K| = p$.

Work over the field $\mathbb{F} = \mathbb{F}_{2^b}$. Introduce variables $v_{u,d}$ for $u \in V(H)$ and $d \in \Sigma_{\mathrm{col}(u)}$, variables $\zeta_{d,j}$ for $d \in \Sigma$ and $j \in \Lambda$, and one more variable λ. Put $v_{u,j} = \sum_{d\in\Sigma_{\mathrm{col}(u)}} v_{u,d}\zeta_{d,j}$, $v_u^A = \sum_{j\in A} v_{u,j}$ for $A \subseteq \Lambda$, and

$$Q(\vec{v}, \vec{\zeta}, \vec{y}, \lambda) = \sum_{A\subseteq\Lambda} P_p\big((\lambda^{\mathrm{cost}(u)} v_u^A)_{u\in V(H)}, \vec{y}\big). \tag{35}$$

This is the sieve (2) of [BKK16] applied to $P_p(\lambda^{\mathrm{cost}(1)}x_1, \dots, \lambda^{\mathrm{cost}(n_H)}x_{n_H}, \vec{y})$. Write $Q = \sum_\sigma Q_\sigma(\vec{v}, \vec{\zeta}, \vec{y})\lambda^\sigma$.

Lemma D.10.2 (Weighted sieve identity)**.** *Let H, D, w', p and the coloring be as above, and let $\sigma \geq 0$ be an integer. The polynomial Q_σ is nonzero if and only if some set $K \subseteq V(H)$ with $|K| = p$, $D \subseteq K$ and $H[K]$ connected has $\mathrm{cost}(K) = \sigma$, that is, $w'(K) = pw_{\max} - \sigma$. Moreover Q_σ has total degree at most $3p - 1$, and $Q_\sigma = 0$ for $\sigma > 2pw_{\max}$.*

Proof. The idea is that the sieve of [BKK16] acts on each monomial of P_p separately, and the substitution $x_u \mapsto \lambda^{\mathrm{cost}(u)}x_u$ only multiplies each monomial by a power of λ determined by its $\vec{x}$-part. Write $P_p = \sum_m a_m \vec{x}^{\,\delta_m}\vec{y}^{\,\epsilon_m}$ with distinct exponent pairs (δ_m, ϵ_m) and nonzero coefficients a_m. Then

$$Q = \sum_m a_m\, \lambda^{\sum_u \mathrm{cost}(u)\delta_{m,u}}\, \vec{y}^{\,\epsilon_m} \sum_{A\subseteq\Lambda} \prod_u (v_u^A)^{\delta_{m,u}}.$$

Fix m and write $\delta = \delta_m$; then $\sum_u \delta_u = p = |\Lambda|$. The proofs of Lemmas 1 and 2 of [BKK16] (equations (5)–(8)) show, in characteristic 2, that $\sum_A \prod_u (v_u^A)^{\delta_u} = 0$ unless $\delta \in \{0,1\}^{V(H)}$; that argument uses only $v_u^A = \sum_{j\in A} v_{u,j}$ and $\sum_u \delta_u = |\Lambda|$. If δ is the indicator vector of a set K, then by (8) and (9) of [BKK16] and the paragraph after (9),

$$\sum_{A\subseteq\Lambda} \prod_{u\in K} v_u^A = R_K := \sum_h \sum_g \prod_{u\in K} v_{u,h(u)}\zeta_{h(u),g(u)},$$

where h ranges over the injective maps $K \to \Sigma$ with $h(u) \in \Sigma_{\mathrm{col}(u)}$ and g over the bijections $K \to \Lambda$; the triple (K, h, g) can be read off from the monomial $\prod_{u\in K} v_{u,h(u)}\zeta_{h(u),g(u)}$, so these monomials are distinct and $R_K \neq 0$ if and only if such an h exists, that is, if and only if K is properly colored.

By the facts quoted from [BKK16, Lemma 7], the monomials of P_p that are multilinear in $\vec{x}$ are $\prod_{u\in K_W} x_u\, \vec{y}^{\,\epsilon_W}$ with coefficient 1, one for each simple properly ordered branching walk W of size p, where K_W is the set that W spans and $\vec{y}^{\,\epsilon_W}$ the $\vec{y}$-part of its fingerprint. Hence

$$Q_\sigma = \sum_{W\,:\,\mathrm{cost}(K_W)=\sigma,\ K_W \text{ properly colored}} \vec{y}^{\,\epsilon_W}\, R_{K_W}.$$

The monomials $\vec{y}^{\epsilon_W} \prod_{u \in K_W} v_{u,h(u)} \zeta_{h(u),g(u)}$ over all such W and all (h, g) are pairwise distinct: the $\vec{v}$-variables determine K_W and h, the $\vec{\zeta}$-variables then determine g, and K_W together with $\vec{y}^{\epsilon_W}$ is the monomial of W in P_p, which determines W. So no cancellation occurs, and $Q_\sigma \neq 0$ if and only if some simple walk of size p spans a properly colored set K with $\mathrm{cost}(K) = \sigma$. By the facts quoted above and the characterization of properly colored sets, this holds if and only if some K with $|K| = p$, $D \subseteq K$ and $H[K]$ connected has $\mathrm{cost}(K) = \sigma$. Each monomial of Q_σ has degree $p - 1$ in $\vec{y}$ and $2p$ in $(\vec{v}, \vec{\zeta})$, and $\mathrm{cost}(K) \leq 2pw_{\max}$. □

Algorithm D.10.3 (Weighted sieve)

Input: H, D, w', p as above and an integer θ'.

If $p > n_H$ or $\theta' > pw_{\max}$, reject. Let $b = \lceil \log(6pw_{\max}) \rceil$, and find an irreducible polynomial of degree b over $\mathbb{F}_2$ by trying all 2^b monic polynomials of degree b and testing each by division by all polynomials of degree between 1 and $\lfloor b/2 \rfloor$; it defines $\mathbb{F} = \mathbb{F}_{2^b}$. Draw values for all variables $\vec{v}, \vec{\zeta}, \vec{y}$ independently and uniformly from $\mathbb{F}$. For each of $D_\lambda + 1$ distinct nonzero elements $\lambda_0 \in \mathbb{F}$, where $D_\lambda = 2pw_{\max}$, compute $Q(\vec{v}, \vec{\zeta}, \vec{y}, \lambda_0)$ by evaluating, for every $A \subseteq \Lambda$, the polynomial P_p at the point $x_u = \lambda_0^{\mathrm{cost}(u)} v_u^A$ with the recurrence (11)–(13) of [BKK16], and summing over A. Recover the coefficients $Q_\sigma(\vec{v}, \vec{\zeta}, \vec{y})$, $0 \leq \sigma \leq D_\lambda$, of the univariate polynomial $\lambda \mapsto Q(\vec{v}, \vec{\zeta}, \vec{y}, \lambda)$ by Lagrange interpolation. Accept if $Q_\sigma(\vec{v}, \vec{\zeta}, \vec{y}) \neq 0$ for some $\sigma \leq pw_{\max} - \theta'$; otherwise reject.

Lemma D.10.4 (Weighted sieve, guarantees)**.** *Let H have n_H vertices. Algorithm D.10.3 runs in polynomial space and, for every outcome of its coins, in time $2^p (n_H w_{\max})^{O(1)}$. It rejects if no $K \subseteq V(H)$ with $|K| = p$, $D \subseteq K$ and $H[K]$ connected has $w'(K) \geq \theta'$, and if such a K exists, it accepts with probability at least $1/2$.*

Proof. Since $w_{\max} \geq 1$, we have $2^b \geq 6pw_{\max} \geq \max(6p, D_\lambda + 2)$, so the field has $D_\lambda + 1$ nonzero distinct elements. By Lemma D.10.2, $\lambda \mapsto Q(\vec{v}, \vec{\zeta}, \vec{y}, \lambda)$ is a polynomial of degree at most D_λ whose coefficients are the values $Q_\sigma(\vec{v}, \vec{\zeta}, \vec{y})$, so the interpolation recovers them. If the algorithm accepts, some Q_σ with $\sigma \leq pw_{\max} - \theta'$ is a nonzero polynomial, and Lemma D.10.2 gives a set K as in the lemma with $w'(K) = pw_{\max} - \sigma \geq \theta'$. Conversely, let K^* be such a set and $\sigma^* = \mathrm{cost}(K^*) = pw_{\max} - w'(K^*) \leq pw_{\max} - \theta'$. By Lemma D.10.2, Q_{σ^*} is a nonzero polynomial of total degree at most $3p - 1$, so by the lemma of DeMillo, Lipton, Schwartz and Zippel in the form of [BKK16, Lemma 8] its value at a uniformly random point is zero with probability at most $(3p-1)/2^b < 1/2$.

The search for the irreducible polynomial takes time $2^{3b/2} b^{O(1)} = (pw_{\max})^{O(1)}$, as $2^b < 12pw_{\max}$, and every field operation takes time $b^{O(1)}$. For fixed λ_0 and A, the point $(\lambda_0^{\mathrm{cost}(u)} v_u^A)_u$ is computed with $O(p^2 n_H + n_H w_{\max})$ field operations from the values $v_{u,j}$, which are computed once, and P_p is evaluated with $O(p^2 |E(H)|)$ field operations by the recurrence (11)–(13) ([BKK16, Section 3.5]). There are $(D_\lambda + 1) 2^p$ such pairs, and the interpolation takes $D_\lambda^{O(1)}$ field operations. The algorithm stores the $D_\lambda + 1$ sums and the tables of one evaluation of the recurrence, which is polynomial space. □

The sieve only decides. A set is found by deleting vertices, as follows.

Algorithm D.10.5 (Weighted finishing with a witness)

Input: H, D, w', p, θ' as in Algorithm D.10.3.

Put $\rho = \lceil \log(4n_H) \rceil$ and $H_{\mathrm{cur}} = H$. For every $u \in V(H) \setminus D$ in a fixed order, run Algorithm D.10.3 on $(H_{\mathrm{cur}} - u, D, w', p, \theta')$ independently ρ times, and if one run accepts, replace

H_{cur} by $H_{\mathrm{cur}} - u$. Afterward, if $K = V(H_{\mathrm{cur}})$ satisfies $|K| = p$, $H[K]$ connected and $w'(K) \geq \theta'$, output K; otherwise reject.

Lemma D.10.6 (Weighted finishing step)**.** *Algorithm D.10.5 runs in polynomial space and, for every outcome of its coins, in time $2^p (n_H w_{\max})^{O(1)}$. Every output K satisfies $|K| = p$, $D \subseteq K$, $H[K]$ connected and $w'(K) \geq \theta'$. If such a set exists, the algorithm outputs one with probability at least $3/4$.*

Proof. The output is checked explicitly. Suppose that a set as in the lemma exists. If $V(H) = D$, then this set is $V(H)$, the loop is empty, and the final check outputs it. Otherwise, with Z the set of deleted vertices, so that $H_{\mathrm{cur}} = H - Z$, the algorithm is the self-reduction of Lemma D.10.1 with $\Omega = V(H) \setminus D$, $Z_0 = \emptyset$, the family $\mathcal{P}$ of the sets $V(H) \setminus K$ over the sets K with $|K| = p$, $D \subseteq K$, $H[K]$ connected and $w'(K) \geq \theta'$, the test $\mathsf{T}(Z)$ given by Algorithm D.10.3 on $(H - Z, D, w', p, \theta')$, $\delta = 1/4$ and $\rho = \lceil \log(4 n_H) \rceil \geq \log(|\Omega|/\delta)$. The family is nonempty by assumption, and Z_0 lies in every member. By Lemma D.10.4, the test rejects Z if $H - Z$ contains no such set K, that is, if Z lies in no member of $\mathcal{P}$, and it accepts with probability at least $1/2$ if this holds, since then $K \subseteq V(H - Z)$ and $(H - Z)[K] = H[K]$. So, with probability at least $3/4$, the final Z is $V(H) \setminus K$ for such a K, and K is output. There are at most $n_H \rho$ runs of Algorithm D.10.3, each within the bounds of Lemma D.10.4. □

The extension problem. For an instance $I = (G, w, k)$ and an integer θ, let Φ_θ be the implicit set system with universe $V(G)$ whose solutions are the connected k-sets S with $w(S) \geq \theta$. Membership is decided in polynomial time, and every solution has size k. For a nonempty $X \subseteq V(G)$, let $X_1, \ldots, X_q$ be the vertex sets of the components of $G[X]$, and let G_X be obtained from G by contracting each X_i to a new vertex d_i, as in Appendix D.9.2; here G need not be connected, and $\hat{X} = \{d_1, \ldots, d_q\}$ is an independent set of G_X. For $X = \emptyset$ put $G_X = G$ and $\hat{X} = \emptyset$. Define $w_X \colon V(G_X) \to \{-w_{\max}, \ldots, w_{\max}\}$ by $w_X = w$ on $V(G) \setminus X$ and $w_X(d_i) = 0$.

For $Y \subseteq V(G) \setminus X$, the graph $G_X[\hat{X} \cup Y]$ is obtained from $G[X \cup Y]$ by contracting the connected sets $X_1, \ldots, X_q$: the vertex d_i is adjacent to $y \in Y$ exactly when y has a neighbor in X_i, edges inside Y are unchanged, and there are no edges between different X_i. So, by Lemma D.9.3(1), $G[X \cup Y]$ is connected if and only if $G_X[\hat{X} \cup Y]$ is. Moreover $|\hat{X} \cup Y| = |Y| + q$ and $w_X(\hat{X} \cup Y) = w(X \cup Y) - w(X)$. Every set $K \subseteq V(G_X)$ with $\hat{X} \subseteq K$ is $\hat{X} \cup Y$ for $Y = K \setminus \hat{X} \subseteq V(G) \setminus X$. Consequently, for every integer $\ell \geq 0$, some $Y \subseteq V(G) \setminus X$ with $|Y| \leq \ell - |X|$ makes $X \cup Y$ a solution of Φ_θ if and only if $|X| \leq k \leq \ell$ and G_X has a set $K \supseteq \hat{X}$ with $|K| = k - |X| + q$, $G_X[K]$ connected and $w_X(K) \geq \theta - w(X)$, and then $Y = K \setminus \hat{X}$ is such a completion.

The extension algorithm $\mathcal{A}_{\mathrm{mw}}$, given (I, X, ℓ) for Φ_θ, where the size bound ℓ of the extension problem is distinct from the size k in the instance, rejects if $|X| > k$ or $\ell < k$; otherwise it computes G_X, $\hat{X}$ and w_X, runs Algorithm D.10.5 on $(G_X, \hat{X}, w_X, k - |X| + q, \theta - w(X))$, where $q = \mathrm{comp}(G[X])$, and if this outputs K, accepts with $Y = K \setminus \hat{X}$. (The input of Algorithm D.10.5 satisfies $p = k - |X| + q \geq \max(1, q)$ because $k \geq 1$.) Its running-time exponent is

$$g_{\mathrm{mw}}(I, X, \ell) = \begin{cases} (k - |X|) + \mathrm{comp}(G[X]) & \text{if } |X| \leq k \leq \ell, \\ 0 & \text{otherwise.} \end{cases} \tag{36}$$

Lemma D.10.7 (Extension for Maximum-Weight Connected k-Subgraph)**.** *Let $1 \leq w_{\max} \leq n^{O(1)}$. The algorithm $\mathcal{A}_{\mathrm{mw}}$ solves Φ_θ-Extension with one-sided error: if it accepts, $X \cup Y$ is a connected k-set of weight at least θ with $Y \subseteq V(G) \setminus X$ and $|Y| \leq \ell - |X|$, and on a* yes*-instance it accepts with probability at least $3/4$. It runs in polynomial space and, for every outcome of its coins, in time $2^{g_{\mathrm{mw}}(I,X,\ell)} n^{O(1)}$.*

Proof. Correctness follows from the equivalence before (36) and Lemma D.10.6. If the algorithm runs Algorithm D.10.5, then $p = k - |X| + \mathrm{comp}(G[X]) = g_{\mathrm{mw}}(I, X, \ell)$ and G_X has at most n vertices, so Lemma D.10.6 gives time $2^{g_{\mathrm{mw}}}(nw_{\max})^{O(1)} = 2^{g_{\mathrm{mw}}}n^{O(1)}$; otherwise it takes polynomial time. The running-time exponent is an integer computable in polynomial time. □

Proof of Theorem 11.16. A connected k-set exists if and only if some component of G has at least k vertices; then the first k vertices of a breadth-first search order of that component form one. If none exists, report this. Otherwise, for every $\theta \in \{-kw_{\max}, \ldots, kw_{\max}\}$, run Algorithm MLS-$\theta$ for Φ_θ and the size k as in Corollary 11.6, with the extension algorithm $\mathcal{A}_{\mathrm{mw}}$, $c = 2$, $\gamma = 0$, the rate $\psi = 0.71755$ of Lemma 11.12, $\Gamma_0 = 16$ and $\kappa = 1$. Output the heaviest set returned by these runs, and, if no run returns a set, the connected k-set found by breadth-first search.

By Lemma D.10.7, $\mathcal{A}_{\mathrm{mw}}$ satisfies the hypotheses of Corollary 3.15, with running-time exponent g_{mw}. It has a component-count bound with constants $(2, 0)$: for a solution S with $|S| = k$, let F be a spanning tree of $G[S]$; for $X \subseteq S$ and the size bound $\ell = k$ we have $|X| \le k \le \ell$, so $g_{\mathrm{mw}}(I, X, k) = (k - |X|) + \mathrm{comp}(G[X]) \le (k - |X|)\log 2 + |X| - e(F[X])$, because $F[X]$ is a spanning forest of a subgraph of $G[X]$. By Lemma 11.12, every $\sigma \in (0, 1]$ satisfies one of (R1)–(R3) with these constants. So, by Corollary 11.6, every run takes time $2^{0.71755n}n^{O(1)}$ and polynomial space, every returned set is a solution of Φ_θ, hence a connected k-set, and the run for θ equal to the maximum weight returns a set with probability at least $1 - 1/e$; that set has maximum weight. There are $2kw_{\max} + 1 \le n^{O(1)}$ runs, so the total time is $2^{0.71755n}n^{O(1)}$, which is $O(1.6444^n)$ because $0.71755 < \log 1.6444$ (Lemma 11.12). □

D.10.2 Vertex-weighted Connected Feedback Vertex Set

Theorem 11.17 (Vertex-weighted Connected Feedback Vertex Set)**.** *Let $w_{\max} \le n^{O(1)}$. There is a randomized algorithm that, given a graph G with n vertices and weights $w\colon V(G) \to \{1, \ldots, w_{\max}\}$, reports correctly that G has no connected feedback vertex set or outputs one, in time $O^*(1.8^n)$ and polynomial space. The output has minimum weight with probability at least $1 - 1/e$. In particular, the decision version ("is there a connected feedback vertex set of weight at most θ?") is solved with one-sided error.*

The weighted extension algorithm combines two ingredients. The unweighted extension algorithm (Lemma 11.7) provides a connected feedback vertex set of small excess, which serves as a structured modulator; it is only needed as a modulator, so the intermediate graphs of its iterative compression never need weights. The weighted test then runs the counting of (CC1) and (CC3) with a weight function that combines the isolating weights with the vertex weights.

The counted objects. We use the families of the proof of Theorem A.13 of [CNP+11], with a general weight function, and we prove the two properties of them that the weighted test needs, so that this subsection does not depend on their dynamic program. Let $\tilde{G}$ be a graph with $\tilde{n}$ vertices, $Y_{\mathrm{in}} \subseteq V(\tilde{G})$ nonempty and $r \in Y_{\mathrm{in}}$, and let $\tilde{U} = V(\tilde{G}) \times \{\mathrm{F}, \mathrm{M}\}$. For a weight function $\mathrm{wt}\colon \tilde{U} \to \{1, \ldots, N\}$ and integers a, e, c, W, let $\mathcal{R}_W^{a,e,c}$ be the family of pairs (Z, M) with $Z \subseteq V(\tilde{G}) \setminus Y_{\mathrm{in}}$, $|Z| = a$, $|E(\tilde{G}[Z])| = e$, $M \subseteq Z$, $|M| = c$ and $\mathrm{wt}(Z \times \{\mathrm{F}\}) + \mathrm{wt}(M \times \{\mathrm{M}\}) = W$. Let $\mathcal{S}_W^{a,e,c}$ consist of the pairs in $\mathcal{R}_W^{a,e,c}$ such that $\tilde{G}[Z]$ is a forest with a vertex of M in each of its components and $\tilde{G} - Z$ is connected. Let $\mathcal{C}_W^{a,e,c}$ consist of the triples of a pair $(Z, M) \in \mathcal{R}_W^{a,e,c}$, a consistent cut (Z_1, Z_2) of $\tilde{G}[Z]$ with $M \subseteq Z_1$, and a consistent cut (Y_1, Y_2) of $\tilde{G} - Z$ with $r \in Y_1$; a cut of a graph is *consistent* if no edge joins its two sides. (We write Z for the candidate forest, which [CNP+11] call X, and Y_{in} for their S.)

Lemma D.10.8 (Cut parity)**.** *For every weight function* wt *and all integers* a, e, c, W *with* $c \le a-e$, $|\mathcal{S}_W^{a,e,c}| \equiv |\mathcal{C}_W^{a,e,c}| \pmod 2$.

Proof. Fix $(Z, M) \in \mathcal{R}_W^{a,e,c}$. A cut of $\tilde{G}[Z]$ is consistent exactly when it puts each component of $\tilde{G}[Z]$ on one side. The components that contain a vertex of M must lie on side 1, and each of the ℓ other components can lie on either side, so there are 2^ℓ cuts (Z_1, Z_2). Likewise there are $2^{\mathrm{comp}(\tilde{G}-Z)-1}$ cuts (Y_1, Y_2), since the component of $r \in Y_{\mathrm{in}} \subseteq V(\tilde{G}) \setminus Z$ must lie on side 1. So (Z, M) lies in $2^{\mathrm{comp}(\tilde{G}-Z)-1+\ell}$ triples, and this number is odd if and only if $\tilde{G}-Z$ is connected and every component of $\tilde{G}[Z]$ contains a vertex of M. In that case $\mathrm{comp}(\tilde{G}[Z]) \le |M| = c \le a - e = |Z| - |E(\tilde{G}[Z])|$. Every graph H satisfies $\mathrm{comp}(H) \ge |V(H)| - |E(H)|$, with equality if and only if H is a forest, so $\tilde{G}[Z]$ is a forest and $(Z, M) \in \mathcal{S}_W^{a,e,c}$. Conversely, every pair in $\mathcal{S}_W^{a,e,c}$ lies in exactly one triple. Summing over the pairs gives the claim. □

A triple in $\mathcal{C}_W^{a,e,c}$ is determined by its *state map* $\sigma\colon V(\tilde{G}) \to \Sigma$, $\Sigma = \{\mathsf{y}_1, \mathsf{y}_2, \mathsf{z}_1, \mathsf{z}_2\}$, which sends v to y_j if $v \in Y_j$ and to z_j if $v \in Z_j$, together with the marker set M. A map σ and a set M come from a triple, in the family with the indices computed from them, if and only if

(S1) $\sigma(v) \in \{\mathsf{y}_1, \mathsf{y}_2\}$ for $v \in Y_{\mathrm{in}}$, and $\sigma(r) = \mathsf{y}_1$;

(S2) no edge uv has $\{\sigma(u), \sigma(v)\} = \{\mathsf{y}_1, \mathsf{y}_2\}$ or $\{\sigma(u), \sigma(v)\} = \{\mathsf{z}_1, \mathsf{z}_2\}$;

(S3) $M \subseteq \sigma^{-1}(\mathsf{z}_1)$.

The indices are $a = |\sigma^{-1}(\{\mathsf{z}_1, \mathsf{z}_2\})|$, the number e of edges with both ends in $\sigma^{-1}(\{\mathsf{z}_1, \mathsf{z}_2\})$, $c = |M|$ and $W = \sum_{v \in \sigma^{-1}(\{\mathsf{z}_1, \mathsf{z}_2\})} \mathrm{wt}(v, \mathrm{F}) + \sum_{v \in M} \mathrm{wt}(v, \mathrm{M})$. These are the four states $0_1, 0_2, 1_1, 1_2$ of the dynamic program of [CNP+11] (see (CC2)). For $\mathrm{Mod} \subseteq V(\tilde{G})$ and a coloring $\varsigma\colon \mathrm{Mod} \to \Sigma$, the *core evaluation* is the family of numbers $|\mathcal{C}_W^{a,e,c}(\varsigma)| \bmod 2$, where $\mathcal{C}_W^{a,e,c}(\varsigma)$ consists of the triples in $\mathcal{C}_W^{a,e,c}$ whose state map agrees with ς on Mod. A coloring that violates (S1) or (S2) on Mod has empty families, so $|\mathcal{C}_W^{a,e,c}|$ is the sum of $|\mathcal{C}_W^{a,e,c}(\varsigma)|$ over the reasonable colorings ς of Mod (Appendix D.9.3).

Lemma D.10.9 (Weighted core evaluation)**.** *Let* $\mathrm{wt}\colon \tilde{U} \to \{1, \dots, N\}$*, let* $\mathrm{Mod} \subseteq V(\tilde{G})$ *be such that* $\tilde{G} - \mathrm{Mod}$ *is a forest, and let* $\varsigma\colon \mathrm{Mod} \to \Sigma$*. Then the numbers* $|\mathcal{C}_W^{a,e,c}(\varsigma)| \bmod 2$ *for all* $0 \le a, c \le \tilde{n}$*,* $0 \le e \le \tilde{n}^2$ *and* $0 \le W \le 2N\tilde{n}$ *(all other values are* 0*) can be computed in time and space polynomial in* $\tilde{n}$ *and* N*. The dynamic program has four states for each vertex of the forest, whatever the weight function;* wt *affects only the range of the index* W *and the values added to it.*

Proof. We compute the counts exactly and reduce them modulo 2. A *table* is an array $T(a, e, c, W)$ of integers indexed as in the lemma, and $T * T'$ denotes the convolution $(T * T')(a, e, c, W) = \sum T(a', e', c', W')\, T'(a-a', e-e', c-c', W-W')$, the sum over all index vectors with both arguments in range. For an index vector ι, $[\iota]$ is the table with entry 1 at ι and 0 elsewhere, and T^{+e} is T shifted by 1 in the index e.

Write $Z(\sigma) = \sigma^{-1}(\{\mathsf{z}_1, \mathsf{z}_2\})$. We assign each vertex and each edge of $\tilde{G}$ to one part of the computation. Let $F = \tilde{G} - \mathrm{Mod}$, root each tree of F at a vertex, and let T_v be the subtree of F rooted at v. Then every edge of $\tilde{G}$ lies inside Mod, joins a forest vertex to Mod, or joins a forest vertex u to its parent v in F. We call the last kind the *parent edge* of u.

The modulator. If ς violates (S1) or (S2) on Mod, all counts are 0. Otherwise let e_0 be the number of edges of $\tilde{G}[\mathrm{Mod}]$ with both ends in $\varsigma^{-1}(\{\mathsf{z}_1, \mathsf{z}_2\})$, and put

$$T_{\mathrm{Mod}} = [(0, e_0, 0, 0)] * \mathop{*}_{m \in \mathrm{Mod}} L_m,$$

where $L_m = [(0,0,0,0)]$ if $\varsigma(m) \in \{\mathsf{y}_1, \mathsf{y}_2\}$, $L_m = [(1,0,0,\mathrm{wt}(m,\mathrm{F}))]$ if $\varsigma(m) = \mathsf{z}_2$, and $L_m = [(1,0,0,\mathrm{wt}(m,\mathrm{F}))] + [(1,0,1,\mathrm{wt}(m,\mathrm{F}) + \mathrm{wt}(m,\mathrm{M}))]$ if $\varsigma(m) = \mathsf{z}_1$. The two terms for z_1 are the choices $m \notin M$ and $m \in M$.

A forest vertex alone. For a vertex v of F and $s \in \Sigma$, let $L_v[s] = 0$ if s violates (S1) at v (that is, $v \in Y_{\mathrm{in}}$ and $s \notin \{\mathsf{y}_1, \mathsf{y}_2\}$, or $v = r$ and $s \neq \mathsf{y}_1$), or if some edge vm with $m \in \mathrm{Mod}$ has $\{s, \varsigma(m)\} = \{\mathsf{y}_1, \mathsf{y}_2\}$ or $\{\mathsf{z}_1, \mathsf{z}_2\}$. Otherwise let d_v be the number of neighbors $m \in \mathrm{Mod}$ of v with $\varsigma(m) \in \{\mathsf{z}_1, \mathsf{z}_2\}$, and let $L_v[s]$ be $[(0,0,0,0)]$ for $s \in \{\mathsf{y}_1, \mathsf{y}_2\}$, $[(1, d_v, 0, \mathrm{wt}(v,\mathrm{F}))]$ for $s = \mathsf{z}_2$, and $[(1, d_v, 0, \mathrm{wt}(v,\mathrm{F}))] + [(1, d_v, 1, \mathrm{wt}(v,\mathrm{F}) + \mathrm{wt}(v,\mathrm{M}))]$ for $s = \mathsf{z}_1$.

Subtrees. Call $s, s' \in \Sigma$ *compatible* if $\{s, s'\}$ is neither $\{\mathsf{y}_1, \mathsf{y}_2\}$ nor $\{\mathsf{z}_1, \mathsf{z}_2\}$. Going up from the leaves of F, for each vertex v with children $u_1, \ldots, u_d$ and each $s \in \Sigma$ put

$$A_v[s] = L_v[s] * \overset{d}{\underset{i=1}{*}} \sum_{s' \text{ compatible with } s} A_{u_i}[s']^{(s,s')},$$

where $A^{(s,s')} = A^{+e}$ if $s, s' \in \{\mathsf{z}_1, \mathsf{z}_2\}$ and $A^{(s,s')} = A$ otherwise. The shift counts the parent edge of u_i when both of its ends lie in $Z(\sigma)$. Finally, with $\rho_1, \ldots, \rho_t$ the roots of the trees of F, output

$$T_\varsigma = T_{\mathrm{Mod}} * \overset{t}{\underset{j=1}{*}} \sum_{s \in \Sigma} A_{\rho_j}[s]$$

reduced modulo 2.

Correctness. By induction from the leaves, $A_v[s](a,e,c,W)$ is the number of pairs (σ_v, M_v) with the following properties. Here $\sigma_v \colon V(T_v) \to \Sigma$ satisfies $\sigma_v(v) = s$, satisfies (S1) on $V(T_v)$, and satisfies (S2) on the edges inside T_v and on the edges from T_v to Mod, with ς on Mod. The set M_v is a subset of $\sigma_v^{-1}(\mathsf{z}_1)$. The index a is $|Z(\sigma_v)|$, and e is the number of those edges with both ends in $Z(\sigma_v) \cup \varsigma^{-1}(\{\mathsf{z}_1, \mathsf{z}_2\})$. The index c is $|M_v|$, and W is the wt-weight of $Z(\sigma_v) \times \{\mathrm{F}\}$ and $M_v \times \{\mathrm{M}\}$. The induction step holds because such a pair for T_v is exactly a choice at v, counted by $L_v[s]$, together with one such pair for each T_{u_i} whose root state is compatible with s, so that (S2) holds on the parent edge of u_i. All four indices add over these parts, and the parent edge of u_i is added to e exactly when both of its ends lie in $Z(\sigma_v)$. In the same way, T_ς before reduction counts the pairs (σ, M) with $\sigma|_{\mathrm{Mod}} = \varsigma$ that satisfy (S1)–(S3), by their indices. Each vertex of $\tilde{G}$ is counted in exactly one of the parts, and so is each edge: an edge inside Mod in e_0, an edge from a forest vertex v to Mod in L_v, and a parent edge in the shift. By the correspondence between triples and pairs (σ, M), this number is $|\mathcal{C}_W^{a,e,c}(\varsigma)|$. All indices stay in range, since $a, c \leq \tilde{n}$, $e \leq \tilde{n}^2$ and $W \leq 2N\tilde{n}$, so the truncation in the convolution loses nothing.

Time and space. A table has $(\tilde{n}+1)^2(\tilde{n}^2+1)(2N\tilde{n}+1)$ entries. Each entry is a nonnegative integer at most $5^{\tilde{n}}$, a bound on the number of pairs (σ, M) (each vertex has state y_1, y_2 or z_2, or state z_1 with or without a marker), so it has $O(\tilde{n})$ bits. One convolution therefore takes time polynomial in $\tilde{n}$ and N. The computation keeps four tables $A_v[s]$ for each vertex whose parent has not been processed, at most $4\tilde{n}$ tables in all, and performs $O(\tilde{n} + |\mathrm{Mod}|)$ convolutions and sums for each of the at most $\tilde{n}$ vertices. The state set Σ, the marker choice and the recursion do not depend on wt. The weight function enters only through the values $\mathrm{wt}(v,\mathrm{F})$ and $\mathrm{wt}(v,\mathrm{M})$ placed in the index W by the tables L_m and L_v, and through the range $0 \leq W \leq 2N\tilde{n}$ of that index. □

For $N = 4\tilde{n}$ and the weights of (CC1), Lemma D.10.9 is a direct proof of the core evaluation of (CC3). In the weighted test we use it with a weight function that also encodes the vertex weights.

Algorithm D.10.10 (Weighted test)

Input: a graph $\tilde{G}$ with $\tilde{n}$ vertices, sets $\emptyset \neq \hat{X} \subseteq Y_{\text{in}} \subseteq V(\tilde{G})$, weights $\tilde{w}\colon V(\tilde{G}) \to \{0, \dots, w_{\max}\}$, integers $k' \geq 0$ and θ', and a structured modulator Mod for $(\tilde{G}, \hat{X}, k')$.

Fix $r \in \hat{X}$ and put $\beta = 8\tilde{n}^2 + 1$ and $N' = \beta w_{\max} + 4\tilde{n}$. For every η with $|Y_{\text{in}}| \leq \eta \leq \min(|\hat{X}| + k', \tilde{n})$:

1. draw $\text{iso}\colon \tilde{U} \to \{1, \dots, 4\tilde{n}\}$ uniformly at random, and put $\text{wt}'(v, \text{F}) = \beta\tilde{w}(v) + \text{iso}(v, \text{F})$ and $\text{wt}'(v, \text{M}) = \text{iso}(v, \text{M})$, so that wt' takes values in $\{1, \dots, N'\}$;
2. compute $|\mathcal{C}_W^{a,e,a-e}| \bmod 2$ for $a = \tilde{n} - \eta$, all e and all W, for the weight function wt', as the sum over the reasonable colorings of Mod, listed by Lemma D.9.5, of the core evaluations of Lemma D.10.9;
3. if $|\mathcal{C}_W^{a,e,a-e}|$ is odd for some e and some W with $\tilde{w}(V(\tilde{G})) - \lfloor W/\beta \rfloor \leq \theta'$, accept.

If no η leads to acceptance, reject.

Lemma D.10.11 (Weighted test)**.** *Algorithm D.10.10 runs in polynomial space and, for every outcome of its coins, in time* $3^{k'}2^{|\hat{X}|}(\tilde{n}w_{\max})^{O(1)}$*. If it accepts, then* $\tilde{G}$ *has a connected feedback vertex set* $Y \supseteq Y_{\text{in}}$ *with* $|Y| \leq |\hat{X}| + k'$ *and* $\tilde{w}(Y) \leq \theta'$*. If* $\tilde{G}$ *has such a set, the algorithm accepts with probability at least* $1/2$*.*

Proof. For a pair (Z, M) write $\text{iso}(Z, M) = \text{iso}(Z \times \{\text{F}\}) + \text{iso}(M \times \{\text{M}\})$, and similarly $\text{wt}'(Z, M)$. Then $\text{wt}'(Z, M) = \beta\tilde{w}(Z) + \text{iso}(Z, M)$ with $0 \leq \text{iso}(Z, M) \leq 2\tilde{n} \cdot 4\tilde{n} < \beta$, so $\lfloor \text{wt}'(Z, M)/\beta \rfloor = \tilde{w}(Z)$ and $\text{wt}'(Z, M) \bmod \beta = \text{iso}(Z, M)$. Fix η and put $a = \tilde{n} - \eta$. By Lemma D.10.9 the algorithm computes the parities $|\mathcal{C}_W^{a,e,a-e}| \bmod 2$ for wt', which equal $|\mathcal{S}_W^{a,e,a-e}| \bmod 2$ by Lemma D.10.8.

Soundness. Suppose the algorithm accepts at η, e and W. Then $\mathcal{S}_W^{a,e,a-e}$ contains a pair (Z, M). The set $Y = V(\tilde{G}) \setminus Z$ contains Y_{in}, has size $\eta \leq |\hat{X}| + k'$, $\tilde{G} - Y = \tilde{G}[Z]$ is a forest and $\tilde{G}[Y] = \tilde{G} - Z$ is connected; so Y is a connected feedback vertex set. Its weight is $\tilde{w}(Y) = \tilde{w}(V(\tilde{G})) - \tilde{w}(Z) = \tilde{w}(V(\tilde{G})) - \lfloor W/\beta \rfloor \leq \theta'$.

Completeness. Let Y^* be a set as in the lemma and $\eta = |Y^*|$; then $|Y_{\text{in}}| \leq \eta \leq \min(|\hat{X}| + k', \tilde{n})$, so this η is tried. Let $\mathcal{P}$ be the family of pairs (Z, M) such that $Y = V(\tilde{G}) \setminus Z$ is a connected feedback vertex set with $Y \supseteq Y_{\text{in}}$, $|Y| = \eta$ and $\tilde{w}(Y) \leq \theta'$, and M contains exactly one vertex of each component of the forest $\tilde{G}[Z]$. Identify (Z, M) with $(Z \times \{\text{F}\}) \cup (M \times \{\text{M}\}) \subseteq \tilde{U}$; this is injective and maps $\text{iso}(Z, M)$ to the weight of the set. The family $\mathcal{P}$ is nonempty, since Y^* with one marker per component of $\tilde{G} - Y^*$ belongs to it. As $|\tilde{U}| = 2\tilde{n}$, the Isolation Lemma [CNP+22, Lemma 2.5] with weights drawn from $\{1, \dots, 4\tilde{n}\}$ shows that with probability at least $1/2$ a unique pair $(Z^\circ, M^\circ) \in \mathcal{P}$ minimizes iso over $\mathcal{P}$. Assume this, and put $e^\circ = |E(\tilde{G}[Z^\circ])|$ and $W^\circ = \text{wt}'(Z^\circ, M^\circ)$. We claim $\mathcal{S}_{W^\circ}^{a,e^\circ,a-e^\circ} = \{(Z^\circ, M^\circ)\}$. The pair (Z°, M°) belongs to it, since $|M^\circ| = \text{comp}(\tilde{G}[Z^\circ]) = |Z^\circ| - e^\circ$ for a forest. Let (Z, M) belong to it. Then $\tilde{G}[Z]$ is a forest with $|Z| - e^\circ$ components, each containing a vertex of M, and $|M| = |Z| - e^\circ$, so each component contains exactly one vertex of M; the set $Y = V(\tilde{G}) \setminus Z$ is a connected feedback vertex set containing Y_{in} with $|Y| = \eta$; and $\tilde{w}(Z) = \lfloor W^\circ/\beta \rfloor = \tilde{w}(Z^\circ)$, so $\tilde{w}(Y) = \tilde{w}(V(\tilde{G}) \setminus Z^\circ) \leq \theta'$. Hence $(Z, M) \in \mathcal{P}$ with $\text{iso}(Z, M) = W^\circ \bmod \beta = \text{iso}(Z^\circ, M^\circ)$, and uniqueness gives $(Z, M) = (Z^\circ, M^\circ)$. So $|\mathcal{C}_{W^\circ}^{a,e^\circ,a-e^\circ}|$ is odd, and $\tilde{w}(V(\tilde{G})) - \lfloor W^\circ/\beta \rfloor = \tilde{w}(V(\tilde{G}) \setminus Z^\circ) \leq \theta'$, so the algorithm accepts.

Time and space. There are at most $\tilde{n} + 1$ values of η. By Lemma D.9.5, Mod has at most $8 \cdot 3^{k'}2^{|\hat{X}|}$ reasonable colorings, listed in that time and polynomial space; its hypotheses hold because

Mod $= C \cup C'$ is a structured modulator and $r \in \hat{X} \subseteq Y_{\text{in}} \cap C$. By Lemma D.10.9 each core evaluation takes time and space polynomial in $\tilde{n}$ and $N' \leq (8\tilde{n}^2+1)w_{\max} + 4\tilde{n}$, and the parities are accumulated coloring by coloring. □

We now solve the forced problem of Appendix D.9.4 with weights.

Algorithm D.10.12 (Forced weighted extension)

Input: a forced instance $(\tilde{G}, \hat{X}, \tilde{w}, k', \theta')$, with $\tilde{G}$ on $\tilde{n}$ vertices.

If $\theta' < 0$, reject.

1. Run the algorithm $\mathcal{A}$ of Lemma 11.7 on $(\tilde{G}, \hat{X}, |\hat{X}| + k')$. If it rejects, reject. Otherwise it outputs Y_0; put $A_0 = \hat{X} \cup Y_0$ and Mod $= C \cup C'$ with $C = A_0$ and $C' = \emptyset$.
2. Start with $Y_{\text{in}} = \hat{X}$ and go through the vertices $v \in V(\tilde{G}) \setminus \hat{X}$ in a fixed order: run Algorithm D.10.10 on $(\tilde{G}, Y_{\text{in}} \cup \{v\}, \hat{X}, \tilde{w}, k', \theta', \text{Mod})$ independently $\lceil \log(8\tilde{n}) \rceil$ times, and if one run accepts, add v to Y_{in}.
3. If Y_{in} is a connected feedback vertex set of $\tilde{G}$ with $|Y_{\text{in}} \setminus \hat{X}| \leq k'$ and $\tilde{w}(Y_{\text{in}}) \leq \theta'$, output $A = Y_{\text{in}}$; otherwise reject.

Lemma D.10.13 (Forced weighted extension)**.** *Algorithm D.10.12 runs in polynomial space and, for every outcome of its coins, in time* $3^{k'} 2^{|\hat{X}|} (\tilde{n} w_{\max})^{O(1)}$*. Every output is a solution of the input instance, and if one exists, the algorithm outputs one with probability at least* $5/8$*.*

Proof. The output is checked in step 3.

The modulator. If step 1 succeeds, then by Lemma 11.7 (with $G = \tilde{G}$ and $X = \hat{X}$) the set A_0 is a connected feedback vertex set of $\tilde{G}$ with $\hat{X} \subseteq A_0$ and $|A_0 \setminus \hat{X}| \leq k'$. So $\tilde{G} - \text{Mod}$ is a forest, $\tilde{G}[C]$ is connected, $\hat{X} \subseteq C$, $C' = \emptyset$ and $|\text{Mod} \setminus \hat{X}| \leq k' \leq k'+2$: Mod is a structured modulator for $(\tilde{G}, \hat{X}, k')$, and every set $Y_{\text{in}} \cup \{v\}$ tested in step 2 contains $\hat{X}$, as Algorithm D.10.10 requires.

Success. Let A^* be a solution of the instance. Then $Y_0 = A^* \setminus \hat{X}$ shows that $(\tilde{G}, \hat{X}, |\hat{X}| + k')$ is a YES-instance of Connected Feedback Vertex Set Extension, so step 1 succeeds with probability at least $3/4$ by Lemma 11.7. Step 2 is the self-reduction of Lemma D.10.1 with $\Omega = V(\tilde{G})$, $Z_0 = \hat{X}$, the family $\mathcal{P}$ of the connected feedback vertex sets $Y \supseteq \hat{X}$ of $\tilde{G}$ with $|Y \setminus \hat{X}| \leq k'$ and $\tilde{w}(Y) \leq \theta'$, which contains A^*, the test $\mathsf{T}(Y_{\text{in}})$ given by Algorithm D.10.10 on $(\tilde{G}, Y_{\text{in}}, \hat{X}, \tilde{w}, k', \theta', \text{Mod})$, $\delta = 1/8$ and $\rho = \lceil \log(8\tilde{n}) \rceil \geq \log(\tilde{n}/\delta)$. By Lemma D.10.11, the test rejects Y_{in} if no member of $\mathcal{P}$ contains Y_{in} (for $Y \supseteq \hat{X}$ the conditions $|Y| \leq |\hat{X}| + k'$ and $|Y \setminus \hat{X}| \leq k'$ agree), and with probability at least $1/2$ if one does. So, with probability at least $7/8$, the final Y_{in} is a member of $\mathcal{P}$, and step 3 outputs it. Hence the algorithm fails with probability at most $1/4 + 1/8$.

Time and space. Step 1 takes time $3^{k'} 2^{\text{comp}(\tilde{G}[\hat{X}])} \tilde{n}^{O(1)} \leq 3^{k'} 2^{|\hat{X}|} \tilde{n}^{O(1)}$ and polynomial space by Lemma 11.7. Step 2 runs Algorithm D.10.10 at most $\tilde{n} \lceil \log(8\tilde{n}) \rceil$ times, each within the bounds of Lemma D.10.11. □

For an instance $I = (G, w)$ with G connected and not a forest, and an integer θ, let Φ^{w}_θ be the implicit set system with universe $V(G)$ whose solutions are the connected feedback vertex sets S of G with $w(S) \leq \theta$. Membership is decided in polynomial time, and every solution is nonempty.

Proof of Theorem 11.17. If G is a forest, output $\emptyset$. If G is disconnected, then, as in Section 11.3.1, a connected feedback vertex set exists only if exactly one component G_1 of G contains a cycle (as

G is not a forest, at least one does), and then the connected feedback vertex sets of G are those of G_1; report that none exists or replace G by G_1. So let G be connected and not a forest.

Let $\mathcal{A}_{\mathrm{w}}$ be the algorithm of Lemma D.9.7 with Algorithm D.10.12 as Forced; by Lemma D.10.13 the hypotheses of Lemma D.9.7 hold with $p = 5/8$, as $(\tilde{n}w_{\max})^{O(1)} \le (\tilde{n} + w_{\max})^{O(1)}$. So $\mathcal{A}_{\mathrm{w}}$ solves Φ^{w}_θ-Extension with one-sided error, accepts a YES-instance with probability at least $5/8$, and runs in polynomial space and in time $3^{k-|X|}2^{\mathrm{comp}(G[X])}n^{O(1)}$ for every outcome of its coins, because $w_{\max} \le n^{O(1)}$. For every $\theta \in \{1, \dots, nw_{\max}\}$ and every $k \in \{1, \dots, n\}$, run Algorithm MLS-θ for Φ^{w}_θ and k as in Corollary 11.6, with the extension algorithm $\mathcal{A}_{\mathrm{w}}$, $c = 3$, $\gamma = 0$, $\psi = \log 1.8$ and $\kappa = 1$. Output the lightest set returned by these runs, and $V(G)$ if no run returns a set; $V(G)$ is a connected feedback vertex set because G is connected.

The algorithm $\mathcal{A}_{\mathrm{w}}$ satisfies the hypotheses of Corollary 3.15 with the running-time exponent $g(G, X, k) = (k - |X|)\log 3 + \mathrm{comp}(G[X])$ of (22), whose comparisons with the thresholds of Algorithm MLS-θ reduce to integer comparisons (Lemma 11.7). Every solution S induces a connected subgraph, so a spanning tree of $G[S]$ gives the component-count bound with constants $(3, 0)$, as in the proof of Theorem 11.8. By the last claim of Corollary 11.6, every run takes time $1.8^n n^{O(1)}$ and polynomial space, and every returned set is a connected feedback vertex set of weight at most θ. Let S^* be a connected feedback vertex set of minimum weight $\theta^* \le nw_{\max}$; the run for $\theta = \theta^*$ and $k = |S^*|$ returns a set with probability at least $1 - 1/e$, and then the output has weight θ^*. There are $n^2 w_{\max} \le n^{O(1)}$ runs. □

D.11 The further applications

This subsection contains the lemmas and proofs for Sections 12.1 to 12.4, the analyses of Directed Steiner Tree and Paired Domination (Appendices D.11.6 and D.11.7), and the proofs of their weighted versions.

D.11.1 Red–blue states

For a finite set V let $V^{\mathrm{b}} = \{v^{\mathrm{b}} : v \in V\}$ and $V^{\mathrm{r}} = \{v^{\mathrm{r}} : v \in V\}$ be two disjoint copies of V, the *blue* and the *red* copy, and for $X \subseteq V$ write $X^{\mathrm{b}} = \{x^{\mathrm{b}} : x \in X\}$. A *red–blue graph* on V is a graph H with vertex set $V^{\mathrm{b}} \cup V^{\mathrm{r}}$ in which $v^{\mathrm{b}}v^{\mathrm{r}}$ is an edge for every $v \in V$. A set $S \subseteq V$ is *connected red–blue dominating* in H if $S \neq \emptyset$, $H[S^{\mathrm{b}}]$ is connected and every red vertex has a neighbor in S^{b}. This is the problem that Abu-Khzam, Mouawad and Liedloff [AKML11] solve in time $O^*(1.3645^{|V^{\mathrm{b}}|+|V^{\mathrm{r}}|})$ and polynomial space; here it serves to formulate the extension problems of the variants as states of Definition 5.4.

Lemma D.11.1 (Red–blue states)**.** *Let H be a red–blue graph on a set V with $n \ge 1$ elements, and for $X \subseteq V$ put*

$$g_H(X) = \big|\{v \in V : v^{\mathrm{r}} \text{ has no neighbor in } X^{\mathrm{b}}\}\big| + \mathrm{comp}(H[X^{\mathrm{b}}]).$$

Then g_H is computed in polynomial time and takes integer values in $\{0, \dots, n\}$, with $g_H(\emptyset) = n$ and $g_H(S) = 1$ for every connected red–blue dominating set S. For every $X \subseteq V$ and every $\varphi\colon V \setminus X \to \{0, 1, \dots, \varphi_{\max}\}$, the least weight $\varphi(Y)$ over the sets $Y \subseteq V \setminus X$ for which $X \cup Y$ is connected red–blue dominating in H (∞ if there is none) is computed deterministically in time $2^{g_H(X)}(n + \varphi_{\max})^{O(1)}$ and polynomial space; with $\varphi \equiv 1$, this decides for every integer $k \ge 0$, in time $2^{g_H(X)}n^{O(1)}$, whether some such Y has $|Y| \le k - |X|$. Consequently, if Φ is an implicit set system whose instance I determines in polynomial time a red–blue graph H_I on U_I whose connected red–blue dominating sets are the solutions, then this decision algorithm solves Φ-Extension deterministically in polynomial space, with the running-time exponent $(I, X, k) \mapsto g_{H_I}(X)$ (Definition 3.2).

Proof. Let $\mathcal{D}$ be the family of vertex sets of the components of $H[X^{\mathrm{b}}]$, let $B = (V \setminus X)^{\mathrm{b}}$, and let R be the set of red vertices without a neighbor in X^{b}. Then $J = (\mathcal{D}, B, R)$ is a state of H in the sense of Definition 5.4: the members of $\mathcal{D}$ are disjoint, nonempty and connected, no edge of H joins two distinct components of an induced subgraph, $V(\mathcal{D}) = X^{\mathrm{b}}$, B avoids X^{b}, and R avoids $N_H[X^{\mathrm{b}}]$ because a red vertex is not in X^{b} and the vertices of R are not adjacent to X^{b}. A subset of B has the form Y^{b} with $Y \subseteq V \setminus X$, and as no red vertex lies in Y^{b}, the condition $R \subseteq N_H[Y^{\mathrm{b}}]$ says that every red vertex without a neighbor in X^{b} has one in Y^{b}, that is, that every red vertex has a neighbor in $(X \cup Y)^{\mathrm{b}}$. So the solutions of J are exactly the sets Y^{b} with $Y \subseteq V \setminus X$ for which $X \cup Y$ is connected red–blue dominating, and $|\mathrm{T}(J)| = |R| + |\mathcal{D}| = g_H(X)$. Give the blue vertex y^{b} of B the weight $\varphi(y)$. By Lemma 5.6, applied to the state J of the graph H with $2n$ vertices, the value $\mathrm{OPT}_\varphi(J)$, which is the least weight in question, is computed deterministically in time $2^{g_H(X)}(2n + \varphi_{\max})^{O(1)}$ and polynomial space. With $\varphi \equiv 1$ this is $\mathrm{OPT}(J)$, the least $|Y|$, computed in time $2^{g_H(X)}n^{O(1)}$ by the first statement of Corollary 5.8, and one accepts if and only if $\mathrm{OPT}(J) \le k - |X|$.

For the range, every $v \in X$ has the neighbor $v^{\mathrm{b}} \in X^{\mathrm{b}}$ of v^{r}, so the first term of $g_H(X)$ is at most $n - |X|$, and $\mathrm{comp}(H[X^{\mathrm{b}}]) \le |X|$. For $X = \emptyset$ the first term is n and the second is 0. For a connected red–blue dominating set S the first term is 0 and the second is 1. Both terms are computed in polynomial time from H and X.

For the last claim, membership in $\mathcal{F}_I$ is tested in polynomial time through H_I, so Φ is as in Definition 3.1, and g_{H_I} is computable in polynomial time and bounds the running time of the algorithm as Definition 3.2 requires. □

D.11.2 Connected r-domination

Theorem 12.2 (Connected $\mathcal{N}$-domination)**.** *Let $\mathcal{N}$ be a domination relation. For connected n-vertex graphs G, the value $\gamma_{\mathrm{c}}^{\mathcal{N}}(G)$, in particular the minimum size of a connected r-dominating set for every $r \ge 1$ (with r part of the input), can be computed*

(a) by a randomized algorithm with one-sided error in time $O(1.7204^n)$ and polynomial space;

(b) deterministically in time $O((\sqrt{5} - \frac{1}{2})^n\, 2^{O(n(\log\log n)^2/\log n)}) = O(1.7361^n)$ and polynomial space.

Throughout, $\mathcal{N}$ is a domination relation (Definition 12.1) on a connected graph G with $n \ge 2$ vertices.

The Steiner extension. For $X \subseteq V$ put

$$g_{\mathcal{N}}(X) = |\,\mathrm{und}_{\mathcal{N}}(X)| + \mathrm{comp}(G[X]).$$

For $\mathcal{N}[v] = N[v]$ this is $|\mathrm{T}(J_X)|$ by (13). The instance I consists of G and the description of $\mathcal{N}$, which are fixed, and $g_{\mathcal{N}}(X)$ does not depend on k; in the notation of Definition 3.2, the running-time exponent is $g(I, X, k) = g_{\mathcal{N}}(X)$. The next lemma shows that the Steiner extension transfers with this running-time exponent.

Lemma D.11.2 (Steiner extension for domination relations)**.** *Let $H_{\mathcal{N}}$ be the red–blue graph on V with the edges $u^{\mathrm{b}}v^{\mathrm{b}}$ for the edges uv of G and the edges $u^{\mathrm{b}}v^{\mathrm{r}}$ for all u, v with $u \in \mathcal{N}[v]$. The connected red–blue dominating sets of $H_{\mathcal{N}}$ are the connected $\mathcal{N}$-dominating sets of G, and $g_{H_{\mathcal{N}}} = g_{\mathcal{N}}$. So Lemma D.11.1 gives a deterministic extension algorithm for the implicit set system of connected $\mathcal{N}$-dominating sets, in time $2^{g_{\mathcal{N}}(X)}n^{O(1)}$ and polynomial space, with the running-time exponent $(I, X, k) \mapsto g_{\mathcal{N}}(X)$, which takes values in $\{0, \dots, n\}$.*

Proof. The graph $H_{\mathcal{N}}$ is built in polynomial time, because every $\mathcal{N}[v]$ is, and $v^{\mathrm{b}}v^{\mathrm{r}}$ is an edge since $v \in N[v] \subseteq \mathcal{N}[v]$. For $S \subseteq V$ the graph $H_{\mathcal{N}}[S^{\mathrm{b}}]$ is a copy of $G[S]$, and v^{r} has a neighbor in S^{b} if and only if $\mathcal{N}[v] \cap S \neq \emptyset$. So S is connected red–blue dominating in $H_{\mathcal{N}}$ if and only if S is a connected $\mathcal{N}$-dominating set, and $g_{H_{\mathcal{N}}}(X) = g_{\mathcal{N}}(X)$ for every $X \subseteq V$. Now Lemma D.11.1 applies. □

Anchored trees. The analyses of Sections 5.3 and 7.2 assign every vertex outside the solution to a neighbor inside it. For a domination relation we assign it to a solution vertex that dominates it, which need not be a neighbor, and we treat the assignment as a pendant edge of a virtual tree. In this tree the solution is a connected dominating set, and the running-time exponent of a sample is at most its number of terminals in the tree. For a graph T' on the vertex set V and $X \subseteq V$ we write

$$g_{T'}(X) = n - |N_{T'}[X]| + \mathrm{comp}(T'[X])$$

for the number of terminals $|\mathrm{T}(J_X)|$ of the state J_X of (12) formed in the graph T', by (13).

Lemma D.11.3 (Anchored tree)**.** *Let S be a connected $\mathcal{N}$-dominating set, let F be a spanning tree of $G[S]$, and let $o : V \setminus S \to S$ be an* anchor map*, that is, $o(v) \in \mathcal{N}[v] \cap S$ for every $v \notin S$. Let T' be the graph on V formed by F and the edges $v\,o(v)$ for $v \notin S$. Then T' is a tree, S is a connected dominating set of T', every vertex outside S is a leaf of T', and*

$$g_{\mathcal{N}}(X) \;\leq\; g_{T'}(X) \qquad \text{for every } X \subseteq S.$$

Consequently, in the notation of Definition 3.3, the exponential moments $\mu_t(S)$ for the running-time exponent $g_{\mathcal{N}}$ and $\mu_t^{T'}(S)$ for the running-time exponent $g_{T'}$ of Φ_{CDS} on the graph T' satisfy $\mu_t(S) \geq \mu_t^{T'}(S)$ termwise, and every integer certificate (Theorem 9.10) for S with respect to $g_{T'}$ is one with respect to $g_{\mathcal{N}}$.

Proof. The graph T' is the tree F with the $n - |S|$ vertices outside S attached as pendant vertices, so it is a tree, $T'[S] = F$ is connected, every $v \notin S$ has exactly one neighbor, namely $o(v) \in S$, and an anchor map exists because S is $\mathcal{N}$-dominating. Let $X \subseteq S$. We show $\mathrm{und}_{\mathcal{N}}(X) \subseteq V \setminus N_{T'}[X]$. Let $v \in N_{T'}[X]$. If $v \in X$, then $v \in N[v] \subseteq \mathcal{N}[v]$ meets X. Otherwise v is joined in T' to some $x \in X \subseteq S$; if the edge vx belongs to F, then $x \in N[v] \subseteq \mathcal{N}[v]$, and if it is a pendant edge, then $v \notin S$ and $x = o(v) \in \mathcal{N}[v]$, because the only pendant edge at v is $v\,o(v)$. In both cases v is dominated by X. Finally $\mathrm{comp}(G[X]) \leq \mathrm{comp}(F[X]) = \mathrm{comp}(T'[X])$, because $F[X]$ is a spanning subgraph of $G[X]$ and $T'[X] = F[X]$ for $X \subseteq S$. Adding the two bounds gives $g_{\mathcal{N}}(X) \leq |V \setminus N_{T'}[X]| + \mathrm{comp}(T'[X]) = g_{T'}(X)$. The last claims follow from $2^{-g_{\mathcal{N}}(X)} \geq 2^{-g_{T'}(X)}$ for every $X \subseteq S$, and from $g_{\mathcal{N}}(X) \leq g_{T'}(X) \leq \alpha - w(X)$ for every nonempty $X \subseteq S$ when w is an integer certificate for S with respect to $g_{T'}$. □

Proof of Theorem 12.2. Let $\Phi^{\mathcal{N}}$ be the implicit set system of connected $\mathcal{N}$-dominating sets, with the extension algorithm and the running-time exponent $g_{\mathcal{N}}$ of Lemma D.11.2; G has a solution, and $\gamma_{\mathrm{c}}^{\mathcal{N}}(G)$ is the least solution size. Let S be a minimum connected $\mathcal{N}$-dominating set, $s^* = |S|$, and let T' be an anchored tree of Lemma D.11.3 for S.

(a) Let $\mathrm{Dec}(G, s)$ be Algorithm MLS-θ for $\Phi^{\mathcal{N}}$, with $k = s$, $\psi = \psi^*$, $\Gamma = 2$ and confidence κ, that is, Algorithm RS-θ (Algorithm 7.1) with the extension algorithm of Lemma D.11.2 in place of $\mathcal{A}_{\mathrm{St}}$ and $g_{\mathcal{N}}$ in place of $|\mathrm{T}(J_X)|$. By Theorem 3.5, it takes time $2^{\psi^* n} n^{O(1)}$ and polynomial space, and it rejects if $\gamma_{\mathrm{c}}^{\mathcal{N}}(G) > s$. As T' is a connected graph and S is a connected dominating set of T', Proposition 7.8 applied to T' gives $\max_t \mu_t^{T'}(S)/\binom{n}{t} \geq 2^{-\psi^* n}/2$, and by Lemma D.11.3 the set S satisfies condition ($\star$) for $g_{\mathcal{N}}$; so $\mathrm{Dec}(G, s^*)$ accepts with probability at least $1 - e^{-\kappa}$, by Theorem 3.5. Now Lemma 5.2 gives the claim; the bound $2^{\psi^*} < 1.7204$ is as in Theorem 7.2.

(b) Run the procedure Det of Lemma 10.3 for $\Phi^{\mathcal{N}}$, the extension algorithm and running-time exponent of Lemma D.11.2 and $(\alpha, \Xi) = (n-1, n-2)$, for $s = 1, \ldots, n$, and output the least s at which it answers YES. By Lemma 10.3, it answers NO if $\gamma_c^{\mathcal{N}}(G) > s$. At $s = s^*$ the set S has size exactly s, and it has an integer certificate for $(n-1, n-2)$: the tree T' is a spanning tree of the graph T' in which every vertex outside S is a leaf, so the weights $w(u) = d_{T'}(u) - 1$ on S form the certificate (19) for S in the graph T', that is, they are nonnegative integers with $w(S) = n-2$ and $g_{T'}(X) \le n - 1 - w(X)$ for every nonempty $X \subseteq S$, by Lemma 5.10 and (13) applied to T', and by Lemma D.11.3 they form an integer certificate for S with respect to $g_{\mathcal{N}}$. So Det answers YES at $s = s^*$, and Lemma 5.2, in its deterministic form and with the time and space bounds of Lemma 10.3, gives the theorem. □

D.11.3 Vertex-weighted connected domination

Theorem 12.3 (Vertex-weighted CONNECTED DOMINATING SET and weighted MaxLeaf)**.** *For connected n-vertex graphs with vertex weights in $\{0, \ldots, \varphi_{\max}\}$, the minimum weight of a connected dominating set, and the maximum leaf weight of a spanning tree (by Lemma D.11.4 for $n \ge 3$; for $n \le 2$ it is 0 if $n = 1$ and $\varphi(V)$ if $n = 2$), can be computed*

(a) by a randomized algorithm with one-sided error in time $O(1.7204^n)(n + \varphi_{\max})^{O(1)}$ and polynomial space;

(b) deterministically in time $(\sqrt{5} - \frac{1}{2})^n 2^{O(n(\log\log n)^2/\log n)}(n + \varphi_{\max})^{O(1)} = O(1.7361^n)(n + \varphi_{\max})^{O(1)}$ and polynomial space.

The notation is that of Section 12.2.

Lemma D.11.4 (Weighted max leaf and connected domination)**.** *If $n \ge 3$, the maximum leaf weight of a spanning tree of G equals $\varphi(V) - \gamma_c^{\varphi}(G)$.*

Proof. If F is a spanning tree, then $V \setminus \text{leaves}(F)$ is a connected dominating set, by the first half of the proof of Lemma 5.1, so $\varphi(\text{leaves}(F)) = \varphi(V) - \varphi(V \setminus \text{leaves}(F)) \le \varphi(V) - \gamma_c^{\varphi}(G)$. Conversely, for a connected dominating set S the second half of that proof gives a spanning tree in which every vertex of $V \setminus S$ is a leaf; as $\varphi \ge 0$, its leaf weight is at least $\varphi(V \setminus S) = \varphi(V) - \varphi(S)$. □

The weighted Steiner extension. For $X \subseteq V$ let

$$\text{OPT}_{\varphi}(X) = \min\{\varphi(Y) : Y \subseteq V \setminus X,\ X \cup Y \text{ is a connected dominating set}\}.$$

Since the solutions of the state J_X of (12) are the sets $Y \subseteq V \setminus X$ for which $X \cup Y$ is a connected dominating set (also for $X = \emptyset$), this is $\text{OPT}_{\varphi}(J_X)$ in the notation of Definition 5.5, with φ restricted to the blue vertices $V \setminus X$, and Lemma 5.6 computes it deterministically in time $2^{|\mathrm{T}(J_X)|}(n + \varphi_{\max})^{O(1)}$ and polynomial space.

Relaxed extension. The weighted extension does not control the size of the completion: Lemma 5.3 minimizes weight, and a completion of least weight may be large. We show that this does not affect Theorems 3.5 and 9.10: their proofs use the correctness of the extension algorithm in only two places. The following lemma extends the two theorems accordingly.

Lemma D.11.5 (Relaxed extension algorithms)**.** *Let Φ be an implicit set system, and let $\mathcal{A}$ be a deterministic algorithm on inputs (I, X, k) such that*

(R1) *$\mathcal{A}$ accepts (I, X, k) whenever some $Y \subseteq U_I \setminus X$ with $|Y| \leq k - |X|$ has $X \cup Y \in \mathcal{F}_I$, and*

(R2) *$\mathcal{A}$ rejects (I, X, k) if $\mathcal{F}_I = \emptyset$.*

Let g satisfy Definition 3.2 for $\mathcal{A}$. Then Theorem 3.5, with its ranges for the cap and the trial counts, and Theorem 9.10 hold for $\mathcal{A}$ and g, with the soundness statement "answers NO *if $\mathcal{F}_I$ contains no set of size at most k" replaced by "answers* NO *if $\mathcal{F}_I = \emptyset$."*

Proof. An algorithm for Φ-Extension satisfies (R1) and (R2), and the proofs of the two theorems use the correctness of $\mathcal{A}$ only through these two properties, in the following places. The soundness parts (item (ii) of Theorem 3.5, and the paragraph called soundness in the proof of Theorem 9.10) conclude from an accepting run of $\mathcal{A}$ that a solution exists; by (R2), $\mathcal{F}_I \neq \emptyset$. The completeness parts (item (iii) of Theorem 3.5, and the paragraph called completeness) run $\mathcal{A}$ on (I, X, k) with $X \subseteq S$, $S \in \mathcal{F}_I$ and $|S| = k$, and conclude that it accepts because $Y = S \setminus X$ is a witness; this is (R1). The statements about time and space use only that $\mathcal{A}$ runs in time $2^{g(I,X,k)}N^{O(1)}$ on every input (I, X, k), and the early answer of Algorithm 9.9, the rounding arguments, Lemma 3.8, Theorem 9.6 and Lemma 9.7 do not involve $\mathcal{A}$. □

Budget loop. The weighted problems of this paper are solved by running a decision algorithm for every weight budget and every solution size. The following lemma states this wrapper once.

Lemma D.11.6 (Budget loop)**.** *Let Φ be an implicit set system whose instances I include a weight function $w\colon U_I \to \{0, 1, \ldots, w_{\max}\}$, written in unary, and write $w(S) = \sum_{u \in S} w(u)$. For an integer $\varpi \geq 0$ let Φ_ϖ be the implicit set system with the same instances and universes whose solutions are the solutions S of Φ with $w(S) \leq \varpi$. Let $p \in (0, 1]$, and suppose that for every ϖ an algorithm $\mathcal{B}_\varpi$, given I and an integer $k \in \{0, \ldots, n\}$, answers* YES *or* NO *within time T and polynomial space, such that*

(B1) *$\mathcal{B}_\varpi(I, k)$ answers* NO *if Φ_ϖ has no solution on I, and*

(B2) *if Φ has a solution on I, then some solution S^* of minimum weight has the property that $\mathcal{B}_{w(S^*)}(I, |S^*|)$ answers* YES *with probability at least p.*

Let the algorithm $\mathcal{B}$ run $\mathcal{B}_\varpi(I, k)$ for every $\varpi \in \{0, \ldots, w(U_I)\}$ and every $k \in \{0, \ldots, n\}$, output the least ϖ for which some run answers YES*, and report that Φ has no solution on I if no run answers* YES*. Then $\mathcal{B}$ runs in time $(w(U_I) + 1)(n + 1)T + N^{O(1)}$ and polynomial space, it never outputs a value below the minimum weight of a solution, and with probability at least p it outputs the minimum weight of a solution (respectively reports correctly that there is none).*

Proof. If Φ has no solution on I, then no Φ_ϖ has one, so by (B1) no run answers YES and the report is correct. Otherwise let S^* be as in (B2) and $\varpi^* = w(S^*)$. Every solution has weight at least ϖ^*, so for $\varpi < \varpi^*$ the system Φ_ϖ has no solution, and by (B1) no run for such a ϖ answers YES; hence the output, if any, is at least ϖ^*. The run $\mathcal{B}_{\varpi^*}(I, |S^*|)$ is among the runs, because $\varpi^* \leq w(U_I)$ and $|S^*| \leq n$, and it answers YES with probability at least p by (B2); then the output is ϖ^*. There are $(w(U_I) + 1)(n + 1)$ runs, and $w(U_I) \leq N$ because w is written in unary. □

In the applications, $\mathcal{B}_\varpi$ is Algorithm RS-θ, Algorithm 9.9 or Algorithm MLS-θ for Φ_ϖ, with a relaxed extension algorithm (Lemma D.11.5) or an extension algorithm with one-sided error (Corollary 3.15); property (B1) is the soundness statement of these theorems, and property (B2) is their completeness at $k = |S^*|$, from condition $(\star)$ with $p = 1 - e^{-\kappa}$ or from an integer certificate with $p = 1$; the completeness is used only at the size of S^*.

The algorithms. For a budget $\varpi \in \{0, \dots, \varphi(V)\}$ let Φ_{wCDS} be the implicit set system whose instances are triples (G, φ, ϖ), with universe V and with the connected dominating sets S with $\varphi(S) \le \varpi$ as solutions. Let $\mathcal{A}_\varphi$ be the algorithm that, on $((G, \varphi, \varpi), X, k)$, computes $\mathrm{OPT}_\varphi(X)$ by Lemma 5.6 and accepts if and only if $\varphi(X) + \mathrm{OPT}_\varphi(X) \le \varpi$; it ignores k. It satisfies (R1) and (R2): a witness Y in (R1) gives $\varphi(X) + \mathrm{OPT}_\varphi(X) \le \varphi(X \cup Y) \le \varpi$, and an acceptance yields a connected dominating set $X \cup Y$ of weight at most ϖ. By Lemma 5.6, the function g with $g((G, \varphi, \varpi), X, k) = |\mathrm{T}(J_X)|$ is a running-time exponent for $\mathcal{A}_\varphi$ in the sense of Definition 3.2, with integer values in $\{0, \dots, n\}$; it depends neither on k nor on φ and ϖ, and in the shorthand declared after Definition 3.2 it is $g(X) = |\mathrm{T}(J_X)|$.

Proof of Theorem 12.3. For $n \le 2$ the maximum leaf weight is output directly, as stated in the theorem, since G is its own only spanning tree. For $n \ge 3$, by Lemma D.11.4 it suffices to compute $\gamma_{\mathrm{c}}^{\varphi}(G)$, which we do by Lemma D.11.6 for the implicit set system of connected dominating sets with the weights φ, so that Φ_ϖ is Φ_{wCDS} on (G, φ, ϖ) and $w(U_I) = \varphi(V) \le n\varphi_{\max}$. In (a), $\mathcal{B}_\varpi(G, \varphi, s)$ is the run for the candidate size s of Theorem 7.2 (Algorithm RS-θ, Algorithm 7.1) for Φ_{wCDS} on (G, φ, ϖ), with the relaxed extension algorithm $\mathcal{A}_\varphi$ in place of $\mathcal{A}_{\mathrm{St}}$ and the running-time exponent $|\mathrm{T}(J_X)|$. In (b), $\mathcal{B}_\varpi(G, \varphi, s)$ is the procedure Det of Lemma 10.3 for Φ_{wCDS} on (G, φ, ϖ) with $\mathcal{A}_\varphi$, its running-time exponent $|\mathrm{T}(J_X)|$ and $(\alpha, \Xi) = (n-1, n-2)$: by Lemma D.11.5, Theorem 9.10 holds for $\mathcal{A}_\varphi$ with the soundness statement "only if Φ_{wCDS} has a solution," so Lemma 10.3, whose proof uses $\mathcal{A}_\varphi$ only through Theorem 9.10 and Lemma 10.2, holds for $\mathcal{A}_\varphi$ with the same change; its enumeration tests membership in Φ_{wCDS}, that is, connected domination and weight at most ϖ, in polynomial time per set.

Property (B1). By Lemma D.11.5 with Theorem 3.5 in (a), and by Lemma 10.3 as just stated in (b), $\mathcal{B}_\varpi(G, \varphi, s)$ answers NO if Φ_{wCDS} has no solution on (G, φ, ϖ), that is, if G has no connected dominating set of weight at most ϖ.

Property (B2). Let S^* be a connected dominating set with $\varphi(S^*) = \gamma_{\mathrm{c}}^{\varphi}(G)$, and put $s^* = |S^*|$ and $\varpi^* = \varphi(S^*)$; S^* is a solution of Φ_{wCDS} on (G, φ, ϖ^*) of size exactly s^*. In (a), Proposition 7.8 gives condition $(\star)$ for S^* with the running-time exponent $|\mathrm{T}(J_X)|$, which does not involve the weights, so by Lemma D.11.5 with Theorem 3.5(iii) the run $\mathcal{B}_{\varpi^*}(G, \varphi, s^*)$ answers YES with probability at least $1 - e^{-\kappa}$. In (b), the certificate (19) for S^* is an integer certificate for $(n-1, n-2)$, so $\mathcal{B}_{\varpi^*}(G, \varphi, s^*)$ answers YES by Lemma 10.3.

Time and space. By Lemma D.11.5, each run in (a) has the time and space bounds of the corresponding run in Theorem 7.2, and each run in (b) those of Lemma 10.3, with $N^{O(1)} = (n + \varphi_{\max})^{O(1)}$. So Lemma D.11.6 gives the claims, with $p = 1 - e^{-\kappa}$ in (a) and $p = 1$ in (b), and a factor $(n\varphi_{\max} + 1)(n + 1)$ in the time. □

D.11.4 Weakly connected domination

Theorem 12.4 (Weakly Connected Dominating Set)**.** *For connected n-vertex graphs, $\gamma_{\mathrm{w}}(G)$ can be computed deterministically in time $O((\sqrt{5} - \frac{1}{2})^n\, 2^{O(n(\log\log n)^2/\log n)}) = O(1.7361^n)$ and polynomial space.*

The notation is that of Section 12.3. For $X \subseteq V$ the running-time exponent is

$$g_{\mathrm{w}}(X) = n - |N[X]| + \mathrm{comp}(G^2[X]), \tag{37}$$

the analogue of (13) with $G^2[X]$ in place of $G[X]$.

Lemma D.11.7 (Characterization)**.** *A set $S \subseteq V$ is a weakly connected dominating set of G if and only if S is dominating and $G^2[S]$ is connected.*

Proof. Let S be a weakly connected dominating set, and let $x, x' \in S$. The graph $\langle S\rangle_{\mathrm{w}}$ is connected, so it contains a path from x to x'. A vertex $v \notin S$ on this path is an end of its two path edges, each of which has an end in S; so both path neighbors of v lie in S. Hence any two consecutive vertices of S on the path are adjacent in G or have a common neighbor, that is, they are adjacent in $G^2[S]$, and x and x' lie in one component of $G^2[S]$.

Conversely, let S be dominating with $G^2[S]$ connected. Then $\langle S\rangle_{\mathrm{w}}$ has vertex set $N[S] = V$. Every $v \notin S$ has a neighbor $u \in S$, and uv is an edge of $\langle S\rangle_{\mathrm{w}}$. If $x, x' \in S$ are adjacent in $G^2[S]$, then xx' is an edge of $\langle S\rangle_{\mathrm{w}}$, or x and x' have a common neighbor u and xu, ux' are edges of $\langle S\rangle_{\mathrm{w}}$. So all of S lies in one component of $\langle S\rangle_{\mathrm{w}}$, and every other vertex is adjacent to S in it. □

By Lemma D.11.7 and the graph H_{w} of Lemma D.11.8 below, the weakly connected dominating sets of G are the connected red–blue dominating sets of H_{w}, so the algorithm of Abu-Khzam, Mouawad and Liedloff [AKML11] for Connected Red-Blue Dominating Set solves Weakly Connected Dominating Set in time $O^*(1.3645^{2n}) = O^*(1.8619^n)$ and polynomial space.

Lemma D.11.8 (Extension for weak connectivity)**.** *Let H_{w} be the red–blue graph on V with the edges $u^{\mathrm{b}}v^{\mathrm{b}}$ for all $u \neq v$ at distance at most 2 in G and the edges $u^{\mathrm{b}}v^{\mathrm{r}}$ for all u, v with $u \in N[v]$. The connected red–blue dominating sets of H_{w} are the weakly connected dominating sets of G, and $g_{H_{\mathrm{w}}} = g_{\mathrm{w}}$. So Lemma D.11.1 gives a deterministic extension algorithm for the implicit set system of weakly connected dominating sets, in time $2^{g_{\mathrm{w}}(X)}n^{O(1)}$ and polynomial space, with the running-time exponent $(G, X, k) \mapsto g_{\mathrm{w}}(X)$, which takes values in $\{0, \dots, n\}$.*

Proof. The graph H_{w} is built in polynomial time. For $S \subseteq V$ the graph $H_{\mathrm{w}}[S^{\mathrm{b}}]$ is a copy of $G^2[S]$, and v^{r} has a neighbor in S^{b} if and only if $N[v] \cap S \neq \emptyset$; so by Lemma D.11.7 the set S is connected red–blue dominating in H_{w} if and only if it is weakly connected dominating, and $g_{H_{\mathrm{w}}}(X) = n - |N[X]| + \mathrm{comp}(G^2[X]) = g_{\mathrm{w}}(X)$ by (37). Now Lemma D.11.1 applies, for the implicit set system of weakly connected dominating sets. □

The next lemma is the analogue of Lemma 5.10; unlike (19), it also holds for the empty sample.

Lemma D.11.9 (Pointwise certificate for weak connectivity)**.** *Let S be a weakly connected dominating set, let T be a spanning tree of $\langle S\rangle_{\mathrm{w}}$, rooted anywhere, and for $u \in S$ let $w(u)$ be $d_T(u)$ minus the number of children of u in T that lie in S. Then w takes nonnegative integer values, $w(S) = n - 1$, and*

$$g_{\mathrm{w}}(X) \;\leq\; n - w(X) \qquad \text{for every } X \subseteq S.$$

Proof. Every edge of $\langle S\rangle_{\mathrm{w}}$ has an end in S, so S is a vertex cover of T. Charge every edge of T, joining a vertex v to its parent p, to v if $v \in S$ and to p otherwise; in the second case $p \in S$, because S covers the edge. A vertex $u \in S$ is charged with the edge to its parent (if u is not the root) and with the edges to its children outside S, that is, with exactly $w(u) \geq 0$ edges. So $w(S) = |E(T)| = n - 1$. Let $E_T(X)$ be the set of edges of T with an end in X. An edge charged to a vertex of X has that vertex as an end, so $w(X) \leq |E_T(X)|$.

It remains to show $g_{\mathrm{w}}(X) \leq n - |E_T(X)|$. Let T_X be the graph with vertex set $N_T[X]$ and edge set $E_T(X)$; every edge of $E_T(X)$ has both ends in $N_T[X]$, and T_X is a forest, being a subgraph of T, so it has $|N_T[X]| - |E_T(X)|$ components. Let $x, x' \in X$ lie in one component of T_X, and take a path of T_X from x to x'. A vertex $v \notin X$ on this path is an end of its two path edges, each of which has an end in X; so both path neighbors of v lie in X. Hence consecutive vertices of X on the path are at distance at most 2 in T, and so in G, that is, they are adjacent in $G^2[X]$; thus x

and x' lie in one component of $G^2[X]$. Therefore $\mathrm{comp}(G^2[X]) \le \mathrm{comp}(T_X) = |N_T[X]| - |E_T(X)|$, and with $N_T[X] \subseteq N[X]$, as $T \subseteq G$,

$$\begin{aligned} g_{\mathrm{w}}(X) &= n - |N[X]| + \mathrm{comp}(G^2[X]) \le n - |N_T[X]| + |N_T[X]| - |E_T(X)| \\ &= n - |E_T(X)| \le n - w(X). \end{aligned}$$

□

Proof of Theorem 12.4. Run the procedure Det of Lemma 10.3 for the implicit set system of weakly connected dominating sets, the extension algorithm and running-time exponent g_{w} of Lemma D.11.8 and $(\alpha, \Xi) = (n, n-1)$, for $s = 1, \dots, n$, and output the least s at which it answers YES; V is a solution, and $\gamma_{\mathrm{w}}(G)$ is the least solution size. By Lemma 10.3, Det answers NO if $\gamma_{\mathrm{w}}(G) > s$. At $s = \gamma_{\mathrm{w}}(G)$ a minimum weakly connected dominating set S has size exactly s, and the weights of Lemma D.11.9 are an integer certificate for S and $(n, n-1)$: they are nonnegative integers with $w(S) = n - 1 = \Xi$ and $g_{\mathrm{w}}(X) \le \alpha - w(X)$ for every $X \subseteq S$. So Det answers YES at $s = \gamma_{\mathrm{w}}(G)$, and Lemma 5.2, in its deterministic form and with the bounds of Lemma 10.3, gives the theorem. □

Unlike for Connected Dominating Set, we do not expect the exponential moments of the running-time exponent g_{w} to give a smaller running time than its affine certificate: on the following instances the inequality of Lemma D.11.9 is an equality. For integers $s \ge 2$ and $d_1, \dots, d_s \ge 2$, the *spider* T is the tree with a center z adjacent to vertices $v_1, \dots, v_s$, where v_i carries $d_i - 1$ pendant vertices; so T has $n = 1 + \sum_i d_i$ vertices.

Proposition D.11.10 (Spiders)**.** *Let $s \ge 2$ and $d_1, \dots, d_s \ge 2$ be integers, let T be the spider with these parameters, and let $S = \{v_1, \dots, v_s\}$. Then:*

(a) S is the unique minimum weakly connected dominating set of T.

(b) $g_{\mathrm{w}}(X) = n - \sum_{v_i \in X} d_i$ for every $X \subseteq S$, so the generating function of Definition 3.3 for the running-time exponent g_{w} is $W_S(y) = 2^{-n} \prod_{i=1}^{s} (1 + y 2^{d_i})$.

Proof. (a) The sets $\{v_i\} \cup \mathrm{pend}(v_i)$, where $\mathrm{pend}(v_i)$ is the set of pendant vertices of v_i, are disjoint, and each meets every weakly connected dominating set, since a pendant vertex must be dominated. So a set of size at most s has exactly one vertex in each of them and does not contain z. If such a set S' contains a pendant vertex of v_i instead of v_i, then no edge at v_i other than those to $\mathrm{pend}(v_i)$ has an end in S', so $\{v_i\} \cup \mathrm{pend}(v_i)$ is a union of components of $\langle S' \rangle_{\mathrm{w}}$ that misses z, and $\langle S' \rangle_{\mathrm{w}}$ is disconnected. Hence $S' = S$. And S dominates T and every edge of T has an end in S, so $\langle S \rangle_{\mathrm{w}} = T$ is connected.

(b) For $X = \emptyset$ both sides equal n. For nonempty $X \subseteq S$, $N[X]$ consists of z and the sets $\{v_i\} \cup \mathrm{pend}(v_i)$ with $v_i \in X$, so $|N[X]| = 1 + \sum_{v_i \in X} d_i$, and $G^2[X]$ is connected, as any two vertices of X have the common neighbor z. Now (37) gives the formula, and the product form of W_S follows by expanding the product over the subsets of S. □

By Proposition D.11.10(b), on spiders g_{w} equals the integer affine surrogate exponent $n - w(X)$ of Lemma D.11.9 for T rooted at z, whose weights $d_i \in \{2, 3\}$ form the worst integer profile of Lemma 9.7 when $s/\Xi \in [\frac{1}{3}, \frac{1}{2}]$. Let β_s be the largest value of $\left(\inf_{y>0} (1+y)^n / W_S(y)\right)^{1/n}$ over $d_1, \dots, d_s \in \{2, 3\}$. Numerical computation (Appendix F.5) gives values of β_s that appear to decrease toward $\sqrt{5} - \frac{1}{2}$; this is not proved. By Proposition D.11.10(a), Lemma 3.9 and Proposition 3.10, if β_s tends to $\sqrt{5} - \frac{1}{2}$, then Algorithm MLS-θ with the running-time exponent g_{w} needs time $(\sqrt{5} - \frac{1}{2})^{n - o(n)}$ on spiders and cannot improve on Theorem 12.4 by more than a subexponential factor.

D.11.5 Directed Max Leaf Spanning Tree

Theorem 12.5 (Directed Max Leaf Spanning Tree)**.** *For digraphs with n vertices, the maximum number of leaves of an out-branching (or the fact that there is none) can be computed deterministically in time $O((\sqrt{5}-\frac{1}{2})^n\, 2^{O(n(\log\log n)^2/\log n)}) = O(1.7361^n)$ and polynomial space.*

The notation is that of Section 12.4. Write $N^-(v)$ for the set of in-neighbors of v, and $\mathrm{ch}_F(x)$ for the set of children of x in an out-branching F.

Lemma D.11.11 (Internal sets)**.** *Let $r \in V(\vec{G})$. Then:*

(a) The set of non-leaves of an out-branching rooted at r is r-internal.

(b) From an r-internal set S one computes in polynomial time an out-branching rooted at r in which every vertex outside S is a leaf.

(c) Every superset of an r-internal set is r-internal.

Hence the maximum number of leaves of an out-branching is $n - \min|S|$, the minimum over all r and all r-internal sets S, and an out-branching rooted at r exists if and only if $V(\vec{G})$ is r-internal.

Proof. (a) Let F be an out-branching rooted at r and S its set of non-leaves. As $n \geq 2$, r has a child, so $r \in S$. For $v \in S$, every vertex of the path of F from r to v has a child or is v, so the path lies in $\vec{G}[S]$. A leaf v is not r, and its parent is a non-leaf and an in-neighbor of v. (b) Take a breadth-first search out-tree of $\vec{G}[S]$ from r, which spans S, and attach every $v \notin S$ by an arc from an in-neighbor in S. (c) If $S \subseteq S'$, then a vertex of $S' \setminus S$ has an in-neighbor in S, which is reachable from r in $\vec{G}[S] \subseteq \vec{G}[S']$; the other conditions are inherited. The final statements follow from (a) and (b), and from (c) applied to $S' = V(\vec{G})$. □

For a fixed root r, let Φ_r be the implicit set system whose instances are pairs $(\vec{G}, r)$, with universe $V(\vec{G})$ and the r-internal sets as solutions. For $X \subseteq V(\vec{G})$ let

$$\mathrm{und}_r(X) = \{v \notin X \cup \{r\} : N^-(v) \cap X = \emptyset\},$$

let $\mathrm{Src}_r(X)$ be the set of strong components C of $\vec{G}[X]$ that are *sources* (no arc of $\vec{G}[X]$ enters C from $X \setminus C$) and do not contain r, and put

$$g_r(X) = |\,\mathrm{und}_r(X)| + |\mathrm{Src}_r(X)|.$$

The members of $\mathrm{Src}_r(X)$ are disjoint, and each contains a vertex of $X \setminus \{r\}$, so $g_r(X) \leq (n - |X \cup \{r\}|) + |X \setminus \{r\}| = n - 1$; also $g_r(\emptyset) = n - 1$, and $g_r(S) = 0$ for every r-internal set S, because $\mathrm{und}_r(S) = \emptyset$ and the strong component of r is the only source of $\vec{G}[S]$.

The extension problem for Φ_r is an instance of Directed Steiner Tree, on a digraph with a blue and a red copy of every vertex, and Lemma 12.7 solves it. Let $\vec{G}^r$ be the digraph with a vertex v^{b} for every $v \in V(\vec{G})$ and a vertex v^{r} for every $v \neq r$, with the arcs $u^{\mathrm{b}}v^{\mathrm{b}}$ for every arc uv of $\vec{G}$, the arcs $u^{\mathrm{b}}v^{\mathrm{r}}$ for every arc uv of $\vec{G}$ with $v \neq r$, and the arcs $v^{\mathrm{b}}v^{\mathrm{r}}$ for $v \neq r$. We consider the instance $(\vec{G}^r, r^{\mathrm{b}}, K)$ of Directed Steiner Tree with root r^{b} and terminals $K = \{v^{\mathrm{r}} : v \neq r\}$; its non-terminals are $U = \{v^{\mathrm{b}} : v \neq r\}$, and we write g_{DST} for the running-time exponent g of Section 12.5 on this instance. Red vertices have no outgoing arcs. So for $S \subseteq V(\vec{G})$ with $r \in S$, in the digraph $\vec{G}^r[\{r^{\mathrm{b}}\} \cup K \cup (S \setminus \{r\})^{\mathrm{b}}]$ the blue vertices reachable from r^{b} are the copies of the vertices reachable from r in $\vec{G}[S]$, and a red vertex v^{r} is reachable if and only if one of its in-neighbors in this digraph, which are v^{b} if $v \in S$ and u^{b} for $u \in S \cap N^-(v)$, is reachable. Hence $(S \setminus \{r\})^{\mathrm{b}}$ is a rooted Steiner set of $(\vec{G}^r, r^{\mathrm{b}}, K)$ if and only if S is r-internal: the blue vertices of $S \setminus \{r\}$ are reachable if and only if every vertex of S is reachable from r in $\vec{G}[S]$, and then a red vertex v^{r} is reachable if and only if $v \in S$ or v has an in-neighbor in S.

Lemma D.11.12 (Directed extension)**.** *For every $X \subseteq V(\vec{G})$ and integer $k \geq 0$ one can decide in time $2^{g_r(X)}n^{O(1)}$ and polynomial space, deterministically, whether some $Y \subseteq V(\vec{G}) \setminus X$ with $|Y| \leq k - |X|$ makes $X \cup Y$ an r-internal set. So $((\vec{G}, r), X, k) \mapsto g_r(X)$ is a running-time exponent (Definition 3.2) for Φ_r, with values in $\{0, \ldots, n-1\}$.*

Proof. Put $X' = (X \setminus \{r\})^{\mathrm{b}} \subseteq U$ and $k' = k - |X| - [r \notin X]$; the root r belongs to every r-internal set, so if $r \notin X$ it must be added, using one of the $k - |X|$ further elements. If $k' < 0$, reject. Otherwise run the algorithm of Lemma 12.7 on the instance $(\vec{G}^{\,r}, r^{\mathrm{b}}, K)$ with the set X' and the size bound $|X'| + k'$, and return its answer. This is correct: the sets $Y \subseteq V(\vec{G}) \setminus X$ with $|Y| \leq k - |X|$ for which $X \cup Y$ is r-internal correspond, by $Y \mapsto Y' = (Y \setminus \{r\})^{\mathrm{b}}$, to the sets $Y' \subseteq U \setminus X'$ with $|Y'| \leq k'$ for which $X' \cup Y'$ is a rooted Steiner set. Indeed, if $X \cup Y$ is r-internal, then $r \in X \cup Y$, so $r \in Y$ exactly when $r \notin X$, $|Y'| = |Y| - [r \notin X]$, and $(X \cup Y \setminus \{r\})^{\mathrm{b}} = X' \cup Y'$ is rooted by the equivalence above; conversely, for $Y' \subseteq U \setminus X'$ put $Y = \{v : v^{\mathrm{b}} \in Y'\} \cup (\{r\} \setminus X)$, so that $|Y| = |Y'| + [r \notin X]$, $r \in X \cup Y$ and $(X \cup Y \setminus \{r\})^{\mathrm{b}} = X' \cup Y'$, and $X \cup Y$ is r-internal when $X' \cup Y'$ is rooted.

The algorithm of Lemma 12.7 runs in time $2^{g_{\mathrm{DST}}(X')}n^{O(1)}$ and polynomial space, as $\vec{G}^{\,r}$ has $2n - 1$ vertices. We show $g_{\mathrm{DST}}(X') \leq g_r(X)$. Let $W = \{r^{\mathrm{b}}\} \cup K \cup X'$, whose blue vertices are the copies of $X \cup \{r\}$, and let W^r be the set of vertices reachable from r^{b} in $\vec{G}^{\,r}[W]$. Let A be the set of vertices reachable from r in $\vec{G}[X \cup \{r\}]$; since red vertices have no outgoing arcs, the blue vertices of W^r are exactly A^{b}, and the blue vertices of $W \setminus W^r$ are $(X \setminus A)^{\mathrm{b}}$, which induce a copy of $\vec{G}[X \setminus A]$. Let C be a source component of $\vec{G}^{\,r}[W \setminus W^r]$; we assign to it an element of $\mathrm{und}_r(X)$ or of $\mathrm{Src}_r(X)$, injectively.

If C contains a red vertex v^{r}, then $C = \{v^{\mathrm{r}}\}$, because every vertex of a strong component with at least two vertices has an outgoing arc inside it. Let u^{b} be an in-neighbor of v^{r} in $\vec{G}^{\,r}$ with $u \in X \cup \{r\}$, so that $u^{\mathrm{b}} \in W$. If $u^{\mathrm{b}} \in W^r$, then $v^{\mathrm{r}} \in W^r$, and otherwise the arc $u^{\mathrm{b}}v^{\mathrm{r}}$ enters C from $(W \setminus W^r) \setminus C$; both contradict the choice of C. So no in-neighbor of v^{r} has its underlying vertex in $X \cup \{r\}$: $v \notin X$ (as v^{b} is an in-neighbor), $v \neq r$, and $N^-(v) \cap X = \emptyset$. Hence $v \in \mathrm{und}_r(X)$.

Otherwise C consists of blue vertices, $C = D^{\mathrm{b}}$ with $D \subseteq X \setminus A$, and D is strongly connected in $\vec{G}[X \setminus A]$. No arc of $\vec{G}[X]$ enters D from $X \setminus D$: an arc from $X \setminus (A \cup D)$ would enter C from $(W \setminus W^r) \setminus C$ in $\vec{G}^{\,r}[W \setminus W^r]$, and an arc from a vertex of A to a vertex $d \in D$ would make d reachable from r in $\vec{G}[X \cup \{r\}]$, that is, $d \in A$. So D is a strong component of $\vec{G}[X]$ (a vertex of $X \setminus D$ in the same strong component would have a path to D in $\vec{G}[X]$, whose last arc enters D), it is a source, and $r \notin D$ because $r \in A$. Hence $D \in \mathrm{Src}_r(X)$.

Distinct source components give distinct elements, so $g_{\mathrm{DST}}(X') \leq |\,\mathrm{und}_r(X)| + |\mathrm{Src}_r(X)| = g_r(X)$. The value $g_r(X)$, and membership in Φ_r, are computed in polynomial time, and the range was shown before the lemma. □

The proof of the certificate is the directed analogue of Lemma D.11.14: the source components of $\vec{G}[X]$ are bounded by the vertices of X whose parent is not in X.

Lemma D.11.13 (Pointwise certificate for out-branchings)**.** *Let S be an r-internal set, and let F be an out-branching rooted at r in which every vertex outside S is a leaf (Lemma D.11.11(b)). Then for every $X \subseteq S$,*

$$g_r(X) \;\leq\; n - 1 - \sum_{x \in X} |\mathrm{ch}_F(x)|.$$

Consequently the weights $w(x) = |\mathrm{ch}_F(x)| - [x = r]$ on S are nonnegative integers with $w(S) = n-2$, and $g_r(X) \leq n - 1 - w(X)$ for every $X \subseteq S$; this is the certificate (19)*.*

Proof. Let Q be the set of vertices $v \neq r$ whose parent in F is not in X. Every vertex other than r has exactly one parent, and the children sets $\mathrm{ch}_F(x)$ are disjoint, so $|Q| = n - 1 - \sum_{x \in X} |\mathrm{ch}_F(x)|$.

We show that $\mathrm{und}_r(X) \subseteq Q \setminus X$ and that $\mathrm{Src}_r(X)$ maps injectively into $Q \cap X$; adding the two bounds gives $g_r(X) \le |Q|$.

A vertex $v \in \mathrm{und}_r(X)$ is not r and not in X, and its parent is an in-neighbor of v, so the parent is not in X; hence $v \in Q \setminus X$. Let $C \in \mathrm{Src}_r(X)$, let $x \in C$, and let u be the ancestor of x in F closest to r such that the path of F from u to x lies in X. This path is a path of $\vec{G}[X]$ ending in C, and no arc of $\vec{G}[X]$ enters the source component C from outside, so $u \in C$. As $r \notin C$, $u \neq r$, and by the choice of u its parent is not in X; so $u \in Q \cap X \cap C$. As the members of $\mathrm{Src}_r(X)$ are disjoint, choosing such a u for each of them is injective.

Every vertex other than r has a parent, which is a non-leaf and so lies in S; hence $\sum_{x \in S} |\mathrm{ch}_F(x)| = n - 1$ and $w(S) = n - 2$. As $n \ge 2$, r has a child, so $w \ge 0$, and $w(X) \le \sum_{x \in X} |\mathrm{ch}_F(x)|$ gives $g_r(X) \le n - 1 - w(X)$. □

Proof of Theorem 12.5. For $n = 1$ the answer is 1; let $n \ge 2$. For every r for which $V(\vec{G})$ is r-internal (by Lemma D.11.11, the other roots carry no out-branching), compute the least size s_r of an r-internal set, and output $n - \min_r s_r$; if there is no such r, report that there is no out-branching. This is correct by Lemma D.11.11.

To compute s_r, run the procedure Det of Lemma 10.3 for Φ_r, the extension algorithm and running-time exponent g_r of Lemma D.11.12 and $(\alpha, \Xi) = (n-1, n-2)$, for $s = 1, \ldots, n$, and let s_r be the least s at which it answers YES; $V(\vec{G})$ is a solution of Φ_r, and s_r is the least solution size. By Lemma 10.3, Det answers NO if $s_r > s$. At $s = s_r$ a minimum r-internal set S has size exactly s, and Lemma D.11.13 gives its integer certificate for $(n-1, n-2)$, so Det answers YES. Hence Lemma 5.2, in its deterministic form, computes s_r within the time and space bounds of Lemma 10.3, and running the computation for each of the at most n roots multiplies the running time by at most n. □

D.11.6 Directed Steiner Tree

This subsection proves Theorem 12.8, with the notation of Section 12.5. Fix an inclusion-minimal rooted Steiner set S, put $s = |S|$ and $p = |K|$, and let A be an out-arborescence of $\vec{G}[\{r\} \cup K \cup S]$ rooted at r; it exists because S is rooted. By minimality every leaf of A other than r is a terminal: if a vertex $v \in S$ were a leaf, $A - v$ would show that $S \setminus \{v\}$ is rooted. For $u \neq r$ write $\mathrm{par}(u)$ for the parent of u in A. The trees of the forest $A[\bar{K}_X]$ that do not contain r bound the running-time exponent $g(X)$ of a sample $X \subseteq S$ from above.

Lemma D.11.14 (Pointwise bound)**.** *For every $X \subseteq S$,*

$$g(X) \;\le\; \Psi_A(X) := \big|\{u \in K \cup X : \; \mathrm{par}(u) \notin \bar{K}_X\}\big|.$$

Equality holds if $\vec{G}[\bar{K}_X] = A[\bar{K}_X]$.

Proof. The forest $A[\bar{K}_X]$ consists of out-trees; the tree containing r lies in $\bar{K}^r_X$, and every other tree has a root $u \in K \cup X$ with $\mathrm{par}(u) \notin \bar{K}_X$. So at most $\Psi_A(X)$ trees miss r. A tree whose root lies in $\bar{K}^r_X$ lies entirely in $\bar{K}^r_X$, as its arcs are arcs of $\vec{G}[\bar{K}_X]$. So every $v \in \bar{K}_X \setminus \bar{K}^r_X$ lies in a tree with root $u \in \bar{K}_X \setminus \bar{K}^r_X$, and the tree path from u to v avoids $\bar{K}^r_X$; hence v is reachable from u within $\vec{G}[\bar{K}_X \setminus \bar{K}^r_X]$. If v lies in a source component C, then so does u, since nothing outside C reaches C. Assigning to each source component the root of the tree of one of its vertices is therefore injective, and $g(X) \le \Psi_A(X)$. If $\vec{G}[\bar{K}_X] = A[\bar{K}_X]$, then $\vec{G}[\bar{K}_X \setminus \bar{K}^r_X]$ is a forest of out-trees whose source components are exactly the roots counted by Ψ_A. □

Consequently, with the quantities of Definition 3.3 for the universe U,

$$W_S(y) = \sum_{X \subseteq S} y^{|X|} 2^{-g(X)} \;\geq\; W_A(y) := \sum_{X \subseteq S} y^{|X|} 2^{-\Psi_A(X)} \qquad (y \geq 0). \tag{38}$$

Whether a vertex u is counted in $\Psi_A(X)$ depends only on whether u and $\mathrm{par}(u)$ lie in $\bar{K}_X$. This locality lets us compute W_A by a recursion along A (Lemma D.11.15) with two states: a vertex lies in $\bar{K}_X$ (a terminal, the root, or a sampled vertex), or it is an unsampled vertex of S. For $x, x' \in \mathbb{R}^2$ let

$$x \circ x' = \Big(x_1(x'_1 + x'_2),\; x_2(\tfrac{x'_1}{2} + x'_2) \Big),$$

a bilinear map with nonnegative coefficients. For a vertex v of A let A_v be the subtree of v and its descendants. For $y \geq 0$ we define vectors $\mathrm{msg}_v \in \mathbb{R}^2$ for the vertices v of A, from the leaves up. Put $\mathrm{Init}_v = (1,0)$ if $v \in K \cup \{r\}$ and $\mathrm{Init}_v = (y,1)$ if $v \in S$, and for v with children $o_1, \ldots, o_\ell$, in any order, let $\mathrm{msg}_v = (\cdots(\mathrm{Init}_v \circ \mathrm{msg}_{o_1}) \circ \cdots) \circ \mathrm{msg}_{o_\ell}$.

Lemma D.11.15 (Two-state recursion)**.** *Let $y \geq 0$, and let the vectors msg_v be defined with this y and any order of the children. Then $W_A(y) = (\mathrm{msg}_r)_1$.*

Proof. For a vertex v and $X \subseteq S \cap V(A_v)$ put $\Psi_v(X) = |\{u \in V(A_v) \setminus \{v\} : u \in K \cup X,\ \mathrm{par}(u) \notin \bar{K}_X\}|$; this depends only on X, as $\mathrm{par}(u) \in V(A_v)$ for these u. Say that X puts v in state 1 if $v \in \bar{K}_X$ and in state 2 otherwise. We show by induction that $(\mathrm{msg}_v)_i$ is the sum of $y^{|X|}2^{-\Psi_v(X)}$ over the sets $X \subseteq S \cap V(A_v)$ that put v in state i; more precisely, that the same holds after the first j children, for the subtree A_v^j formed by v and the subtrees of $o_1, \ldots, o_j$, and for Ψ_v^j defined like Ψ_v on A_v^j.

For $j = 0$ the subtree is $\{v\}$ and $\Psi_v^0 = 0$. If $v \in K \cup \{r\}$, the only set is $\emptyset$, in state 1, giving $(1,0)$; if $v \in S$, the set $\{v\}$ is in state 1 with weight y and $\emptyset$ in state 2 with weight 1, giving $(y,1)$. For the step to j, write $o = o_j$ and split $X = X' \cup X_o$ with $X' \subseteq V(A_v^{j-1})$ and $X_o \subseteq V(A_o)$. Then $\Psi_v^j(X) = \Psi_v^{j-1}(X') + \Psi_o(X_o) + [o \in \bar{K}_X \text{ and } v \notin \bar{K}_X]$, because o is the only vertex counted in Ψ_v^j whose parent lies in the other part. Summing the weights, a state-1 vertex v receives the factor $(\mathrm{msg}_o)_1 + (\mathrm{msg}_o)_2$ and a state-2 vertex v the factor $\frac{1}{2}(\mathrm{msg}_o)_1 + (\mathrm{msg}_o)_2$, which is the product $\circ$. Finally $r \in \bar{K}_X$ always and r is not counted, so $W_A(y) = \sum_X y^{|X|}2^{-\Psi_r(X)} = (\mathrm{msg}_r)_1$. □

Antinorm certificates. As for RS-θ (Lemma D.4.2), a region of the plane that $\circ$ maps into itself, up to known scale factors, bounds W_A from below for all arborescences at once. With two states the region lies in the plane, and the cone added to the convex hull can be the nonnegative quadrant. For a finite set $Z \subseteq \mathbb{R}^2_{\geq 0}$ whose points have positive first coordinate, let $\mathcal{R}_Z = \mathrm{conv}(Z) + \mathbb{R}^2_{\geq 0}$ and $c_Z = \min_{z \in Z} z_1$. Minimality of S enters through the vertices of S: since none of them is a leaf of A, their initial vector $\mathrm{Init}_v = (y,1)$ is never the vector msg_v of a complete subtree, and it is needed only as a left factor of $\circ$ (condition (C3) below).

Lemma D.11.16 (Certificate)**.** *Let $y \geq 0$, $e_1, e_2 > 0$, and let $Z \subseteq \mathbb{R}^2_{\geq 0}$ be a finite set whose points have positive first coordinate, with*

(C1) *$z \circ z' \in \mathcal{R}_Z$ for all $z, z' \in Z$;*

(C2) *$(1,0) \in e_2 \mathcal{R}_Z$;*

(C3) *$(y,1) \circ z \in e_1 \mathcal{R}_Z$ for all $z \in Z$.*

Then $W_A(y) \geq c_Z\, e_1^s e_2^{p+1}$ for every out-arborescence A as above, that is, rooted at r, with s vertices in S, p terminals, and no leaf in S.

Proof. First, if $x \in \eta\mathcal{R}_Z$ and $x' \in \eta'\mathcal{R}_Z$ with $\eta, \eta' > 0$, then $x \circ x' \in \eta\eta'\mathcal{R}_Z$, and $(y,1) \circ x' \in e_1\eta'\mathcal{R}_Z$. Indeed, write $x = \eta(z+q)$ and $x' = \eta'(z'+q')$ with $z, z' \in \mathrm{conv}(Z)$ and $q, q' \in \mathbb{R}^2_{\geq 0}$. By bilinearity, $x \circ x' = \eta\eta'(z \circ z' + z \circ q' + q \circ (z'+q'))$. Writing z and z' as convex combinations of points of Z shows that $z \circ z'$ is a convex combination of products of points of Z, so it lies in the convex set $\mathcal{R}_Z$ by (C1); the other two terms are nonnegative, because $\circ$ maps pairs of nonnegative vectors to nonnegative vectors, and $\mathcal{R}_Z + \mathbb{R}^2_{\geq 0} = \mathcal{R}_Z$. Likewise $(y,1) \circ x' = \eta'((y,1) \circ z' + (y,1) \circ q')$ with $(y,1) \circ z' \in e_1\mathcal{R}_Z$ by (C3) and convexity, and $(y,1) \circ q' \geq 0$.

Let s_v and p_v be the numbers of vertices of A_v in S and in $K \cup \{r\}$. We show $\mathrm{msg}_v \in e_1^{s_v} e_2^{p_v} \mathcal{R}_Z$ by induction from the leaves. If $v \in K \cup \{r\}$, then $\mathrm{Init}_v = (1,0) \in e_2\mathcal{R}_Z$ by (C2), and each child o multiplies the scale by that of msg_o, by the first claim. If $v \in S$, then v is not a leaf; with its first child o_1, the second claim gives $\mathrm{Init}_v \circ \mathrm{msg}_{o_1} \in e_1 e_1^{s_{o_1}} e_2^{p_{o_1}} \mathcal{R}_Z$, and the remaining children are handled by the first claim. At the root, $s_r = s$ and $p_r = p+1$. Every point of $\mathcal{R}_Z$ has first coordinate at least c_Z, so Lemma D.11.15 gives $W_A(y) = (\mathrm{msg}_r)_1 \geq c_Z e_1^s e_2^{p+1}$. □

Proposition D.11.17 (Certificates for Directed Steiner Tree)**.** *The certificates of Appendix F.7 are three rational certificates $(y_j, e_{1,j}, e_{2,j}, Z_j)$, $j = 0,1,2$, each of which satisfies* (C1)–(C3) *with $y = y_j$, $e_1 = e_{1,j}$, $e_2 = e_{2,j}$ and $Z = Z_j$, and rational weights $\lambda_j \geq 0$ with $\sum_j \lambda_j = 1$. With the exponent vectors $\boldsymbol{\alpha}_j = (\alpha_{j,1}, \alpha_{j,2}, \alpha_{j,3}) = \big(\log(1+y_j) - \log e_{1,j},\ -\log e_{2,j},\ \log(1+y_j)\big)$, the data are, approximately,*

j	$\lambda_j \approx$	$y_j \approx$	$e_{1,j} \approx$	$e_{2,j} \approx$	$\lvert Z_j\rvert$	$c_{Z_j} e_{2,j} \geq$	$(\alpha_{j,1}, \alpha_{j,2}, \alpha_{j,3}) \approx$
0	0.7458	0.45494	1.003883	0.78	27	0.40	(0.535367, 0.358454, 0.540958)
1	0.0858	0.48223	1.041122	0.78	10	0.41	(0.509635, 0.358454, 0.567774)
2	0.1684	0.02	0.963015	0.52	28	0.011	(0.082938, 0.943416, 0.028569)

and

$$\max_{i \in \{1,2,3\}} \sum_{j=0}^{2} \lambda_j \alpha_{j,i} \;=\; 0.4569669\ldots \;<\; 0.457.$$

Proof. The hypotheses (C1)–(C3), the displayed maximum, which lies more than $3.3 \cdot 10^{-5}$ below 0.457, and the inequality $0.457 < \log 1.3727$ are checked in Appendix F.7. □

Theorem 12.8 (Directed Steiner Tree)**.** Directed Steiner Tree *on digraphs with n vertices can be solved by a randomized algorithm with one-sided error in time $2^{0.457n} n^{O(1)} = O(1.3727^n)$ and polynomial space.*

Proof. Let $\psi = 0.457$; we may assume $n_U \geq 1$. Run Algorithm MLS-θ for Φ_{DST}, the extension algorithm of Lemma 12.7 and the running-time exponent g, with universe size n_U, rate $\psi n / n_U$, $\Gamma = 100$ and confidence κ, for $k = 0, 1, \ldots, n_U$, and output the least accepted k, or report that no Steiner set exists if no k is accepted. By Theorem 3.5 and Remark 3.6, this takes time $2^{\psi n} n^{O(1)}$ and polynomial space, and no accepted k is below the optimum; in particular, no k is accepted if there is no Steiner set. Let S be a minimum Steiner set; it is rooted and inclusion-minimal, so for $k = |S|$ it remains to verify condition $(\star)$. Let $r_0 = n_U - s$, so that $s + p + r_0 = n - 1$. For each j, Equation (38) and Lemma D.11.16 and the first inequality of Lemma 3.9 give

$$\max_{0 \leq t \leq s} \frac{\mu_t(S)}{\binom{n_U}{t}} \;\geq\; \frac{W_S(y_j)}{(1+y_j)^{n_U}} \;\geq\; c_{Z_j} e_{2,j}\, \frac{e_{1,j}^s e_{2,j}^p}{(1+y_j)^{s+r_0}} \;=\; c_{Z_j} e_{2,j}\, 2^{-(\alpha_{j,1} s + \alpha_{j,2} p + \alpha_{j,3} r_0)}.$$

Choose j with the least exponent. By Proposition D.11.17, $\alpha_{j,1} s + \alpha_{j,2} p + \alpha_{j,3} r_0 \leq \sum_i \lambda_i (\alpha_{i,1} s + \alpha_{i,2} p + \alpha_{i,3} r_0) \leq \psi(n-1)$, and $c_{Z_j} e_{2,j} \geq 1/100$. So $\max_t \mu_t(S) / \binom{n_U}{t} \geq 2^{-(\psi n / n_U) n_U} / \Gamma$, which is $(\star)$. Finally $2^{0.457} = 1.37268\ldots$. □

D.11.7 Paired Domination

This subsection proves Lemma 12.9, Theorem 12.10, and Proposition 12.11, with the notation of Section 12.6.

Lemma 12.9 (Tutte-matrix finishing)**.** *There is a randomized algorithm that, given G, weights $w\colon V \to \{0,\dots,w_{\max}\}$, a budget $\varpi \geq 0$, $X \subseteq V$ and $k \geq 0$, never accepts if no $Y \subseteq V \setminus X$ with $|Y| \leq k - |X|$ makes $X \cup Y$ a paired dominating set of weight at most ϖ, accepts with probability at least $\frac{1}{2}$ otherwise, and runs in time $2^{|\operatorname{und}(X)|}(n + w_{\max})^{O(1)}$ and polynomial space for every outcome of its coins. With $w \equiv 0$ and $\varpi = 0$ this is an algorithm for Φ_{PD}-Extension with the same error bounds, in time $2^{|\operatorname{und}(X)|}n^{O(1)}$.*

Proof. Fix a target size $s \in \{|X|,\dots,\min(n,k)\}$ and a target weight

$$b \in \{0,\dots,\lfloor \min(\varpi, w(V)) \rfloor\};$$

there are at most $(n+1)(nw_{\max}+1)$ such pairs. A set $A \supseteq X$ dominates G if and only if it dominates every vertex of $\operatorname{und}(X)$, that is, $A \cap N[u] \neq \emptyset$ for all $u \in \operatorname{und}(X)$. For $F \subseteq \operatorname{und}(X)$ let $V_F = V \setminus N[F]$, and consider the polynomial

$$\mathsf{F}_{s,b} = \sum_{F \subseteq \operatorname{und}(X)} (-1)^{|F|} \sum_{\substack{X \subseteq A \subseteq V_F \\ |A|=s,\ w(A)=b}} \det \mathbf{T}_A.$$

Exchanging the sums, the coefficient of $\det \mathbf{T}_A$, for $A \supseteq X$ with $|A| = s$ and $w(A) = b$, is

$$\sum_{F \subseteq \operatorname{und}(X)} (-1)^{|F|} \prod_{u \in F} [A \cap N[u] = \emptyset] = \prod_{u \in \operatorname{und}(X)} (1 - [A \cap N[u] = \emptyset]),$$

which is 1 if A dominates $\operatorname{und}(X)$ and 0 otherwise. So

$$\mathsf{F}_{s,b} = \sum_{A} \operatorname{Pf}(\mathbf{T}_A)^2,$$

summed over the dominating sets $A \supseteq X$ of size s and weight b.

No cancellation. Let M be a perfect matching of $G[A]$ for such an A. In $\operatorname{Pf}(\mathbf{T}_{A'})^2 = \sum_{M_1,M_2} \pm x_{M_1} x_{M_2}$, a product $x_{M_1}x_{M_2}$ equals x_M^2 only if every edge of M lies in both M_1 and M_2 and no other edge occurs, that is, $M_1 = M_2 = M$ and $A' = V(M) = A$; this term has coefficient $(\pm 1)^2 = 1$. So the coefficient of x_M^2 in $\mathsf{F}_{s,b}$ is exactly 1, and $\mathsf{F}_{s,b} \neq 0$ if and only if some dominating $A \supseteq X$ of size s and weight b has a perfect matching in $G[A]$, that is, if $A \setminus X$ is a completion of size $s - |X|$ with $w(A) = b$.

Evaluation. For $F \subseteq \operatorname{und}(X)$ we have $X \subseteq V_F$, because no vertex of F lies in $N[X]$. Let $\mathbf{E}$ be the diagonal matrix on V_F with $\mathbf{E}_{vv} = z^{w(v)}$ for $v \notin X$ and $\mathbf{E}_{vv} = 0$ for $v \in X$, with a further indeterminate z. Expanding the determinant along the diagonal, with one more indeterminate $x_\star$,

$$\det(\mathbf{T}_{V_F} + x_\star \mathbf{E}) = \sum_{Q \subseteq V_F \setminus X} x_\star^{|Q|} z^{w(Q)} \det \mathbf{T}_{V_F \setminus Q},$$

so the inner sum of $\mathsf{F}_{s,b}$ is the coefficient of $x_\star^{|V_F|-s} z^{w(V_F)-b}$. Choose a prime q with $4n + nw_{\max} \leq q \leq 8n + 2nw_{\max}$, by trial division, and substitute independent uniform elements of $\mathbb{F}_q$ for the x_e. The determinant is then a polynomial in $x_\star$ and z of degree at most n in $x_\star$ and at most $nw_{\max} < q$ in z, so evaluating it at $(n+1)(nw_{\max}+1)$ points and interpolating gives the coefficient over $\mathbb{F}_q$ in

polynomial time, and summing over F gives $\mathsf{F}_{s,b}$ at the random point, in time $2^{|\operatorname{und}(X)|}(n+w_{\max})^{O(1)}$ and polynomial space. Accept if the value is nonzero for some s and b.

If no completion of weight at most ϖ exists, every $\mathsf{F}_{s,b}$ with $b \le \varpi$ is the zero polynomial and the algorithm never accepts. If a completion of size $s - |X|$ and weight $b \le \varpi$ exists, then $\mathsf{F}_{s,b}$ is a nonzero polynomial of degree $s \le n$ in the x_e with a coefficient 1, so it is nonzero over $\mathbb{F}_q$, and by the Schwartz–Zippel lemma [Sch80, Zip79] it vanishes at the random point with probability at most $n/q \le \frac{1}{4}$. With $w \equiv 0$ and $\varpi = 0$ only $b = 0$ occurs, the determinant does not involve z, and the bounds are polynomial in n. □

Fix a paired dominating set S and a perfect matching M of $G[S]$, and choose an *anchor* $\operatorname{anc}(v) \in S \cap N(v)$ for every $v \notin S$. For $u \in S$ let $d_u = |\operatorname{anc}^{-1}(u)|$ be the number of vertices anchored at u, and for a pair $uv \in M$ let $n_{uv} = 2 + d_u + d_v$, the number of vertices the pair is responsible for; so $\sum_{uv \in M} n_{uv} = n$.

Lemma D.11.18 (Pointwise certificate)**.** *For every $X \subseteq S$,*

$$|\operatorname{und}(X)| \;\le\; \big|\{v \notin S : \operatorname{anc}(v) \notin X\}\big| + 2\,\big|\{uv \in M : u, v \notin X\}\big|,$$

and consequently, for every $y \ge 0$,

$$W_S(y) = \sum_{X \subseteq S} y^{|X|} 2^{-|\operatorname{und}(X)|} \;\ge\; \prod_{uv \in M} \Omega_{uv}(y), \qquad \Omega_{uv}(y) = y^2 + y\big(2^{-d_u} + 2^{-d_v}\big) + 2^{-n_{uv}}.$$

Proof. A vertex $v \notin S$ is dominated once $\operatorname{anc}(v) \in X$. A vertex $u \in S$ is dominated once u or its partner in M lies in X. This gives the first inequality. The right-hand side is a sum over the pairs of M of a term that depends only on $X \cap \{u, v\}$: it is 0 if $u, v \in X$, it is d_v if only $u \in X$, it is d_u if only $v \in X$, and it is n_{uv} if neither is in X. So $\sum_{X \subseteq S} y^{|X|} 2^{-(\text{right-hand side})}$ factorizes over the pairs into the $\Omega_{uv}(y)$. □

By the inequality of arithmetic and geometric means, $2^{-d_u} + 2^{-d_v} \ge 2 \cdot 2^{-(n_{uv}-2)/2}$, so $\Omega_{uv}(y) \ge y^2 + 4 \cdot 2^{-n_{uv}/2} y + 2^{-n_{uv}}$ for $y \ge 0$. The functions $h_{\bar n}$, the point y^* and the rate ψ_{pr} are those of Section 12.6.

Lemma D.11.19 (Rate)**.** *The functions $h_{\bar n}$ and the point y^* defined above satisfy*

$$\max_{\bar n \in \{2,3,\dots\}} h_{\bar n}(y^*) = h_6(y^*) = \psi_{\mathrm{pr}} = 0.7050097\ldots,$$

and for every graph G without isolated vertices and every paired dominating set S of G,

$$W_S(y^*) \;\ge\; (1 + y^*)^n\, 2^{-\psi_{\mathrm{pr}} n}.$$

Proof. For $\bar n \ge 41$ we have $y^2 + 4 \cdot 2^{-\bar n/2} y + 2^{-\bar n} \ge y^2$ and $0 < y^* < 1$, so $h_{\bar n}(y^*) \le \log(1 + y^*) + \frac{2}{41} \log \frac{1}{y^*} < 0.431$. For $2 \le \bar n \le 40$, $\bar n \ne 6$, direct evaluation (Appendix F.8) gives $h_6(y^*) - h_{\bar n}(y^*) > 0.00297$, the smallest gap occurring at $\bar n = 7$; it also gives $\log(1 + y^*) + \frac{2}{41} \log \frac{1}{y^*} < 0.431 < h_6(y^*)$, which completes the tail bound. This proves the first claim. By Lemma D.11.18 and the bound $\Omega_{uv}(y) \ge y^2 + 4 \cdot 2^{-n_{uv}/2} y + 2^{-n_{uv}}$,

$$\begin{aligned} \log \frac{(1 + y^*)^n}{W_S(y^*)} \;&\le\; \sum_{uv \in M} \Big(n_{uv} \log(1 + y^*) - \log \Omega_{uv}(y^*) \Big) \\ &\le\; \sum_{uv \in M} n_{uv}\, h_{n_{uv}}(y^*) \;\le\; \psi_{\mathrm{pr}} \sum_{uv \in M} n_{uv} = \psi_{\mathrm{pr}}\, n. \end{aligned}$$

□

Theorem 12.10 (Paired Domination)**.** *For graphs G with n vertices and no isolated vertex, $\gamma_{\mathrm{pr}}(G)$ can be computed by a randomized algorithm with one-sided error in time $O(1.6302^n)$ and polynomial space.*

Proof. Let $\psi = 0.70502$, a rational with $\psi_{\mathrm{pr}} < \psi$ and $2^\psi < 1.6302$. For $k = 2, 4, \ldots$ up to n, run Algorithm MLS-θ for Φ_{PD}, the extension algorithm of Lemma 12.9 (with $w \equiv 0$ and $\varpi = 0$), the running-time exponent $|\operatorname{und}(X)|$, rate ψ, $\Gamma = 1$ and confidence κ; as this extension algorithm has one-sided error, the cap and the trial counts are those of Corollary 3.15, that is, those of Theorem 3.5 for $2\Gamma = 2$. Output the least accepted k, or n if no k is accepted (as in Lemma 5.2). By Theorem 3.5 with Corollary 3.15, each run takes time $2^{\psi n} n^{O(1)}$ and polynomial space and rejects if no paired dominating set of size at most k exists. For $k = \gamma_{\mathrm{pr}}(G)$, a minimum paired dominating set satisfies the generating-function condition of Theorem 3.5 at $y = y^*$ with $\Gamma = 1$, by Lemma D.11.19, so by Corollary 3.15 this k is accepted with probability at least $1 - e^{-\kappa}$. □

Proposition 12.11 (Tightness)**.** *The only critical point of h_6 in $(0, \infty)$ is y^*, and $\min_{y \geq 0} h_6(y) = h_6(y^*)$. Let G_L be the disjoint union of L double stars, each an edge uv with two pendant vertices at u and two at v. Then the centers form the only paired dominating set S^* of G_L of size at most $2L$, and if Algorithm MLS-θ of Theorem 12.10 is run with a rate $\psi' < \psi_{\mathrm{pr}}$, it accepts $k = 2L$ with probability tending to 0 as $L \to \infty$.*

Proof. The derivative of h_6 vanishes exactly where $\frac{1}{1+y} = \frac{2y+1/2}{6(y^2+y/2+1/64)}$, that is, where $4y^2 + \frac{y}{2} - \frac{13}{32} = 0$; the only positive root is y^*. As $h_6(0) = 1$ and $h_6(y) \to \infty$ as $y \to \infty$, while $h_6(y^*) < 1$, the minimum is attained at y^*.

In G_L, a pendant vertex at u is dominated only by itself or by u; if it lies in a paired dominating set, its partner is its only neighbor u. So every paired dominating set contains all centers, and the centers form one. Hence S^* is the only one of size at most $2L$, and a trial for $k = 2L$ accepts only if its sample lies in S^*. With the anchors at the centers, Lemma D.11.18 holds with equality, and $W_{S^*}(y) = (y^2 + \frac{y}{2} + \frac{1}{64})^L$, so $(1+y)^n / W_{S^*}(y) = 2^{n\, h_6(y)} \geq 2^{n\, \psi_{\mathrm{pr}}}$ for all $y \geq 0$, with $n = 6L$. By the second inequality of Lemma 3.9, $\max_t \mu_t(S^*)/\binom{n}{t} \leq (n+1)2^{-\psi_{\mathrm{pr}} n}$, and by Proposition 3.10 the expected number of trials that run the extension on a subset of S^*, and so the probability of acceptance, is at most $n^{O(1)} 2^{(\psi' - \psi_{\mathrm{pr}})n} \to 0$. □

D.11.8 Vertex weights for Directed Steiner Tree and Paired Domination

The two weighted results stated in Sections 12.5 and 12.6 follow from the unweighted proofs through the budget loop of Lemma D.11.6, with the notation of those subsections.

The proof of Theorem 12.8 uses only two properties of S, that S is rooted and that A has no leaf in S, and not that $|S|$ is least.

Corollary D.11.20 (Vertex-weighted Directed Steiner Tree)**.** *For non-terminal weights $w \colon U \to \{0, 1, \ldots, w_{\max}\}$, written in unary, the least weight of a Steiner set can be computed by a randomized algorithm with one-sided error in time $2^{0.457n}(n + w_{\max})^{O(1)}$ and polynomial space.*

Proof. We apply Lemma D.11.6 to Φ_{DST} with the weights w on its universe U; so Φ_ϖ has as solutions the rooted Steiner sets of weight at most ϖ, and its least solution weight is the least weight of a Steiner set, because the vertices of a Steiner set that are reachable from r form a rooted Steiner set of no larger weight. The extension algorithm $\mathcal{A}_w$ computes the least weight OPT of a completion of X by Lemma 12.7 and accepts if and only if $w(X) + \mathrm{OPT} \leq \varpi$; it ignores k. A witness Y in (R1) gives $w(X) + \mathrm{OPT} \leq w(X \cup Y) \leq \varpi$, and an acceptance yields a rooted Steiner set of weight at most ϖ; so $\mathcal{A}_w$ satisfies (R1) and (R2) of Lemma D.11.5, and g is a running-time

exponent for it, with $N^{O(1)} = (n + w_{\max})^{O(1)}$. Let $\mathcal{B}_\varpi(I, k)$ be Algorithm MLS-θ for Φ_ϖ with $\mathcal{A}_w$ and g, with the parameters of the proof of Theorem 12.8. By Lemma D.11.5 with Theorem 3.5, it runs in time $2^{0.457n}(n + w_{\max})^{O(1)}$ and polynomial space, and it answers NO if Φ_ϖ has no solution, which is property (B1). For property (B2), let S be a Steiner set of least weight ϖ^* that is inclusion-minimal among those; it is rooted, and A has no leaf in S, since deleting such a leaf would keep a Steiner set of no larger weight. The proof of Theorem 12.8 uses only these two properties of S to verify condition $(\star)$ at $k = |S|$, so $\mathcal{B}_{\varpi^*}(I, |S|)$ answers YES with probability at least $1 - e^{-\kappa}$. Now Lemma D.11.6 gives the claim, with $w(U) + 1 \le n_U w_{\max} + 1$ budgets. □

For unit weights this is Theorem 12.8. We know of no algorithm for vertex-weighted DIRECTED STEINER TREE, or for its undirected special case, with running time $O(c^n)$ for a constant $c < \sqrt{2}$; for undirected VERTEX-WEIGHTED STEINER TREE, Buchanan, Wang and Butenko [BWB18] give $O(1.5296^n)$.

Let $w\colon V \to \{0, \ldots, w_{\max}\}$ be vertex weights, and let $\gamma^w_{\mathrm{pr}}(G)$ be the least weight of a paired dominating set.

Theorem D.11.21 (Weighted Paired Domination)**.** *For graphs G with n vertices and no isolated vertex and weights $w\colon V \to \{0, \ldots, w_{\max}\}$ in unary, $\gamma^w_{\mathrm{pr}}(G)$ can be computed by a randomized algorithm with one-sided error in time $O(1.6302^n)(n + w_{\max})^{O(1)}$ and polynomial space.*

Proof. We apply Lemma D.11.6 to Φ_{PD} with the weights w, so that the solutions of Φ_ϖ are the paired dominating sets of weight at most ϖ; Lemma 12.9 with the weights w and the budget ϖ is a one-sided-error extension algorithm for Φ_ϖ with running-time exponent $|\operatorname{und}(X)|$, as in Corollary 3.15. Let $\mathcal{B}_\varpi(G, k)$ be Algorithm MLS-θ for Φ_ϖ with this extension algorithm and the parameters of the proof of Theorem 12.10, for even k (and the answer NO for odd k, as a paired dominating set has even size). By Theorem 3.5 with Corollary 3.15, it runs in time $2^{\psi n}(n+w_{\max})^{O(1)} = O(1.6302^n)(n+w_{\max})^{O(1)}$, with $\psi = 0.70502$ as in that proof, and in polynomial space. It answers NO if Φ_ϖ has no solution, which is property (B1). For property (B2), let S be a paired dominating set of least weight; Lemma D.11.19 holds for every paired dominating set, so S satisfies condition $(\star)$ at $k = |S|$ and $\mathcal{B}_{w(S)}(G, |S|)$ answers YES with probability at least $1 - e^{-\kappa}$. Now Lemma D.11.6 gives the claim, with $w(V) + 1 \le nw_{\max} + 1$ budgets. □

E Audit of the applications

This appendix collects, for each algorithmic theorem of the paper, what its proof imports, where its randomness comes from, how it treats weights, and what it outputs. Every algorithm in the tables runs in polynomial space, and every randomized one has one-sided error: it never outputs a value below the optimum, or never accepts a NO-instance, and it succeeds with probability at least $1 - e^{-\kappa}$ for the confidence κ of the theorem, or at least $1 - 1/e$ where the theorem says so.

Polynomial factors. Every extension algorithm runs in time $2^{g(I,X,k)}N^{O(1)}$ on every input, with one polynomial for all inputs (Definition 3.2); for randomized extension algorithms the bound holds for every outcome of the coins. The polynomial factor of Theorems 3.5 and 3.14 depends only on N, since the rate, the loss factor and the confidence are at most $N^{O(1)}$, and the trials of RSC run in time $2^\theta r_0(n)$ for one polynomial r_0 that does not depend on G, s, t or θ (Lemma 8.4). The deterministic algorithms use Lemma 10.2, which holds uniformly for every implicit set system, extension algorithm and running-time exponent with values in $\{0, \ldots, n\}$, and they carry the further factor $2^{O(n(\log\log n)^2/\log n)}$ of the balanced families. Weighted versions carry the factor $(n+w_{\max})^{O(1)}$ stated in their theorems, which is $n^{O(1)}$ when $w_{\max} \le n^{O(1)}$.

From decisions to optimal values. The algorithms for Connected Dominating Set and its variants run the decision procedure for $s = 1, 2, \ldots$ and output the first accepted s (Lemma 5.2); vertex-weighted versions run it for every budget ϖ and output the least accepted budget (Lemma D.11.6). The randomized ones output a value that is never below the optimum and equals it with the success probability of their theorem; the deterministic ones output the optimal value. The algorithms for Connected Feedback Vertex Set, its weighted version, Connected Vertex Cover and Maximum-Weight Connected k-Subgraph also output a solution, which is optimal with probability at least $1 - 1/e$ (at least $1 - e^{-\kappa}$ for Connected Vertex Cover), through the witness of the extension algorithm and, where the black box gives only a decision, a self-reduction (Lemma D.10.1). Graph Motif is a decision problem.

Theorem	Extension step and imported results	Randomness	Weights
Theorems 6.4 and 7.2	Steiner extension Corollary 5.8, from Lemma 5.3, proved in Appendix D.3; parameter: the number of terminals	uniform samples	none
Theorem 8.1	the same, on the closed sample	uniform samples and the closure	none
Theorem 10.4	the same	deterministic (balanced families, Theorem 9.6)	none
Theorem 12.2	Lemma 5.3 on the red–blue graph with $2n$ vertices (Lemma D.11.1)	(a) uniform samples; (b) deterministic	none
Theorem 12.3	Lemma 5.3 with vertex weights, encoded as arc lengths (Appendix D.3)	(a) uniform samples; (b) deterministic	$\{0, \ldots, \varphi_{\max}\}$, unary
Theorem 12.4	Lemma 5.3 via Lemma D.11.1	deterministic	none
Theorem 12.5	Lemma 12.7, from Lemma 5.3, once for every root	deterministic	none

Table 6: Algorithms for Connected Dominating Set, Max Leaf Spanning Tree and their variants. Each outputs the optimal value, with probability at least $1 - e^{-\kappa}$ for the randomized ones (Lemmas 5.2 and D.11.6). No external algorithm is used as a black box: the vertex-weighted directed Steiner lemma (Lemma 5.3) is proved in this paper (Appendix D.3), following inclusion–exclusion over branching walks [Ned13].

Imported results. The results that the proofs use as black boxes are: the Cut & Count facts (CC1)–(CC3) for Connected Feedback Vertex Set, which its weighted version uses only through the unweighted extension algorithm that supplies the modulator, together with the isolation lemma in the form of [CNP+22, Lemma 2.5]; the algorithm of Björklund, Kaski and Kowalik for Graph Motif; Cygan's algorithm for weighted Connected Vertex Cover; and the Tutte matrix with the identity test of Schwartz and Zippel. The Steiner computations for connected domination and Directed Steiner Tree, the balanced families, the weighted counting for Connected Feedback Vertex Set (Lemmas D.10.8 and D.10.9), and the weighted sieve for Maximum-Weight Connected k-Subgraph are proved in this paper.

Theorem	Extension step and imported results	Randomness	Weights
Theorem 11.8	Cut & Count with forced vertices (Lemma 11.7), from the facts (CC1)–(CC3): Theorem A.13 of [CNP$^+$11], its proof, and the proof of Theorem 5.1 of [CNP$^+$22] (Appendix D.9.3)	uniform samples; Cut & Count	none
Theorem 11.17	Lemma 11.7 as a modulator, the cut parity and weighted core evaluation (Lemmas D.10.8 and D.10.9), proved in Appendix D.10, and the isolation lemma [CNP$^+$22, Lemma 2.5]	uniform samples; Cut & Count; isolation	1 to $w_{\max} \leq n^{O(1)}$; forced vertices weigh 0
Theorem 11.13	[BKK16, Theorem 4 and Section 3.7] on the instance in which each component of the sample is contracted to a vertex with a fresh color (Lemma 11.11)	uniform samples; the black box	none
Theorem 11.15	[Cyg12, Theorem 3], deterministic, on the contracted graph with pendant vertices (Lemma 11.14)	uniform samples only	1 to $w_{\max} \leq n^{O(1)}$; contracted and pendant vertices weigh 1
Theorem 11.16	a weighted sieve adapted from [BKK16], proved in Appendix D.10	uniform samples; polynomial identity test	$-w_{\max}$ to $w_{\max} \leq n^{O(1)}$; contracted vertices weigh 0

Table 7: Algorithms with component-count running times. The algorithms for Connected Feedback Vertex Set and Maximum-Weight Connected k-Subgraph output a solution that is optimal with probability at least $1 - 1/e$, the one for Connected Vertex Cover a weight and a solution that are optimal with probability at least $1 - e^{-\kappa}$, and the one for Graph Motif a decision with one-sided error. Each extension step runs in time exponential in the number $k - |X|$ of elements still to be added and in the number $\mathrm{comp}(G[X])$ of components of the sample (Section 11).

Theorem	Extension step and imported results	Randomness	Weights
Theorem 12.8	Lemma 12.7, from Lemma 5.3	uniform samples	none
Corollary D.11.20	the same with weights, and Lemma D.11.6	uniform samples	$\{0, \ldots, w_{\max}\}$, unary
Theorem 12.10	Lemma 12.9: Tutte matrix [Tut47] and the identity test of Schwartz and Zippel [Sch80, Zip79]	uniform samples; identity test	none
Theorem D.11.21	the same with an indeterminate for the weights, and Lemma D.11.6	uniform samples; identity test	$\{0, \ldots, w_{\max}\}$, unary

Table 8: Directed Steiner Tree and Paired Domination. Each algorithm outputs a value that is never below the optimum and equals it with probability at least $1 - e^{-\kappa}$; the one for Directed Steiner Tree may instead report, correctly, that no Steiner set exists.

F Computer certificates

Several constants in the bases of our running times rest on finite computations. The first table lists every ancillary file that a proof in the paper depends on, together with what it certifies and the arithmetic it uses; Appendices F.1 to F.8 describe, result by result, what is checked and how. Every check is either in exact rational arithmetic or, for logarithms, exponentials and square roots, in ball arithmetic with outward rounding (Arb [Joh17], through python-flint, at 256 bits): each operation returns a ball that provably contains the exact real result, and an inequality is accepted only if it holds for the whole ball. Inputs such as decimal endpoints and parameters are read as exact rationals. So each certificate is a proof, up to the correctness of the software.

Files that a proof depends on

File	Certifies	Arithmetic
`rs_closed_form_interval.py`	the constant 1.744835 of RS and the two bounds on F^* used by RS-θ (Proposition 6.1(c)), by adaptive bisection into 9608, 60 and 73 intervals	exact rationals; Arb
`cert_final_0.json`, `cert_final_1.json`, `verify_cert.py`, `verify_cert_indep.py`, `v_cert_primal.py`, `cz_bounds.py`	the two antinorm certificates of RS-θ (Proposition 7.7), checked three independent ways, and the bounds on c_{Z_i}	exact rationals
`rst_rate_interval.py`	the rates of the RS-θ certificates at the endpoints of their ranges and $2^{\psi^*} < 1.7204$ (Propositions 7.7 and 7.8)	exact rationals; Arb
`lobster_rate_certificate.py`	the bounds 1.71927 and 1.71928 for the uniform lobsters (Proposition 7.9(c)), on 4764 intervals	exact rationals
`rsc/rsc-certificate.json.gz`, `rsc/rsc-verify.py` (with `rsc_verify_core.py` and `rsc_modelg.py` in `rsc/`), `rsc/rsc-credits.py`, `rsc/rsc-verify-independent.py`	the antinorm certificate of RSC (Proposition D.5.23), checked by two independent programs, and the credit inequalities of Lemma D.5.22	exact rationals
`rates.py` (part 4), `v_rates.py` (part V1)	the inequalities at the ten endpoints of the five intervals of the deterministic rate certificate (Lemma 10.1)	exact rationals
`component_count_rate_interval.py`	the constant 1.6444 for Graph Motif, Connected Vertex Cover and its variants (Lemma 11.12, with the rate $0.71755 < \log 1.6444$)	exact rationals, including Collatz–Wielandt bounds on spectral radii; Arb
`dst_certificates.json`, `dst_certificate_verify.py`, `cz_bounds.py`	the three certificates, the bounds on $c_{Z_j} e_{2,j}$ and the constant 1.3727 for Directed Steiner Tree (Proposition D.11.17)	exact rationals; Arb for logarithms
`paired_rate_certificate.py`	the constant 1.6302 for Paired Domination (Lemma D.11.19)	Arb

The programs that produce the data files search in floating-point or decimal arithmetic; only the checks in the table are used (of these producers, only `dst_certificate_build.py` is included). The second table lists the programs behind the numerical statements of the paper. Each of these statements is marked as numerical where it is made, and no proof uses them.

Numerical evidence, not used in any proof	
File	Computes
`rs_closed_form_interval.py` (numerical part)	the maximum of the closed form F^* for RS, about 0.8030893, so that the constant 1.744835 of Proposition 6.1(c) exceeds the base $2^{\max F^*}$ by less than $2 \cdot 10^{-6}$
`orthant/orthant_try.py`	the orthant antinorm search for RS-θ that did not close (Remark D.4.3)
`lower_bound_caterpillar.py`	the finite values for uniform lobsters (Proposition 7.9)
`families.py`, `v_extremal.py`	the bases of stars of stars and paths, and the search over 58 800 periodic families (Section 7.6)
`v_rates.py` (part V5: 1.9440 at $\sigma = 0.4172629$, the value used in the text, and 1.9443 at $\sigma = 0.4176$, as in `rates.py` part 3)	the black-box lower bound of about 1.9440^n for RS-θ (Theorem 9.4)
`rates.py` (part 4, grid), `v_rates.py` (part V2; floating point, tolerance 10^{-11})	numerical evidence that other choices of y would not lower the bound of Lemma 10.1: $\max_u \inf_{y>0} \chi_y(u) \approx 0.795823$, with grid maximizer $u \approx 0.436$
`size_demo.py`	the sizes of the balanced product families for $n \le 20$ and block lengths 4 to 6: they contain 80–90% of all t-sets (Appendix D.7)
`motif_path_extremal.py`	$\mathcal{Z}_F \ge \mathcal{Z}_{\text{path}}$ on trees with at most 8 vertices (after Theorem 11.13)
`wcds_spider.py`	the spider values after Proposition D.11.10 (Appendix F.5)
`dst_obstruction.py`	the bound $\Omega(1.3716^n)$ for a family of instances of Directed Steiner Tree (Section 12.5)
`maxleaf-measure/optimise.py`, `maxleaf-measure/sensitivity.py`	the measure optimization for the algorithm of Binkele-Raible and Fernau (after Proposition C.4)

The files of the measure optimization, of the orthant search for RS-θ and of RSC are in the subfolders `maxleaf-measure/`, `orthant/` and `rsc/` of the ancillary folder; the tables give the paths of these files. All other files are at its top level. Each program is run as `python3 <program>` from the folder that contains it, with the data files of its table row as arguments where it reads them. `verify_cert.py`, `verify_cert_indep.py` and `v_cert_primal.py` take `cert_final_0.json` and `cert_final_1.json`; `rsc-verify.py` and `rsc-verify-independent.py` take `rsc-certificate.json.gz` and an optional number of processes; and `dst_certificate_build.py` takes `dst_certificate_params.json`. The other programs find their data themselves. The output of each program is recorded in the file of the same name with extension `.out`. For RSC, the outputs are `rsc-verify.out`, `rsc-verify-independent.out` (with `rsc-verify-independent-negative.out` for the perturbed copies) and `rsc-credits.out`. The programs need Python 3 with python-flint (tested with version 0.9.0); some also need NumPy and SciPy. The file `SHA256SUMS` lists the SHA-256 checksums of

all ancillary files; the certificate of RSC, `rsc/rsc-certificate.json.gz`, has SHA-256 checksum `9465bd0f7f74dc3a6376724645bfbde97b417035583c89ba787e347813177651`. This file (45 MB) is not among the ancillary files on arXiv, which would store it unpacked; it is in the archived copy of this release [Gas26] and on the companion site, at the same path. The script `verify_all.sh` checks the checksums and runs every program of the first table except the two that check the certificate of RSC with one command; with the option `-full` it also runs these two, which took about 46 and 16 minutes with four processes in our runs, while all other checks together took about a minute and a half on one core. The file `README.md` lists which file checks which statement. The ancillary files are released under the MIT license (file `LICENSE`).

F.1 The closed form of RS

For Proposition 6.1(c), the program `rs_closed_form_interval.py` covers $[0.2, 0.8]$ by 9608 closed intervals with exact rational endpoints, obtained by adaptive bisection, and encloses F^* on each of them in ball arithmetic; every enclosure lies below the enclosure of $\log 1.744835$. The largest upper bound obtained is 0.80309061427 (rounded up), while $\log 1.744835 = 0.80309061469\ldots$, a margin above $4.2 \cdot 10^{-10}$. The same program shows $F^* \le 0.7807$ on $[0.15, 0.31]$ and $F^* \le 0.7802$ on $[0.54, 0.8]$ in the same way, by two further coverings of these intervals, with 60 and 73 intervals, separate from the first one because the targets are lower; Proposition 6.1(c) states all three bounds, and RS-θ uses the last two (Table 3).

F.2 The certificates of RS-θ

For $i \in \{0, 1\}$, the file `cert_final_`i`.json` stores the data $(y_i, e_{1,i}, e_{2,i}, Z_i)$ of certificate i of Proposition 7.7. Three independent programs check the hypotheses of Lemma 7.5 for both certificates in exact rational arithmetic: `verify_cert.py`, `verify_cert_indep.py`, which uses linear programming duality instead of the stored convex combinations, and `v_cert_primal.py`. The program `cz_bounds.py` computes $c_{Z_i} = \min_{z \in Z_i} \mathrm{fin}(z)$ exactly, using the code of `verify_cert.py`, and checks the bounds $c_{Z_0} \ge 0.6212$ and $c_{Z_1} \ge 0.6293$ of Proposition 7.7 (the values are $0.62120\ldots$ and $0.62935\ldots$), hence $c_{Z_i} \ge 0.62$, the bound used in Proposition 7.8. The program `rst_rate_interval.py` encloses the coefficients of ψ_0 and ψ_1 and their values at the four endpoints 0.31, 0.4172629 (twice) and 0.54 in ball arithmetic, starting from the exact rationals $y_i, e_{1,i}, e_{2,i}$; every enclosure lies below $0.78270152 < \psi^*$, with a margin $\psi^* - \max > 8 \cdot 10^{-8}$. It also checks $2^{\psi^*} < 1.7204$.

F.3 Uniform lobsters

For Proposition 7.9(c), the program `lobster_rate_certificate.py` produces the 4764 intervals $[a, b]$ covering $[0, \frac{3}{4}]$, the rational vectors $w > 0$ and the rationals $\lambda > 0$, and the rational data $v > 0$ and $\lambda' > 0$ at $y^* = 0.3453$, and checks all the inequalities stated in the proof of Proposition 7.9 for these data in exact rational arithmetic.

The following computations are in floating point and are not used in any proof. The minimum over $y > 0$ of $\log(1 + y) - \frac{1}{7} \log \lambda_{\max}(\mathbf{M}_y)$, for the nonnegative matrix $\mathbf{M}_y$ of the proof of Proposition 7.9 and its largest real eigenvalue $\lambda_{\max}(\mathbf{M}_y)$, which is its spectral radius by the Perron–Frobenius theorem, is 0.7817982553 at $y^* = 0.34531$, which gives the numerical value $1.7192725\ldots$ of Proposition 7.9(c). The programs `families.py` and `v_extremal.py` compute the bases of stars of stars and of paths stated after the proof, and search 58 800 periodic families; none has a larger base than Λ_L.

F.4 The certificate of RSC

The data of Proposition D.5.23 are stored in `rsc/rsc-certificate.json.gz`. The points of Z and of the stages are stored as exact rationals. For each vector P whose membership in $\mathrm{conv}(\mathcal{Y}) + \mathcal{K}_{\mathrm{pos}}$ is asserted, the file lists the points z_i and stores the weights α_i as floating-point numbers, which the checking program converts to rationals with denominators at most 10^{12} (the closest such rationals, by `Fraction.limit_denominator`), dropping nonpositive weights; these rationals are the coefficients α_i of the proof of Proposition D.5.23, and the program checks exactly that $\sum_i \alpha_i \geq 1$ and that $P \succeq \sum_i \alpha_i z_i$, the latter by evaluating all up-set sums; the poset cone has 43 nonempty up-sets. There are $1024^2 = 1{,}048{,}576$ pair conditions for (C1); 26 conditions for (C2); 1848 conditions for (C4) ($4 \leq \nu \leq 36$, for the points $(0, c_3)$ and the hull vertices with $c_2 \geq 1$ of the lattice for $\nu - 1$, which the program recomputes exactly); and 1962 staged chains for (C5) (the points $(0, c_3)$, $1 \leq c_3 \leq \nu$, and the hull vertices with $c_2 \geq 1$ of the lattice for ν, $1 \leq \nu \leq 36$), with 2,636,800 conditions $x \circ z$ in these chains. The file stores the values of $d^{(3)}_{\min}(\nu)$ and $d^{(2)}_{\min}(\nu)$ stated in Proposition D.5.23 for $1 \leq \nu \leq 4$, and the value 0 for $5 \leq \nu \leq 36$, where Proposition D.5.23 states the value 1; the proof of Proposition D.5.23 shows that both values give the same conditions. The check also confirms that all points of Z are nonnegative and that $d^{(3)}_{\min} \leq d^{(2)}_{\min}$, and it computes $c_Z = 0.60605\ldots$ exactly. Both programs described below also check exactly the inequality $(q_3/\hat{e})^{37} \geq \hat{e}^{-25}$ of (C6), which covers the tail $\nu > 36$.

The checking program (`rsc/rsc-verify.py`, with `rsc_verify_core.py` and `rsc_modelg.py` in the same folder) works in rational arithmetic throughout; it also evaluates the inequalities of Definition D.5.10 and (30) in 60-digit decimal arithmetic; this evaluation is not used, since Lemmas D.5.21 and D.5.22 prove these inequalities. The truncated-series bounds in the proof of Lemma D.5.22 are evaluated exactly by `rsc/rsc-credits.py`, which also rechecks the light pairs of Lemma D.5.21. A second program (`rsc/rsc-verify-independent.py`), written from the definitions of Section 8 without reusing any code of the first, re-derives the eight-state recursion, checks it exactly against a brute-force enumeration of the relaxed model and checks $\widetilde{W} \leq W^{\mathrm{tm}}$ on random trees, recomputes the lattice hulls, confirms that all 613 stage points are nonnegative, and verifies the same certificate in rational arithmetic, with rigorous rational bounds for the logarithms in Definition D.5.10 and (30); perturbed copies of the certificate are rejected by it at the expected condition. The first program checks every relation with the rationals above as the coefficients α_i of the proof. The second accepts a relation if it holds with these rationals or with the exact binary values of the stored weights; either set of nonnegative coefficients proves the relation, so both programs confirm the same relations, possibly with different coefficients.

The points were found by an iterative floating-point search and then rounded to rationals; only the exact check is used.

F.5 The deterministic rate certificate

For Lemma 10.1, the inequalities between rational numbers at the ten endpoints of the five rational intervals, against $\log \frac{1736}{1000}$, are checked in exact rational arithmetic by `rates.py` (part 4) and, independently, by `v_rates.py` (part V1).

Separately, for the spiders of Proposition D.11.10, the program `wcds_spider.py` computes β_s, the largest value of $\left(\inf_{y>0}(1+y)^n / W_S(y)\right)^{1/n}$ over $d_1, \ldots, d_s \in \{2, 3\}$, by one-dimensional minimization over y in floating point. It gives $\beta_{50} = 1.743507$, $\beta_{200} = 1.737898$ and $\beta_{1000} = 1.736432$, decreasing toward $\sqrt{5} - \frac{1}{2} = 1.736068\ldots$. These values are numerical evidence and are not used in any proof.

F.6 The component-count certificate

For Lemma 11.12, the program `component_count_rate_interval.py` checks the inequalities (o)–(iii) of its proof with exact rational parameters and ball arithmetic for the logarithms. In (iii), $\lambda_{\max}(M_{y_j})$ is bracketed between rationals by the Collatz–Wielandt inequalities $\min_i (M_{y_j} v)_i / v_i \le \lambda_{\max}(M_{y_j}) \le \max_i (M_{y_j} v)_i / v_i$ for positive vectors v, evaluated exactly; the lower bracket enters $\phi_{2,\sigma}(y_j)$, which is decreasing in $\lambda_{\max}(M_{y_j})$, and the upper one the prefactor. Part B of the same program rechecks the identities and the inequality behind the value 1.8 (Corollary 11.5 and Lemma D.9.8), including $5^{45} > 3^{64}$ in exact integer arithmetic; the proofs of these statements do not use it. The largest value of $\max(\phi_{2,\sigma_j^-}(y_j), \phi_{2,\sigma_j^+}(y_j))$ in (iii) is $0.7174350\ldots$, against $\psi = 0.71755$ and $\log 1.6444 = 0.7175613\ldots$; the largest prefactor $\lambda_{\max}(M_{y_j})/\min(1, y_j)$ is at most 5.933. The program `motif_path_extremal.py` checks the inequality $\mathcal{Z}_F \ge \mathcal{Z}_{\text{path}}$ raised after Theorem 11.13 for all trees on at most 8 vertices at $\zeta \in \{\frac{1}{8}, \frac{1}{4}, \frac{1}{2}, 1, 2, 4\}$ in exact rational arithmetic; this is evidence only.

F.7 Directed Steiner Tree

The file `dst_certificates.json` stores the three certificates of Proposition D.11.17 and the weights λ_j. The program `dst_certificate_verify.py` checks the hypotheses (C1)–(C3) in exact rational arithmetic and that the weights λ_j are nonnegative and sum to 1. The program `cz_bounds.py` computes c_{Z_j} exactly ($0.52305\ldots$, $0.53544\ldots$ and $0.02153\ldots$) and checks $c_{Z_j} e_{2,j} \ge 0.40$, 0.41 and 0.011 (Proposition D.11.17), so $c_{Z_j} e_{2,j} \ge 1/100$ for every j, as the proof of Theorem 12.8 uses. Membership of a point x in $\mathcal{R}_Z$ is certified by a witness, a point of Z or a rational convex combination of two points of Z that lies coordinatewise below x; in the plane such a witness exists whenever $x \in \mathcal{R}_Z$. The logarithms in $\boldsymbol{\alpha}_j$ and the weighted sums $\sum_j \lambda_j \alpha_{j,i}$ are enclosed by the same program in ball arithmetic, starting from the exact rationals $y_j, e_{1,j}, e_{2,j}, \lambda_j$; the enclosure of their maximum $0.4569669\ldots$ lies more than $3.3 \cdot 10^{-5}$ below 0.457, and $0.457 < \log 1.3727$ is checked in the same arithmetic. The program `dst_obstruction.py` computes, in floating point, the value $1.37168\ldots$ behind the bound $\Omega(1.3716^n)$ stated after Theorem 12.8; this is evidence only. The family consists of a directed path $r \to p_1 \to \cdots \to p_{2L}$ in which each odd p_{2i-1} has one terminal out-neighbor of out-degree 0 and one further out-neighbor with two such terminals, each even p_{2i} has two such terminals, and $\lfloor \varkappa L \rfloor$ padding vertices have arcs to and from both terminals of p_2. The spine and its further out-neighbors form the only Steiner set of size at most $3L$, and as $L \to \infty$ the bound $\left(\min_t \binom{n_U}{t} / \mu_t(S^*)\right)^{1/n}$ for this set is largest at $\varkappa \approx 2.9663$, where it equals $2^{5/(8+\varkappa)} = 1.37168\ldots$.

F.8 Paired Domination

For the first claim of Lemma D.11.19, the program `paired_rate_certificate.py` encloses y^* and $h_{\bar{n}}(y^*)$ for $2 \le \bar{n} \le 40$ in ball arithmetic and checks $h_6(y^*) - h_{\bar{n}}(y^*) > 0.00297$ for $\bar{n} \ne 6$, and it checks the tail bound $\log(1+y^*) + \frac{2}{41} \log \frac{1}{y^*} < 0.431 < h_6(y^*)$ in the same arithmetic, starting from $y^* = (3\sqrt{3} - 1)/16$. It also checks $h_6(y^*) < 0.70502$ and $2^{0.70502} < 1.6302$, the two inequalities used in the proof of Theorem 12.10.

F.9 Formal verification

Part of the analysis is also checked in Lean 4 with Mathlib. Except for the data of the RSC certificate (see the end of this subsection), it is checked by the Lean kernel, using only the standard axioms (propositional extensionality, choice and quotient soundness) and without compiled

evaluation (`native_decide`). The statements below refer to the Lean files in the folder `formal/` of the ancillary files, a copy of the development as of 1 November 2026; the files `lean-toolchain` and `lake-manifest.json` there pin the versions of Lean and Mathlib. They cover the following statements.

- Soundness and success probability, items (ii) and (iii), of Theorem 3.5, for the cap and trial counts stated after Algorithm 3.4; Lemma 3.8; and soundness and success probability of Theorem 3.14 for abstract trials. Also both inequalities of Lemma 3.9: the first in the form of the last sentence of Theorem 3.5, the second in the pointwise form of its proof; and Proposition 3.10 for the same cap and trial counts, with the probability that some trial runs $\mathcal{A}$ on a subset of S in place of the expected number of such trials.

- The Jensen step of the proof of Theorem 4.1 and the success probability it gives; the minimization in Remark 4.4 (f_σ is strictly convex on $[0, \sigma]$ and attains its minimum exactly at τ^*); the inequality of part (1) of Corollary 4.3, and its part (2).

- Lemma 5.1, with the spanning tree shown to exist rather than computed; Lemmas 5.10 and 5.11; the inequality of Lemma 5.9, without its equality case; the claims of Lemma 5.2 about the output; and Corollary 8.5 except (T1), running time and space, for abstract trials, with its claim for Max Leaf Spanning Tree when $n \geq 3$.

- For RS-θ: Lemmas 7.3 to 7.5 and D.4.1; both certificates of Proposition 7.7, checked in exact rational arithmetic, with the numerical claims of Proposition 7.7 and $2^{\psi^*} < 1.7204$; the constants of Theorem 7.2; and Proposition 7.8.

- For RS: Proposition 6.1, with part (c) by interval covers checked in exact rational arithmetic; Corollary 6.2; and the moment bound in the proof of Theorem 6.4, with $\log 1.744835 < \psi_{\mathrm{RS}}$.

- For RSC: Proposition 8.6 from conditions (C1), (C2), (C4) and (C5) of a certificate with $c_Z > 0.6060$, through all statements of Appendix D.5 except Proposition D.5.23; the claim of Lemma 8.4(a) that a trial rejects if $\gamma_{\mathrm{c}}(G) > s$, and Lemma 8.4(b); and, from the same conditions, that the output of Theorem 8.1 is never below $\gamma_{\mathrm{c}}(G)$, its success probability, and its claim for Max Leaf Spanning Tree when $n \geq 3$. The bound $\varrho < 1.6991$ is checked as an exact rational comparison. The two Lean definitions of the closure generating function of Definition 8.3 that these results use are proved equal.

- For the Steiner extension: the correctness of the computed value in Lemmas 5.3 and 5.6 and Corollary 5.8, that is, the inclusion–exclusion over branching walks on the directed encoding returns $\mathrm{OPT}(J)$ for every state J, and $\mathcal{A}_{\mathrm{St}}$ solves Φ_{CDS}-Extension. So the acceptance test of step (c) of Algorithm 8.2, as the algorithm computes it, accepts exactly the samples that its specification accepts, and the two RSC results above are also proved for that test. No Lean function returns a connected dominating set or a spanning tree.

- Lemma 10.1; Lemma 11.12, with $\lambda_{\max}(M_\zeta)$ given by its closed form in Definition 11.1; Lemma 11.2 for every finite tree and Lemma 11.3 for paths, both with the same closed form, which is proved to be the largest eigenvalue of M_ζ; part (a) of Corollary 11.5; the final inequality $5^{45} > 3^{64}$ in the proof of Lemma D.9.8; Proposition D.11.17, Lemmas D.11.14 to D.11.16 and Equation (38), which give the lower bound on $W_S(y_j)$ in the proof of Theorem 12.8, and the constants of that proof; Lemmas D.11.18 and D.11.19, the constants of Theorem 12.10 and condition ($\star$) for every paired dominating set, and Proposition 12.11, whose last claim

is checked for the cap and trial counts stated after Algorithm 3.4 and an extension algorithm that accepts exactly the YES-instances (the step to the algorithm of Lemma 12.9 is not in Lean: that algorithm accepts only YES-instances, so a run with it accepts only on samples on which the run with the exact algorithm accepts); and the two bounds of part (c) of Proposition 7.9, for Λ_L and for every graph that has the same edges as Λ_L at the vertices of S^*.

The data of the RSC certificate are checked in Lean as well, but not by the kernel alone. The program `formal/scripts/export_rsc_cert.py` translates `rsc/rsc-certificate.json.gz` into Lean data files. Checking functions written in Lean, whose correctness is proved in the kernel with the standard axioms, are evaluated on these data by compiled code (`native_decide`, in 472 generated theorems). From this Lean proves Proposition D.5.23 as stated: the 1024 points of Z satisfy (C1)–(C6), and for the same Z there are nonnegative rational stages of the stated sizes that satisfy the relations defining stages. The same theorem contains the conditions on the parameters in Lemma D.5.20, which the proof of Proposition D.5.23 checks. Lean also proves the exact value of c_Z, a fraction equal to $0.60605\ldots$, and proves (C5) directly for every sequence of the required length, the form that Lemma D.5.20 uses. With the RSC item above, this gives the success probability in Theorem 8.1 with no assumption on the certificate. This check trusts the Lean compiler in addition to the kernel; it is a third check of the data, beside the two programs of Appendix F.4. The large generated data files (about 170 MB) are not included: the program regenerates them, and `formal/MaxLeafFormal/RSCData/MANIFEST.sha256` lists the SHA-256 checksums of the files that Lean checked. Files regenerated with the exporter shipped here differ from those only in one header comment, the checksum of the exporter; the `README.md` of the ancillary files shows how to restore it, after which all checksums match.

Not checked in Lean are running times and space bounds, the algorithms other than the Steiner tree computation, parts (b) and (c) of Corollary 11.5, parts (a) and (b) of Proposition 7.9, the last sentence of its part (c) (the bracketing up to factors and the numerical limit) and the consequence stated after it, and the graph-theoretic parts of the other applications.

AI methodology

The author used Anthropic's Claude (Claude Opus 5.5 as the main research collaborator, and Claude Fable 5.1 for part of the analysis of Algorithm RSC), OpenAI Codex and GitHub Copilot. These tools materially affected every section of the paper.

- Claude proposed and developed ideas in discussion with the author, carried out mathematical derivations and proofs, wrote the code for the experiments, the computer certificates and their checking programs (Appendix F) and the Lean development (Appendix F.9), and drafted and revised the text.
- OpenAI Codex was used for independent reviews of the manuscript and the code, and for developing the research-workflow tooling of the project.
- GitHub Copilot was used for code completion, reviews of pull requests and commit messages. Pull requests were also reviewed by Claude Haiku 5.5.

The author directed the research, contributed ideas, among them graded closure (Section 8), checked the proofs and the Lean statements, edited the manuscript and extended the literature search. The project started from work of the author from 2020: the W[1]-hardness of MaxLeaf-Extension

(Theorem C.2) and the product form of multivariate Monotone Local Search, which were the basis of an Honours thesis [Lu20]. The author verified the correctness and originality of all content, including references.

References

[ABK+24] Nikita Andreev, Ivan Bliznets, Madhumita Kundu, Saket Saurabh, Vikash Tripathi, and Shaily Verma. Parameterized complexity of paired domination. In *Combinatorial Algorithms, 35th International Workshop, IWOCA 2024*, volume 14764 of *Lecture Notes in Computer Science*, pages 523–536. Springer, 2024. doi:10.1007/978-3-031-63021-7_40.

[AFG+09] Noga Alon, Fedor V. Fomin, Gregory Z. Gutin, Michael Krivelevich, and Saket Saurabh. Spanning directed trees with many leaves. *SIAM Journal on Discrete Mathematics*, 23(1):466–476, 2009. doi:10.1137/070710494.

[AGM+96] Helmut Alt, Leonidas J. Guibas, Kurt Mehlhorn, Richard M. Karp, and Avi Wigderson. A method for obtaining randomized algorithms with small tail probabilities. *Algorithmica*, 16(4/5):543–547, 1996. doi:10.1007/BF01940879.

[AKFG+25] Faisal N. Abu-Khzam, Henning Fernau, Benjamin Gras, Mathieu Liedloff, and Kevin Mann. Enumerating minimal connected dominating sets. *SIAM Journal on Discrete Mathematics*, 39(2):946–972, 2025. doi:10.1137/23M1611233.

[AKML11] Faisal N. Abu-Khzam, Amer E. Mouawad, and Mathieu Liedloff. An exact algorithm for connected red-blue dominating set. *Journal of Discrete Algorithms*, 9(3):252–262, 2011. doi:10.1016/j.jda.2011.03.006.

[BKK16] Andreas Björklund, Petteri Kaski, and Łukasz Kowalik. Constrained multilinear detection and generalized graph motifs. *Algorithmica*, 74(2):947–967, 2016. arXiv:1209.1082, doi:10.1007/s00453-015-9981-1.

[BRF12] Daniel Binkele-Raible and Henning Fernau. An exact exponential-time algorithm for the Directed Maximum Leaf Spanning Tree problem. *Journal of Discrete Algorithms*, 15:43–55, 2012. doi:10.1016/j.jda.2012.03.006.

[BRF14] Daniel Binkele-Raible and Henning Fernau. A parameterized measure-and-conquer analysis for finding a k-leaf spanning tree in an undirected graph. *Discrete Mathematics & Theoretical Computer Science*, 16(1):179–200, 2014. doi:10.46298/dmtcs.1256.

[BRFF+12] Daniel Binkele-Raible, Henning Fernau, Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Yngve Villanger. Kernel(s) for problems with no kernel: On out-trees with many leaves. *ACM Transactions on Algorithms*, 8(4):38:1–38:19, 2012. doi:10.1145/2344422.2344428.

[BvBF+11] Nadja Betzler, René van Bevern, Michael R. Fellows, Christian Komusiewicz, and Rolf Niedermeier. Parameterized algorithmics for finding connected motifs in biological networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(5):1296–1308, 2011. doi:10.1109/TCBB.2011.19.

[BWB18] Austin Buchanan, Yiming Wang, and Sergiy Butenko. Algorithms for node-weighted Steiner tree and maximum-weight connected subgraph. *Networks*, 72(2):238–248, 2018. doi:10.1002/net.21825.

[CDL+16] Marek Cygan, Holger Dell, Daniel Lokshtanov, Dániel Marx, Jesper Nederlof, Yoshio Okamoto, Ramamohan Paturi, Saket Saurabh, and Magnus Wahlström. On problems

as hard as CNF-SAT. *ACM Transactions on Algorithms*, 12(3):41:1–41:24, 2016. doi:10.1145/2925416.

[CFK+15] Marek Cygan, Fedor V. Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.

[CL14] Péter Csikvári and Zhicong Lin. Graph homomorphisms between trees. *The Electronic Journal of Combinatorics*, 21(4):P4.9, 2014. doi:10.37236/4096.

[Cla26] Claude (Anthropic). A $2^{o(n)}$ time algorithm for 3-SAT. Paper dated 26 October 2026, with a Lean 4 formalization, 2026. URL: https://github.com/anthropics/formal-math/tree/16a5ce41304fcdaea4e0be7c132e1c361dc26da5/eth.

[CNP+11] Marek Cygan, Jesper Nederlof, Marcin Pilipczuk, Michał Pilipczuk, Johan M. M. van Rooij, and Jakub Onufry Wojtaszczyk. Solving connectivity problems parameterized by treewidth in single exponential time. *CoRR*, abs/1103.0534, 2011. arXiv:1103.0534.

[CNP+22] Marek Cygan, Jesper Nederlof, Marcin Pilipczuk, Michał Pilipczuk, Johan M. M. van Rooij, and Jakub Onufry Wojtaszczyk. Solving connectivity problems parameterized by treewidth in single exponential time. *ACM Transactions on Algorithms*, 18(2):17:1–17:31, 2022. doi:10.1145/3506707.

[CWY00] Yair Caro, Douglas B. West, and Raphael Yuster. Connected domination and spanning trees with many leaves. *SIAM Journal on Discrete Mathematics*, 13(2):202–211, 2000. doi:10.1137/S0895480199353780.

[Cyg12] Marek Cygan. Deterministic parameterized connected vertex cover. In *Algorithm Theory – SWAT 2012, 13th Scandinavian Symposium and Workshops*, volume 7357 of *Lecture Notes in Computer Science*, pages 95–106. Springer, 2012. arXiv:1202.6642, doi:10.1007/978-3-642-31155-0_9.

[DGH+97] Jean E. Dunbar, Jerrold W. Grossman, Johannes H. Hattingh, Stephen T. Hedetniemi, and Alice A. McRae. On weakly connected domination in graphs. *Discrete Mathematics*, 167–168:261–269, 1997. doi:10.1016/S0012-365X(96)00233-6.

[DGKY10] Jean Daligault, Gregory Gutin, Eun Jung Kim, and Anders Yeo. FPT algorithms and kernels for the Directed k-Leaf problem. *Journal of Computer and System Sciences*, 76(2):144–152, 2010. doi:10.1016/j.jcss.2009.06.005.

[EK25] Baris Can Esmer and Ariel Kulik. Sampling with a black box: Faster parameterized approximation algorithms for vertex deletion problems. In *52nd International Colloquium on Automata, Languages, and Programming, ICALP 2025*, volume 334 of *LIPIcs*, pages 39:1–39:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025. arXiv:2407.12654, doi:10.4230/LIPIcs.ICALP.2025.39.

[EKM+22] Baris Can Esmer, Ariel Kulik, Dániel Marx, Daniel Neuen, and Roohani Sharma. Faster exponential-time approximation algorithms using approximate monotone local search. In *30th Annual European Symposium on Algorithms, ESA 2022*, volume 244 of *LIPIcs*, pages 50:1–50:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. arXiv:2206.13481, doi:10.4230/LIPIcs.ESA.2022.50.

[EKM+24] Baris Can Esmer, Ariel Kulik, Dániel Marx, Daniel Neuen, and Roohani Sharma. Optimally repurposing existing algorithms to obtain exponential-time approximations. In *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms, SODA 2024*, pages 314–345. SIAM, 2024. arXiv:2306.15331, doi:10.1137/1.9781611977912.13.

[EKM+26] Baris Can Esmer, Ariel Kulik, Dániel Marx, Daniel Neuen, and Roohani Sharma. Approximate monotone local search for weighted problems. *Algorithmica*, 88(5):59, 2026. doi:10.1007/s00453-026-01402-7.

[FFHV11] Michael R. Fellows, Guillaume Fertin, Danny Hermelin, and Stéphane Vialette. Upper and lower bounds for finding connected motifs in vertex-colored graphs. *Journal of Computer and System Sciences*, 77(4):799–811, 2011. doi:10.1016/j.jcss.2010.07.003.

[FGK08] Fedor V. Fomin, Fabrizio Grandoni, and Dieter Kratsch. Solving connected dominating set faster than 2^n. *Algorithmica*, 52(2):153–166, 2008. doi:10.1007/s00453-007-9145-z.

[FGK09] Fedor V. Fomin, Fabrizio Grandoni, and Dieter Kratsch. A measure & conquer approach for the analysis of exact algorithms. *Journal of the ACM*, 56(5):25:1–25:32, 2009. doi:10.1145/1552285.1552286.

[FGK+13] Fedor V. Fomin, Fabrizio Grandoni, Dieter Kratsch, Daniel Lokshtanov, and Saket Saurabh. Computing optimal Steiner trees in polynomial space. *Algorithmica*, 65(3):584–604, 2013. doi:10.1007/s00453-012-9612-z.

[FGLS19] Fedor V. Fomin, Serge Gaspers, Daniel Lokshtanov, and Saket Saurabh. Exact algorithms via monotone local search. *Journal of the ACM*, 66(2):8:1–8:23, 2019. doi:10.1145/3284176.

[FGPR08] Fedor V. Fomin, Serge Gaspers, Artem V. Pyatkin, and Igor Razgon. On the minimum feedback vertex set problem: Exact and enumeration algorithms. *Algorithmica*, 52(2):293–307, 2008. doi:10.1007/s00453-007-9152-0.

[FGPS08] Fedor V. Fomin, Fabrizio Grandoni, Artem V. Pyatkin, and Alexey A. Stepanov. Combinatorial bounds via measure and conquer: Bounding minimal dominating sets and applications. *ACM Transactions on Algorithms*, 5(1):9:1–9:17, 2008. doi:10.1145/1435375.1435384.

[FK10] Fedor V. Fomin and Dieter Kratsch. *Exact Exponential Algorithms*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2010. doi:10.1007/978-3-642-16533-7.

[FK20] Guillaume Fertin and Christian Komusiewicz. Graph motif problems parameterized by dual. *Journal of Graph Algorithms and Applications*, 24(3):371–396, 2020. doi:10.7155/jgaa.00538.

[FKK+11] Henning Fernau, Joachim Kneis, Dieter Kratsch, Alexander Langer, Mathieu Liedloff, Daniel Raible, and Peter Rossmanith. An exact algorithm for the Maximum Leaf Spanning Tree problem. *Theoretical Computer Science*, 412(45):6290–6302, 2011. doi:10.1016/j.tcs.2011.07.011.

[FKL+19] Fedor V. Fomin, Petteri Kaski, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Parameterized single-exponential time polynomial space algorithm for Steiner tree. *SIAM Journal on Discrete Mathematics*, 33(1):327–345, 2019. doi:10.1137/17M1140030.

[FLPS16] Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Efficient computation of representative families with applications in parameterized and exact algorithms. *Journal of the ACM*, 63(4):29:1–29:60, 2016. doi:10.1145/2886094.

[FM23] Andreas Emil Feldmann and Dániel Marx. The complexity landscape of fixed-parameter directed Steiner network problems. *ACM Transactions on Computation Theory*, 15(3-4):4:1–4:28, 2023. doi:10.1145/3580376.

[Gas26] Serge Gaspers. Monotone Local Search with Sample-Dependent Running Times: Companion site with ancillary programs, certificates and Lean 4 formalization. Software and data, version v1.1, 2026. URL: https://serggasp.github.io/sample-MLS-site/, doi:10.5281/zenodo.23851002.

[GHK18] Petr A. Golovach, Pinar Heggernes, and Dieter Kratsch. Enumeration and maximum number of minimal connected vertex covers in graphs. *European Journal of Combinatorics*, 68:132–147, 2018. doi:10.1016/j.ejc.2017.07.015.

[GL17] Serge Gaspers and Edward J. Lee. Exact algorithms via multivariate subroutines. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017*, volume 80 of *LIPIcs*, pages 69:1–69:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPIcs.ICALP.2017.69.

[GMM+25] David Galvin, Phillip Marmorino, Emily McMillon, JD Nir, and Amanda Redlich. Hoffman–London graphs: When paths minimize H-colorings among trees, 2025. arXiv:2512.23828.

[GMNR25] David Galvin, Emily McMillon, JD Nir, and Amanda Redlich. Long paths need not minimize H-colorings among trees, 2025. arXiv:2510.18770.

[GP13] Nicola Guglielmi and Vladimir Protasov. Exact computation of joint spectral characteristics of linear operators. *Foundations of Computational Mathematics*, 13(1):37–97, 2013. doi:10.1007/s10208-012-9121-0.

[GS13] Sylvain Guillemot and Florian Sikora. Finding and counting vertex-colored subtrees. *Algorithmica*, 65(4):828–844, 2013. arXiv:1002.1880, doi:10.1007/s00453-011-9600-8.

[IPZ01] Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001. doi:10.1006/jcss.2001.1774.

[JLR+17] Mark Jones, Daniel Lokshtanov, M. S. Ramanujan, Saket Saurabh, and Ondřej Suchý. Parameterized complexity of directed Steiner tree on sparse graphs. *SIAM Journal on Discrete Mathematics*, 31(2):1294–1327, 2017. arXiv:1210.0260, doi:10.1137/15M103618X.

[Joh74] David S. Johnson. Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences*, 9(3):256–278, 1974. doi:10.1016/S0022-0000(74)80044-9.

[Joh17] Fredrik Johansson. Arb: Efficient arbitrary-precision midpoint-radius interval arithmetic. *IEEE Transactions on Computers*, 66(8):1281–1292, 2017. doi:10.1109/TC.2017.2690633.

[KI04] Valentine Kabanets and Russell Impagliazzo. Derandomizing polynomial identity tests means proving circuit lower bounds. *Computational Complexity*, 13(1-2):1–46, 2004. doi:10.1007/s00037-004-0182-6.

[KLR11] Joachim Kneis, Alexander Langer, and Peter Rossmanith. A new algorithm for finding trees with many leaves. *Algorithmica*, 61(4):882–897, 2011. doi:10.1007/s00453-010-9454-5.

[Kou12] Ioannis Koutis. Constrained multilinear detection for faster functional motif discovery. *Information Processing Letters*, 112(22):889–892, 2012. arXiv:1206.3483, doi:10.1016/j.ipl.2012.08.008.

[KW09] Ioannis Koutis and Ryan Williams. Limits and applications of group algebras for parameterized problems. In *Automata, Languages and Programming, 36th International Colloquium, ICALP 2009, Part I*, volume 5555 of *Lecture Notes in Computer Science*, pages 653–664. Springer, 2009. doi:10.1007/978-3-642-02927-1_54.

[KW16] Ioannis Koutis and Ryan Williams. Limits and applications of group algebras for parameterized problems. *ACM Transactions on Algorithms*, 12(3):31:1–31:18, 2016. doi:10.1145/2885499.

[Lee21] Edward J. Lee. *Exponential Time Algorithms via Separators and Random Subsets*. PhD thesis, University of New South Wales, Sydney, Australia, 2021. URL: http://hdl.handle.net/1959.4/71134, doi:10.26190/unsworks/22740.

[LFS06] Vincent Lacroix, Cristina G. Fernandes, and Marie-France Sagot. Motif search in graphs: Application to metabolic networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 3(4):360–368, 2006. doi:10.1109/TCBB.2006.55.

[LN10] Daniel Lokshtanov and Jesper Nederlof. Saving space by algebraization. In *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010*, pages 321–330. ACM, 2010. doi:10.1145/1806689.1806735.

[Lov75] László Lovász. On the ratio of optimal integral and fractional covers. *Discrete Mathematics*, 13(4):383–390, 1975. doi:10.1016/0012-365X(75)90058-8.

[LPS18] Daniel Lokshtanov, Michał Pilipczuk, and Saket Saurabh. Below all subsets for minimal connected dominating set. *SIAM Journal on Discrete Mathematics*, 32(3):2332–2345, 2018. doi:10.1137/17M1138753.

[LSZ93] Michael Luby, Alistair Sinclair, and David Zuckerman. Optimal speedup of Las Vegas algorithms. *Information Processing Letters*, 47(4):173–180, 1993. doi:10.1016/0020-0190(93)90029-9.

[Lu20] Edward Lu. The k-leaf spanning tree problem. Honours thesis, UNSW Sydney, 2020.

[MPR+12] Neeldhara Misra, Geevarghese Philip, Venkatesh Raman, Saket Saurabh, and Somnath Sikdar. FPT algorithms for Connected Feedback Vertex Set. *Journal of Combinatorial Optimization*, 24(2):131–146, 2012. doi:10.1007/s10878-011-9394-2.

[Ned13] Jesper Nederlof. Fast polynomial-space algorithms using inclusion-exclusion. *Algorithmica*, 65(4):868–884, 2013. doi:10.1007/s00453-012-9630-x.

[Ope26] OpenAI. The Exponential Time Hypothesis is false. OpenAI Math Release preprint, revised 25 October 2026, 2026. URL: https://github.com/openai/math/blob/main/preprints/The-Exponential-Time-Hypothesis-is-False-October-25-2026/paper.pdf.

[PPSZ05] Ramamohan Paturi, Pavel Pudlák, Michael E. Saks, and Francis Zane. An improved exponential-time algorithm for k-SAT. *Journal of the ACM*, 52(3):337–364, 2005. doi:10.1145/1066100.1066101.

[PVC10] PVC (Research Infrastructure), UNSW Sydney. Katana. UNSW, Sydney, 2010. doi:10.26190/669X-A286.

[Ros21] Matthieu Rosenfeld. The growth rate over trees of any family of sets defined by a monadic second order formula is semi-computable. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021*, pages 776–795. SIAM, 2021. doi:10.1137/1.9781611976465.49.

[Rot19] Günter Rote. The maximum number of minimal dominating sets in a tree. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019*, pages 1201–1214. SIAM, 2019. doi:10.1137/1.9781611975482.73.

[Sca23] Kevin Scaman. Black-box acceleration of Las Vegas algorithms and algorithmic reverse Jensen's inequalities. *CoRR*, abs/2304.11017, 2023. arXiv:2304.11017, doi:10.48550/arXiv.2304.11017.

[Sch80] Jacob T. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM*, 27(4):701–717, 1980. doi:10.1145/322217.322225.

[Sid94] Alexander Sidorenko. A partially ordered set of functionals corresponding to graphs. *Discrete Mathematics*, 131(1–3):263–277, 1994. doi:10.1016/0012-365X(94)90388-3.

[Tut47] W. T. Tutte. The factorization of linear graphs. *Journal of the London Mathematical Society*, s1-22(2):107–111, 1947. doi:10.1112/jlms/s1-22.2.107.

[Zam23] Or Zamir. The wrong direction of Jensen's inequality is algorithmically right. In *50th International Colloquium on Automata, Languages, and Programming, ICALP 2023*, volume 261 of *LIPIcs*, pages 107:1–107:10. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPIcs.ICALP.2023.107.

[Zeh15] Meirav Zehavi. The k-leaf spanning tree problem admits a klam value of 39, 2015. Version 2. arXiv:1502.07725.

[Zeh18] Meirav Zehavi. The k-leaf spanning tree problem admits a klam value of 39. *European Journal of Combinatorics*, 68:175–203, 2018. doi:10.1016/j.ejc.2017.07.018.

[Zip79] Richard Zippel. Probabilistic algorithms for sparse polynomials. In *Symbolic and Algebraic Computation, EUROSAM '79*, volume 72 of *Lecture Notes in Computer Science*, pages 216–226. Springer, 1979. doi:10.1007/3-540-09519-5_73.