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

1 Introduction

Many exact exponential-time algorithms [22] 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 $O^*(1.7204^n)$.

Monotone Local Search [18] 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 [27] 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 3.3). 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, these bounds cannot be improved: Esmer, Kulik, Marx, Neuen and Sharma [15] show that every randomized algorithm whose only problem-specific access is a polynomial-time feasibility test and an extension algorithm with running time $O^*(c^{k-|X|})$ takes time at least $(2-1/c)^n$ divided by a polynomial in n, and remark that the same technique shows the bound of Gaspers and Lee to be optimal.

This paper lets the running time of the extension algorithm depend on the sample. Our main theorem (Theorem 2.5) 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 2.2). 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 2.5)**.** *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 2.5 with $\Gamma = \kappa = 1$. The algorithm 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. Unlike restart and cutoff strategies for Las Vegas algorithms (Section 1.3), the running-time exponent is computed *before* the extension call, so a sample above the threshold is skipped at polynomial cost, and it depends on the structure of the sample, not only on its size.

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. 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 (Section 2.4), and in a black-box model the theorem cannot in general be derandomized without exponential loss (Section 4.5). 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. Lemma 2.7 collects the moments of all sample sizes in one *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 $\inf_{y \geq 0} (1+y)^n / W_S(y)$. The theorem extends to randomized trials, such as extension algorithms with one-sided error and samples to which random vertices are added (Theorem 2.9 and Corollary 2.10).

This version. This is a short version of the paper. It states all main results, proves the main theorem, its generating-function form and its general form for budgeted trials, the mean-exponent and multivariate corollaries, the mean bound and the tree reduction for Connected Dominating Set, the partition-function bound for component-count running times and the bound 1.8^n for Connected Feedback Vertex Set given its extension algorithm; it describes the other proofs and keeps the applications short. All omitted proofs are in the full version [25]; the programs that check the computer certificates and the Lean development are on the companion site [26].

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 translate into each other in polynomial time (Lemma 4.1). Table 1 lists the previous bounds for the two problems 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 [2]: the terminals are the vertices that X does not dominate and the components of $G[X]$. Inclusion–exclusion over branching walks [37] solves it in time $2^{|\mathrm{T}(J_X)|} n^{O(1)}$ and polynomial space through a directed Steiner encoding, where J_X is the residual instance left by X (Section 4.1). 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 4.4. 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 (the full version [25]), so it has no algorithm with running time $O^*(c^{\ell'})$ for a constant c unless FPT = W[1].

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 4.7: 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

Corollaries. A bound λ on the *mean* of g over the t-subsets of every solution gives running time $O^*(\binom{n}{t}\binom{k}{t}^{-1}2^{\lambda})$ (Theorem 3.1 and Corollary 3.2). For running times of product form $b^{n-|X|}c^{k-|X|}d^{|X|}$ the main theorem gives the multivariate bound $\big(b+\min(d,bc)(1-1/c)\big)^n$ for all $b,c,d \geq 1$, and hence the theorems of [18] ($b = d = 1$) and [27] ($d = 1$); conversely, the main theorem certifies nothing better for such running times (Corollary 3.3). Every gain in this paper therefore comes from the dependence of the running time on the samples.

Max Leaf Spanning Tree and Connected Dominating Set. All four algorithms complete a sample with the Steiner tree algorithm described above and use polynomial space. The randomized ones have one-sided error: they never output a value below $\gamma_c(G)$, and they output $\gamma_c(G)$ with constant probability.

Theorem 1.2 (Theorems 4.12 and 4.14)**.** 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 4.7), through the linear-savings corollary, and runs in time $O(1.7449^n)$ (Theorem 4.8).
- Algorithm RS-θ is the same algorithm run with a smaller target running time $2^{\psi n}$; the improvement lies in the analysis, which bounds the generating function of a potential that is local along a spanning tree of the solution by a three-state recursion and a convex region that the merge rule maps into itself. This gives $O(1.7204^n)$ (Theorem 4.9); on the uniform $(2,2)$-lobsters the algorithm needs time $\Omega(1.7192^n)$.
- 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 4.12).
- In a black-box model, deterministic algorithms cannot in general match the randomized ones: on the instances that satisfy the mean bounds of Connected Dominating Set at solution size $n/2$, every deterministic black-box algorithm has cost at least $1.7547^n/\operatorname{poly}(n)$, against $\sqrt{3}^n \operatorname{poly}(n)$ for the randomized algorithm (Section 4.5). But the number of Steiner terminals is at most an affine function of the sample with nonnegative integer weights, and balanced set-inclusion families derandomize the exponential moments of such *integer affine certificates* (Section 4.5).

Bound	Type	Space	Year	Source	Method
1.9407^n	deterministic	polynomial	2008	[19]	measure and conquer
1.8966^n	deterministic	polynomial	2011	[17]	measure and conquer
1.8962^n	deterministic	polynomial	2014	[6]	measure and conquer
1.8619^n	deterministic	polynomial	2011	[2]	branching, Steiner tree
1.8139^n	deterministic	exponential	2012	[5]	algorithm for digraphs
1.7449^n	randomized	polynomial		Theorem 4.8	mean number of terminals
$\mathbf{1.7361^n}$	**deterministic**	polynomial		Theorem 4.14	affine certificate
1.7204^n	randomized	polynomial		Theorem 4.9	exponential moments
$\mathbf{1.6991^n}$	**randomized**	polynomial		Theorem 4.12	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 (Section 1.3).

Component-count running times. Our extension algorithm for Connected Feedback Vertex Set is a Cut & Count algorithm, built on the $O^*(3^k)$ algorithm of Cygan et al. [12], 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 W_S from below by a partition function on a spanning tree of S (Corollary 5.4), and a weighted form of a lower bound of Csikvári and Lin [9] 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 5.2). The main theorem then gives time $O^*(1.8^n)$ (Theorem 5.7), against the baseline $O(1.9526^n)$ obtained by balancing $O^*(3^k)$ against enumeration; the analysis is tight for this algorithm. More generally, for a constant $c > 1$ and a problem whose solutions induce connected subgraphs, every extension algorithm with running time $O^*(c^{k-|X|}2^{\mathrm{comp}(G[X])})$ gives running time $O^*(((4c-3)/(2c-1))^n)$ (Corollary 5.5). The case $c = 2$, where a sharper analysis gives 1.6444 (Lemma 5.8), covers Connected Vertex Cover, with Cygan's weighted $O^*(2^k)$ algorithm [10] as the extension step, and Graph Motif (Theorem 5.9).

Theorem 1.3 (Theorems 5.7 and 5.9 and Section 5.3)**.** (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* $w_{\max} \leq n^{O(1)}$*.*

The analyses of Algorithm RS-θ and of the deterministic algorithm extend to the variants of Connected Dominating Set in Table 2; similar finishing steps give the bounds for Directed Steiner Tree and Paired Domination (Section 6).

1.3 Techniques and related work

Monotone Local Search. Esmer et al. [14, 16] developed approximate Monotone Local Search, after Lee's thesis [33] had applied Monotone Local Search with exact and approximate extension algorithms to approximation, and showed it to be optimal in the extension-oracle model described

Problem	Previous	New	Type	Source
Main applications of the component-count principle				
Connected Feedback Vertex Set	1.9526^n [12]*	1.8^n	rand.	Thm. 5.7
Connected Vertex Cover	1.7088^n [28]	1.6444^n	rand.	Thm. 5.9
weighted	1.7088^n [10]*	1.6444^n	rand.	Thm. 5.9
Further consequences				
Graph Motif	1.7088^n [7]*	1.6444^n	rand.	Thm. 5.9
Connected r-Dominating Set, $r \geq 2$	1.8619^n [2]†	1.7361^n	det.	Section 6
		1.7204^n	rand.	Section 6
Connected Dominating Set, weighted	1.9896^n [1]†	1.7361^n	det.	Section 6
		1.7204^n	rand.	Section 6
Weakly Connected Dominating Set	1.8619^n [2]†	1.7361^n	det.	Section 6
Directed Max Leaf Spanning Tree	1.9044^n [5]	1.7361^n	det.	Section 6
Directed Steiner Tree	$2^{n/2}$ (see Section 6)	1.3727^n	rand.	Section 6
Paired Domination	1.7159^n [4]	1.6302^n	rand.	Section 6

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 (Section 1.3). Weighted problems have polynomially bounded integer weights. *: baseline derived by balancing the cited algorithm against enumeration. †: by a reduction to the cited algorithm (see the full version [25]). Undirected Steiner Tree has the bound 1.3533^n [21, Theorem 5].

above [15]. Esmer and Kulik [13] sample with a black box in which the number of correct samples is random, but the running time of each call is fixed by the algorithm. Theorem 2.5 reads a sample-dependent running-time exponent before the extension call, which is information outside the oracle model of [15], so their lower bounds do not apply; for product-form running times it gives the classical bound up to a polynomial factor, so it does not contradict them.

Restarts, cutoffs and generating functions. The probabilistic core of Theorem 2.5 has counterparts in restart strategies with cutoffs for Las Vegas algorithms [35, 3, 42], in the Markov cutoff of Zamir [45], and in the exponential moment of the number of guessed variables in the analysis of Paturi et al. [38]. Generating functions were already used in Monotone Local Search for product-form running times [18, 27]; our W_S is a different series, the binomial series over the subsets of one solution.

Certificates. We prove the lower bounds on W_S for RS-θ and RSC by antinorm certificates in a cumulative cone, adapting the method that Rote [41] introduced to count minimal dominating sets in trees, generalized by Rosenfeld [40], from upper to lower bounds; antinorms in the sense of Guglielmi and Protasov [29] take the place of norms. Measure-and-conquer analyses [20] bound the number of leaves of a search tree from above; our certificates bound generating functions from below. The certificates are found by computer and checked by programs in exact rational arithmetic, or in ball arithmetic with outward rounding where logarithms, exponentials and square roots occur; the two certificates of RS-θ are checked by three independent programs and the certificate of RSC by two, and 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 [26]. The base 1.8 for Connected Feedback Vertex Set has a proof without computation.

A caveat. Koutis and Williams [31] 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 [6, footnote (i)], [46, footnote 1], and the k-leaf application does not appear in the journal version [32]. We know of no repair; our 1.6991^n is below 1.7088^n in any case.

Derandomization. Esmer et al. [14, 16] generalize the set-inclusion families of [18] 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 $2^{o(n)}$ [15, 16]. Our black-box barrier (Section 4.5) uses the same planting idea and shows that, for sample-dependent running times, the two do not match in general.

2 Monotone Local Search with sample-dependent running times

2.1 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, and 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; decimal bases are rounded up as explained in Section 1, and exact and parametric bases, such as 1.8^n, are stated with the O^*-notation. 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 $\log M$. For an optimization problem, an algorithm with one-sided error never outputs a value better than the optimum, and outputs the optimum with the stated probability.

The framework is that of [18]: a problem is given by a polynomial-time membership test for its solutions.

Definition 2.1 (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 $X \cup Y$.

Definition 2.2 (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 $k \geq |X|$ 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 $b, c, d \geq 1$; 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. Once I and a target size k are fixed, we write $g(X) = g(I, X, k)$ for $|X| \leq k$. The algorithms below compare $g(I, X, k)$ with thresholds that are integers or rationals of polynomial bit length, and such a comparison is decided in time $N^{O(1)}$ for all running-time exponents of this paper; running-time exponents that can only be approximated are rounded up by at most 1, which costs a factor 2 in the moments below (see the full version [25]).

Definition 2.3 (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 $0 \leq t \leq k$ and real $y \geq 0$ 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 $\mu_t(S) \leq \binom{k}{t}$, because $g \geq 0$.

2.2 The algorithm and the main theorem

The algorithm monitors 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 $t \in \{0, \ldots, k\}$ and an integer threshold $\theta \geq 0$. A *trial* of the pair (t, θ) draws a uniformly random t-set X. If $g(I, X, k) \leq \theta$, the trial runs $\mathcal{A}$ on (I, X, k), in time $O^*(2^\theta)$; otherwise it discards X in polynomial time. To balance fast and successful trials, 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, and thresholds above a cap $\Theta \approx \psi n$ are not used. The *rate* $\psi \geq 0$ is the exponent of the target running time, the *loss factor* $\Gamma \geq 1$ weakens the success condition $(\star)$ by a factor, typically polynomial, that an application may lose in its lower bound on $\mu_t(S)$, and the *confidence* $\kappa > 0$ sets the error probability $e^{-\kappa}$.

Algorithm 2.4 (MLS-$\theta(\psi, \Gamma, \kappa)$)

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

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

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

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

1. draw X uniformly from $\binom{U_I}{t}$ and compute $g(I, X, k)$;
2. if $g(I, X, k) \leq \theta$, 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.

Theorem 2.5 (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 $\psi \geq 0$, $\Gamma \geq 1$ 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 $k \leq n$,*

(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*

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

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

Running the algorithm for $k = 0, 1, \ldots, n$ decides Φ-Subset with one-sided error 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 $k = |S|$. For optimization problems, the algorithms of later sections run it for $k = 1, 2, \ldots, n$ and output the first accepted size, which is never below the optimum and is the optimum with probability at least $1 - e^{-\kappa}$.

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 $\mu_t(S) \geq \binom{n}{t} 2^{-\psi n}/\Gamma$, some threshold $\theta \leq \Theta$ is passed by at least $2^{\theta} \mu_t(S)/(4(\Theta+1))$ sets $X \in \binom{S}{t}$. *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 2.5 can use it as well. We write $2^{-\infty} = 0$.

Lemma 2.6 (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 $\theta \in \{0, \ldots, \Theta\}$ 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 $\theta_\xi \in \{0, \ldots, \Theta\}$, 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 of Theorem 2.5. 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 $|Y|\le k-t$, 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 $g(I,X,k)\le\theta$, so the trial takes time $2^{g(I,X,k)}N^{O(1)}\le 2^{\theta}N^{O(1)}$. The trials of a pair (t,θ) therefore take time at most

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

and $\Theta\le\psi n+\log\Gamma+3$ gives $2^{\Theta}\le 8\Gamma 2^{\psi n}$. There are $(k+1)(\Theta+1)$ pairs. Since κ, Γ and ψ are at most $N^{O(1)}$ and $n\le N$, also $\Theta+1\le N^{O(1)}$, 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 $\mu_t(S)\ge\binom{n}{t}2^{-\psi n}/\Gamma$. 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, $\Pr[\xi\le\theta]=q_\theta/\binom{n}{t}$, where q_θ is the number of $X\in\binom{S}{t}$ with $g(I,X,k)\le\theta$. The hypothesis of Lemma 2.6(a) holds, because $\Theta\ge\psi n+\log\Gamma+1$ gives $2^{-\Theta}\le 2^{-\psi n}/(2\Gamma)\le\mu_t(S)/(2\binom{n}{t})$. So some $\theta\in\{0,\dots,\Theta\}$ satisfies

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

A trial of the pair (t,θ) draws an $X\in\binom{S}{t}$ with $g(I,X,k)\le\theta$ 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 $m_{t,\theta}\ge M_{t,\theta}=4\kappa\Gamma(\Theta+1)2^{\psi n-\theta}\ge\kappa/p$ trials of this pair are independent, so all of them fail with probability at most $(1-p)^{m_{t,\theta}}\le e^{-pm_{t,\theta}}\le e^{-\kappa}$.

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

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

so $W_S(y)\ge(1+y)^n2^{-\psi n}/\Gamma$ implies $(\star)$. □

2.3 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. The two forms agree up to a factor $n+1$.

Lemma 2.7 (Generating function versus exponential moments)**.** *For every instance I, running-time exponent g and solution S with $|S|=k$,*

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

Proof. The first inequality is the last display of the proof of Theorem 2.5, read as $(1+y)^n/W_S(y)\ge 1/\max_t(\mu_t(S)/\binom{n}{t})$; note that $W_S(y)\ge\mu_0(S)>0$. For the second, fix t; we exhibit $y\ge 0$ with $(1+y)^n/W_S(y)\le(n+1)\binom{n}{t}/\mu_t(S)$. If $t=0$, take $y=0$. 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 $W_S(y)\ge\mu_t(S)\,y^t$,

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

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 $y = n$ gives $(1+y)^n/W_S(y) \leq (1+1/n)^n/\mu_n(S) \leq (n+1)/\mu_n(S)$. □

2.4 Necessity and optimality in its class

Exponential moments are the right quantity for Algorithm MLS-θ. The full version [25] proves two statements, which we only describe here. First, condition ($\star$) is necessary up to a polynomial factor: if the left-hand side of ($\star$) is smaller than $2^{-\psi n}$ by a suitable polynomial factor, then the algorithm runs $\mathcal{A}$ on a subset of S with probability at most $1/4$, by Lemma 2.6(b) and Markov's inequality. Second, Algorithm MLS-θ is optimal, up to polynomial factors and on every instance, among all algorithms that perform trials which draw uniform samples of a chosen size, run $\mathcal{A}$ only on samples whose running-time exponent is at most a chosen threshold θ, and are charged 2^θ per trial, even when the sizes and thresholds are chosen adaptively: such an algorithm with total charge B answers YES with probability at most $\mathsf{B} \max_{t,\theta} 2^{-\theta} q_\theta(t)/\binom{n}{t}$, where $q_\theta(t)$ is the number of t-sets that extend to a solution of size at most k and have running-time exponent at most θ, while Algorithm MLS-θ with $\Gamma = 1$ and $2^{\psi n}$ equal to the reciprocal of this maximum answers YES with probability at least $1 - e^{-\kappa}$. So a sharper analysis of Algorithm MLS-θ cannot certify a smaller running time, and further gains need trials outside this class, such as the closure sampling of Section 4.4.

2.5 Budgeted trials

The proof of Theorem 2.5 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. The following version for arbitrary budgeted trials covers randomized extension algorithms and the closure sampling of Section 4.4. 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 2.8 (Budgeted trials)

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

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

Theorem 2.9 (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 2.8, 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 $k \leq n$, every $\iota \in \mathcal{T}_{I,k}$ and every integer $\theta \geq 0$*

(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$] \geq \Pr[\xi_\iota \leq \theta]$ for every integer $\theta \geq 0$.*

Let ψ, Γ, κ be as in Theorem 2.5. Then Algorithm 2.8, run with these trials and parameters, runs in time $2^{\psi n}N^{O(1)}$ and space $N^{O(1)}+\mathrm{sp}$, 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*

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

Proof. Time, space and soundness follow from (T1) and (T2) exactly as items (i) and (ii) of Theorem 2.5; the number of pairs is now $|\mathcal{T}|(\Theta+1)\leq N^{O(1)}$, as $\mathcal{T}$ is listed in time $N^{O(1)}$. Fix ι with $\mathbb{E}[2^{-\xi_\iota}]\geq 2^{-\psi n}/\Gamma$. Then $2^{-\Theta}\leq 2^{-\psi n}/(2\Gamma)\leq \mathbb{E}[2^{-\xi_\iota}]/2$, so Lemma 2.6(a) gives $\theta\leq\Theta$ with $\Pr[\xi_\iota\leq\theta]\geq 2^{\theta}\mathbb{E}[2^{-\xi_\iota}]/(4(\Theta+1))\geq 2^{\theta-\psi n}/(4\Gamma(\Theta+1))$. By (T3) each trial of (ι,θ) accepts with probability $\pi_{\iota,\theta}\geq\Pr[\xi_\iota\leq\theta]$, the trial count satisfies $m_{\iota,\theta}\geq M_{\iota,\theta}\geq\kappa/\pi_{\iota,\theta}$, and by independence all these trials fail with probability at most $(1-\pi_{\iota,\theta})^{m_{\iota,\theta}}\leq e^{-\kappa}$. □

Theorem 2.5 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. The next instance allows extension algorithms with one-sided error.

Corollary 2.10 (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 2.5 holds for $\mathcal{A}$ with Γ replaced by 2Γ in the cap Θ and in the trial counts $m_{t,\theta}$.*

Proof. Apply Theorem 2.9 with $\mathcal{T}=\{0,\dots,k\}$ and the trials of Algorithm MLS-θ with the randomized $\mathcal{A}$; (T1) and (T2) hold because the time bound holds for every outcome of the coins and $\mathcal{A}$ never accepts a NO-instance. Let ξ'_t equal ξ_t if an independent fair coin shows heads and ∞ otherwise. Then $\Pr[\mathrm{Tr}_{t,\theta}\text{ accepts}]\geq\frac{1}{2}\Pr[\xi_t\leq\theta]=\Pr[\xi'_t\leq\theta]$ and $\mathbb{E}[2^{-\xi'_t}]=\mu_t(S)/(2\binom{n}{t})$, so Theorem 2.9 with 2Γ in place of Γ gives the claim. □

3 Corollaries: mean exponents, linear savings and product form

This section derives earlier and intermediate forms of Monotone Local Search from Theorem 2.5. For a solution S with $|S|=k$ put

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

so that $M(S)\leq G(S)\leq(n+1)M(S)$ by Lemma 2.7. Condition $(\star)$ holds exactly when $2^{\psi n}\Gamma\geq M(S)$; thus, with $\Gamma=1$, $M(S)$ is the smallest running-time bound $2^{\psi n}$ (up to the factor $N^{O(1)}$) that Theorem 2.5 certifies for S, and $G(S)$ is the smallest one that the generating-function condition certifies.

3.1 Mean exponents

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.

Theorem 3.1 (Mean-exponent Monotone Local Search)**.** *Let $t(I,k) \in \{0,\dots,k\}$ and $\lambda(I,k) \geq 0$ be computable in time $N^{O(1)}$, with values $\lambda(I,k) \leq N^{O(1)}$, such that for every instance I, every $k \leq n$ and every $S \in \mathcal{F}_I$ with $|S| = k$, writing $t = t(I,k)$,*

$$\mathbb{E}_X[g(I,X,k)] \leq \lambda(I,k), \qquad X \text{ uniform in } \binom{S}{t}. \tag{3}$$

Then there is a randomized algorithm that, given I and $k \leq n$, runs in time $\binom{n}{t}\binom{k}{t}^{-1}2^{\lambda(I,k)}N^{O(1)}$ 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.*

Proof. If $n = 0$, the membership test decides the question. Otherwise define ψ by $2^{\psi n} = \binom{n}{t}\binom{k}{t}^{-1}2^{\lambda}$, with $t = t(I,k)$ and $\lambda = \lambda(I,k)$, and take $\Gamma = \kappa = 1$; then $0 \leq \psi \leq 1 + \lambda/n \leq N^{O(1)}$. For $S \in \mathcal{F}_I$ with $|S| = k$, Jensen's inequality for the convex function $x \mapsto 2^{-x}$ and (3) give

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

with X uniform in $\binom{S}{t}$. Hence $(\star)$ holds with $\Gamma = 1$, and Theorem 2.5 gives the claims. □

The same bound has a direct proof by the Markov-cutoff mechanism of Lemma 2 of Zamir [45]: use only the sample size t and the single threshold $\lambda + 1$; by Markov's inequality a sample inside S passes the threshold with probability at least $1/(\lambda+1)$.

3.2 Linear mean savings

Often the mean running-time exponent of a sample X of size t inside a solution of size k is at most $\beta n - \rho t + C$ for reals $\beta, \rho, C \geq 0$: at solution fraction $\sigma = k/n$, each further sampled element divides the probability that the sample lies inside the solution by about $1/\sigma$, and the running-time bound of an extension call by 2^{ρ}. For $\sigma \in (0,1)$, reals $\beta, \rho \geq 0$ and $p = 2^{-\rho}$ define

$$F^*(\sigma) = \begin{cases} \beta & \text{if } \sigma \leq p, \\ \beta + \sigma\log\dfrac{p}{\sigma} + (1-\sigma)\log\dfrac{1-p}{1-\sigma} = \beta - \sigma\rho + H(\sigma) + (1-\sigma)\log(1-p) & \text{if } \sigma > p. \end{cases} \tag{4}$$

By the weighted arithmetic–geometric mean inequality, $F^*(\sigma) \leq \beta$, with equality exactly when $\sigma \leq p$.

Corollary 3.2 (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 $\beta n + C \leq N^{O(1)}$. 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 $|S| = k$,*

$$\mathbb{E}_X[g(I,X,k)] \leq \beta n - \rho t + C, \qquad X \text{ uniform in } \binom{S}{t}. \tag{5}$$

Let $F^*(\sigma)$ be as in (4), and let $\kappa > 0$ be as in Theorem 2.5. Then:

(1) Every such S satisfies $G(S) \leq 2^{C + nF^*(\sigma)}$. 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 proof bounds the moments $\mu_t(S)$ from below by Jensen's inequality, as in the proof of Theorem 3.1, 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 (see the full version [25]).

3.3 Product-form running times and multivariate Monotone Local Search

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

$$g(I, X, k) = (n - |X|)\log b + (k - |X|)\log c + |X|\log d, \tag{6}$$

and we put

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

If $S \in \mathcal{F}_I$ with $|S| = k$ and g has the form (6) on all inputs (I, X, k) with $X \subseteq S$, then $\mu_t(S) = \binom{k}{t} b^{-(n-t)} c^{-(k-t)} d^{-t}$ for $0 \le t \le k$, and by the binomial theorem

$$W_S(y) = \sum_{t=0}^{k} \binom{k}{t} y^t\, b^{-(n-t)} c^{-(k-t)} d^{-t} = b^{-n} c^{-k} \Big(1 + \frac{ybc}{d}\Big)^k. \tag{7}$$

At $y^* = (c-1)d/(bc)$ we have $1 + y^* bc/d = c$, so $W_S(y^*) = b^{-n}$ and $(1+y^*)^n / W_S(y^*) = (b + d - d/c)^n$; at $y = 0$ the ratio is $b^n c^k \le (bc)^n$. Hence $G(S) \le \mathrm{base}(b,c,d)^n$. Conversely, the full version [25] shows that $\max_{0 \le k \le n} M(S) \ge \mathrm{base}(b,c,d)^n / ((n+1)\mathsf{K})$ for a constant K that depends only on b, c, d: for such running-time exponents the main theorem certifies, up to a polynomial factor, no smaller running-time bound for the worst solution size.

Corollary 3.3 (Multivariate Monotone Local Search)**.** *Let $b, c, d \ge 1$ 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*

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

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

Proof. The product-form exponent (6) is nonnegative, computable in time $N^{O(1)}$ (if $\log b$, $\log c$ or $\log d$ is irrational, rounded up by at most 1, which replaces Γ by 2), and by hypothesis a running-time exponent for the given extension algorithm. Fix $k \le n$ and put $2^\psi = \mathrm{base}(b,c,d)$ (with a rational rate in $[\psi, \psi + 1/n]$ if ψ is irrational), $\Gamma = 2$ and $\kappa = 1$. For every $S \in \mathcal{F}_I$ with $|S| = k$, the computation after (7) gives the generating-function condition of Theorem 2.5 with $2^{\psi n} = \mathrm{base}(b,c,d)^n$, at $y = y^*$ if $d \le bc$ and at $y = 0$ if $d \ge bc$. So Theorem 2.5 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 $bc \ge 1$, the case $d = 1$ gives $(1 + b - 1/c)^n N^{O(1)}$, the theorem of Gaspers and Lee [27], and the case $b = d = 1$ gives $(2 - 1/c)^n N^{O(1)}$, the randomized theorem of [18].

4 Connected Dominating Set and Max Leaf Spanning Tree

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 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, 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 asks for the maximum number of leaves of a spanning tree of a connected graph G. The following equivalence is folklore [8, 19, 17, 2].

Lemma 4.1 (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 $n = 2$.*

For the second claim, take a spanning tree of $G[S]$ and attach every vertex of $V \setminus S$ by one edge to a neighbor in S; this gives, whenever $n \geq 2$, a spanning tree of G in which every vertex outside S is a leaf. All algorithms of this section run, for $s = 1, \ldots, n$ in turn, a procedure that rejects s if no connected dominating set of size at most s exists and accepts $s = \gamma_c(G)$ (with probability at least $1 - e^{-\kappa}$, for the randomized ones), and output the first accepted s, or n if there is none; this output is never below $\gamma_c(G)$, and with Lemma 4.1 it also solves Max Leaf Spanning Tree, where graphs with $n \leq 2$ vertices are handled directly.

4.1 Red–blue states and the Steiner extension

Abu-Khzam, Mouawad and Liedloff [2] 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.

Definition 4.2 (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 $Y \subseteq B$ with $R \subseteq N[Y]$, $G[V(\mathcal{D}) \cup Y]$ connected and $V(\mathcal{D}) \cup Y \neq \emptyset$, 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 $|\mathrm{T}(J)| = |R| + |\mathcal{D}|$.

For a nonempty set $X \subseteq V$ let $\mathcal{D}_X$ be the family of vertex sets of the components of $G[X]$, put $J_X = (\mathcal{D}_X,\ V \setminus X,\ V \setminus N[X])$, and let $J_\emptyset = (\emptyset, V, V)$. The solutions of J_X are exactly the sets $Y \subseteq V \setminus X$ for which $X \cup Y$ is a connected dominating set, and

$$|\mathrm{T}(J_X)| = n - |N[X]| + \mathrm{comp}(G[X]) \qquad \text{for every } X \subseteq V, \tag{8}$$

an integer in $\{0, \ldots, n\}$; $|\mathrm{T}(J_\emptyset)| = n$, and $|\mathrm{T}(J_X)| = 1$ if X is a connected dominating set. A state is encoded as a directed Steiner tree instance in which each red vertex is a sink that can be entered only from a blue vertex dominating it, one dominating component is the root (if there is none, every blue vertex is tried as the first vertex of the solution), and the other components and the red

vertices are the terminals. Nederlof's inclusion–exclusion over branching walks [37], with arcs used only in their own direction [36, 30], solves it in time $2^{|K|}n^{O(1)}$ and polynomial space, for K the set of terminals (proofs in the full version [25]).

Definition 4.3 (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 $\mathrm{OPT}(J_X) \le k - |X|$.

Corollary 4.4 (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 4.3 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 2.2, with values in $\{0, \ldots, n\}$.*

This running-time exponent does not depend on k; we write $g(X) = |\mathrm{T}(J_X)|$.

4.2 Sampling inside a connected dominating set

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.

Lemma 4.5 (Tree inequality)**.** *Let F be a tree and let $X \subseteq V(F)$ be nonempty. Then*

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

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 $e_X + e_{XY} \le |X| + |Y| - 1$, with equality if and only if $F\{X\}$ is connected. Therefore

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

Lemma 4.6 (Pointwise form)**.** *Let G be a connected graph with $n \ge 2$ 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. Put $w(u) = d_F(u) - 1$ for $u \in S$. Then $\sum_{u \in S} w(u) = n - 2$, and for every nonempty $X \subseteq S$,*

$$|N[X]| - \mathrm{comp}(G[X]) \;\ge\; 1 + \sum_{x \in X} w(x), \qquad \textit{that is,} \qquad |\mathrm{T}(J_X)| \le n - 1 - w(X).$$

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 $\mathrm{comp}(G[X]) \le \mathrm{comp}(F[X])$ because $F[X]$ is a spanning subgraph of $G[X]$. Now Lemma 4.5 and (8) give the claim. $\square$

Lemma 4.7 (Mean number of terminals)**.** *Let G be a connected graph with $n \ge 2$ vertices, let S be a connected dominating set of G with $|S| = s$, let $1 \le t \le s$ and let X be uniform in $\binom{S}{t}$. Then*

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

Proof. Take F and w as in Lemma 4.6. Each $u \in S$ lies in X with probability t/s, so by linearity of expectation $\mathbb{E}[w(X)] = t(n-2)/s$, and Lemma 4.6 gives the claim. □

The bound of Lemma 4.7 cannot be improved in general: equality holds for every t when $G[S]$ is a star and every vertex of $V \setminus S$ has exactly one neighbor in S (see the full version [25]).

4.3 Algorithms RS and RS-θ

Algorithm RS runs, for $s = 1, \dots, n$, Algorithm MLS-$\theta(\psi_{\mathrm{RS}}, 2, \kappa)$ on (G, s) for Φ_{CDS} with the extension algorithm $\mathcal{A}_{\mathrm{St}}$ and $g(X) = |\mathrm{T}(J_X)|$, where $\psi_{\mathrm{RS}} = 0.80309062$, and outputs the first accepted s (or n). With $\sigma = s/n$, Lemma 4.7 gives the hypothesis (5) of Corollary 3.2 with $\beta = 1$, $\rho = 1/\sigma$ and $C = 1$ (for $t = 0$, $|\mathrm{T}(J_\emptyset)| = n$), and the resulting rate is

$$F^*(\sigma) = H(\sigma) + (1-\sigma)\log(1 - 2^{-1/\sigma}) \qquad (0 < \sigma < 1), \tag{9}$$

the second case of (4), since $2^{-1/\sigma} < \sigma$; at $\sigma = \frac{1}{2}$ the base $2^{F^*(\sigma)}$ is exactly $\sqrt{3}$. So every connected dominating set S of size $s < n$ satisfies $\max_t \mu_t(S)/\binom{n}{t} \geq 2^{-nF^*(\sigma)}/2$ by Corollary 3.2 and Lemma 2.7, and for $s = n$ we have $\mu_n(V) = \frac{1}{2}$. A check of F^* on $[0.2, 0.8]$ in ball arithmetic, with $F^*(\sigma) < H(\sigma) \leq H(0.2) < 0.73$ outside, shows $F^*(\sigma) \leq \log 1.744835$ for all $\sigma \in (0,1)$, and $F^*(\sigma) \leq 0.7807$ for $\sigma \in (0, 0.31] \cup [0.54, 1)$ (the full version [25]). As $2^{\psi_{\mathrm{RS}}} = 1.744835\ldots$, Theorem 2.5 with $\Gamma = 2$ gives:

Theorem 4.8 (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 $1 - e^{-\kappa}$.*

By the tightness of Lemma 4.7, the mean cannot be improved, but how it is used can: when the number of terminals fluctuates, Jensen's inequality $\mathbb{E}[2^{-|\mathrm{T}(J_X)|}] \geq 2^{-\mathbb{E}|\mathrm{T}(J_X)|}$ is strict. Algorithm RS-θ is Algorithm RS with the rate $\psi^* = 0.7827016$ in place of ψ_{RS}, and its analysis bounds W_S directly.

Theorem 4.9 (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 $1 - e^{-\kappa}$.*

The first step is a potential that is local along a spanning tree of the solution. An *owner map* of a connected dominating set S is a map $o : V \setminus S \to S$ with $o(u) \in N(u)$ for every $u \notin S$; let $\nu_v = |o^{-1}(v)|$ for $v \in S$. For a tree F with $V(F) = S$ and $X \subseteq S$ put

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

Lemma 4.10 (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 $|\mathrm{T}(J_X)| \leq \Pi_{F,\nu}(X)$ for every $X \subseteq S$. Consequently $W_S(y) \geq W_{F,\nu}(y)$ for every $y \geq 0$.*

Proof. By (8), $|\mathrm{T}(J_X)| = |(V \setminus S) \setminus N[X]| + |S \setminus N[X]| + \mathrm{comp}(G[X])$. (1) If $u \in V \setminus S$ is not in $N[X]$, then $o(u) \notin X$; so $|(V \setminus S) \setminus N[X]| \leq \sum_{v \in S \setminus X} \nu_v$. (2) If $v \in S$ is not in $N[X]$, then $N_F[v] \cap X = \emptyset$. (3) $F[X]$ is a spanning subgraph of $G[X]$, so $\mathrm{comp}(G[X]) \leq \mathrm{comp}(F[X])$. □

Since $\mathrm{comp}(F[X]) = |X| - e(F[X])$, the potential is a sum of terms that each depend on one vertex, its count ν_v and the sampled vertices of its closed neighborhood in F, so $W_{F,\nu}(y)$ is computed by a three-state recursion from the leaves of F (the state of a vertex records whether it is sampled and, if not, whether a child is), with a bilinear merge rule. Adapting the method of Rote [41] and Rosenfeld [40] from upper to lower bounds, we bound the recursion for all trees at once by a region $\mathrm{conv}(Z) + \mathcal{K}$, for a finite set Z of rational points and the *cumulative cone* $\mathcal{K} = \{P \in \mathbb{R}^3 : P_1 \geq 0,\ P_1 + P_2 \geq 0,\ P_1 + P_2 + P_3 \geq 0\}$, that the merge rule maps into itself up to known scales e_1 per tree vertex and e_2 per owned vertex, an antinorm in the sense of Guglielmi and Protasov [29]. Such a certificate gives $\max_t \mu_t(S)/\binom{n}{t} \geq c_Z 2^{-n(\log(1+y) - \sigma \log e_1 - (1-\sigma)\log e_2)}$ for every connected dominating set S with $|S| = \sigma n$, where $c_Z > 0$ depends only on Z. Two certificates, with $y = 69/200$ and $17/50$ and with 27 and 29 points, cover $\sigma \in [0.31, 0.54]$ with rate at most $0.78270152 < \psi^*$ and $c_Z \geq 0.62$, and the bound for RS covers the other σ; this proves Theorem 4.9 (details and the certificate data in the full version [25]). The analysis is close to tight for this algorithm: on the uniform $(2,2)$-lobsters, whose unique minimum connected dominating set is a spine path with two legs at every spine vertex, and which have two further pendant vertices at every leg, RS-θ run with a rate ψ succeeds on all of them with probability bounded away from 0 only if $2^\psi \geq 1.71927$ (the full version [25]).

4.4 Closure sampling: Algorithm RSC

On the instances where the analysis of RS-θ is nearly attained, some vertices of the solution have many neighbors inside the solution, and such a 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. For $X' \subseteq V$ let $C(X') = \{v \in V \setminus X' : |N(v) \cap X'| \geq 2\}$, and write $p_k := p_2$ if $k = 2$ and $p_k := p_3$ if $k \geq 3$.

Algorithm 4.11 (RSC)

Input: a connected graph G with $n \geq 1$ 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 $s = 1, 2, \ldots, n$, decide whether to accept s as follows, and output the first accepted s (and n if there is none). For every $t \in \{0, \ldots, s\}$ and every integer $\theta \in \{0, \ldots, \Theta\}$, 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 $|X| \leq s$ and $|\mathrm{T}(J_X)| = n - |N[X]| + \mathrm{comp}(G[X]) \leq \theta$, compute $\mathrm{OPT}(J_X)$ by Corollary 4.4, and accept s if $\mathrm{OPT}(J_X) \leq s - |X|$.

Theorem 4.12 (Algorithm RSC)**.** *For every constant $\kappa > 0$, Algorithm RSC runs in time $O^*(\varrho^n)$, where $\varrho = 990436373/582919832 = 1.6990\ldots < 1.6991$, 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*, can be solved by a randomized algorithm with one-sided error in time $O(1.6991^n)$ and polynomial space.*

The trials of RSC are budgeted trials in the sense of Theorem 2.9, with $\xi_t = |\mathrm{T}(J_X)|$ if the closed sample X lies in S and $\xi_t = \infty$ otherwise. Grouping by $|X'|$ reduces Theorem 4.12 to the

bound

$$\sum_{X' \subseteq S} y^{|X'|}\, \mathbb{E}_{\Upsilon}\left[\mathbf{1}[\Upsilon \subseteq S]\, 2^{-|\mathrm{T}(J_{X' \cup \Upsilon})|}\right] \;\geq\; 0.6060\, \hat{e}^{n}$$

for every connected dominating set S with $|S| = \gamma_{\mathrm{c}}(G)$, with the parameters of Algorithm 4.11. Its proof (the full version [25]) replaces the graph by a spanning tree of $G[S]$ with an owner map, bounds most outside factors by owner factors with an *exposure* $1 - p_\tau$ when the owner is in the closed sample and a *credit* when it is not, evaluates the result by a ten-state recursion, uses the minimality of S at the leaves, and bounds a relaxation of the recursion from below by an antinorm in an eight-dimensional cone. This certificate has 1024 rational points and about 3.7 million relations, each checked in exact rational arithmetic by two independently written programs; the reduction of the bound to these finite conditions is checked in Lean 4, where the conditions are checked as well, by compiled evaluation [26].

4.5 Derandomization and a deterministic algorithm

Fomin et al. [18] derandomize classical Monotone Local Search with set-inclusion families, at the price of a factor $2^{o(n)}$. With sample-dependent running times this is not enough, since the member of the family inside the solution may be one of the slowest samples. In a black-box model, an algorithm sees the instance only through calls $\mathrm{COST}(X, \ell)$, which returns $g(X, \ell)$ at cost 1, $\mathrm{EXT}(X, \ell)$, which decides extendability at cost $2^{g(X,\ell)}$, and membership tests at cost 1; computation is free. The randomized algorithms of Theorems 2.5 and 3.1 run in this model. An adversary in the style of the planting argument of [14, 15] plants a solution among the k-sets that a deterministic run leaves untouched. For the mean bounds $\lambda_0 = n$ and $\lambda_j = n - 1 - j(n-2)/k$ $(1 \leq j \leq k)$ of Lemma 4.7, this gives the following (proof in the full version [25]).

Corollary 4.13 (The mean bounds of Connected Dominating Set)**.** *Let n be even and $k = n/2$. Every deterministic black-box algorithm that answers* NO *on every instance with no set of size at most k, and* YES *on every instance that has a solution of size k and in which every solution S of size k satisfies the mean bounds $\mathbb{E}_{X \in \binom{S}{j}} g(X, k) \leq \lambda_j$ for all $j \in \{0, \ldots, k\}$, has cost at least $2^{H(1/4)n}/(6(n+1)(n+2)) \geq 1.7547^n/(6(n+1)(n+2))$ on some no-instance, while Algorithm* MLS-θ *with the parameters of Corollary 3.2(1) decides the same class with one-sided error at cost $\sqrt{3}^{\,n} n^{O(1)}$.*

This is a statement about the abstract class, not about Connected Dominating Set. By Lemma 4.6, $|\mathrm{T}(J_X)| \leq n-1-w(X)$ for nonempty $X \subseteq S$ with unknown nonnegative integer weights of known total $w(S) = n - 2$, an *integer affine certificate*. We list deterministically, in polynomial space, *balanced families*: set-inclusion families that contain, for every k-set S and every weighting of S, a t-subset X of S with $w(X) \geq \frac{t}{k} w(S)$. Running them for guesses of the sizes and total weights of the light and heavy weight classes of the unknown solution achieves deterministically, up to a factor $2^{O(n(\log\log n)^2/\log n)}$, the exponential-moment bound of the worst integer weight profile, which takes only the two values $\lfloor (n-2)/k \rfloor$ and $\lfloor (n-2)/k \rfloor + 1$. A rate certificate with explicit values of y, checked by one algebraic identity and ten rational inequalities in exact arithmetic, then gives the following (proofs in the full version [25]).

Theorem 4.14 (Deterministic Connected Dominating Set and MaxLeaf)**.** Connected Dominating Set*, and hence* Max Leaf Spanning Tree*, can be solved by a deterministic algorithm 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, where $\sqrt{5} - \frac{1}{2} = 1.7360\ldots$.*

Theorem 4.14 derandomizes the exponential-moment bound of the surrogate $n-1-w(X)$, not Theorem 2.5 for the true running-time exponent; it improves on Algorithm RS but not on RS-θ or RSC.

5 Component-count running times

This section treats problems whose solutions S induce connected subgraphs $G[S]$ and whose extension algorithms run in time $O^*\big(c^{k-|X|}\,2^{\mathrm{comp}(G[X])}\big)$ on a sample X, with $c=3$ for Connected Feedback Vertex Set and $c=2$ for Graph Motif and Connected Vertex Cover. A sample of size t can have t components, and the product-form bound $c^{k-t}2^t$ gives through Corollary 3.3 with $(b,c,d)=(1,c,2)$ the base $3-2/c\ge 2$ when $c\ge 2$. For a graph F and $X\subseteq V(F)$ we write $e(F[X])=|E(F[X])|$, and $H_e(\mathbb{P})=-\sum_z \mathbb{P}(z)\ln\mathbb{P}(z)$ is the entropy of a probability distribution $\mathbb{P}$ on a finite set, with the natural logarithm.

5.1 The partition function of a tree

Definition 5.1 (Tree partition function)**.** For a tree F and a real $\zeta\ge 0$ 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 $\lambda_{\max}(M_\zeta)=\frac{1}{2}\big(1+2\zeta+\sqrt{1+4\zeta^2}\big)$ be its largest eigenvalue.

Both eigenvalues of M_ζ are real and nonnegative, and $\lambda_{\max}(M_\zeta)>1$ for $\zeta>0$. 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. So $\mathcal{Z}_F(\zeta)$ is a weighted count of homomorphisms from F to a two-vertex target. For graphs without loops or weights, Csikvári and Lin [9, Theorem 4.3] proved that every tree T with m vertices has $\hom(T,G)\ge e^{\eta}\lambda^{m-1}$ homomorphisms into a connected graph G, where λ is the largest adjacency eigenvalue of G and $\eta\ge 0$ is an entropy; our proof follows theirs. Whether the path minimizes such counts exactly depends on the target [43, 24, 23]; 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 5.2 (Gibbs bound, after Csikvári and Lin)**.** *For every tree F with $k\ge 1$ vertices and every $\zeta>0$,*

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

Proof. Write $\lambda=\lambda_{\max}(M_\zeta)$, $w_0=1$, $w_1=\zeta$, $A_{ij}=2$ 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 $f(z)=\prod_{v\in V(F)}w_{z_v}\prod_{uv\in E(F)}A_{z_uz_v}>0$.

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 symmetric matrix has the same characteristic polynomial $x^2-(1+2\zeta)x+\zeta$ as M_ζ, so its largest eigenvalue is λ, with the eigenvector $(\sqrt{\zeta},\lambda-1)$, whose entries are positive. Let $\mathbf{v}$ be this vector normalized to $\mathbf{v}_0^2+\mathbf{v}_1^2=1$, and put

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

Then $p_{ij}>0$ and $\sum_j p_{ij}=(\mathbf{K}\mathbf{v})_i/(\lambda\mathbf{v}_i)=1$, so (p_{ij}) is a stochastic matrix. Moreover $\pi_{ij}:=\pi_i p_{ij}=\mathbf{v}_i\mathbf{K}_{ij}\mathbf{v}_j/\lambda$ is symmetric in i and j, so 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, every vertex has label distribution π, and every edge, with parent u and child v, has joint label distribution $\Pr[z_u = i, z_v = j] = \pi_{ij}$, which is symmetric with 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

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

For our $\mathbb{P}$, by its product form 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 $\ln p_{ij} = \ln \mathbf{K}_{ij} + \ln \mathbf{v}_j - \ln \mathbf{v}_i - \ln \lambda$ and (π_{ij}) is symmetric, the terms in $\mathbf{v}$ cancel, and $-\sum \pi_{ij} \ln p_{ij} = \ln \lambda - \sum \pi_{ij} \ln \mathbf{K}_{ij}$. Adding up, the terms in $\mathbf{K}$ cancel as well, and

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

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

Paths attain this bound up to a factor depending only on ζ: for a path F with k vertices, $\mathcal{Z}_F(\zeta) = \mathbf{d}^{\mathsf{T}} \mathbf{K}^{k-1} \mathbf{d}$ with $\mathbf{d} = (1, \sqrt{\zeta})$, so $\mathcal{Z}_F(\zeta) \le (1+\zeta)\lambda_{\max}(M_\zeta)^{k-1}$.

5.2 Component-count running times and Monotone Local Search

Definition 5.3 (Component-count bound)**.** Let $c \ge 1$ and $\gamma \ge 0$ 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 $|S| = k \ge 1$ there is a tree F with vertex set S such that

$$g(I, X, k) \;\le\; (k - |X|)\log c + |X| - e(F[X]) + \gamma \qquad \text{for every } X \subseteq S. \tag{10}$$

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 $\mathrm{comp}(G[X]) \le \mathrm{comp}(F[X]) = |X| - e(F[X])$. For $\sigma \in [0,1]$ and $y \ge 0$ put

$$\phi_{c,\sigma}(y) = \log(1+y) - \sigma \log \frac{\lambda_{\max}(M_{cy/2})}{c}. \tag{11}$$

Corollary 5.4 (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 $|S| = k \ge 1$, let F be a tree for S as in Definition 5.3, and put $\sigma = k/n$.*

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

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

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

(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 (10), 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 5.2 the second; the last form follows by dividing $(1+y)^n$ by this bound. (b) The number c is a root of $x^2-(1+2\zeta^*)x+\zeta^*$, because $c^2-c-\zeta^*(2c-1) = 0$. The other root is $\zeta^*/c = (c-1)/(2c-1) < c$, so $\lambda_{\max}(M_{\zeta^*}) = c$. Now (a) gives the claim, as $1+y^* = (4c-3)/(2c-1)$. (c) For $X = \emptyset$ the right-hand side of (10) is $k\log c + \gamma$; for $X = S$ it is $k-(k-1)+\gamma = 1+\gamma$, because F has $k-1$ edges. □

Corollary 5.5 (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 2.10, 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 2.5, Algorithm* MLS-$\theta(\psi,\Gamma,\kappa)$ *with* $\Gamma = 2^{1+\gamma}\Gamma_0$, *and with* 2Γ *in place of* Γ *in the cap and in the trial counts if* $\mathcal{A}$ *is randomized, 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 2.5 give the time, space and soundness; for a randomized $\mathcal{A}$ use Corollary 2.10, which doubles Γ in the cap and in the trial counts. For item (iii), let $S \in \mathcal{F}_I$ with $|S| = k$ and $\sigma = k/n$. If (R1) holds, then by Corollary 5.4(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 5.4(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 5.4(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 generating-function condition of Theorem 2.5 holds, 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 5.4(b). □

For $c = 3$ the closed-form base $(4c-3)/(2c-1) = 1.8$ is the best rate that Corollary 5.5 can certify, since at $\sigma = 26/45$ the infimum of $\phi_{3,\sigma}$ over y equals $\log 1.8$ and both $\sigma\log 3$ and $H(\sigma)$ are larger. For $c = 2$ the base $5/3$ is not optimal: the best rate is $2^{\psi_2} \approx 1.64429$ (numerical), attained at $\sigma \approx 0.71746$.

5.3 Applications

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, and $\mathrm{OPT}(G)$ is the minimum size of one. Running the Monte Carlo $3^k n^{O(1)}$-time algorithm of Cygan et al. [12] or enumerating all

k-subsets, whichever is faster, gives the folklore base 1.9526, and we are not aware of a published exact algorithm in time $O((2-\varepsilon)^n)$ for a constant $\varepsilon > 0$. Contracting each component of the sample to a forced vertex and following the iterative compression of [11, 12] with a Cut & Count test that keeps the forced vertices in the solution gives the following extension algorithm, for the running-time exponent

$$g(G, X, k) = (k - |X|) \log 3 + \operatorname{comp}(G[X]) \tag{12}$$

(construction and proof in the full version [25]).

Lemma 5.6 (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 $3^{k-|X|}2^{\operatorname{comp}(G[X])}n^{O(1)}$. 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 $3/4$.*

Theorem 5.7 (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 $\operatorname{OPT}(G)$, and it has size exactly $\operatorname{OPT}(G)$ with probability at least $1 - 1/e$.*

Proof. If G is a forest, output $\emptyset$. Otherwise $V(G)$ is a solution and every solution is nonempty. The extension algorithm of Lemma 5.6 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 as in (12). Every solution S induces a connected subgraph, so a spanning tree F of $G[S]$ gives the component-count bound (10) with constants $(3, 0)$. By Corollary 5.5 with $c = 3$, $y^* = 4/5$, $\zeta^* = 6/5$ and $\psi = \log 1.8$, Algorithm MLS-θ with $\kappa = 1$ and $\Gamma = 2^{1+\gamma}\Gamma_0 = 6$, where $\Gamma_0 = c/\min(1, \zeta^*) = 3$, hence with $2\Gamma = 12$ in the cap and in the trial counts as $\mathcal{A}$ is randomized, 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 $k = \operatorname{OPT}(G)$; each run takes time $1.8^n\, n^{O(1)}$ and polynomial space. Output the smallest set found, and $V(G)$ if none is found. □

For this algorithm the rate $\log 1.8$ cannot be lowered by a positive constant: the upper bound on $\mathcal{Z}_F$ for paths and the necessity of $(\star)$ give, for every constant $\varepsilon > 0$, an infinite family of graphs on which rates $\psi \leq \log 1.8 - \varepsilon$ succeed with probability $o(1)$ (the full version [25]).

Connected Vertex Cover, Graph Motif and weighted problems. For $c = 2$ the closed form gives only the base $5/3$, so we optimize y for each solution fraction.

Lemma 5.8 (Rate certificate for $c = 2$)**.** *Every $\sigma \in (0, 1]$ satisfies* (R1)*,* (R2) *or* (R3) *of Corollary 5.5 with $c = 2$, $\psi = 0.71755$ and $\Gamma_0 = 16$. Moreover $0.71755 < \log 1.6444$, so $2^{0.71755n}n^{O(1)} = O(1.6444^n)$.*

The proof covers $\sigma \leq 0.7175$ by (R1), $\sigma \geq 0.81$ by (R2), and the range between by 925 intervals of width 10^{-4} with explicit values y_j, on each of which $\phi_{2,\sigma}(y_j)$ is affine in σ; the finitely many inequalities are checked by computer in exact or ball arithmetic (the full version [25]). Contracting each component of the sample to a vertex with a pendant vertex and running Cygan's deterministic algorithm [10], which computes a minimum-weight connected vertex cover of size at most b in time $2^b n^{O(1)}$ for polynomially bounded integer weights, gives an extension algorithm for Connected Vertex Cover in time $2^{k-|X|+\operatorname{comp}(G[X])}n^{O(1)}$; contracting each component to a vertex with a fresh color and running the randomized $O^*(2^k)$ algorithm of Björklund, Kaski and Kowalik [7] does the same for Graph Motif, with one-sided error (the full version [25]).

Theorem 5.9 (Connected Vertex Cover and Graph Motif)**.** *For vertex weights in* $\{1, \ldots, w_{\max}\}$ *with* $w_{\max} \leq n^{O(1)}$*, a minimum-weight connected vertex cover of a connected graph with* $n \geq 2$ *vertices can be computed by a randomized algorithm with one-sided error in time* $O(1.6444^n)$ *and polynomial space.* Graph Motif *can be solved by a randomized algorithm with one-sided error in time* $O(1.6444^n)$ *and polynomial space.*

The best published bound for Connected Vertex Cover in terms of n is $O(1.7088^n)$ [28, Corollary 1]. With weights, enumerating the minimal connected vertex covers gives $O^*(1.8668^n)$ [28, Theorem 3], and balancing Cygan's Theorem 3 [10] against enumeration gives $O(1.7088^n)$, a combination we have not seen stated. 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, in polynomial space and with one-sided error (the full version [25]).

6 Further applications

The analyses of RS-θ and of the deterministic algorithm use the graph only through a spanning tree of the solution and an assignment of every outside vertex to a solution vertex that dominates it, so they extend, with the same bases 1.7204 (randomized) and 1.7361 (deterministic), to Connected r-Dominating Set with r part of the input, to vertex-weighted Connected Dominating Set and maximum leaf weight, to Weakly Connected Dominating Set and to Directed Max Leaf Spanning Tree (the last two deterministically). Closure sampling is claimed for none of these variants. Sampling non-terminals for Directed Steiner Tree, with the number of source components left by the sample as the running-time exponent, gives randomized time $O(1.3727^n)$, against $2^{(n-1)/2}n^{O(1)} = O^*(2^{n/2})$ for the better of inclusion–exclusion and enumeration; and a finishing step with the Tutte matrix [44] gives $O(1.6302^n)$ for Paired Domination. Table 2 lists the previous bounds; statements and proofs are in the full version [25].

7 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 (Theorem 2.5); the criterion is necessary for Algorithm MLS-θ, and the algorithm is optimal within its class (Section 2.4). The gains come from extension algorithms whose running time depends on the structure of the sample. Several questions remain open.

1. *Derandomization.* 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, which are not separable over the blocks of a product construction.

2. *Better trials for Connected Dominating Set.* With the same extension algorithm and running-time exponent, further gains must come from trials outside the class of Algorithm MLS-θ: a non-uniform sampling distribution, 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. We do not know whether any of these improves on Theorem 4.12.

3. *Component-count running times.* For $c = 3$ the base 1.8 is exact for Algorithm MLS-θ, 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 2.5 certifies for a solution whose spanning tree is a path, but we have no family of instances of Graph Motif or Connected Vertex Cover on which this value is tight.

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 $r \geq 2$?

5. *Other problems and enumeration.* It would be interesting to characterize the problems for which Theorem 2.5 is exponentially better than every bound from a product-form running-time exponent. Fomin et al. [18] 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 [39], which was used to build and check the certificate for Theorem 4.12. 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.

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 and the Lean development, and drafted and revised the text; Claude also condensed the full version into this short version.

- 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 4.4), 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 (proved in the full version [25]) and the product form of multivariate Monotone Local Search, which were the basis of an Honours thesis [34]. The author verified the correctness and originality of all content, including references.

References

[1] Faisal N. Abu-Khzam, Henning Fernau, Benjamin Gras, Mathieu Liedloff, and Kevin Mann. Enumerating minimal connected dominating sets. *SIAM J. Discrete Math.*, 39(2):946–972, 2025. doi:10.1137/23M1611233.

[2] Faisal N. Abu-Khzam, Amer E. Mouawad, and Mathieu Liedloff. An exact algorithm for connected red-blue dominating set. *J. Discrete Algorithms*, 9(3):252–262, 2011. doi:10.1016/j.jda.2011.03.006.

[3] 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.

[4] Nikita Andreev, Ivan Bliznets, Madhumita Kundu, Saket Saurabh, Vikash Tripathi, and Shaily Verma. Parameterized complexity of paired domination. In *IWOCA 2024*, pages 523–536, 2024. doi:10.1007/978-3-031-63021-7_40.

[5] Daniel Binkele-Raible and Henning Fernau. An exact exponential-time algorithm for the Directed Maximum Leaf Spanning Tree problem. *J. Discrete Algorithms*, 15:43–55, 2012. doi:10.1016/j.jda.2012.03.006.

[6] Daniel Binkele-Raible and Henning Fernau. A parameterized measure-and-conquer analysis for finding a k-leaf spanning tree in an undirected graph. *Discrete Math. Theor. Comput. Sci.*, 16(1):179–200, 2014. doi:10.46298/dmtcs.1256.

[7] Andreas Björklund, Petteri Kaski, and Łukasz Kowalik. Constrained multilinear detection and generalized graph motifs. *Algorithmica*, 74(2):947–967, 2016. doi:10.1007/s00453-015-9981-1.

[8] Yair Caro, Douglas B. West, and Raphael Yuster. Connected domination and spanning trees with many leaves. *SIAM J. Discrete Math.*, 13(2):202–211, 2000. doi:10.1137/S0895480199353780.

[9] Péter Csikvári and Zhicong Lin. Graph homomorphisms between trees. *Electron. J. Combin.*, 21(4):P4.9, 2014. doi:10.37236/4096.

[10] Marek Cygan. Deterministic parameterized connected vertex cover. In *SWAT 2012*, pages 95–106, 2012. doi:10.1007/978-3-642-31155-0_9.

[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. arXiv preprint, 2011. arXiv:1103.0534.

[12] 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 Trans. Algorithms*, 18(2):17:1–17:31, 2022. doi:10.1145/3506707.

[13] Baris Can Esmer and Ariel Kulik. Sampling with a black box: Faster parameterized approximation algorithms for vertex deletion problems. In *ICALP 2025*, pages 39:1–39:20, 2025. doi:10.4230/LIPIcs.ICALP.2025.39.

[14] Baris Can Esmer, Ariel Kulik, Dániel Marx, Daniel Neuen, and Roohani Sharma. Faster exponential-time approximation algorithms using approximate monotone local search. In *ESA 2022*, pages 50:1–50:19, 2022. doi:10.4230/LIPIcs.ESA.2022.50.

[15] Baris Can Esmer, Ariel Kulik, Dániel Marx, Daniel Neuen, and Roohani Sharma. Optimally repurposing existing algorithms to obtain exponential-time approximations. In *SODA 2024*, pages 314–345, 2024. doi:10.1137/1.9781611977912.13.

[16] 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.

[17] 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. *Theoret. Comput. Sci.*, 412(45):6290–6302, 2011. doi:10.1016/j.tcs.2011.07.011.

[18] Fedor V. Fomin, Serge Gaspers, Daniel Lokshtanov, and Saket Saurabh. Exact algorithms via monotone local search. *J. ACM*, 66(2):8:1–8:23, 2019. doi:10.1145/3284176.

[19] 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.

[20] Fedor V. Fomin, Fabrizio Grandoni, and Dieter Kratsch. A measure & conquer approach for the analysis of exact algorithms. *J. ACM*, 56(5):25:1–25:32, 2009. doi:10.1145/1552285.1552286.

[21] 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.

[22] Fedor V. Fomin and Dieter Kratsch. *Exact Exponential Algorithms*. Springer, 2010. doi:10.1007/978-3-642-16533-7.

[23] David Galvin, Phillip Marmorino, Emily McMillon, JD Nir, and Amanda Redlich. Hoffman–London graphs: When paths minimize H-colorings among trees. arXiv preprint, 2025. arXiv:2512.23828.

[24] David Galvin, Emily McMillon, JD Nir, and Amanda Redlich. Long paths need not minimize H-colorings among trees. arXiv preprint, 2025. arXiv:2510.18770.

[25] Serge Gaspers. Monotone Local Search with Sample-Dependent Running Times. Full version, arXiv preprint, 2026. arXiv:2611.01774.

[26] 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.

[27] Serge Gaspers and Edward J. Lee. Exact algorithms via multivariate subroutines. In *ICALP 2017*, pages 69:1–69:13, 2017. doi:10.4230/LIPIcs.ICALP.2017.69.

[28] Petr A. Golovach, Pinar Heggernes, and Dieter Kratsch. Enumeration and maximum number of minimal connected vertex covers in graphs. *European J. Combin.*, 68:132–147, 2018. doi:10.1016/j.ejc.2017.07.015.

[29] Nicola Guglielmi and Vladimir Protasov. Exact computation of joint spectral characteristics of linear operators. *Found. Comput. Math.*, 13(1):37–97, 2013. doi:10.1007/s10208-012-9121-0.

[30] Mark Jones, Daniel Lokshtanov, Maadapuzhi Sridharan Ramanujan, Saket Saurabh, and Ondřej Suchý. Parameterized complexity of directed Steiner tree on sparse graphs. *SIAM J. Discrete Math.*, 31(2):1294–1327, 2017. doi:10.1137/15M103618X.

[31] Ioannis Koutis and Ryan Williams. Limits and applications of group algebras for parameterized problems. In *ICALP 2009*, pages 653–664, 2009. doi:10.1007/978-3-642-02927-1_54.

[32] Ioannis Koutis and Ryan Williams. Limits and applications of group algebras for parameterized problems. *ACM Trans. Algorithms*, 12(3):31:1–31:18, 2016. doi:10.1145/2885499.

[33] Edward J. Lee. *Exponential Time Algorithms via Separators and Random Subsets*. PhD thesis, UNSW Sydney, 2021. doi:10.26190/unsworks/22740.

[34] Edward Lu. The k-leaf spanning tree problem. Honours thesis, UNSW Sydney, 2020.

[35] Michael Luby, Alistair Sinclair, and David Zuckerman. Optimal speedup of Las Vegas algorithms. *Inform. Process. Lett.*, 47(4):173–180, 1993. doi:10.1016/0020-0190(93)90029-9.

[36] Neeldhara Misra, Geevarghese Philip, Venkatesh Raman, Saket Saurabh, and Somnath Sikdar. FPT algorithms for Connected Feedback Vertex Set. *J. Comb. Optim.*, 24(2):131–146, 2012. doi:10.1007/s10878-011-9394-2.

[37] Jesper Nederlof. Fast polynomial-space algorithms using inclusion-exclusion. *Algorithmica*, 65(4):868–884, 2013. doi:10.1007/s00453-012-9630-x.

[38] Ramamohan Paturi, Pavel Pudlák, Michael E. Saks, and Francis Zane. An improved exponential-time algorithm for k-SAT. *J. ACM*, 52(3):337–364, 2005. doi:10.1145/1066100.1066101.

[39] PVC (Research Infrastructure), UNSW Sydney. Katana. UNSW, Sydney, 2010. doi:10.26190/669X-A286.

[40] Matthieu Rosenfeld. The growth rate over trees of any family of sets defined by a monadic second order formula is semi-computable. In *SODA 2021*, pages 776–795, 2021. doi:10.1137/1.9781611976465.49.

[41] Günter Rote. The maximum number of minimal dominating sets in a tree. In *SODA 2019*, pages 1201–1214, 2019. doi:10.1137/1.9781611975482.73.

[42] Kevin Scaman. Black-box acceleration of Las Vegas algorithms and algorithmic reverse Jensen's inequalities. arXiv preprint, 2023. arXiv:2304.11017.

[43] Alexander Sidorenko. A partially ordered set of functionals corresponding to graphs. *Discrete Math.*, 131(1–3):263–277, 1994. doi:10.1016/0012-365X(94)90388-3.

[44] William Thomas Tutte. The factorization of linear graphs. *J. London Math. Soc.*, s1-22(2):107–111, 1947. doi:10.1112/jlms/s1-22.2.107.

[45] Or Zamir. The wrong direction of Jensen's inequality is algorithmically right. In *ICALP 2023*, pages 107:1–107:10, 2023. doi:10.4230/LIPIcs.ICALP.2023.107.

[46] Meirav Zehavi. The k-leaf spanning tree problem admits a klam value of 39. arXiv preprint, version 2, 2015. arXiv:1502.07725.