Isolation Schemes for Problems on Decomposable Graphs

Jesper Nederlof
Utrecht University, The Netherlands

Michał Pilipczuk
University of Warsaw, Poland

Céline M. F. Swennenhuis
Eindhoven University of Technology, The Netherlands

Karol Węgrzycki
Saarland University, Saarbrücken, Germany
Max Planck Institute for Informatics, Saarbrücken, Germany

Abstract

The Isolation Lemma of Mulmuley, Vazirani and Vazirani [Combinatorica'87] provides a self-reduction scheme that allows one to assume that a given instance of a problem has a unique solution, provided a solution exists at all. Since its introduction, much effort has been dedicated towards derandomization of the Isolation Lemma for specific classes of problems. So far, the focus was mainly on problems solvable in polynomial time.

In this paper, we study a setting that is more typical for NP-complete problems, and obtain partial derandomizations in the form of significantly decreasing the number of required random bits. In particular, motivated by the advances in parameterized algorithms, we focus on problems on decomposable graphs. For example, for the problem of detecting a Hamiltonian cycle, we build upon the rank-based approach from [Bodlaender et al., Inf. Comput.'15] and design isolation schemes that use

- $\mathcal{O}(t \log n + \log^2 n)$ random bits on graphs of treewidth at most t;
- $\mathcal{O}(\sqrt{n})$ random bits on planar or H-minor free graphs; and
- $\mathcal{O}(n)$-random bits on general graphs.

In all these schemes, the weights are bounded exponentially in the number of random bits used. As a corollary, for every fixed H we obtain an algorithm for detecting a Hamiltonian cycle in an H-minor-free graph that runs in deterministic time $2^{\mathcal{O}(\sqrt{n})}$ and uses polynomial space; this is the first algorithm to achieve such complexity guarantees. For problems of more local nature, such as finding an independent set of maximum size, we obtain isolation schemes on graphs of treedepth at most d that use $\mathcal{O}(d)$ random bits and assign polynomially-bounded weights.

We also complement our findings with several unconditional and conditional lower bounds, which show that many of the results cannot be significantly improved.

2012 ACM Subject Classification Theory of computation → Fixed parameter tractability

Keywords and phrases Isolation Lemma, Derandomization, Hamiltonian Cycle, Exact Algorithms

Digital Object Identifier 10.4230/LIPIcs.STACS.2022.50

Related Version *Full Version*: https://arxiv.org/abs/2105.01465

Funding *Jesper Nederlof*: Supported by the project CRACKNP (grant agreement No 853234) that has received funding from the European Research Council (ERC).
Michał Pilipczuk: Supported by the project TOTAL (grant agreement No 677651) that has received funding from the European Research Council (ERC).
Céline M. F. Swennenhuis: Supported by the Netherlands Organization for Scientific Research under project no. 613.009.031b.
Karol Węgrzycki: Supported by the project TIPEA (grant agreement No 850979) that has received funding from the European Research Council (ERC).

1 Introduction

Isolation is a procedure that allows to single out a unique solution to a given problem within a possibly larger solution space, thus effectively reducing the original problem to a variant where one may assume that if a solution exists, then there is a unique one. The classic Isolation Lemma of Mulmuley, Vazirani and Vazirani [26] can be used to achieve this at the cost of allowing randomization. In complexity theory, isolation is used to show that hard problems are not easier to solve on instances with unique solutions [35]. This idea has found numerous applications ranging from structural results in complexity theory (e.g. NL/poly $\subseteq \oplus$L/poly [37] or NL/poly = UL/poly [32]) to the design of parallel algorithms [26, 22, 17, 34].

Since obtaining a general derandomization of the Isolation Lemma is impossible by counting arguments [4, 8, 1], it is natural to ask whether the isolation step can be derandomized for specific problems with explicit representation. In this context, there has recently been an exciting progress in isolation for perfect matchings [2, 7, 13, 21, 3, 22], which culminated in an isolation scheme that uses $\mathcal{O}(\log^3 n)$ random bits, implying a quasi-NC algorithm for detecting a perfect matching [34].

In contrast to this, derandomization of isolation procedures for NP-complete problems is relatively less studied, and not because of a lack of motivation: Many contemporary fixed-parameter algorithms rely on the Isolation Lemma [25, 28, 5, 23, 24, 11, 38]. Usually, the isolation procedure is the only subroutine requiring randomness. Many of the algorithms mentioned above apply the Isolation Lemma in combination with a decomposition-based method such as Divide&Conquer or dynamic programming. This motivates us to study the following:

▶ **Main Question.** *How much randomness is required for isolating problems with decomposable structure?*

More concretely, we focus on graph problems where the underlying graph is *decomposable*, in the sense that it can be decomposed using small separators. Examples of such graphs are planar graphs or graphs of bounded treewidth. It is well-known that for many NP-complete problems, the nice structure of such graphs can be leveraged to solve these problems faster than in general graphs. We show that a similar phenomenon occurs when one considers the amount of randomness needed to isolate a single solution.

The model for isolation schemes. Suppose U is a finite set and $\omega\colon U \to \mathbb{N}$ is a weight function. For $X \subseteq U$ we write $\omega(X) \coloneqq \sum_{e\in X} \omega(e)$. For a set family $\mathcal{F} \subseteq 2^U$ we say that ω *isolates* $\mathcal{F}$ if there is exactly one set $S \in \mathcal{F}$ such that $\omega(S)$ is the minimum possible among the weights of the sets in $\mathcal{F}$. The classic Isolation Lemma of Mulmuley et al. [26] states that a weight function $\omega\colon U \to \{1, \ldots, 2|U|\}$ chosen uniformly at random isolates any family $\mathcal{F} \subseteq 2^U$ with probability at least $\frac{1}{2}$. Note that sampling such ω requires $\mathcal{O}(|U| \log |U|)$ random bits.

Most of our isolation schemes work in a very restricted model inspired by the discussion above, which we explain now. Intuitively, the scheme is not aware of the graph or its decomposition, but is only aware of the vertex count of the graph and the relevant width parameter, such as the treewidth or treedepth.

Formally, a *vertex selection problem* is a function $\mathcal{P}$ that maps every graph G to a family $\mathcal{P}(G) \subseteq 2^{V(G)}$ consisting of subsets of the vertex set of G. Edge selection problems are defined analogously: $\mathcal{P}(G)$ consists of subsets of $E(G)$. For example, we could define a

vertex selection problem $\mathsf{MIS}(\cdot)$ that maps every graph G to the family $\mathsf{MIS}(G)$ comprising all maximum-size independent sets in G, or an edge selection problem $\mathsf{HC}(\cdot)$ that maps every graph G to the family $\mathsf{HC}(G)$ comprising all (edge sets of) Hamiltonian cycles in G. Further, let $\mathcal{C}$ be a class of graphs, that is, a set of graphs that is invariant under isomorphism. For instance, $\mathcal{C}$ could be the class of planar graphs, or the class of graphs of treewidth at most k, for any fixed k. Then our definition of an isolation scheme reads as follows (here, we write $[n] \coloneqq \{1, \ldots, n\}$):

▶ **Definition 1.** *For a graph class $\mathcal{C}$, we say that a vertex selection problem $\mathcal{P}$ admits an* isolation scheme *on $\mathcal{C}$ with $\log \ell$ random bits and maximum weight W if for every $n \in \mathbb{N}$ there exist weight functions $\omega_1, \ldots, \omega_\ell \colon [n] \to [W]$ such that for every $G \in \mathcal{C}$ with vertex set $[n]$, ω_i isolates $\mathcal{P}(G)$ for at least half of the indices $i \in [\ell]$.*

Isolation schemes for edge selection problems are defined analogously: the weight functions $\omega_1, \ldots, \omega_\ell$ have domain $[m]$ and should assign weights to all the edges in m-edge graphs in $\mathcal{C}$, where the edges are assumed to be enumerated with numbers in $[m]$.

The two main parameters of interest for isolation schemes will be the number of *random bits*, which is defined as $\log \ell$, and the *maximum weight*, defined as the maximum value that any of the functions ω_i may take. Although Definition 1 only assumes the *existence* of suitable weight functions, all the isolation schemes proposed in this paper are extremely simple and can be used as an effective derandomization tool.

1.1 Our contribution

In the following discussion we restrict attention to Hamiltonian cycles and maximum-size independent sets for concreteness, that is, to the edge- and vertex-selection problems $\mathsf{HC}(\cdot)$ and $\mathsf{MIS}(\cdot)$ described above. However, our techniques have a wider applicability, which we comment on throughout the presentation. On a very high level, the natural idea that permeates all our arguments is to reduce the randomness using Divide&Conquer along small separators: If a separator X splits the given graph G in a balanced way, then the same random bits can be reused in each part of $G - X$.

Isolation schemes for Hamiltonian cycles. We first consider the problem of detecting a Hamiltonian cycle, since it represents an important class of connectivity problems such as Steiner Tree or k-Path. For these problems, the Isolation Lemma has been particularly useful in the design of parameterized algorithms [25, 28, 5, 23, 24, 11, 38]. Our first results concerns general graphs.

▶ **Theorem 2.** *There is an isolation scheme for Hamiltonian cycles in undirected graphs that uses $\mathcal{O}(n)$ random bits and assigns weights upper bounded by $2^{\mathcal{O}(n)}$.*

Observe that in an n-vertex graph there can be as many as $n!$ different Hamiltonian cycles. Hence, the application of the general-usage isolation scheme of Chari et al. [8] would give an isolation scheme for Hamiltonian cycles in general graphs that uses $\mathcal{O}(\log(n!)) = \mathcal{O}(n \log n)$ random bits. Note that as proved in [8], isolating a family $\mathcal{F}$ over a universe of size n requires $\Omega(\log |\mathcal{F}| + \log n)$ random bits in general, hence the shaving of the $\log n$ factor reported in Theorem 2 required a problem-specific insight into the family of Hamiltonian cycles in a graph. This insight is provided by the *rank-based approach*, a technique introduced in the context of detecting Hamiltonian cycles in graphs of bounded treewidth [6]. The fact that this works is unexpected because all known methods for derandomizing Hamiltonian cycle require at least exponential space (see [6] for overview).

Let us note that isolation of Hamiltonian cycles was used by Björklund [5] in his $\mathcal{O}(1.657^n)$-time algorithm for detecting a Hamiltonian cycle in an undirected graph. This algorithm is randomized due to the usage of the Isolation Lemma, and derandomizing it, even within time complexity $\mathcal{O}((2-\varepsilon)^n)$ for any $\varepsilon > 0$, is a major open problem. While the constant hidden in the $\mathcal{O}(\cdot)$ notation used in Theorem 2 is too large to allow exploring the whole space of random bits within time $\mathcal{O}((2-\varepsilon)^n)$, in principle we show that the amount of randomness needed is of the same magnitude as would be required for derandomization of the algorithm of Björklund.

Next, we show that in the setting of graphs of bounded treewidth the amount of randomness can be reduced dramatically, to a polylogarithm in n.

▶ **Theorem 3.** *For every $t \in \mathbb{N}$, there is an isolation scheme for Hamiltonian cycles in graphs of treewidth at most t that uses $\mathcal{O}(t \log n + \log^2(n))$ random bits and assigns weights upper bounded by $2^{\mathcal{O}(t \log n + \log^2 n)}$.*

The proof of Theorem 3 fully exploits the idea of using small separators to save on randomness. It also uses the rank-based approach to shave off a $\log t$ factor in the number of random bits.

Finally, we use the separator properties of H-minor free graphs to prove the following.

▶ **Theorem 4.** *For every fixed H, there is an isolation scheme for Hamiltonian cycles in H-minor-free graphs that uses $\mathcal{O}(\sqrt{n})$ random bits and assigns weights upper bounded by $2^{\mathcal{O}(\sqrt{n})}$.*

Recently, [28] presented a randomized algorithm for detecting a Hamiltonian cycle in a graph of treedepth at most d that works in time $2^{\mathcal{O}(d)} \cdot (W+n)^{\mathcal{O}(1)}$ time and uses polynomial space; here, W is the maximum weight assigned by isolation scheme[1]. The only source of randomness in the algorithm of [28] is the Isolation Lemma. Since H-minor free graphs have treedepth $\mathcal{O}(\sqrt{n})$, we can use the isolation scheme of Theorem 4 to derandomize this algorithm, thus obtaining the following result.

▶ **Theorem 5.** *For every fixed H, there is a deterministic algorithm for detecting a Hamiltonian cycle in an H-minor-free graph that runs in time $2^{\mathcal{O}(\sqrt{n})}$ and uses polynomial space.*

To the best of our knowledge, this is the first application of a randomness-efficient isolation scheme for a full derandomization of an exponential-time algorithm without a significant loss on complexity guarantees. Further, we are not aware of any previous algorithms that would simultaneously achieve determinism, running time $2^{\mathcal{O}(\sqrt{n})}$, and polynomial space complexity, even in the setting of planar graphs[2]. Finally, let us note that the algorithm of Theorem 5 does not rely on any topological properties of H-minor-free graphs: the existence of balanced separators of size $\mathcal{O}(\sqrt{n})$ is the only property we use.

MSO-definable problems on graphs of bounded treewidth. We observe that the approach used in the proof of Theorem 3 relies only on finite-state properties of the Hamiltonian Cycle problem on graphs of bounded treewidth. The range of problems enjoying such properties is much wider and encompasses all problems definable in CMSO_2: the Monadic

[1] They did not consider the weighted case, but the statement is implied by a standard extension, see the full version of this paper [27] for details.

[2] Deterministic $2^{\mathcal{O}(\sqrt{n})}$-time algorithms were previously known, but all of these use exponential space [6, 18].

Second-Order logic with modular counting predicates. Consequently, we can lift the proof of Theorem 3 to a generic reasoning that yields an analogous result for every CMSO_2-definable problem. This proves the following (see the full full version of this paper [27] for definitions).

▶ **Theorem 6.** *Let $\mathcal{P}$ be a CMSO_2-definable edge (or vertex) selection problem. There exists a computable function f such that for every $k \in \mathbb{N}$, $\mathcal{P}$ admits an isolation scheme on graphs of treewidth at most k that uses $R := f(k) \cdot \log n + \mathcal{O}(\log^2 n)$ random bits and assigns weights upper bounded by 2^R.*

Lower bounds. We show that a significant improvement of the parameters in the isolation schemes presented above is unlikely. First, a counting argument shows that the $\log n$ factor is necessary.

▶ **Theorem 7.** *There does not exist an isolation scheme for Hamiltonian cycles on graphs of treewidth at most 4 that uses $o(\log n)$ random bits and polynomially bounded weights.*

Using similar constructions we also provide analogous $\Omega(\log n)$ lower bounds for isolating other families of combinatorial objects related to NP-hard problems, such as maximum independent sets, minimum Steiner trees, and minimum maximal matchings. These lower bounds hold even in graphs of bounded *treedepth*, which is a more restrictive setting than bounded treewidth.

We also show using existing reductions that a significant improvement over the scheme of Theorem 2 would imply a surprising partial derandomization of isolation schemes for SAT.

▶ **Theorem 8.** *Suppose there is an isolation scheme for Hamiltonian cycles in undirected graphs that uses $o(n)$ random bits and polynomially bounded weights. Then there is a randomized polynomial-time reduction from* SAT *to* Unique SAT *that uses $o(n)$ random bits, where n is the number of variables.*

Observe that since an n-vertex graph has treewidth at most $n-1$, Theorem 8 also implies that in Theorem 3 one cannot expect reducing the number of random bits to $o(t)$. However, we stress that the lower bounds of Theorems 7 and 8 are not completely tight with respect to the upper bounds of Theorems 2 and 3, because the latter allow superpolynomial weights. It remains open whether the weights used by the schemes of Theorems 2, 3, and 4 can be reduced to polynomial.

In the full version of this paper [27] we further discuss consequences of the hypothetical existence of a polynomial-time reduction from SAT to Unique SAT that would use $o(n)$ random bits.

Level-aware isolation schemes for independent sets. In the light of the $\Omega(\log n)$ lower bound of Theorem 7, we consider a relaxation of the model from Definition 1, where the graph is provided together with an *elimination forest* (a decomposition notion suited for the graph parameter *treedepth*), and the weight of a vertex may depend both on the vertex' identifier and its level in the elimination forest. We demonstrate that in this relaxed model, the $\Omega(\log n)$ lower bound can be circumvented.

▶ **Definition 9.** *We say that vertex selection problem $\mathcal{P}$ admits a* level-aware isolation scheme *if for all $n, d \in \mathbb{N}$ there exist functions $\omega_1, \ldots, \omega_\ell \colon [n] \times [d] \to \mathbb{N}$ such that for every graph G on vertex set $[n]$ and elimination forest F of G of height at most d, at least half of the functions $\omega_1, \ldots, \omega_\ell$ isolate $\mathcal{P}(G)$. Here, when evaluating ω_i on a vertex $u \in [n]$, we apply ω_i to u and the index of the level of u in F.*

▶ **Theorem 10.** *For every $d \in \mathbb{N}$, there is a level-aware isolation scheme for maximum-size independent sets in graphs of treedepth at most d that uses $\mathcal{O}(d)$ random bits and assigns weights bounded by $\mathcal{O}(n^6)$.*

In the proof of Theorem 10 we describe an abstract condition, dubbed the *exchange property*, which is sufficient for the argument to go through. This property is enjoyed also by other families of combinatorial objects defined through constraints of local nature, such as minimum dominating sets or minimum vertex covers. Therefore, we can prove analogous isolation results for those families as well.

Also, in the full version of this paper [27] we discuss a similar reasoning for edge-selection problems on the example of maximum matchings, achieving a level-aware isolation scheme that uses $\mathcal{O}(d \log n)$ random bits and assigns weights bounded by $n^{\mathcal{O}(\log n)}$. This provides another natural class of graphs where isolation-based algorithms for finding a maximum matching can be derandomized (see [2, 7, 13, 21]).

We summarize our results with Table 1.

Table 1 Summary of our results based on Theorems 2-10.

Problem	Random Bits	Max Weight	Graph Class
Hamiltonian Cycle	$\mathcal{O}(n)$	$2^{\mathcal{O}(n)}$	General Graphs
	$\Omega(n)$	$\mathrm{poly}(n)$	
	$\mathcal{O}(\sqrt{n})$	$2^{\mathcal{O}(\sqrt{n})}$	H-minor free graphs
	$\Omega(\sqrt{n})$	$\mathrm{poly}(n)$	
	$\mathcal{O}(t \log(n) + \log^2(n))$	$n^{\mathcal{O}(t+\log(n))}$	Treewidth t graphs
	$\Omega(t + \log(n))$	$\mathrm{poly}(n)$	
CMSO_2	$f(t)\log(n) + \mathcal{O}(\log^2(n))$	$n^{f(t)+\mathcal{O}(\log(n))}$	Treewidth t graphs
Max Independent Set	$\mathcal{O}(d)$	$\mathrm{poly}(n)$	Treedepth d graphs
	$\Omega(d)$	$\mathrm{poly}(n)$	

1.2 Organization

In Section 2 we provide preliminaries. Section 3 is dedicated to the formal proof of Theorem 2. In Appendix A, we formally proof Theorem 3. We finish the main part of the paper with possible directions for further research in Section 4.

In the full version of this paper [27] we include the formal proofs of Theorem 4, Theorem 5 and the general CMSO_2-result of Theorem 6. The full version [27] also includes the lower bounds from Theorem 7 and Theorem 8, as well as the level-aware isolation schemes for local vertex (respectively, edge) selection problems.

2 Preliminaries

Notation. For an integer k, we write $[k] \coloneqq \{1, \ldots, k\}$. We use standard graph notation: $V(G)$ and $E(G)$ respectively denote the vertex set and the edge set of a graph G, for $X \subseteq V(G)$ the *closed neighborhood* $N_G[X]$ is X plus all the neighbors of vertices of X, and the *open neighborhood* is $N_G(X) \coloneqq N_G[X] \setminus X$.

Hashing modulo primes. The following standard hashing lemma that dates back to the work of Fredman, Komlós, and Szemerédi [19], will be the main source of randomness in our isolation schemes.

▶ **Lemma 11** (FKS hashing lemma [19]). *Let $S \subseteq \{0, 1, \ldots, 2^n\}$ be a set of k integers, where $n, k \geqslant 1$. Suppose that p is a prime number chosen uniformly at random among prime numbers in the range $\{1, \ldots, M\}$, where $M \geqslant 2$. Then*

$$\mathbb{P}\left[x \not\equiv y \bmod p \quad \textit{for all} \quad x, y \in S, x \neq y\right] \geqslant 1 - \frac{nk^2}{\sqrt{M}}.$$

Proof. Let

$$R := \prod_{x,y \in S, x \neq y} |x - y|.$$

Note that $R \leqslant 2^{n \cdot \binom{k}{2}}$. This implies that R may have at most $n \cdot \binom{k}{2}$ different prime divisors. On the other hand, from the prime number theorem it follows that $\pi(M) \in \Omega(\frac{M}{\log M})$, where $\pi(M)$ denotes the number of primes in the range $\{1, \ldots, M\}$. In fact, using a more precise estimate of Rosser [33], for $M \geqslant 17$ we have $\pi(M) \geqslant \frac{M}{\ln M}$. For $2 \leqslant M \leqslant 17$ a direct check shows that $\pi(M) \geqslant \sqrt{M}/2$. Since $\frac{M}{\ln M} \geqslant \sqrt{M}/2$ for all $M \geqslant 2$, we conclude that the probability that a random prime in the range $\{1, \ldots M\}$ is not among the at most $n \cdot \binom{k}{2}$ prime divisors of R is at least

$$1 - \frac{n \cdot \binom{k}{2}}{\sqrt{M}/2} \geqslant 1 - \frac{nk^2}{\sqrt{M}}.$$ ◀

Graph decompositions. A *rooted forest* is directed acyclic graph F where every node x has at most one outneighbor, called the *parent* of x. A *root* is a node with no parent. If a node y is reachable from x by a directed path, then we write $y \preceq_F x$ and say that y is an *ancestor* of x and x is a *descendant* of y. Note that every vertex is considered its own ancestor and descendant. For $x \in V(F)$, we write

$$\mathsf{tail}_F[x] := \{y \colon y \preceq_F x\}, \qquad \mathsf{subtree}_F[x] := \{z \colon z \succeq_F x\},$$
$$\mathsf{tail}_F(x) := \mathsf{tail}_F[x] \setminus \{x\}, \qquad \mathsf{subtree}_F(x) := \mathsf{subtree}_F[x] \setminus \{x\}.$$

The *level* of a node x in F, denoted $\mathrm{lvl}_F(x)$, is the number of its strict ancestors, that is, $|\mathsf{tail}_F(x)|$. Note that roots have level 0. The *height* of a forest F is the maximum level among its nodes, plus 1. If the forest F is clear from the context, then we may omit it in the above notation.

An *elimination forest* of a graph G is a rooted forest F with $V(F) = V(G)$ such that for every edge uv of G, either u is an ancestor of v in F or vice versa. The *treedepth* of a graph G is the least possible height of an elimination forest of G. Treedepth as a graph parameter plays a central role in the structural theory of sparse graphs, see [29, Chapters 6 and 7]. It also has several applications in parameterized complexity and algorithm design [9, 15, 20, 28, 30, 31], as well as exhibits interesting combinatorial properties [9, 12, 14] and connections to descriptive complexity theory [16]. We refer to the introductory sections of the above works for a wider discussion.

A *tree decomposition* of a graph G is a pair $\mathbb{T} = (T, \beta)$, where T is an (unrooted) tree and $\beta \colon V(T) \to 2^{V(G)}$ is a function that assigns to each node $x \in V(T)$ its *bag* $\beta(x) \subseteq V(G)$ so that the following two conditions are satisfied:

- for each $u \in V(G)$, the set $\{x\colon u \in \beta(x)\}$ induces a nonempty and connected subtree of T; and
- for each $uv \in E(G)$, there exists $x \in V(T)$ such that $\{u, v\} \subseteq \beta(x)$.

The *width* of $\mathbb{T}$ is $\max_{x \in V(T)} |\beta(x)| - 1$ and the *treewidth* of G is the minimum possible width of a tree decomposition of G. It is easy to see that the treedepth of a graph is at least its treewidth plus one. Conversely, the treewidth is upper bounded by the treedepth times the logarithm of the vertex count [29].

For surgery on tree decompositions we will use the following definition and standard lemma.

▶ **Definition 12** (Segment of a tree)**.** *For an unrooted tree T, a* segment *of T is a nonempty and connected subtree I of T such that there are at most two vertices of I that have a neighbor outside of I. The set of those at most two vertices is the* boundary *of I, and is denoted by ∂I. The* size *of I is equal to $|E(I)|$.*

▶ **Lemma 13.** *Let T be an unrooted tree and let I be a segment of T of size $\ell \geqslant 2$. Then there are at most 5 segments $I_1, \ldots, I_t$ of T ($t \leqslant 5$), each of size at most $\ell/2$, such that segments $I_1, \ldots, I_t$ have pairwise disjoint edge sets and $E(I_1) \cup \ldots \cup E(I_t) = E(I)$.*

Proof. For each edge $xy \in E(I)$, let $I_{y,x}$ and $I_{x,y}$ be the connected components of $I - xy$ that contain x and y, respectively. Let $\vec{I}$ be the orientation of I where each edge xy is oriented towards x if $|E(I_{y,x})| > |E(I_{x,y})|$ and towards y if $|E(I_{y,x})| < |E(I_{x,y})|$; in case $|E(I_{y,x})| = |E(I_{x,y})|$, the edge xy is oriented in any way. Since I has ℓ edges and $\ell + 1$ nodes, there is a node z of I that has outdegree 0 in $\vec{I}$. This means that for every neighbor x of z, we have $|E(I_{z,x})| \leqslant |E(I_{x,z})|$, implying $|E(I_{z,x})| < \ell/2$. Denote $I_x \coloneqq I_{z,x}$ and let $\widehat{I}_x$ be I_x with the edge xz added.

We first argue that I can be edge-partitioned into at most 3 subtrees (not necessarily segments), each with at most $\ell/2$ edges. Consider first the corner case when there exists a neighbor x of z such that $\widehat{I}_x$ has more than $\ell/2$ edges. Then both $I_x = I_{z,x}$ and $I_{x,z}$ have exactly $\frac{\ell-1}{2}$ edges each, so we can partition I into $I_{z,x}$, $I_{x,z}$, and a separate subtree consisting only of the edge xz. This case being resolved, we can assume that each tree $\widehat{I}_x$ has at most $\ell/2$ edges. Starting with the set of trees $\mathcal{T} \coloneqq \{\widehat{I}_x\colon x \text{ is a neighbor of } z\}$, iteratively apply the following procedure: take two trees from $\mathcal{T}$ with the smallest edge counts, and replace them with their union, provided this union has at most $\ell/2$ edges. The procedure stops when this assertion fails to be satisfied. Observe that the procedure can be carried out as long as $|\mathcal{T}| \geqslant 4$, for then the two trees from $\mathcal{T}$ that have the smallest edge counts together include at most half of the edges of I. Therefore, at the end we obtain the desired edge-partition of I into at most three subtrees.

All in all, in both cases we edge-partitioned I into at most three subtrees, each having at most $\ell/2$ edges. Since $|\partial I| \leqslant 2$, it is easy to see that all of those subtrees are already segments (i.e. have boundaries of size at most 2) apart from at most one, say J, which may have a boundary of size 3. Supposing that J exists, let $\partial J = \{a, b, c\}$. Then there exists a node d of J such that every connected component of $J - d$ contains at most one of the vertices a, b, c. It is now straightforward to edge-partition J into three trees so that the boundary of each of them consists of d and one of the vertices a, b, c. Thus, replacing J with those three segments yields an edge-partition of I into at most 5 segments, each with at most $\ell/2$ edges. ◀

3 Isolating Hamiltonian cycles

In this section we prove Theorem 2. We begin by defining *configurations* for Hamiltonian cycles, which reflect the states of a natural dynamic programming algorithm for detection of a Hamiltonian cycle in a bounded-treewidth graph. Then we use the rank-based approach to bound the number of *minimum weight compliant edge sets* (see Theorem 19). This technical result captures the essence of the rank-based approach and will be used in all subsections that follow. Next, we prove Theorem 2 in Section 3.3. In Appendix A we also include the full proof of Theorem 3.

3.1 Configurations for Hamiltonian cycles

Let us fix a graph G. An edge set $S \subseteq E(G)$ is called a *partial solution* if every vertex of G is incident to at most two edges of S and S has no cycles. The following notion of a *configuration* describes the behavior of a partial solution with respect to a set of vertices.

▶ **Definition 14** (Configurations). *For $X \subseteq V(G)$, we define the set of* configurations $\mathsf{conf}(X)$ *on X as:*

$$\{\, (V_0, V_1, V_2, M) \,:\, (V_0, V_1, V_2) \text{ is a partition of } X \text{ and } M \text{ is a perfect matching on } V_1 \,\}.$$

Given a subgraph H of G, one can view the configurations on $X \subseteq V(H)$ as all possible different ways that a partial solution may behave on X. A vertex is then in the set V_i if it is incident to exactly i edges of the partial solution. The matching M on V_1 describes the endpoints of each path in the partial solution. This intuition is formalized in the following definition.

▶ **Definition 15.** *Let $X \subseteq V(G)$ be a set of vertices of G and let $S \subseteq E(G)$ be a partial solution. Then define the* configuration of S on X *as $c_X(S) \coloneqq (V_0, V_1, V_2, M) \in \mathsf{conf}(X)$, where*

- $V_0 \coloneqq \{v \in X : v \text{ is not incident to any edge of } S\}$,
- $V_1 \coloneqq \{v \in X : v \text{ is incident to exactly one edge of } S\}$,
- $V_2 \coloneqq \{v \in X : v \text{ is incident to exactly two edges of } S\}$,
- $M \coloneqq \{\{u, v\} \in \binom{V_1}{2} :$ *there is a path with edges from S connecting u and v*$\}$.

We omit X in the notation and write $c(S)$ when X is clear from context.

Note that in the above definition M is indeed a matching, because each $v \in V_1$ is connected to exactly one $u \in V_1$ through S, as any partial solution covers each vertex at most twice. For an example of deriving $c_X(S)$ from a partial solution S, see Figure 1.

We can use configurations to tell whether two partial solutions together form a Hamiltonian cycle. Let H be a subgraph of G and let $X \subseteq V(H)$. Assume that there exists a partial solution S that visits only vertices from $(V(G) \setminus V(H)) \cup X$, where every vertex of $V(G) \setminus V(H)$ is visited exactly twice. Then we only need to know $c_X(S)$ to determine which partial solutions $S' \subseteq E(H)$ would combine with S to a Hamiltonian cycle in G. We say that any such partial solution is *compliant* with $c_X(S)$, as expressed formally in the next definition.

▶ **Definition 16** (Compliant partial solution). *For a graph H let $X \subseteq V(H)$. A configuration $c = (V_0, V_1, V_2, M) \in \mathsf{conf}(X)$ and a partial solution $S \subseteq E(H)$ are* compliant *if $S \cap M = \emptyset$ and $S \cup M$ forms a Hamiltonian cycle on $V(H) \setminus V_2$.*

See Figure 2 for an example of a compliant partial solution.

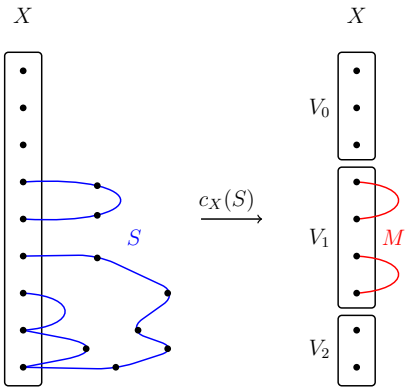

Figure 1 Example partial solution S and its configuration $c_X(S) = (V_0, V_1, V_2, M)$ on a set X.

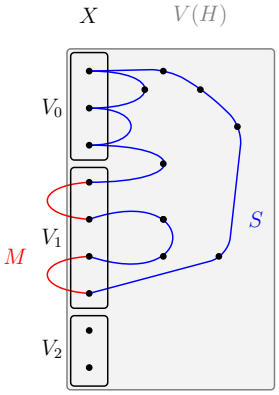

Figure 2 Example compliant partial solution S for a configuration $c = (V_0, V_1, V_2, M) \in \mathsf{conf}(X)$.

In the sequel we will be trying to argue that some weight function ω is isolating the family of Hamiltonian cycles in the given graph G with high probability. In all cases this will be done by induction on larger and larger subgraphs of G, where at each point we argue that a suitable family of partial solutions is isolated with high probability. The following definition facilitates this discussion.

▶ **Definition 17** (Minimum weight compliant partial solution)**.** *Let H be a subgraph of G, $X \subseteq V(H)$, $c \in \mathsf{conf}(X)$, and let $\omega\colon E(G) \to \mathbb{N}$ be a weight function on the edges of G. Then we define the set $\mathsf{Min}(\omega, H, c)$ of* minimum weight partial solutions compliant with c *as the set of those partial solutions $S \subseteq E(H)$ that*

- *are compliant with c, and*
- *subject to the above, have the smallest possible weight $\omega(S)$.*

3.2 Rank-based approach

We will use the *rank-based approach*, introduced by Cygan et al. in [10], as a tool in our analysis of isolation schemes. Let X be a set of vertices. Then define the *compatibility matrix* $\mathcal{H}_X$ as the matrix with entries indexed by $\mathcal{H}_X[M_1, M_2]$ for M_1, M_2 perfect matchings on X, where

$$\mathcal{H}_X[M_1, M_2] = \begin{cases} 1 & \text{if } M_1 \cup M_2 \text{ is a simple cycle,} \\ 0 & \text{otherwise.} \end{cases}$$

Note that $\mathcal{H}_X[M_1, M_2]$ has $2^{\mathcal{O}(|X| \log |X|)}$ rows and columns. The crux of the rank-based approach is that in spite of that, this matrix has a small rank over the two-element field $\mathbb{F}_2$.

▶ **Theorem 18** (Rank-based approach,[10])**.** *For any set X, the rank of $\mathcal{H}_X$ over $\mathbb{F}_2$ is equal to $2^{|X|/2-1}$.*

We use Theorem 18 to prove that the total number of minimum weight compliant solutions is always relatively small, no matter what the weight function is. The following statement will be reused several times in the sequel. Note that a trivial cardinality argument would yield an upper bound of the form $2^{\mathcal{O}(|X| \log |X|)}$; the point of the rank-based approach is to reduce this to $2^{\mathcal{O}(|X|)}$.

▶ **Theorem 19.** *Let G be a graph, $X \subseteq V(G)$, and $\omega\colon V(G) \to \mathbb{N}$ be a weight function such that for all $c \in \mathsf{conf}(X)$, we have $|\mathsf{Min}(\omega, G, c)| \leqslant 1$. Then*

$$\left| \bigcup_{c \in \mathsf{conf}(X)} \mathsf{Min}(\omega, G, c) \right| \leqslant 2^{\mathcal{O}(|X|)}.$$

Proof. Let $K \coloneqq \bigcup_{c \in \mathsf{conf}(X)} \mathsf{Min}(\omega, G, c)$ and let $C \coloneqq \{c(S)\colon S \in K\}$.

We first verify that $|C| = |K|$. By construction, we have $|C| \leqslant |K|$. Assume for contradiction that $|C| < |K|$. Then there are two different partial solutions $S_1, S_2 \in K$ such that $c(S_1) = c(S_2)$. By construction and the assumptions, there are two different configurations $d_1, d_2 \in \mathsf{conf}(X)$ such that $\mathsf{Min}(\omega, G, d_1) = \{S_1\}$ and $\mathsf{Min}(\omega, G, d_2) = \{S_2\}$. However, since $c(S_1) = c(S_2)$, it follows that for any configuration $d \in \mathsf{conf}(X)$, S_1 is compliant with d if and only if S_2 is compliant with d. In particular, S_1 is compliant with d_2 and S_2 is compliant with d_1. This implies that $\omega(S_1) = \omega(S_2)$ and $S_2 \in \mathsf{Min}(\omega, G, d_1)$ and $S_1 \in \mathsf{Min}(\omega, G, d_2)$, a contradiction. Hence $|C| = |K|$.

Define a matrix $\widehat{\mathcal{H}}$ with both coordinates indexed by $\mathsf{conf}(X)$ such that for $c, c' \in \mathsf{conf}(X)$, where $c = (V_0, V_1, V_2, M)$ and $c' = (V_0', V_1', V_2', M')$:

$$\widehat{\mathcal{H}}[c, c'] = \begin{cases} 1 & \text{if } V_0 = V_2',\ V_2 = V_0', \text{ and } M \cup M' \text{ is a simple cycle,} \\ 0 & \text{otherwise.} \end{cases}$$

Notice that if we sort the indices of $\widehat{\mathcal{H}}$ by the partitions (V_0, V_1, V_2), then $\widehat{\mathcal{H}}$ can be seen as a block diagonal matrix with one block for each partition, and this block is a compatibility matrix on V_1. That is,

$$\widehat{\mathcal{H}} = \bigoplus_{V_0 \uplus V_1 \uplus V_2 = X} \mathcal{H}_{V_1},$$

where $\bigoplus$ denotes the operator of combining several matrices into a single block diagonal matrix. By Theorem 18, the rank over $\mathbb{F}_2$ of each of these blocks is bounded by $2^{|X|/2-1}$, hence the rank over $\mathbb{F}_2$ of $\widehat{\mathcal{H}}$ is bounded by $2^{|X|/2-1} \cdot 3^{|X|} \leqslant 2^{\mathcal{O}(|X|)}$.

Next, we claim that the set of rows of $\widehat{\mathcal{H}}$ corresponding to the configurations of C is linearly independent over $\mathbb{F}_2$. Assume not, hence there is a nonempty set of configurations $D \subseteq C$ such that

$$\sum_{d \in D} \widehat{\mathcal{H}}[d, \cdot] = \mathbf{0},$$

where $\mathbf{0}$ is the all-zero vector (all computations are performed in $\mathbb{F}_2$). For each $d \in D$ there is some $S_d \in K$ such that $d = c(S_d)$. Let d_{max} be a configuration of D for which $\omega(S_{d_{\mathsf{max}}})$ is the largest possible. Since $d_{\mathsf{max}} \in C$, we have that $\mathsf{Min}(\omega, G, c) = \{S_{d_{\mathsf{max}}}\}$ for some $c \in \mathsf{conf}(X)$ and hence $\widehat{\mathcal{H}}[d_{\mathsf{max}}, c] = 1$. However, as $\sum_{d \in D} \widehat{\mathcal{H}}[d, \cdot] = \mathbf{0}$, there must be another $d' \in D$, $d' \neq d_{\mathsf{max}}$, such that also $\widehat{\mathcal{H}}[d', c] = 1$. This means that d' is compliant with c, which implies that $\omega(S_{d'}) > \omega(S_{d_{\mathsf{max}}})$ by $\mathsf{Min}(\omega, G, c) = \{S_{d_{\mathsf{max}}}\}$. This contradicts the maximality of $\omega(S_{d_{\mathsf{max}}})$.

We conclude that the set of rows of $\widehat{\mathcal{H}}$ corresponding to C are indeed linearly independent over $\mathbb{F}_2$. Therefore, $|K| = |C|$ is upper bounded by the rank of $\widehat{\mathcal{H}}$ over $\mathbb{F}_2$, which is at most $2^{\mathcal{O}(|X|)}$. ◀

3.3 Hamiltonian cycles in general graphs using $\mathcal{O}(n)$ random bits

We now use the tools prepared so far to prove Theorem 2. The goal is to isolate all Hamiltonian cycles in an undirected graph $G = (V, E)$ using $\mathcal{O}(n)$ random bits, where n is the vertex count. First we give the isolation procedure. Then we analyze the probability of isolating all Hamiltonian cycles using configurations, compliant partial solutions, and the rank-based approach (through Theorem 19). Throughout the subsection we assume without loss of generality that $\log n$ is an integer.

As usual with isolation schemes, we assume that the vertex set of the considered graph G is $V = [n]$. We will apply induction on specific subgraphs of G called *intervals*.

▶ **Definition 20** (Interval of G). *For integers $1 \leqslant s \leqslant t \leqslant n$ and $1 \leqslant s' \leqslant t' \leqslant n$, the* interval $G\langle s, t, s', t'\rangle$ *is the graph (V', E'), where*

$$V' \coloneqq \{s, \ldots, t\} \cup \{s', \ldots, t'\} \qquad \text{and} \qquad E' \coloneqq \{uv \colon u \in \{s, \ldots, t\}, v \in \{s', \ldots, t'\}, uv \in E\}.$$

By $V\langle s, t, s', t'\rangle$ we denote the vertex set V' of the interval $G\langle s, t, s', t'\rangle$.

Note that $G\langle s, t, s, t\rangle$ is just the subgraph of G induced by $\{s, \ldots, t\}$. On the other hand, if $\{s, \ldots, t\} \cap \{s', \ldots, t'\} = \emptyset$, then $G\langle s, t, s', t'\rangle$ is a bipartite graph, with $\{s, \ldots, t\}$ and $\{s', \ldots, t'\}$ being the sides of the bipartition.

Isolation scheme. We first present the isolation scheme. Let $\mathsf{id} \colon E(G) \to \{1, \ldots, |E(G)|\}$ be any bijection that assigns to each edge $e \in E(G)$ its unique *identifier* $\mathsf{id}(e)$. Let C be some large enough constant, to be chosen later. Then independently at random sample $1 + \log n$ primes $p_0, p_1, \ldots, p_{\log n}$ so that p_i is sampled uniformly among primes in the range $\{1, \ldots, M_i\}$, where $M_i \coloneqq 2^{C(\log n + 2^i)}$. Note that choosing each p_i requires $C(\log n + 2^i)$ random bits, hence we have used $\mathcal{O}(n)$ random bits in total.

Next, we inductively define weights functions $\omega_0, \ldots, \omega_{\log n}$ on $E(G)$ as follows:

- Set $\omega_0(e) \coloneqq 2^{\mathsf{id}(e)} \bmod p_0$ for all $e \in E(G)$.
- For each $e \in E(G)$ and $i = 1, \ldots, \log n$, set

$$\omega_i(e) \coloneqq M_{i-1} n \cdot \omega_{i-1}(e) + \left(2^{\mathsf{id}(e)} \bmod p_i\right).$$

Let $\omega \coloneqq \omega_{\log n}$ and observe that ω assigns weights bounded by $2^{\mathcal{O}(n)}$, as required.

Analysis. We will prove the following statement for all $0 \leqslant i \leqslant \log n$ using induction on i.

Induction hypothesis

With probability at least $\left(1 - \frac{1}{n^2}\right)^{i+1}$, for all intervals $G\langle s, t, s', t'\rangle$ s.t. $t - s \leqslant 2^i$ and $t' - s' \leqslant 2^i$ and for each configuration $c \in \mathsf{conf}(V\langle s, t, s', t'\rangle)$, there is at most one minimum weight (w.r.t. ω_i) compliant partial solution, i.e. $|\mathsf{Min}(\omega_i, G\langle s, t, s', t'\rangle, c)| \leqslant 1$.

For $i = \log n$, the induction hypothesis gives us that for the complete interval $G = G\langle 1, 1, n, n\rangle$ and for the configuration $c = (\emptyset, \emptyset, V(G), \emptyset)$, there is at most one minimum weight compliant partial solution w.r.t. ω. In other words, w.r.t. ω there is at most one minimum weight Hamiltonian cycle in G. This happens with probability at least $\left(1 - \frac{1}{n^2}\right)^{\log n + 1} \geqslant 1 - \frac{1}{n}$. So it remains to perform the induction.

Base step. For $i = 0$, we have $t - s \leqslant 1$ and $t' - s' \leqslant 1$. Hence each such interval $G\langle s, t, s', t'\rangle$ has at most 4 edges. Let

$$Y \coloneqq \bigcup_{\substack{t-s\leqslant 1\\ t'-s'\leqslant 1}} 2^{E(G\langle s,t,s',t'\rangle)}$$

and for each $S \in Y$, let

$$x_S \coloneqq \sum_{e\in S} 2^{\mathsf{id}(e)}.$$

Observe that since the identifiers assigned to the edges are unique, the numbers x_S are also pairwise different. Also, note that $|Y| \leqslant 16n^2$ as there are at most n^2 intervals considered, and for each of them there are at most 16 possible subsets of the at most four edges. Recall that $M_0 = 2^{C(\log n+1)}$ and p_0 is drawn uniformly at random among the primes in the range $\{1, \ldots, M_0\}$. Therefore, from Lemma 11 we can conclude that with probability at least

$$\left(1 - \frac{(n^2+1)(16n^2)^2}{2^{(C/2)(\log n+1)}}\right) \geqslant \left(1 - \frac{1}{n^2}\right)$$

all the numbers $\{x_S \colon S \in Y\}$ have pairwise different remainders modulo p_0; here the last inequality holds for a large enough constant C. Since $\omega_0(S) \equiv x_S \bmod p_0$, this means that with probability at least $\left(1 - \frac{1}{n^2}\right)$, all $S \in Y$ receive pairwise different weights with respect to ω_0. Therefore, the induction hypothesis is true for $i = 0$.

Induction step. Assume the induction hypothesis is true for all intervals $G\langle s, t, s', t'\rangle$ such that $t - s \leqslant 2^{i-1}$ and $t' - s' \leqslant 2^{i-1}$. Let

$$Y' \coloneqq \bigcup_{\substack{t-s\leqslant 2^{i-1}\\ t'-s'\leqslant 2^{i-1}}} \bigcup_{c\in \mathsf{conf}(V\langle s,t,s',t'\rangle)} \mathsf{Min}(\omega_{i-1}, G\langle s, t, s', t'\rangle, c)$$

be the set of all the minimal partial solutions for those intervals. Further, let

$$Y \coloneqq \{S_1 \cup S_2 \cup S_3 \cup S_4 \colon S_1, S_2, S_3, S_4 \in Y'\}$$

be the set containing all combinations of four such partial solutions. The strategy is as follows. We first prove in Claim 21 that any relevant minimum weight compliant partial solution should be in Y. Then Claim 22 says that with hight probability, all partial solutions $S \in Y$ have pairwise different weights with respect to ω_i. Hence, proving these two claims will be sufficient to make the induction hypothesis go through.

▷ Claim 21. Let $1 \leqslant a \leqslant b \leqslant n$ and $1 \leqslant a' \leqslant b' \leqslant n$ be such that $b - a \leqslant 2^i$ and $b' - a' \leqslant 2^i$, and let $c \in \mathsf{conf}(a, b, a', b')$. Then $\mathsf{Min}(\omega_i, G\langle a, b, a', b'\rangle, c) \subseteq Y$.

Proof. Take any $S \in \mathsf{Min}(\omega_i, G\langle a, b, a', b'\rangle, c)$. Let

$$r = \lceil (a+b)/2 \rceil \qquad \text{and} \qquad r' = \lceil (a'+b')/2 \rceil$$

and let us select

$$S_1 \subseteq E(G\langle a, r-1, a', r'-1\rangle), \qquad S_2 \subseteq E(G\langle a, r-1, r', b'\rangle),$$
$$S_3 \subseteq E(G\langle r, b, a', r'-1\rangle), \qquad S_4 \subseteq E(G\langle r, b, r', b'\rangle)$$

so that S_1, S_2, S_3, S_4 are disjoint and $S = S_1 \cup S_2 \cup S_3 \cup S_4$. See Figure 3 for an example.

We argue that $S_1 \in \mathsf{Min}(\omega_{i-1}, G\langle a, r-1, a', r'-1\rangle, c_1)$ for some $c_1 \in \mathsf{conf}(V\langle a, r-1, a', r'-1\rangle)$. Let $c = (V_0, V_1, V_2, M)$. Since $S \cup M$ is a simple cycle that visits all vertices of $V\langle a, b, a', b'\rangle$, we see that $R \coloneqq S_2 \cup S_3 \cup S_4 \cup M$ is a partial solution in the graph $G\langle a, b, a', b'\rangle$ with the edges of M added. Letting $(V_0', V_1', V_2', M') \coloneqq c_{V\langle a, r-1, b, r-1\rangle}(R)$, it follows that S_1 is compliant with the configuration

$$c_1 \coloneqq (V_0' \setminus (V_2 \cap V\langle a, r-1, b, r-1\rangle)), V_1', V_2' \cup (V_2 \cap V\langle a, r-1, b, r-1\rangle), M').$$

Moreover, that $S \in \mathsf{Min}(\omega_i, G\langle a, b, a', b'\rangle, c)$ implies that $S_1 \in \mathsf{Min}(\omega_i, G\langle a, r-1, a', r'-1\rangle, c_1)$, for otherwise S_1 could be replaced in S with a smaller-weight partial solution S_1' that would be still compliant with c_1, and this would turn S into a smaller-weight partial solution $S' = S_1' \cup S_2 \cup S_3 \cup S_4$ that would be still compliant with c. Finally, by the construction of ω_i, $S_1 \in \mathsf{Min}(\omega_i, G\langle a, r-1, a', r'-1\rangle, c_1)$ entails $S_1 \in \mathsf{Min}(\omega_{i-1}, G\langle a, r-1, a', r'-1\rangle, c_1)$.

Therefore $S_1 \in Y'$. Analogously we argue that $S_2, S_3, S_4 \in Y'$, hence we conclude that $S \in Y$. ◁

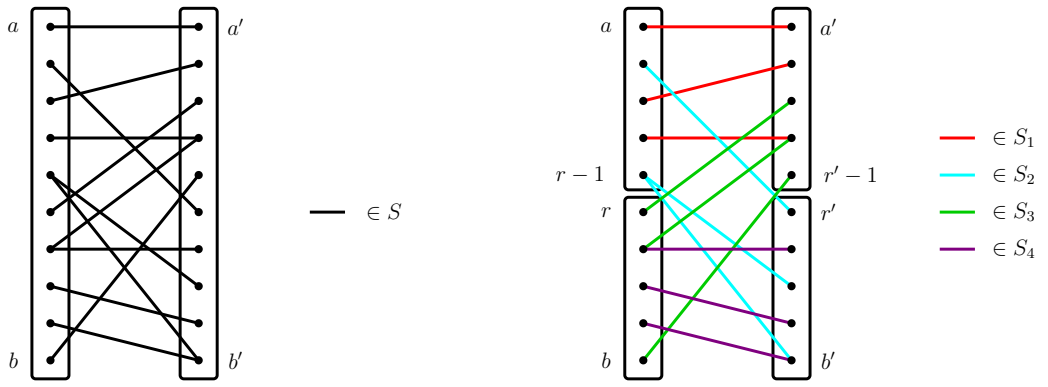

Figure 3 Example of splitting a partial solution $S \in E(G\langle a, b, a', b'\rangle)$ into four partial solutions S_1, S_2, S_3, S_4, where $S_1 \subseteq E(G\langle a, r-1, a', r'-1\rangle)$, $S_2 \subseteq E(G\langle a, r-1, r', b'\rangle)$, $S_3 \subseteq E(G\langle r, b, a', r'-1\rangle)$ and $S_4 \subseteq E(G\langle r, b, r', b'\rangle)$ with $r = \lceil (a+b)/2 \rceil$ and $r' = \lceil (a'+b')/2 \rceil$.

▷ Claim 22. The following event happens with probability at least $\left(1 - \frac{1}{n^2}\right)^{i+1}$: for all different $S, S' \in Y$, it holds that $\omega_i(S) \neq \omega_i(S')$.

Proof. For each $S \in Y$, let

$$x_S \coloneqq \sum_{e \in S} 2^{\mathsf{id}(e)}.$$

Observe that since identifiers assigned to the edges are unique, the numbers x_S are pairwise different. The induction hypothesis gives us that the following event A_{i-1} happens with probability at least $\left(1 - \frac{1}{n^2}\right)^{i}$: for all $1 \leqslant s \leqslant t \leqslant n$ and $1 \leqslant s' \leqslant t' \leqslant n'$ with $t - s \leqslant 2^{i-1}$ and $t' - s' \leqslant 2^{i-1}$, and all $c \in \mathsf{conf}(V\langle s, t, s', t'\rangle)$, we have $|\mathsf{Min}(\omega_{i-1}, G\langle s, t, s', t'\rangle, c)| \leqslant 1$. Assuming now that A_{i-1} indeed happens, by Theorem 19 we conclude that for every fixed choice of s, t, s', t' as above, we have

$$\left| \bigcup_{c \in \mathsf{conf}(V\langle s,t,s',t'\rangle)} \mathsf{Min}(\omega_{i-1}, G\langle s, t, s', t'\rangle, c) \right| \leqslant 2^{\mathcal{O}(2^{i-1})}.$$

Since there are at most n^4 choices of s, t, s', t', this implies that

$$|Y| \leqslant |Y'|^4 \leqslant 2^{\mathcal{O}(2^{i-1})} \cdot n^{16}.$$

Since $M_i = 2^{C(\log n + 2^i)}$ and p_i is drawn uniformly at random among the primes in the range $\{1, \ldots, M_i\}$, from Lemma 11 we can conclude that, for large enough C, with probability at least

$$\left(1 - \frac{(n^2+1)\left(n^{16}2^{\mathcal{O}(2^{i-1})}\right)^2}{2^{(C/2)(\log n + 2^i)}}\right) \cdot \left(1 - \frac{1}{n^2}\right)^i \geqslant \left(1 - \frac{1}{n^2}\right)^{i+1},$$

all the numbers $\{x_S \colon S \in Y\}$ have pairwise different remainders modulo p_i; here, the term $(1 - \frac{1}{n^2})^i$ corresponds to the probability that A_i happens. As a consequence, with the same probability we have that $\omega_i(S) \neq \omega_i(S')$ for all different $S, S' \in Y$. ◁

Now the induction step follows directly from combining Claim 21 with Claim 22.

4 Conclusion and directions for further research

In this paper we presented several isolation schemes for NP-complete problems, and we showed that analogues of decomposition-based methods such as Divide&Conquer can also be used to design more randomness-efficient isolation schemes. While we provide nearly matching lower bounds for all our results, at least as far as the number of random bits is concerned, we still leave open a number of interesting open questions:

1. Can we improve our isolation schemes to have weights that are only polynomial in n, while not increasing the number of used random bits? Note that in our approach, the use of large weights is crucial for the application of Lemma 11 that deals with interactions between different partial solutions in our isolation schemes.[3]
2. Can we shave off the log factors in the number of used random bits in our results? While some of the $\log n$ factors seem to be inherent in our ideas, there still might be a little room. For example, Melkebeek and Prakriya [36] presented an isolation scheme for reachability that uses $\mathcal{O}(\log^{1.5}(n))$-random bits. Perhaps with their ideas one can get the same guarantees for isolating Hamiltonian cycles in constant treewidth graphs.
3. Does the (even more) natural isolation scheme work as well? Many of our isolation schemes draw several random prime numbers and assign a weight that is obtained by concatenating the congruence class of the vertex/edge identifier with respect to the different primes. A more natural, but possibly harder to analyse, scheme would be to sample a single (larger) prime number and define the weights to be the congruence classes of the identifiers with respect to that single prime.
4. Our methods allowed us to derandomize polynomial-space algorithms for H-minor free graphs without significantly increase the running time. Can our methods be used to derandomize other algorithms likewise?

[3] In [8] a similar lemma was used to obtain isolation schemes with polynomial weights, but since the objects of the set family are not decomposed, the authors did not have this issue of interactions between different partial solutions.

References

1 Manindra Agrawal, Rohit Gurjar, and Thomas Thierauf. Impossibility of Derandomizing the Isolation Lemma for all Families. *Electron. Colloquium Comput. Complex.*, 27:98, 2020. URL: https://eccc.weizmann.ac.il/report/2020/098.

2 Manindra Agrawal, Thanh Minh Hoang, and Thomas Thierauf. The Polynomially Bounded Perfect Matching Problem Is in NC^2. In *STACS 2007, 24th Annual Symposium on Theoretical Aspects of Computer Science*, pages 489–499, 2007. doi:10.1007/978-3-540-70918-3_42.

3 Rahul Arora, Ashu Gupta, Rohit Gurjar, and Raghunath Tewari. Derandomizing Isolation Lemma for $K_{3,3}$-free and K_5-free Bipartite Graphs. In *33rd Symposium on Theoretical Aspects of Computer Science, STACS 2016*, pages 10:1–10:15, 2016. doi:10.4230/LIPIcs.STACS.2016.10.

4 Vikraman Arvind and Partha Mukhopadhyay. Derandomizing the isolation lemma and lower bounds for circuit size. In *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*, pages 276–289. Springer, 2008.

5 Andreas Björklund. Determinant sums for undirected hamiltonicity. *SIAM J. Comput.*, 43(1):280–299, 2014. doi:10.1137/110839229.

6 Hans L. Bodlaender, Marek Cygan, Stefan Kratsch, and Jesper Nederlof. Deterministic single exponential time algorithms for connectivity problems parameterized by treewidth. *Inf. Comput.*, 243:86–111, 2015. doi:10.1016/j.ic.2014.12.008.

7 Chris Bourke, Raghunath Tewari, and N. V. Vinodchandran. Directed planar reachability is in unambiguous log-space. *ACM Trans. Comput. Theory*, 1(1):4:1–4:17, 2009. doi:10.1145/1490270.1490274.

8 Suresh Chari, Pankaj Rohatgi, and Aravind Srinivasan. Randomness-Optimal Unique Element Isolation with Applications to Perfect Matching and Related Problems. *SIAM J. Comput.*, 24(5):1036–1050, 1995. doi:10.1137/S0097539793250330.

9 Jiehua Chen, Wojciech Czerwiński, Yann Disser, Andreas Emil Feldmann, Danny Hermelin, Wojciech Nadara, Michał Pilipczuk, Marcin Pilipczuk, Manuel Sorge, Bartlomiej Wróblewski, and Anna Zych-Pawlewicz. Efficient fully dynamic elimination forests with applications to detecting long paths and cycles. *CoRR*, abs/2006.00571, 2020. To appear in the proceedings of SODA 2021. arXiv:2006.00571.

10 Marek Cygan, Stefan Kratsch, and Jesper Nederlof. Fast Hamiltonicity Checking Via Bases of Perfect Matchings. *J. ACM*, 65(3):12:1–12:46, 2018. doi:10.1145/3148227.

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. In *52nd Annual Symposium on Foundations of Computer Science, FOCS 2011*, pages 150–159. IEEE, 2011. doi:10.1109/FOCS.2011.23.

12 Wojciech Czerwiński, Wojciech Nadara, and Marcin Pilipczuk. Improved Bounds for the Excluded-Minor Approximation of Treedepth. In *27th Annual European Symposium on Algorithms, ESA 2019*, volume 144 of *LIPIcs*, pages 34:1–34:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.ESA.2019.34.

13 Samir Datta, Raghav Kulkarni, and Sambuddha Roy. Deterministically isolating a perfect matching in bipartite planar graphs. *Theory Comput. Syst.*, 47(3):737–757, 2010. doi:10.1007/s00224-009-9204-8.

14 Zdenek Dvořák, Archontia C. Giannopoulou, and Dimitrios M. Thilikos. Forbidden graphs for tree-depth. *Eur. J. Comb.*, 33(5):969–979, 2012. doi:10.1016/j.ejc.2011.09.014.

15 Friedrich Eisenbrand, Christoph Hunkenschröder, Kim-Manuel Klein, Martin Koutecký, Asaf Levin, and Shmuel Onn. An algorithmic theory of integer programming. *CoRR*, abs/1904.01361, 2019. arXiv:1904.01361.

16 Michael Elberfeld, Martin Grohe, and Till Tantau. Where first-order and monadic second-order logic coincide. In *27th Annual IEEE Symposium on Logic in Computer Science, LICS 2012*, pages 265–274. IEEE Computer Society, 2012. doi:10.1109/LICS.2012.37.

17 Stephen Fenner, Rohit Gurjar, and Thomas Thierauf. Bipartite perfect matching is in quasi-NC. *SIAM Journal on Computing*, pages STOC16–218, 2019.

18 Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Efficient computation of representative families with applications in parameterized and exact algorithms. *J. ACM*, 63(4):29:1–29:60, 2016. doi:10.1145/2886094.

19 Michael L. Fredman, János Komlós, and Endre Szemerédi. Storing a Sparse Table with $\mathcal{O}(1)$ Worst Case Access Time. *J. ACM*, 31(3):538–544, 1984. doi:10.1145/828.1884.

20 Martin Fürer and Huiwen Yu. Space saving by dynamic algebraization based on tree-depth. *Theory Comput. Syst.*, 61(2):283–304, 2017. doi:10.1007/s00224-017-9751-3.

21 Dima Grigoriev and Marek Karpinski. The matching problem for bipartite graphs with polynomially bounded permanents is in NC (extended abstract). In *28th Annual Symposium on Foundations of Computer Science, Los Angeles, California, USA, 27-29 October 1987*, pages 166–172. IEEE Computer Society, 1987. doi:10.1109/SFCS.1987.56.

22 Chetan Gupta, Vimal Raj Sharma, and Raghunath Tewari. Efficient Isolation of Perfect Matching in $\mathcal{O}(\log n)$ Genus Bipartite Graphs. In *45th International Symposium on Mathematical Foundations of Computer Science (MFCS 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

23 Falko Hegerfeld and Stefan Kratsch. Solving Connectivity Problems Parameterized by Treedepth in Single-Exponential Time and Polynomial Space. In *37th International Symposium on Theoretical Aspects of Computer Science, STACS 2020*, pages 29:1–29:16, 2020. doi:10.4230/LIPIcs.STACS.2020.29.

24 Bart M. P. Jansen and Jesper Nederlof. Computing the chromatic number using graph decompositions via matrix rank. *Theor. Comput. Sci.*, 795:520–539, 2019. doi:10.1016/j.tcs.2019.08.006.

25 Jason Li and Jesper Nederlof. Detecting Feedback Vertex Sets of Size k in $\mathcal{O}^\star(2.7^k)$ Time. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 971–989. SIAM, 2020. doi:10.1137/1.9781611975994.58.

26 Ketan Mulmuley, Umesh V. Vazirani, and Vijay V. Vazirani. Matching is as easy as matrix inversion. *Comb.*, 7(1):105–113, 1987. doi:10.1007/BF02579206.

27 Jesper Nederlof, Michal Pilipczuk, Céline M. F. Swennenhuis, and Karol Wegrzycki. Isolation schemes for problems on decomposable graphs. *CoRR*, abs/2105.01465, 2021. arXiv:2105.01465.

28 Jesper Nederlof, Michał Pilipczuk, Céline M. F. Swennenhuis, and Karol Węgrzycki. Hamiltonian cycle parameterized by treedepth in single exponential time and polynomial space. In *46th International Workshop on Graph-Theoretic Concepts in Computer Science, WG 2020*, volume 12301 of *Lecture Notes in Computer Science*, pages 27–39. Springer, 2020. doi:10.1007/978-3-030-60440-0_3.

29 Jaroslav Nešetřil and Patrice Ossona de Mendez. *Sparsity — Graphs, Structures, and Algorithms*, volume 28 of *Algorithms and combinatorics*. Springer, 2012. doi:10.1007/978-3-642-27875-4.

30 Michał Pilipczuk and Sebastian Siebertz. Polynomial bounds for centered colorings on proper minor-closed graph classes. In *30th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019*, pages 1501–1520. SIAM, 2019. doi:10.1137/1.9781611975482.91.

31 Michał Pilipczuk and Marcin Wrochna. On space efficiency of algorithms working on structural decompositions of graphs. *ACM Trans. Comp. Theory*, 9(4):18:1–18:36, 2018. doi:10.1145/3154856.

32 Klaus Reinhardt and Eric Allender. Making nondeterminism unambiguous. *SIAM J. Comput.*, 29(4):1118–1131, 2000. doi:10.1137/S0097539798339041.

33 Barkley Rosser. Explicit bounds for some functions of prime numbers. *American Journal of Mathematics*, 63(1):211–232, 1941. URL: http://www.jstor.org/stable/2371291.

34 Ola Svensson and Jakub Tarnawski. The matching problem in general graphs is in Quasi-NC. In *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017*, pages 696–707, 2017. doi:10.1109/FOCS.2017.70.

35 Leslie G. Valiant and Vijay V. Vazirani. NP is as Easy as Detecting Unique Solutions. *Theor. Comput. Sci.*, 47(3):85–93, 1986. doi:10.1016/0304-3975(86)90135-0.

36 Dieter van Melkebeek and Gautam Prakriya. Derandomizing Isolation in Space-Bounded Settings. *SIAM J. Comput.*, 48(3):979–1021, 2019. doi:10.1137/17M1130538.

37 Avi Wigderson. NL/poly $\subseteq \oplus$L/poly (preliminary version). In *Proceedings of the Ninth Annual Structure in Complexity Theory Conference*, pages 59–62, 1994. doi:10.1109/SCT.1994.315817.

38 Ryan Williams. Finding paths of length k in $\mathcal{O}^\star(2^k)$ time. *Inf. Process. Lett.*, 109(6):315–318, 2009. doi:10.1016/j.ipl.2008.11.004.

A Hamiltonian cycles in graphs of bounded treewidth

We will now use the same approach to give a proof of Theorem 3. More precisely, assume we are given a graph G of treewidth at most k. Our goal is to isolate the family of Hamiltonian cycles in G using $\mathcal{O}(k \log n + \log^2 n)$ random bits.

The proof follows the same structure as that of Theorem 2. We first describe the isolation scheme and then analyze the scheme using a tree decomposition $\mathbb{T} = (T, \beta)$ of G of width at most k. Note that the actual decomposition is not needed for the isolation procedure, and is only used as a tool in the analysis.

Isolation scheme. We first present the isolation scheme. As before, we assume that $V(G) = [n]$ and n is a power of 2. Let $\mathsf{id}\colon E(G) \to \{1, \ldots, |E(G)|\}$ be any bijection that assigns to each edge $e \in E(G)$ its unique *identifier* $\mathsf{id}(e)$. Let C be some large enough constant, to be chosen later. Then we independently sample $3 \log n$ primes $p_1, \ldots, p_{3 \log n}$ so that each p_i is sampled uniformly among all primes in the interval $\{1, \ldots, M\}$, where $M = 2^{C(k \log n)}$. Note that choosing each p_i requires $C(k + \log n)$ random bits, hence we have used $\mathcal{O}(k \log n + \log^2 n)$ random bits in total, as required.

Next, we inductively define weights functions $\omega_0, \ldots, \omega_{3 \log n}$ on $E(G)$ as follows:

- Set $\omega_0(e) \coloneqq 0$ for all $e \in E(G)$.
- For each $e \in E(G)$ and $i = 1, \ldots, 3 \log n$, set

$$\omega_i(e) \coloneqq Mn \cdot \omega_{i-1}(e) + \left(2^{\mathsf{id}(e)} \bmod p_i\right).$$

We let $\omega \coloneqq \omega_{3 \log n}$ and we observe that ω assigns weights bounded by $2^{\mathcal{O}(k \log n + \log^2 n)}$.

Analysis. Let $\mathbb{T} = (T, \beta)$ be a tree decomposition of G of width at most k. It is well-known that T can be chosen so that it has at most n nodes. Further, let $\eta \coloneqq E(G) \to V(T)$ be any function that assigns to each edge e of G any node x of T such that $e \subseteq \beta(x)$. In the sequel we will assume that η is injective. This can be achieved by adding, for each node $x \in V(T)$, $|\eta^{-1}(x)| - 1$ new nodes with the same bag and adjacent only to x, and appropriately distributing the images of edges of $\eta^{-1}(x)$ among the new nodes. Note that after this modification, the number of nodes of T is bounded by $\binom{k+1}{2} \cdot n \leqslant n^3$.

Compared to the proof of Theorem 2, instead of intervals we will use *segments* in the tree T underlying the tree decomposition $\mathbb{T}$. Recall that segments have been defined and discussed in Section 2. We first observe that there are only few segments.

▷ Claim 23. There are at most n^9 segments of T.

Proof. Note that a segment I in T can be uniquely determined by specifying the at most two vertices of ∂I and any vertex of $V(I) \setminus \partial I$, provided there exists any. Since T has at most n^3 nodes, there are at most n^9 choices for such a specification. ◁

For a set of nodes $Z \subseteq V(T)$, we write $\beta(Z) \coloneqq \bigcup_{z \in Z} \beta(z)$. Further, for a segment I of T we consider the graph

$$G\langle I \rangle \coloneqq \left(\beta(V(I)), \eta^{-1}(V(I))\right).$$

Usually when speaking about partial solutions in $G\langle I \rangle$, we consider their configurations on the vertex subset $\beta(\partial I)$. Note that $G\langle T \rangle = G$.

We proceed to the induction. We will prove the following statement for all $0 \leqslant i \leqslant \log n$.

Induction hypothesis

With probability at least $\left(1 - \frac{1}{n^2}\right)^i$, for all segments I of T of size at most 2^i and for each configuration $c \in \mathsf{conf}(\beta(\partial I))$, there is at most one minimum weight (w.r.t. ω_i) compliant partial solution in $G\langle I \rangle$, i.e. $|\mathsf{Min}(\omega_i, G\langle I \rangle, c)| \leqslant 1$.

Note that since $|V(T)| \leqslant n^3$, for $i = 3 \log n$ the induction hypothesis gives that for $G\langle T \rangle = G$, there is at most one Hamiltonian cycle that has the minimum weight w.r.t. ω with probability at least $\left(1 - \frac{1}{n^2}\right)^{3 \log n} \geqslant \left(1 - \frac{1}{n}\right)$.

Base step. For $i = 0$, we take segments of size at most 1, i.e. we prove the induction hypothesis for every segment I of T that has either one or two nodes. More precisely, we have to prove that (with suitably large probability), for every such segment I and configuration $c \in \mathsf{conf}(\beta(\partial I))$, we have $|\mathsf{Min}(\omega_0, G\langle I \rangle, c)| \leqslant 1$. Note that since I has at most two nodes and η is injective, the edge set $E(G\langle I \rangle)$ consists of at most two edges. Moreover, it cannot be that two different edge subsets $E_1, E_2 \subseteq E(G\langle I \rangle)$ are simultaneously compliant with the same configuration $c \in \mathsf{conf}(\beta(\partial I))$. It follows that sets $\mathsf{Min}(\omega_0, G\langle I \rangle, c)$ have sizes at most 1 always, so the induction hypothesis for $i = 0$ is true.

Induction step. Assume the induction hypothesis is true for all segments of size at most 2^{i-1}. Let

$$Y' \coloneqq \bigcup_{I\colon \text{segment of size } \leqslant 2^{i-1}} \; \bigcup_{c \in \mathsf{conf}(\beta(\partial I))} \mathsf{Min}(\omega_{i-1}, G\langle I \rangle, c).$$

be the set of all minimum weight partial solutions for segments of size at most 2^{i-1}. Further, let

$$Y \coloneqq \{\, S_1 \cup S_2 \cup S_3 \cup S_4 \cup S_5 \;:\; S_1, S_2, S_3, S_4, S_5 \in Y' \,\}$$

be the set comprising all combinations of five such partial solutions.

We first prove with Claim 24 that every relevant minimum weight compliant edge is contained in Y. Then Claim 25 says that with high probability, all $S \in Y$ receive pairwise different weights with respect to ω_i. The induction hypothesis will follow directly from combining these two claims.

▷ Claim 24. Let I be any segment of size at most 2^i and let $c \in \mathsf{conf}(\beta(\partial I))$. Then $\mathsf{Min}(\omega_i, G\langle I \rangle, c) \subseteq Y$.

Proof. Consider any $S \in \mathsf{Min}(\omega_i, G\langle I\rangle, c)$. By Lemma 13, there exist segments $I_1, \ldots, I_t$ ($t \leqslant 5$), each of size at most 2^{i-1}, such that $E(I)$ is the disjoint union of $E(I_1), \ldots, E(I_t)$. For each $j \in \{1, \ldots, t\}$ choose $S_j \in E(G\langle I_j\rangle)$ so that S is the disjoint union of $S_1, \ldots, S_t$. The same argument as that was used in the proof of Claim 21 shows that there exists $c_j \in \mathsf{conf}(\beta(\partial I_j))$ such that $S_j \in \mathsf{Min}(\omega_{i-1}, G\langle I_j\rangle, c_j)$. Hence $S_j \in Y'$ for all $j \in \{1, \ldots, t\}$, so it follows that $S \in Y$. ◁

▷ Claim 25. The probability of the following event is at least $\left(1 - \frac{1}{n^2}\right)^i$: for all different $S, S' \in Y$, it holds that $\omega_i(S) \neq \omega_i(S')$.

Proof. For each $S \in Y$ let us define

$$x_S = \sum_{e \in S} 2^{\mathsf{id}(e)}.$$

Observe that since the identifiers assigned to the edges are unique, the numbers x_S are pairwise different. By the induction hypothesis, the following event A_{i-1} happens with probability at least $\left(1 - \frac{1}{n^2}\right)^{i-1}$: for every segment I of size at most 2^{i-1} and each configuration $c \in \mathsf{conf}(\beta(\partial I))$, we have $|\mathsf{Min}(\omega_{i-1}, G\langle I\rangle, c)| \leqslant 1$. By Theorem 19 it follows that provided A_{i-1} happens, for every fixed segment I of size at most 2^{i-1} we have

$$\left| \bigcup_{c \in \mathsf{conf}(\beta(\partial I))} \mathsf{Min}(\omega_{i-1}, G\langle I\rangle, c) \right| \leqslant 2^{\mathcal{O}(|\beta(\partial I)|)} \leqslant 2^{\mathcal{O}(k)}.$$

By Claim 23 there are at most n^9 different segments, hence this implies that

$$|Y| \leqslant |Y'|^5 \leqslant 2^{\mathcal{O}(k)} \cdot n^{45}.$$

Recall now that $M = 2^{C(k+\log n)}$ and p_i is drawn uniformly at random among the primes in the range $\{1, \ldots, M\}$. Hence, from Lemma 11 we can conclude that, for large enough C, with probability at least

$$\left(1 - \frac{(n^2+1)\left(2^{\mathcal{O}(k)} \cdot n^{45}\right)^2}{2^{(C/2)(k+\log n)}}\right) \cdot \left(1 - \frac{1}{n^2}\right)^{i-1} \geqslant \left(1 - \frac{1}{n^2}\right)^i,$$

all the numbers in $\{x_S \colon S \in Y\}$ have pairwise different remainders modulo p_i. Here, the factor $(1 - \frac{1}{n^2})^{i-1}$ corresponds to the probability that A_{i-1} happens. As a consequence, with the same probability for all different $S, S' \in Y$ we have $\omega_i(S) \neq \omega_i(S')$. ◁

The induction step now follows directly from combining Claims 24 and 25.