Beyond the Lovász Local Lemma: Point to Set Correlations and Their Algorithmic Applications

Dimitris Achlioptas*
Dept. of Computer Science
UC Santa Cruz
Santa Cruz, U.S.A.
optas@cs.ucsc.edu

Fotis Iliopoulos†
Dept. of Electrical Engineering and Computer Science
UC Berkeley
Berkeley, U.S.A.
fotis.iliopoulos@berkeley.edu

Alistair Sinclair‡
Dept. of Electrical Engineering and Computer Science
UC Berkeley
Berkeley, U.S.A.
sinclair@berkeley.edu

***Abstract*—Following the groundbreaking algorithm of Moser and Tardos for the Lovász Local Lemma (LLL), there has been a plethora of results analyzing local search algorithms for various constraint satisfaction problems. The algorithms considered fall into two broad categories: *resampling* algorithms, analyzed via different algorithmic LLL conditions; and *backtracking* algorithms, analyzed via entropy compression arguments. This paper introduces a new convergence condition that seamlessly handles resampling, backtracking, and *hybrid* algorithms, i.e., algorithms that perform both resampling and backtracking steps. Unlike previous work on the LLL, our condition replaces the notion of a dependency or causality graph by quantifying *point-to-set correlations* between bad events. As a result, our condition simultaneously: (i) captures the most general algorithmic LLL condition known as a special case; (ii) significantly simplifies the analysis of entropy compression applications; (iii) relates backtracking algorithms, which are conceptually very different from resampling algorithms, to the LLL; and most importantly (iv) allows for the analysis of hybrid algorithms, which were outside the scope of previous techniques. We give several applications of our condition, including a new hybrid vertex coloring algorithm that extends the recent breakthrough result of Molloy for coloring triangle-free graphs to arbitrary graphs.**

***Keywords*-Lovasz Local Lemma; local search algorithms; backtracking; graph coloring; random graphs**

I. Introduction

Numerous problems in computer science and combinatorics can be formulated as searching for objects that lack certain bad properties, or "flaws". For example, constraint satisfaction problems such as satisfiability and graph coloring can be seen as searching for objects (truth assignments, colorings) that are *flawless*, in the sense that they do not violate any constraint.

The *Lovász Local Lemma (LLL)* [19] is a powerful tool for proving the *existence* of flawless objects that has had far-reaching consequences in computer science and combinatorics [8, 37]. Roughly speaking, it asserts that, given a collection of a bad events in a probability space, if all of them are individually not too likely, and independent of most other bad events, then the probability that none of them occurs is strictly positive; hence a flawless object (i.e., an elementary event that does not belong in any bad event) exists. For example, the LLL implies that every k-CNF formula in which each clause shares variables with fewer than $2^k/\mathrm{e}$ other clauses is satisfiable. Remarkably, this is tight [21].

In seminal work, Moser [38] and Moser-Tardos [39] showed that a simple local search algorithm can be used to make the LLL constructive for product probability spaces, i.e., spaces where elementary events correspond to sequences of independent coin-flips. For example, the Moser-Tardos algorithm for satisfiability starts at a uniformly random truth assignment and, as long as violated clauses exist, selects any such clause and resamples the values of all its variables uniformly at random. Following this work, a large amount of effort has been devoted to making different variants of the LLL constructive [15, 32, 33, 42], and to analyzing sophisticated *resampling* algorithms that extend the Moser-Tardos techniques to non-product probability spaces [2, 3, 27, 28, 30, 35]. Indeed, intimate connections have been established between resampling algorithms and the so-called "lopsided" versions of the LLL: see [3, 28, 30] for more details.

In earlier groundbreaking work aimed at making the LLL algorithmic for k-SAT, Moser [38] introduced the *entropy compression* method. This method has since been used to analyze *backtracking* algorithms, e.g., [18, 20, 23, 24, 25, 41]. These natural and potentially powerful local search algorithms have a very different flavor from resampling algorithms: instead of maintaining a complete value assignment (object) and repeatedly modifying it until it becomes satisfying, they operate on *partial non-violating* assignments, starting with the empty assignment, and try to extend to a

*Research supported by NSF grant CCF-1514434
†Research supported by NSF grants CCF-1514434 and CCF-1815328 and by the Onassis Foundation
‡Research supported by NSF grants CCF-1514434 and CCF-1815328

2575-8454/19/$31.00 ©2019 IEEE
DOI 10.1109/FOCS.2019.00049

complete, satisfying assignment. To do this, at each step they assign a (random) value to a currently unassigned variable; if this leads to the violation of one or more constraints, they backtrack to a partial non-violating assignment by unassigning some set of variables (typically including the last assigned variable).

While there have been efforts to treat certain classes of backtracking algorithms systematically [20, 24], the analysis of such algorithms in general still requires *ad hoc* technical machinery. Moreover, there is no known connection between backtracking algorithms and any known LLL condition, either existential or algorithmic. The main reason for this is that backtracking steps induce non-trivial correlations among bad events, which typically result in very dense dependency graphs that are not amenable to currently known LLL conditions.

The main contribution of this paper is to introduce a new technique for analyzing *hybrid* algorithms, i.e., algorithms that (potentially) use both resampling and backtracking steps. Such algorithms combine the advantages of both approaches by using resampling to explore the state space, while detecting and backing away from unfavorable regions using backtracking steps. To analyze these algorithms, we prove a new algorithmic LLL condition which, unlike previous versions, replaces the notion of a dependency or causality graph by quantifying *point-to-set correlations* between bad events. Notably, our new condition captures the most general algorithmic LLL condition known so far and, moreover, unifies the analysis of all entropy compression applications, connecting backtracking algorithms to the LLL in the same fashion that existing analyses connect resampling algorithms to the LLL.

A new coloring algorithm: Our main application is a new vertex coloring algorithm inspired by the recent breakthrough result of Molloy [35], who proved that any triangle-free graph of maximum degree Δ can be colored using $(1 + o(1))\Delta/\ln\Delta$ colors; this improves the celebrated result of Johannson [31] while, at the same time, dramatically simplifying its analysis. We generalize Molloy's result by establishing a bound on the chromatic number of *arbitrary* graphs, as a function of the maximum number of triangles in the neighborhood of a vertex, and giving an algorithm that produces such a coloring.

Theorem I.1 (Informal Statement)**.** *Let G be any graph with maximum degree Δ in which the neighbors of every vertex span at most $T \geq 0$ edges between them. For every $\epsilon > 0$, if $\Delta \geq \Delta_\epsilon$ and $T \lesssim \Delta^{2\epsilon}$ then*

$$\chi(G) \leq (1+\epsilon)\frac{\Delta}{\ln\Delta - \frac{1}{2}\ln(T+1)} \ , \qquad (1)$$

and such a vertex coloring can be found efficiently. (Here $\lesssim$ hides logarithmic factors.) Moreover, the theorem holds for any $T \geq 0$ if the leading constant $(1+\epsilon)$ is replaced by $(2+\epsilon)$.

Importantly, as explained in Section II-B, the bound (1) *matches* the algorithmic barrier for random graphs [1]. This implies that any improvement on the guarantee of our algorithm for $T \lesssim \Delta^{2\epsilon}$ would amount to an unexpected breakthrough in random graph theory. (Random graphs are only informative in the regime $T \lesssim \Delta^{2\epsilon}$.) For arbitrary graphs our bound is within a factor of 4 of the chromatic number, improving upon a classical result of of Alon, Krivelevich and Sudakov [7] who showed (1) with an unspecified (large) leading constant.

At the heart of Theorem I.1 is a hybrid local search algorithm which we analyze using the techniques introduced in this paper. Molloy's result (and resampling algorithms based on it) breaks down immediately in the presence of triangles; the key to our algorithm is to allow backtracking steps in order to avoid undesirable portions of the search space. We discuss our coloring result and its optimality further in Section II-B.

Applications to backtracking algorithms: Besides our coloring application, we give three representative applications of our techniques applied to pure backtracking algorithms.

Recently, Bissacot and Doin [14] showed that backtracking algorithms can make LLL applications in the variable setting constructive, using the entropy compression method. However, their result applies only to the uniform measure and their algorithms are relatively complicated. Our new algorithmic condition makes applications of the LLL in the variable setting [39], with any measure, constructive via a single, simple backtracking algorithm, i.e., an algorithm of very different flavor from the Moser-Tardos resampling algorithm. For example, in the case of k-SAT the algorithm takes the following form:

Randomized DPLL with single-clause backtracking

while unassigned variables exist **do**
  Assign to the lowest indexed unassigned variable x a value $v \in \{0, 1\}$ with probability p_x^v
  if one or more clauses become violated **then**
    Unassign all k variables in the lowest indexed violated clause

We then show that applying our condition to the algorithm of Esperet and Parreau [20] for *acyclic edge coloring* recovers, in a simple, black-box fashion, the same bound of 4Δ as their highly non-trivial, problem-specific analysis via entropy compression, while guaranteeing an improved running time bound.

Finally, we make constructive in an effortless manner an existential result of Bernshteyn [11] showing improved bounds for the acyclic chromatic index of graphs that do not contain an arbitrary bipartite graph H.

We present our results on backtracking algorithms in Section V.

A. A new algorithmic LLL condition

To state our new algorithmic LLL condition, we need some standard terminology. Let Ω be a finite set and let $F = \{f_1, f_2, \ldots, f_m\}$ be a collection of subsets of Ω, each of which will be referred to as a "flaw." Let $\bigcup_{i \in [m]} f_i = \Omega^*$. For example, for a given CNF formula on n variables with clauses $c_1, \ldots, c_m$, we take $\Omega = \{0,1\}^n$ to be the set of all possible variable assignments, and f_i the set of assignments that fail to satisfy clause c_i. Our goal is to find an assignment in $\Omega \setminus \Omega^*$, i.e., a satisfying ("flawless") assignment. Since we will be interested in algorithms which traverse the set Ω we will also refer to its elements as "states."

We consider algorithms which, in each flawed state $\sigma \in \Omega^*$, choose a flaw f_i present in σ, i.e., $f_i \ni \sigma$, and attempt to leave (or "fix") f_i by moving to a new state τ selected with probability $\rho_i(\sigma, \tau)$; we refer to such an attempt as *addressing* f_i. We make minimal assumptions about how the algorithm choses which flaw to address at each step; e.g., it will be enough for the algorithm to choose the flaw with lowest index according to some fixed permutation. We say that a transition $\sigma \to \tau$, made to address flaw f_i, *introduces* flaw f_j if $\tau \in f_j$, and either $\sigma \notin f_j$ or $j = i$.

For an arbitrary state $\tau \in \Omega$, flaw f_i, and set of flaws S, let

$$\begin{aligned}\mathrm{In}_i^S(\tau) :=\{\sigma \in f_i : &\text{ the set of flaws introduced} \\ &\text{by the transition } \sigma \to \tau \text{ includes } S\}. \end{aligned} \tag{2}$$

For any fixed probability distribution $\mu > 0$ on Ω (either inherited from an application of the probabilistic method, as in the classical LLL, or introduced by the algorithm designer), we define the *charge* of the pair (i, S) with respect to μ to be

$$\gamma_i^S := \max_{\tau \in \Omega} \left\{ \frac{1}{\mu(\tau)} \sum_{\sigma \in \mathrm{In}_i^S(\tau)} \mu(\sigma) \rho_i(\sigma, \tau) \right\}. \tag{3}$$

That is, the charge γ_i^S is an upper bound on the ratio between the ergodic flow into a state via transitions that introduce every flaw in S (and perhaps more), and the probability of the state under μ.

Our condition may now be stated informally as follows:

Theorem I.2. *If there exist positive real numbers $\{\psi_i\}_{i=1}^m$ such that, for all $i \in [m]$,*

$$\frac{1}{\psi_i} \sum_{S \subseteq [m]} \gamma_i^S \prod_{j \in S} \psi_j \ < 1 \ , \tag{4}$$

then a local search algorithm as above reaches a flawless object quickly with high probability.

The phrase "quickly with high probability" essentially means that the running time has expectation linear in the number of flaws and an exponential tail; we spell this out more formally in Section II.

A key feature of Theorem I.2 is the absence of a causality/dependency graph, present in all previous LLL conditions. This is because considering *point-to-set correlations*, i.e., how each flaw interacts with every other set of flaws, frees us from the traditional view of dependencies between individual events. In our condition, every flaw may interact with every other flaw, as long as the interactions are sufficiently weak. Notably, this is achieved without any loss when specialized to the traditional setting of a causality/dependency graph. To see this, note that if S contains any flaw that is never introduced by addressing flaw f_i, then $\gamma_i^S = 0$. Thus, in the presence of a causality/dependency graph, the only terms contributing to the summation in (4) are those that correspond to subsets of the graph neighborhood of flaw f_i, recovering the traditional setting.

Besides relaxing the traditional notion of dependence, our condition is also quantitatively more powerful, even in the traditional setting. To get a feeling for this, we observe that the previously most powerful algorithmic LLL condition, due to Achlioptas, Iliopoulos and Kolmogorov [3], can be derived from (4) by replacing γ_i^S by $\gamma_i^\emptyset$ for every S (and restricting S to subsets of the neighborhood of flaw f_i, as discussed in the previous paragraph). Since the charges γ_i^S are decreasing in S, replacing $\gamma_i^\emptyset$ with γ_i^S can lead to a significant improvement. For example, if the flaws in S are never introduced simultaneously when addressing flaw f_i, then $\gamma_i^S = 0$ and S does not contribute to our sum in (4); in contrast, S contributes $\gamma_i^\emptyset$, i.e., the maximum possible charge, to the corresponding sum in [3]. For a more detailed discussion of how our new condition subsumes existing versions of the LLL, see the full version of this paper [4].

A natural question is whether Theorem I.2 can be improved by replacing the word "includes" with the word "equals" in (2), thus shrinking the sets $\mathrm{In}_i^S(\tau)$. The short answer is "No," i.e., such a change invalidates the theorem. The reason for this is that in resampling algorithms we must allow for the possibility that flaws introduced when addressing f_i may later be fixed "collaterally," i.e., as the result of addressing other flaws rather than by being specifically addressed by the algorithm. While it may seem that such collateral fixes cannot possibly be detrimental, they are problematic from an analysis perspective as they can potentially increase the intensity of correlations between flaw f_i

and S. Perhaps more convincingly, tracking collateral fixes and taking them into account also appears to be a bad idea in practice [9, 10, 43, 44]: for example, local search satisfiability algorithms that select which variable to flip (among those in the targeted violated clause) based *only* on which clauses will become violated, fare much better than algorithms that weigh this damage against the benefit of the collaterally fixed clauses.

Motivated by the above considerations, we introduce the notion of *primary* flaws. These are flaws which, once present, can only be eradicated by being addressed by the algorithm, i.e., they cannot be fixed collaterally. Primary flaws allow us to change the definition of the sets $\mathrm{In}_i^S(\tau)$ in the desired direction. Specifically, say that a set of flaws T *covers* a set of flaws S if:

1) the set of primary flaws in T *equals* the set of primary flaws in S; and
2) the set of non-primary flaws in T *includes* the set of non-primary flaws in S.

In other words, we demand equality at least for the primary flaws.

Theorem I.3. *Theorem I.2 continues to hold if $\mathrm{In}_i^S(\tau)$ is redefined by replacing "includes" by "covers" in equation* (2).

The notion of primary flaws is one of our main conceptual contributions. Crucially for our applications, backtracking steps always introduce only primary flaws and thus, for such steps, we achieve an ideal level of control. The full version of our new algorithmic LLL condition, incorporating primary flaws, is spelled out formally in Theorem II.4 in Section II.

B. Technical overview: the Lovász Local Lemma as a spectral condition

We conclude this introduction by sketching the techniques we use to prove our convergence criterion. The main new insight is that LLL-inspired convergence arguments for local search algorithms can be viewed as methods for bounding the *spectral radius* of an associated matrix.

As above, let Ω be a (large) finite set of states and let $\Omega^* \subseteq \Omega$ be the "bad" part of Ω, comprising the flawed states. Imagine a particle trying to escape Ω^* by following a Markov chain on Ω with transition matrix P. Our task is to develop conditions under which the particle eventually escapes, thus establishing in particular that $\Omega^* \neq \Omega$. Letting A be the $|\Omega^*| \times |\Omega^*|$ submatrix of P that corresponds to transitions from Ω^* to Ω^*, and B the submatrix that corresponds to transitions from Ω^* to $\Omega \setminus \Omega^*$, we may, after a suitable permutation of its rows and columns, write P as:

$$P = \left[\begin{array}{c|c} A & B \\ \hline 0 & I \end{array}\right] .$$

Here I is the identity matrix, since we assume that the particle stops after reaching a flawless state.

Let $\theta = [\theta_1 \mid \theta_2]$ be the row vector corresponding to the probability distribution of the starting state, where θ_1 and θ_2 are the vectors that correspond to states in Ω^* and $\Omega \setminus \Omega^*$, respectively. Then, the probability that after t steps the particle is still inside Ω^* is exactly $\|\theta_1 A^t\|_1$. Therefore, for any initial distribution θ, the particle escapes Ω^* if and only if the spectral radius, $\rho(A)$, of A is strictly less than 1. Moreover, the rate of convergence is dictated by $1 - \rho(A)$. Unfortunately, since A is huge and defined implicitly by an algorithm, the magnitude of its largest eigenvalue, $\rho(A)$, is not readily available.

To sidestep the inaccessibility of the spectral radius $\rho(A)$, one can instead bound some *operator norm* $\|\cdot\|$ of A and appeal to the fact that $\rho(A) \leq \|A\|$ for any such norm. Moreover, instead of bounding an operator norm of A itself, one often first performs a "change of basis" $A' = MAM^{-1}$ for some invertible matrix M and then bounds $\|A'\|$, justified by the fact that $\rho(A) = \rho(A') \leq \|A'\|$. The purpose of the change of basis is to cast A "in a good light" in the eyes of the chosen operator norm, in the hope of minimizing the cost of replacing the spectral norm with the operator norm.

As we explain in Appendix B, essentially all known analyses of LLL-inspired local search algorithms can be recast in the above framework of matrix norms. More significantly, this viewpoint allows the extension of such analyses to a wider class of algorithms. In Section III, we will use this linear-algebraic machinery to prove our new convergence condition. The role of the measure μ will be reflected in the change of basis M, while the charges γ_i^S will correspond to norms $\|MA_i^S M^{-1}\|_1$, where the A_i^S are submatrices of the transition matrix A.

II. Statement of Results

A. A new algorithmic LLL condition

Below we state our main result, which is a formal version of Theorem I.3 discussed in Section I-A.

Recall that Ω is a finite set, $F = \{f_1, f_2, \ldots, f_m\}$ is a collection of subsets of Ω which we refer to as "flaws", and $\bigcup_{i \in [m]} f_i = \Omega^*$. Our goal is to find a flawless object, i.e., an object in $\Omega \setminus \Omega^*$. For a state σ, we denote by $U(\sigma) = \{j \in [m] : f_j \ni \sigma\}$ the set of (indices of) flaws present in σ. (Here and elsewhere, we shall blur the distinction between flaws and their indices.) We consider algorithms which start in a state sampled from a probability distribution θ and, in each flawed state $\sigma \in \Omega^*$, choose a flaw $f_i \in U(\sigma)$, and attempt to leave ("fix") f_i by moving to a new state τ selected with probability $\rho_i(\sigma, \tau)$. We refer to an attempt to fix a flaw, successful or not, as *addressing* it. We say that a transition $\sigma \to \tau$, made to address flaw f_i, *introduces* flaw $f_j \in U(\tau)$ if $f_j \notin U(\sigma)$ or if $j = i$. (Thus, a flaw (re)introduces itself when a transition fails to address it.)

Recall that θ denotes the probability distribution of the starting state. We denote by $\mathrm{Span}(\theta)$ the set of flaw indices that may be present in the initial state, i.e., $\mathrm{Span}(\theta) = \bigcup_{\sigma \in \Omega: \theta(\sigma) > 0} U(\sigma)$.

Let π be an arbitrary permutation over $[m]$. We say that an algorithm follows the *π-strategy* if at each step the flaw it chooses to address is the one corresponding to the element of $U(\sigma)$ of lowest index according to π.

We now formalize the definitions of primary flaws and charges introduced informally in the introduction.

Definition II.1. *A flaw f_i is* primary *if for every $\sigma \in f_i$ and every $j \neq i$, addressing f_j at σ always results in some $\tau \in f_i$, i.e., f_i is never eradicated collaterally. For a given set $S \subseteq [m]$, we write S^P and S^N to denote the indices that correspond to primary and non-primary flaws in S, respectively.*

Definition II.2. *We say that a set of flaws T* covers *a set of flaws S if $T^P = S^P$ and $T^N \supseteq S^N$.*

Definition II.3. *For a state $\tau \in \Omega$, flaw f_i, and set of flaws S, let*

$$\mathrm{In}_i^S(\tau) = \{\sigma \in f_i : \text{the set of flaws introduced by the transition } \sigma \to \tau \text{ covers } S\}.$$

Let $\mu > 0$ be an arbitrary measure on Ω. For every $i \in [m]$ and $S \subseteq [m]$, the charge *of (i, S) with respect to μ is,*

$$\gamma_i^S = \max_{\tau \in \Omega} \left\{ \frac{1}{\mu(\tau)} \sum_{\sigma \in \mathrm{In}_i^S(\tau)} \mu(\sigma) \rho_i(\sigma, \tau) \right\}. \quad (5)$$

We now state the formal version of our main result, Theorem I.2 of the introduction.

Theorem II.4 (Main Result)**.** *If there exist positive real numbers $\{\psi_i\}_{i \in [m]}$ such that, for every $i \in [m]$,*

$$\zeta_i := \frac{1}{\psi_i} \sum_{S \subseteq [m]} \gamma_i^S \prod_{j \in S} \psi_j \; < 1, \quad (6)$$

then, for every permutation π over $[m]$, the probability that an algorithm following the π-strategy fails to reach a flawless state within $(T_0 + s)/\delta$ steps is 2^{-s}, where $\delta = 1 - \max_{i \in [m]} \zeta_i$, and

$$T_0 = \log_2 \mu_{\min}^{-1} + m \log_2 \left(\frac{1 + \psi_{\max}}{\psi_{\min}} \right),$$

with $\mu_{\min} = \min_{\sigma \in \Omega} \mu(\sigma)$, $\psi_{\max} = \max_{i \in [m]} \psi_i$ and $\psi_{\min} = \min_{i \in [m]} \psi_i$.

Remark II.1. *In typical applications, μ and $\{\psi_i\}_{i \in [m]}$ are such that $T_0 = O(\log |\Omega| + m)$ and the sum in* (6) *is easily computable, as $\gamma_i^S = 0$ for the vast majority of subsets S.*

Remark II.2. *For any fixed permutation π, the charges γ_i^S can be reduced by removing from $\mathrm{In}_i^S(\tau)$ every state for which i is not the lowest indexed element of $U(\sigma)$ according to π.*

Remark II.3. *Theorem II.4 also holds for algorithms that use flaw choice strategies other than π-strategies. We discuss some such strategies in Section III-D. However, there is good reason to expect that it does* not *hold for* arbitrary *flaw choice strategies (see [34]).*

Finally, we state a refinement of our running time bound that will be important in order to get the best convergence guarantees in the applications of pure backtracking algorithms in Section V.

Remark II.4. *The upper bound on T_0 in Theorem II.4 can be replaced by the more refined bound:*

$$T_0 = \log_2 \left(\max_{\sigma \in \Omega} \frac{\theta(\sigma)}{\mu(\sigma)} \right) + \log_2 \left(\sum_{S \subseteq \mathrm{Span}(\theta)} \prod_{j \in S} \psi_j \right) + \log_2 \left(\max_{S \subseteq [m]} \frac{1}{\prod_{j \in S} \psi_j} \right) .$$

Moreover, if (as in pure backtracking algorithms) every flaw is primary, and (as is typical in pure backtracking algorithms) every flaw is present in the initial state, and if $\psi_i \in (0, 1]$ for all i, then $T_0 = \log_2 \mu_{\min}^{-1}$.

B. Application to graph coloring

In graph coloring one is given a graph $G = (V, E)$ and the goal is to find a mapping of V to a set of q colors so that no edge in E is monochromatic. The *chromatic number*, $\chi(G)$, of G is the smallest integer q for which this is possible. Given a set $\mathcal{L}_v$ of colors for each vertex v (called a *list*), a list-coloring maps each $v \in V$ to a color in $\mathcal{L}_v$ so that no edge in E is monochromatic. A graph is q-list-colorable if it has a list-coloring no matter how one assigns a list of q colors to each vertex. The *list chromatic number*, $\chi_\ell(G)$, is the smallest q for which G is q-list-colorable. Clearly $\chi_\ell(G) \geq \chi(G)$. A celebrated result of Johansson [31] established that there exists a large constant $C > 0$ such that every *triangle-free* graph with maximum degree $\Delta \geq \Delta_0$ can be list-colored using $C\Delta/\ln \Delta$ colors. Very recently, using the entropy compression method, Molloy [35] improved Johansson's result, replacing C with $(1 + \epsilon)$ for any $\epsilon > 0$ and all $\Delta \geq \Delta_\epsilon$. (Soon thereafter, Bernshteyn [12] established the same bound for the list chromatic number, non-constructively, via the Lopsided LLL, and Iliopoulos [29] showed that the algorithm of Molloy can be analyzed using the algorithmic LLL condition of [3], avoiding the need for a problem-specific entropy compression argument.)

Our first result related to graph coloring is a generalization of Molloy's result, bounding the list-chromatic number as a function of the number of triangles in each neighborhood. Specifically, in Section IV we establish the following theorem which is a key ingredient in the proof of Theorem I.1. Note that, in order to comply with the standard notation used in results in the area, we express the bound on the number of triangles as Δ^2/f; the triangle-free case then corresponds to $f = \Delta^2 + 1$. We stress that Molloy's proof breaks in the presence of even a single triangle per vertex.

Theorem II.5. *Let G be any graph with maximum degree Δ in which the neighbors of every vertex span at most Δ^2/f edges. For all $\epsilon > 0$, there exists Δ_ϵ such that if $\Delta \geq \Delta_\epsilon$ and $f \in [\Delta^{\frac{2+2\epsilon}{1+2\epsilon}}(\ln \Delta)^2, \Delta^2 + 1]$, then*

$$\chi_\ell(G) \leq (1+\epsilon)\Delta/\ln\sqrt{f} \ .$$

Furthermore, if G is a graph on n vertices then, for every $c > 0$, there exists an algorithm that constructs such a coloring in polynomial time with probability at least $1 - \frac{1}{n^c}$.

Theorem II.5 is interesting for several reasons. First, random graphs suggest that it is sharp, i.e., that no efficient algorithm can color graphs satisfying the conditions of the theorem with $(1-\epsilon)\Delta/\ln\sqrt{f}$ colors. More precisely, Proposition II.1 below, proved in Appendix D, implies that any such algorithm would entail coloring random graphs using fewer than twice as many colors as their chromatic number. This would be a major (and unexpected) breakthrough in random graph theory, where beating this factor of two has been an elusive goal for over 30 years. Besides the lack of progress, further evidence for the optimality of this factor of two is that it corresponds precisely to a phase transition in the geometry of the set of colorings [1], known as the *shattering threshold*. Second, Theorem II.5 establishes the existence of an algorithm that is robust enough to apply to worst-case graphs, while at the same time matching the performance of the best known (and highly tuned) algorithms for random graphs:

Proposition II.1. *For every $\epsilon > 0$ and $d \in (d_\epsilon \ln n, (n \ln n)^{\frac{1}{3}})$, there exist $\Delta = \Delta(d,\epsilon)$ and $f = f(d,\epsilon)$ such that with probability tending to 1 as $n \to \infty$, a random graph $G = G(n, d/n)$ satisfies the conditions of Theorem II.5 and $\chi(G) \geq (\frac{1}{2} - \epsilon)\Delta/\ln\sqrt{f}$.*

Third, armed with Theorem II.5, we are able to prove the following result concerning the chromatic number of *arbitrary* graphs, as a function of the maximum degree and the maximum number of triangles in any neighborhood:

Theorem II.6. *Let G be a graph with maximum degree Δ in which the neighbors of every vertex span at most Δ^2/f edges. For all $\epsilon > 0$, there exist $\Delta_\epsilon, f_\epsilon$ such that if $\Delta \geq \Delta_\epsilon$ and $f \in [f_\epsilon, \Delta^2 + 1]$, then*

$$\chi(G) \leq (2+\epsilon)\Delta/\ln\sqrt{f} \ . \tag{7}$$

Furthermore, if G is a graph on n vertices then, for every $c > 0$, there exists an algorithm that constructs such a coloring in polynomial time with probability at least $1 - \frac{1}{n^c}$.

Theorem II.6 improves a classical result of Alon, Krivelevich and Sudakov [7] which established (7) with an unspecified (large) constant in place of $2 + \epsilon$. The main idea in their analysis is to break down the input graph into triangle-free subgraphs, and color each one of them separately using distinct sets of colors by applying the result of Johansson [31]. Note that even if one used Molloy's recent result [35] in place of Johansson's in this scheme, the corresponding constant would still be in the thousands. Instead, we break down the graph into subgraphs with *few* triangles per neighborhood, and use Theorem II.5 to color the pieces. The proof of Theorem II.6 can be found in the full version of this paper [4].

As final remark, we note that Vu [45] proved the analogue of the main result of [7] (again with a large constant) for the list chromatic number. While we don't currently see how to sharpen Vu's result to an analogue of Theorem II.6 for the list chromatic number using our techniques, we note that our Theorem II.5 improves over [45] for all $f \geq \Delta^{\frac{2+2\epsilon}{1+2\epsilon}}(\ln \Delta)^2$.

III. Proof of Main Theorem

In Sections III-B and III-C we present the proof of our main result, Theorem II.4. In Section III-D we show how to extend the theorem to allow flaw choice strategies other than following a fixed permutation over flaws.

Throughout this section we use standard facts about operator norms, summarized briefly in Appendix A.

A. Charges as norms of transition matrices

We will first show how charges can be seen as the norms of certain transition matrices. For more concrete examples of this connection, see Appendix B.

Recall that for any $S \subseteq [m]$, we denote by S^P and S^N the subsets of S that correspond to primary and non-primary flaws, respectively.

Definition III.1. *For every $i \in [m]$ and every set of flaw indices $S \subseteq [m]$, let A_i^S be the $|\Omega| \times |\Omega|$ matrix where $A_i^S[\sigma, \tau] = \rho_i(\sigma, \tau)$ if the set of flaws introduced by $\sigma \to \tau$ covers S, i.e., the set of primary flaws introduced by the transition $\sigma \to \tau$* equals *S^P and the set of non-primary flaws introduced by $\sigma \to \tau$ contains S^N; otherwise $A_i^S[\sigma, \tau] = 0$.*

Let $\| \cdot \|_1$ denote the matrix norm induced by the L^1-vector-norm, and recall that it is equal to the max column sum. Let also $M = \mathrm{diag}(\mu(\sigma))$ denote the $|\Omega| \times |\Omega|$ diagonal matrix whose entries correspond to the probability measure μ. Our key observation is that the charges γ_i^S introduced in (5) can be expressed as

$$\gamma_i^S = \|M A_i^S M^{-1}\|_1 \ . \tag{8}$$

The reader is encouraged to verify this equivalence, which is an immediate consequence of the definitions.

Remark III.1. *Although we are specializing here to the $\| \cdot \|_1$ norm and matrix $M = \mathrm{diag}(\mu(\sigma))$, Theorem II.4 holds for any choice of matrix norm and invertible matrix M. It is an interesting research direction whether using other norms can be useful in applications.*

B. Tracking the set of current flaws

We say that a trajectory $\Sigma = (\sigma_1, \sigma_2, \dots, \sigma_{t+1})$ followed by the algorithm is a *bad t-trajectory* if every state σ_i, $i \in [t+1]$, is flawed. Thus, our goal is to bound the probability that the algorithm follows a bad t-trajectory.

Given a bad trajectory, we will track the flaws introduced into the state at each step, where a flaw is said to "introduce itself" whenever addressing it fails to remove it. Of the flaws introduced at each step, we disregard those that later get eradicated collaterally, i.e., by an action addressing some other flaw. The rest form the "witness sequence" of the trajectory, i.e., a sequence of sets of flaws.

Fix any permutation π on $[m]$. For any $S \subseteq [m]$, let $\pi(S) = \min_{j \in S} \pi(j)$, i.e., the lowest index in S according to π. Recalling that $U(\sigma)$ is the set of indices of flaws present in σ, in the following we assume that the index of the flaw addressed in state σ is $\pi(U(\sigma))$, which we sometimes abbreviate as $\pi(\sigma)$. Also, to lighten notation, we will denote $A \setminus \{\pi(B)\}$ by $A - \pi(B)$.

Definition III.2. *Let $\Sigma = (\sigma_1, \sigma_2, \dots, \sigma_{t+1})$ be any bad t-trajectory. Let $B_0 = U(\sigma_1)$. For $1 \le i \le t$, let*

$$B_i = U(\sigma_{i+1}) \setminus [U(\sigma_i) - \pi(\sigma_i)] \ ,$$

i.e., B_i comprises the indices of the flaws introduced in the i-th step. For $0 \le i \le t$, let

$$\begin{aligned} C_i = \{k \in B_i \mid \exists j \in [i+1, t] : \\ k \notin U(\sigma_{j+1}) \wedge \forall \ell \in [i+1, j] : k \neq \pi(\sigma_\ell)\} \ , \end{aligned}$$

i.e., C_i comprises the indices of the flaws introduced in the i-th step that get eradicated collaterally. The witness sequence *of Σ is the sequence of sets*

$$w(\Sigma) = (B_0 \setminus C_0, B_1 \setminus C_1, \dots, B_t \setminus C_t) \ .$$

A crucial feature of witness sequences is that they allow us to recover the sequence of flaws addressed.

Definition III.3. *Given an* arbitrary *sequence $S_0, \dots, S_t$ of subsets of $[m]$, let $S_1^* = S_0$, while for $1 \le i \le t$, let*

$$S_{i+1}^* = \begin{cases} [S_i^* - \pi(S_i^*)] \cup S_i & \textit{if } S_i^* \neq \emptyset \ ; \\ \emptyset & \textit{otherwise} \ . \end{cases}$$

If $S_i^* \neq \emptyset$ for all $1 \le i \le t$, then we say that $(S_i)_{i=0}^t$ is plausible *and write $\pi(S_i^*) = (i)$.*

Lemma III.4. *If $\Sigma = (\sigma_1, \sigma_2, \dots, \sigma_{t+1})$ is any bad t-trajectory, then $w(\Sigma) = (S_0, \dots, S_t)$ is plausible, $\pi(\sigma_i) = \pi(S_i^*) = (i)$ for all $1 \le i \le t$, and for every flaw index $z \in [m]$, the number of times z occurs in the multiset $\bigcup_{i=0}^t S_i$ minus the number of times it occurs in the multiset $\bigcup_{i=1}^t (i)$ equals $\mathbf{1}_{z \in S_{t+1}^*}$.*

Proof: Recall that $S_i = B_i \setminus C_i$. For $1 \le i \le t+1$, let L_i comprise the elements of $U(\sigma_i)$ eradicated collaterally during the i-th step, and let H_i comprise the elements of $U(\sigma_i)$ eradicated collaterally during any step $j \ge i$. Observe that $H_{i+1} = (H_i \setminus L_i) \cup C_i$. We will prove, by induction, that for all $1 \le i \le t+1$,

$$S_i^* \subseteq U(\sigma_i); \tag{9}$$

$$U(\sigma_i) \setminus S_i^* = H_i. \tag{10}$$

Observe that if (9) and (10) hold for a given i, then $\pi(\sigma_i) = \pi(S_i^*)$, since $\pi(\sigma_i) \notin H_i$ by the definition of H_i, and $\pi(A) = \pi(A \setminus B)$ whenever $\pi(A) \notin B$. Moreover, $S_i^* \neq \emptyset$, because otherwise $U(\sigma_i) = H_i$, an impossibility. To complete the proof it suffices to note that for any $z \in [m]$, the difference in question equals $\mathbf{1}_{z \in U(\sigma_{t+1})}$ and that $U(\sigma_{t+1}) = S_{t+1}^*$ since, by definition, $H_{t+1} = \emptyset$. The inductive proof is as follows.

For $i = 1$, (9) and (10) hold since $S_1^* = B_0 \setminus C_0$, while $U(\sigma_1) = B_0$. If (9) and (10) hold for some $i \ge 1$, then $S_{i+1}^* = [S_i^* - \pi(\sigma_i)] \cup S_i$ while, by definition, $U(\sigma_{i+1}) = [(U(\sigma_i) - \pi(\sigma_i)) \setminus L_i] \cup B_i$. Thus, the fact that $S_i^* \subseteq U(\sigma_i)$ trivially implies $S_{i+1}^* \subseteq U(\sigma_{i+1})$, while

$$\begin{aligned} U(\sigma_{i+1}) \setminus S_{i+1}^* &= ((U(\sigma_i) \setminus S_i^*) \setminus L_i) \cup (B_i \setminus S_i) \\ &= (H_i \setminus L_i) \cup C_i = H_{i+1} \ . \end{aligned}$$

This concludes the proof. ∎

The first step in our proof of Theorem II.4 is to give an upper bound on the probability that a given witness sequence occurs in terms of the charges γ_i^S. In particular, and in order to justify Remark III.1, we will use an arbitrary norm $\|\cdot\|$ and invertible matrix M.

Recall that $\|\cdot\|_*$ denotes the *dual* of norm $\|\cdot\|$ and let $\theta^\top \in [0,1]^{|\Omega|}$ denote the row vector corresponding to the probability distribution of the initial state σ_1. Moreover, for a state σ, let e_σ denote the indicator vector of σ, i.e., $e_\sigma[\sigma] = 1$ and $e_\sigma[\tau] = 0$ for all $\tau \in \Omega \setminus \{\sigma\}$.

Lemma III.5. *Fix any integer $t \geq 0$ and let Σ be the random variable $(\sigma_1, \dots, \sigma_{t+1})$. Fix any arbitrary invertible matrix M and operator norm $\|\cdot\|$, and let $\lambda_i^S = \|M A_i^S M^{-1}\|$. For any plausible sequence $\phi = (S_0, \dots, S_t)$,*

$$\Pr[w(\Sigma) = \phi] \leq \|\theta^\top M^{-1}\|_* \left(\sum_{\tau \in \Omega} \|M e_\tau\| \right) \prod_{i=1}^{t} \lambda_{(i)}^{S_i}. \tag{11}$$

Proof: Recall that for any $S \subseteq [m]$, we denote by S^P and S^N the subsets of S that correspond to primary and non-primary flaws, respectively. By Definition III.2 and Lemma III.4, a necessary condition for $w(\Sigma) = \phi$ to occur is that $(i) \in U(\sigma_i)$ and $S_i \subseteq B_i$, for every $1 \leq i \leq t$. Moreover, since primary flaws are never eradicated collaterally, i.e., $C_i^P = \emptyset$ always, it must also be that $S_i^P = B_i^P$ for $1 \leq i \leq t$. Fix any state $\tau \in \Omega$. The probability that $(1) \in U(\sigma_1) \wedge S_1^P = B_1^P(\Sigma) \wedge S_1^N \subseteq B_1^N(\Sigma) \wedge \sigma_2 = \tau$ equals the τ-column (coordinate) of the row-vector $\theta^\top A_{(1)}^{S_1}$. More generally, we see that for any $t \geq 1$,

$$\Pr\left[\bigwedge_{i=1}^{t} \left((i) \in U(\sigma_i)\right) \bigwedge_{i=1}^{t} \left(S_i^P = B_i^P\right) \bigwedge_{i=1}^{t} \left(S_i^N \subseteq B_i^N\right) \bigwedge \sigma_{t+1} = \tau \right] = \theta^\top \prod_{i=1}^{t} A_{(i)}^{S_i} e_\tau. \tag{12}$$

Consider now any vector norm $\|\cdot\|$ and the corresponding operator norm. By (29),

$$\begin{aligned} \theta^\top \prod_{i=1}^{t} A_{(i)}^{S_i} e_\tau &= \theta^\top M^{-1} \left(\prod_{i=1}^{t} M A_{(i)}^{S_i} M^{-1} \right) M e_\tau \\ &\leq \left\| \theta^\top M^{-1} \left(\prod_{i=1}^{t} M A_{(i)}^{S_i} M^{-1} \right) \right\|_* \|M e_\tau\|. \end{aligned} \tag{13}$$

Summing (13) over all $\tau \in \Omega$ we conclude that

$$\begin{aligned} \Pr[w(\Sigma) = \phi] &= \sum_{\tau \in \Omega} \Pr[w(\Sigma) = \phi \wedge \sigma_{t+1} = \tau] \\ &\leq \left\| \theta^\top M^{-1} \prod_{i=1}^{t} M A_{(i)}^{S_i} M^{-1} \right\|_* \sum_{\tau \in \Omega} \|M e_\tau\| . \end{aligned} \tag{14}$$

Applying (31) and then (30) to (14), and recalling the definition of $\lambda_{(i)}^{S_i}$, we conclude that

$$\begin{aligned} &\Pr[w(\Sigma) = \phi] \\ &\leq \|\theta^\top M^{-1}\|_* \left(\sum_{\tau \in \Omega} \|M e_\tau\| \right) \prod_{i=1}^{t} \|M A_{(i)}^{S_i} M^{-1}\| \\ &= \|\theta^\top M^{-1}\|_* \left(\sum_{\tau \in \Omega} \|M e_\tau\| \right) \prod_{i=1}^{t} \lambda_{(i)}^{S_i} , \end{aligned}$$

as claimed. ∎

Now define the set

$$\mathcal{F}_t = \{w(\Sigma) : \Sigma \text{ is a bad } t\text{-trajectory of the algorithm}\}.$$

Since $\mathcal{F}_t$ contains only plausible sequences, an immediate corollary of Lemma III.5 is a bound on the probability that the algorithm fails in t steps.

Corollary III.6. *The probability that the algorithm fails to reach a flawless state within t steps is at most*

$$\left(\max_{\sigma \in \Omega} \frac{\theta(\sigma)}{\mu(\sigma)} \right) \cdot \sum_{\phi \in \mathcal{F}_t} \prod_{i=1}^{t} \gamma_{(i)}^{S_i} . \tag{15}$$

Proof: We apply Lemma III.5 with $M = \mathrm{diag}(\mu(\sigma))$ and the $\|\cdot\|_1$-norm. Since the dual norm of $\|\cdot\|_1$ is $\|\cdot\|_\infty$, we have $\|\theta^\top M^{-1}\|_* = \max_{\sigma \in \Omega} \frac{\theta(\sigma)}{\mu(\sigma)}$. Combining this with the fact that $\sum_{\tau \in \Omega} \|M e_\tau\|_1 = 1$ concludes the proof. ∎

Thus, to complete the proof of Theorem II.4 we are left with the task of bounding the sum in (15).

C. Bounding the sum

Given $\psi_1, \dots, \psi_m > 0$ and $S \subseteq [m]$, let $\Psi(S) = \prod_{j \in S} \psi_j$, with $\Psi(\emptyset) = 1$. For each $i \in [m]$, let

$$\zeta_i = \frac{1}{\psi_i} \sum_{S \subseteq [m]} \gamma_i^S \Psi(S) .$$

Finally, for each $i \in [m]$ consider the distribution on $2^{[m]}$ that assigns to each $S \subseteq [m]$ the probability

$$p(i, S) := \frac{\gamma_i^S \Psi(S)}{\sum_{S \subseteq [m]} \gamma_i^S \Psi(S)} = \frac{\gamma_i^S \Psi(S)}{\zeta_i \psi_i} .$$

For any $S_0 \subseteq [m]$, let $\mathcal{F}_t(S_0)$ comprise the witness sequences in $\mathcal{F}_t$ whose first set is S_0. Consider the probability distribution on sequences of subsets of $[m]$ generated as follows: $R_1 = S_0$; for $i \geq 1$, if

$R_i \neq \emptyset$, then $R_{i+1} = (R_i - \pi(R_i)) \cup S_i$, where $\Pr[S_i = S] = p(\pi(R_i), S)$ for any $S \subseteq [m]$. Under this distribution, by Lemma III.4, each $\phi = (S_0, \ldots, S_t) \in \mathcal{F}_t(S_0)$ receives probability $p_\phi = \prod_{i=1}^{t} p((i), S_i)$, while $\sum_{\phi \in \mathcal{F}_t(S_0)} p_\phi \leq 1$. At the same time, by the last claim in Lemma III.4,

$$p_\phi = \prod_{i=1}^{t} p((i), S_i) = \left(\prod_{i=1}^{t} p((i), S_i) \frac{\psi_{(i)}}{\Psi(S_i)} \right) \frac{\Psi(S_{t+1}^*)}{\Psi(S_0)}$$

$$= \frac{\Psi(S_{t+1}^*)}{\Psi(S_0)} \prod_{i=1}^{t} \frac{\gamma_{(i)}^{S_i}}{\zeta_{(i)}} \quad . \tag{16}$$

Combining (16) with the fact that $\sum_{\phi \in \mathcal{F}_t(S_0)} p_\phi \leq 1$, we obtain

$$\sum_{\phi \in \mathcal{F}_t(S_0)} \prod_{i=1}^{t} \frac{\gamma_{(i)}^{S_i}}{\zeta_{(i)}} \leq \max_{S \subseteq [m]} \frac{\Psi(S_0)}{\Psi(S)} \quad . \tag{17}$$

Let $\zeta = \max_{i \in [m]} \zeta_i$. Then, summing equation (17) over all possible sets S_0 yields

$$\sum_{\phi \in \mathcal{F}_t} \prod_{i=1}^{t} \gamma_{(i)}^{S_i} = \sum_{S_0 \subseteq \mathrm{Span}(\theta)} \sum_{\phi \in \mathcal{F}_t(S_0)} \prod_{i=1}^{t} \gamma_{(i)}^{S_i}$$

$$\leq \zeta^t \sum_{S_0 \subseteq \mathrm{Span}(\theta)} \sum_{\phi \in \mathcal{F}_t(S_0)} \prod_{i=1}^{t} \frac{\gamma_{(i)}^{S_i}}{\zeta_{(i)}}$$

$$\leq \max_{S \subseteq [m]} \sum_{S_0 \subseteq \mathrm{Span}(\theta)} \frac{\Psi(S_0)}{\Psi(S)} \quad . \tag{18}$$

Proofs of Theorem II.4 and Remark II.4 : Combining (18) with Corollary III.6, we see that the binary logarithm of the probability that the algorithm does not encounter a flawless state within t steps is at most $t \log_2 \zeta + T_0$, where

$$T_0 = \log_2 \left(\max_{\sigma \in \Omega} \frac{\theta(\sigma)}{\mu(\sigma)} \right) +$$
$$\log_2 \left(\sum_{S \subseteq \mathrm{Span}(\theta)} \Psi(S) \right) + \log_2 \left(\max_{S \subseteq [m]} \frac{1}{\Psi(S)} \right).$$

Therefore, if $t = (T_0 + s)/\log_2(1/\zeta) \leq (T_0 + s)/\delta$, the probability that the algorithm does not reach a flawless state within t steps is at most 2^{-s}. This concludes the proofs of the first part of Remark II.4 and Theorem II.4, since $\max_{\sigma \in \Omega} \theta(\sigma) \leq 1$ and

$$\log_2 \Big(\sum_{S \subseteq \mathrm{Span}(\theta)} \Psi(S) \Big) + \log_2 \left(\max_{S \subseteq [m]} \frac{1}{\Psi(S)} \right)$$
$$\leq \log_2 \frac{\prod_{i=1}^{m} (1 + \psi_i)}{(\psi_{\min})^m} \leq m \log_2 \left(\frac{1 + \psi_{\max}}{\psi_{\min}} \right) \quad .$$

To see the second part of Remark II.4, let $\mathcal{I}(\theta)$ denote the set comprising the sets of flaw-indices that may be present in a state selected according to θ. Recall now that when every flaw is primary, the only equivalence classes of $\mathcal{F}_t$ that contribute to the sum in (18) are those for which $S_0 \in \mathcal{I}(\theta)$. Thus, for backtracking algorithms, the sum over $S \subseteq \mathrm{Span}(\theta)$ in the definition of T_0 can be restricted to $S \in \mathcal{I}(\theta)$. Finally, if every flaw is always present in the initial state and $\psi_i \in (0, 1]$ for every $i \in [m]$, then $I(\theta) = \{F\}$ and $\log \left(\frac{1}{\max_{S \subseteq [m]} \prod_{j \in S} \psi_j} \right) = -\log_2 \prod_{j \in [m]} \psi_j$. This implies that the second and third term in the expression for T_0 in Remark II.4 cancel out, concluding its proof. ∎

D. Other flaw choice strategies

The only place where we used the fact that the flaw choice is based on a fixed permutation was to assert, in Lemma III.4, that the witness sequence of a trajectory determines the sequence of addressed flaws. Thus, our analysis is in fact valid for every flaw choice strategy that shares this property.

One example of such a strategy is "pick a random occurring flaw and address it". To implement this, we can fix a priori an infinite sequence of uniformly random permutations $\pi_1, \pi_2, \ldots$ and at the i-th step address the lowest indexed flaw present according to π_i. It is straightforward to see that Lemma III.4 still holds if we replace π by π_i therein and in Definition III.3.

As a second example, consider the following recursive way to chose which flaw to address at each step (which makes the algorithm non-Markovian). The algorithm now maintains a stack. The flaws present in σ_1, ordered according to some permutation π, comprise the initial stack contents. The algorithm starts by addressing the flaw at the top of the stack, i.e., $\pi(\sigma_1)$, as before. Now, though, any flaws introduced in the i-th step, i.e., the elements of B_i, go on the top of the stack (ordered by π), while all eradicated flaws are removed from the stack. The algorithm terminates when the stack empties. It is not hard to see that, by taking S_0 to be the initial stack contents, popping the flaw at the top of the stack at each step, and adding S_i to the top of the stack (ordered by π), the sequence of popped flaws is the sequence of addressed flaws.

IV. Graph coloring proofs

A. The algorithm

To prove Theorem II.5 we will generalize the algorithm of Molloy [35] for coloring triangle-free graphs. The main issue we have to address is that in the presence of triangles, the natural generalization of Molloy's algorithm introduces monochromatic edges when the neighborhood of a vertex is recolored. As a result, the existing analysis fails completely even if each vertex participates in just one triangle. To get around this problem, we introduce backtracking steps into the algorithm, whose analysis is enabled by our new convergence condition, Theorem II.4.

For each vertex $v \in V$, let N_v denote the neighbors of v and let $E_v = \{\{u_1, u_2\} : u_1, u_2 \in N_v\}$ denote the edges spanned by them. Recall that the color-list of v is denoted by $\mathcal{L}_v$. It will be convenient to treat Blank also as a color. Indeed, the initial distribution θ of our algorithm assigns all its probability mass to the state where every vertex is colored Blank. Whenever assigning a color to a vertex creates monochromatic edges, the algorithm will immediately uncolor enough vertices so that no monochromatic edge remains. Edges with two Blank endpoints are not considered monochromatic. To uncolor a vertex v, the algorithm picks a monochromatic edge e incident to v and assigns e to v instead of a color, thus also creating a record of the reason for the uncoloring. Thus,

$$\Omega \subseteq \prod_{v \in V} \{\mathcal{L}_v \cup \{\mathrm{Blank}\} \cup E_v\} \quad .$$

Let $L = (1+\epsilon)\frac{\Delta}{\ln f} f^{-\frac{1}{2+2\epsilon}}$ and assume Δ is sufficiently large so that $L \geq 10$.

The flaws. We let $L_v(\sigma) \subseteq (\mathcal{L}_v \cup \{\mathrm{Blank}\})$ be the set of colors we can assign to v in state σ without creating any monochromatic edge. We call these the *available colors for v in σ* and note that Blank is always available. For each $v \in V$, we define a flaw expressing the fact that there are "too few available colors for v," namely

$$B_v = \{\sigma \in \Omega : |L_v(\sigma)| < L\} \quad .$$

For each color c other than Blank, let $T_{v,c}(\sigma)$ be the set of Blank neighbors of v for which c is available in σ, i.e., the vertices that may "compete" with v for color c. For each $v \in V$, we define a flaw expressing the fact that there is "too much competition for v's available (real) colors," namely

$$Z_v = \left\{ \sigma \in \Omega : \sum_{c \in L_v(\sigma) \setminus \mathrm{Blank}} |T_{v,c}(\sigma)| > \frac{L}{10} |L_v(\sigma)| \right\} .$$

Finally, for each $v \in V$ and $e \in E$ we define a flaw for the fact that v is uncolored (because of e), namely

$$f_v^e = \{\sigma \in \Omega : \sigma(v) = e\} \quad .$$

Let $F_v = B_v \cup Z_v \cup \bigcup_{e \in E} f_v^e$ and let $\Omega^+ = \Omega - \bigcup_{v \in V} F_v$; thus Ω^+ denotes the partial colorings that do not suffer from any of the above flaws.

Lemma IV.1 ([35]). *Given $\sigma \in \Omega^+$, a complete list-coloring of G can be found efficiently.*

The proof of Lemma IV.1 is a fairly standard application of the (algorithmic) LLL, showing that σ can be extended to a complete list-coloring by coloring all Blank vertices with actual colors. Thus, the heart of the matter is reaching a state in Ω^+ (i.e., a partial coloring avoiding all the above flaws).

The flaw choice. The algorithm can use any π-strategy in which every B-flaw has priority over every f-flaw.

The actions. To address f_v^e at σ, i.e., to color v, the algorithm simply chooses a color from $L_v(\sigma)$ uniformly at random and assigns it to v. The fact that B-flaws have higher priority than f-flaws implies that there are always at least L such choices.

Addressing B- and Z- flaws is significantly more sophisticated. For each vertex v, for each vertex $u \in N_v$, let $R_u^v(\sigma) \supseteq L_u(\sigma)$ comprise those colors having the property that assigning them to u in state σ creates no monochromatic edge except, perhaps, in E_v. To address either B_v or Z_v in σ, the algorithm selects an action according to the following procedure:

1: **procedure** RECOLOR(v, σ)
2: Assign to each colored vertex u in N_v a uniformly random color from $R_u^v(\sigma)$
3: **while** monochromatic edges exist **do**
4: Let u be the lowest indexed vertex participating in a monochromatic edge
5: Let e be the lowest indexed monochromatic edge with u as an endpoint
6: Uncolor u by assigning e to u

Lemma IV.2. *Let $S'(v, \sigma)$ be the set of colorings that can be reached at the end of Step 2 of* RECOLOR(v, σ) *and let $S''(v, \sigma)$ be the set of possible final colorings. Then $|S'(v, \sigma)| = |S''(v, \sigma)|$.*

Proof: Since Steps 4–6 are deterministic, $|S''(v, \sigma)| \leq |S'(v, \sigma)|$. To prove that $|S''(v, \sigma)| \geq |S'(v, \sigma)|$, we will prove that if $u \in N_v$ has distinct colors in $\sigma_1', \sigma_2' \in S'$, then there exists $z \in V$ such that $\sigma_1''(z) \neq \sigma_2''(z)$. Imagine that in Step 6 we also oriented e to point away from u. Then, in the resulting partial orientation, every vertex would have outdegree at most 1 and there would be no directed cycles. Consider the (potentially empty) oriented paths starting at u in σ_1'' and σ_2'', and let z be their last common vertex. If z is uncolored, then $\sigma_1''(z) = e_1$ and $\sigma_2''(z) = e_2$, where $e_1 \neq e_2$; if z is colored, then $\sigma_i''(z) = \sigma_i'(u)$. ∎

B. Proving termination

Let D_v be the set of vertices at distance 1 or 2 from v and let

$$S_v = \{B_u\}_{u \in D_v} \cup \{Z_u\}_{u \in D_v} \cup \{f_u^{\{u,w\}}\}_{u,w \in N_v} \quad .$$

To lighten notation, in the following we write $\gamma^S(f)$ instead of γ_f^S. Let $q = (1+\epsilon)\frac{\Delta}{\ln \sqrt{f}} \geq 1$.

Lemma IV.3. *For every vertex $v \in V$ and edge $e \in E$:*

(a) if $S \not\subseteq S_v$, then $\gamma^S(B_v) = \gamma^S(Z_v) = \gamma^S(f_v^e) = 0$;
(b) if $S \supseteq \{f_{u_1}^{\{u_1,u_2\}}, f_{u_2}^{\{u_1,u_2\}}\}$, then $\gamma^S(B_v) = \gamma^S(Z_v) = \gamma^S(f_v^e) = 0$;
(c) $\max_{S\subseteq F} \gamma^S(f_v^e) \le \frac{1}{L} =: \gamma(f_v^e)$;
(d) $\max_{S\subseteq F} \gamma^S(B_v) \le 2\mathrm{e}^{-\frac{L}{6}} =: \gamma(B_v)$;
(e) $\max_{S\subseteq F} \gamma^S(Z_v) \le 3q\mathrm{e}^{-\frac{L}{60}} =: \gamma(Z_v)$,

where the charges are computed with respect to the uniform measure over Ω.

We note that, while we give uniform bounds on the charges corresponding to each flaw, the analysis of our algorithm cannot be captured by the algorithmic LLL framework of [3]. This is because we will crucially exploit the existence of *primary* flaws.

Before we give the proof of Lemma IV.3, we first use it to derive Theorem II.5.

Proof of Theorem II.5: For every flaw $f \in F$, we will take $\psi_f = \gamma(f)\psi$, where $\psi > 0$ will be chosen later.

For any vertex $v \in V$, flaw $f \in \{B_v, Z_v, f_v^e\}$, and set of flaws $S \subseteq F$, Lemma IV.3 implies that $\gamma^S(f) = 0$ unless all B- and Z-flaws in S correspond to vertices in D_v, per part (a), and every edge $e \in E_v$ contributes at most one flaw to S, per part (b). Therefore, for $f \in \{B_v, Z_v, f_v^e\}$,

$$\begin{aligned}
&\frac{1}{\psi_f}\sum_{S\subseteq F}\gamma^S(f)\prod_{g\in S}\psi_g \\
&\le \frac{1}{\psi}\prod_{u\in D_v}(1+\gamma(B_u)\psi)(1+\gamma(Z_u)\psi) \ \times \\
&\quad \prod_{e=\{u_1,u_2\}\in E_v}\left(1+\gamma(f_{u_1}^e)\psi+\gamma(f_{u_2}^e)\psi\right) \ . \qquad (19)
\end{aligned}$$

To bound the right hand side of (19) we use parts (c)–(e) of Lemma IV.3 along with the facts $|D_v| \le \Delta^2+1$ and $|E_v| \le \Delta^2/f$ to derive (20) below. To derive (21), we use the facts that $2\mathrm{e}^{-\frac{L}{6}} \le 3q\mathrm{e}^{-\frac{L}{60}}$, since $q \ge 1$, and that $1 + x \le \mathrm{e}^x$ for all x. Thus, for $f \in \{B_v, Z_v, f_v^e\}$, we conclude

$$\begin{aligned}
&\frac{1}{\psi_f}\sum_{S\subseteq F}\gamma^S(f)\prod_{g\in S}\psi_g \\
&\le \frac{1}{\psi}\left(1+2\mathrm{e}^{-\frac{L}{6}}\psi\right)^{\Delta^2+1} \\
&\quad \times\left(1+3q\mathrm{e}^{-\frac{L}{60}}\psi\right)^{\Delta^2+1}\left(1+\frac{2\psi}{L}\right)^{\frac{\Delta^2}{f}} \qquad (20) \\
&\le \frac{1}{\psi}\exp\left(\frac{2\psi\Delta^2}{fL}+6q\mathrm{e}^{-\frac{L}{60}}\psi(\Delta^2+1)\right) \\
&:= \frac{1}{\psi}\exp(Q) \ . \qquad (21)
\end{aligned}$$

Setting $\psi = (1+\epsilon)$, we see that the right hand side of (21) is strictly less than 1 for all $\Delta \ge \Delta_\epsilon$, since $Q \xrightarrow{\Delta\to\infty} 0$ for all $f \in [\Delta^{\frac{2+2\epsilon}{1+2\epsilon}}(\ln\Delta)^2, \Delta^2+1]$. To see this last claim, recall that $L = (1+\epsilon)\frac{\Delta}{\ln f}f^{-\frac{1}{2+2\epsilon}}$ and $q = (1+\epsilon)\frac{\Delta}{\ln\sqrt{f}}$, and note that $\ln f < 3\ln\Delta$ and $f^{\frac{1+2\epsilon}{2+2\epsilon}} \ge \Delta(\ln\Delta)^{\frac{2+4\epsilon}{2+2\epsilon}}$. Thus,

$$\begin{aligned}
\frac{2\psi\Delta^2}{fL} &= \frac{2\Delta^2}{f\frac{\Delta}{\ln f}f^{-\frac{1}{2+2\epsilon}}} = \frac{2\Delta\ln f}{f^{\frac{1+2\epsilon}{2+2\epsilon}}} \le \frac{2\ln f}{(\ln\Delta)^{\frac{2+4\epsilon}{2+2\epsilon}}} \\
&\le \frac{6\ln\Delta}{(\ln\Delta)^{\frac{2+4\epsilon}{2+2\epsilon}}} = \frac{6}{(\ln\Delta)^{\frac{\epsilon}{1+\epsilon}}} \xrightarrow{\Delta\to\infty} 0 \ , \qquad (22)
\end{aligned}$$

while the facts $L = \Omega(\Delta^{\frac{\epsilon}{1+2\epsilon}})$ and $q \le (1+\epsilon)\Delta$ imply that $6q\mathrm{e}^{-\frac{L}{60}}\psi(\Delta^2+1) \xrightarrow{\Delta\to\infty} 0$. ■

Proof of Lemma IV.3.

Proof of part (a): Addressing B_v or Z_v by executing RECOLOR$(v,\cdot)$ only changes the color of vertices in N_v, with any resulting uncolorings being due to edges in E_v. Thus, only flaws in S_v may be introduced. Addressing f_v^e by coloring v trivially can only introduce flaws B_u, Z_u, where $u \in N_v$. ■

Proof of part (b): Since addressing an f-flaw never introduces another f-flaw, we only need to discuss procedure RECOLOR. Therein, vertices are uncolored serially in time, so that any time a vertex w is uncolored there exists, at the time of w's uncoloring, a monochromatic edge $e = \{w, u\}$. Therefore, an edge $e = \{u_1, u_2\}$ can never be the reason for the uncoloring of both its endpoints, i.e., $f_{u_1}^e \cap f_{u_2}^e = \emptyset$. ■

Proof of part (c): If addressing f_v^e results in τ, then the previous state σ must be the mutation of τ that results from assigning e to v. Since $\pi(\sigma) = f_v^e$ implies $\sigma \notin B_v$, it follows that $|L_v(\sigma)| \ge L$. Since colors are chosen uniformly from $L_v(\sigma)$, it follows that $\gamma(f_v^e) \le 1/L$. ■

Proof of parts (d) and (e): Observe that every flaw corresponding to an uncolored vertex is primary, since procedure RECOLOR never colors an uncolored vertex and addressing f_v^e only colors v. Thus, when computing $\gamma^S(f)$, for $f \in \{B_v, Z_v\}$ and $S \subseteq F$, we can restrict to pairs (σ, τ) such that the set of uncolored vertices in τ is exactly the union of the set of uncolored vertices in σ and the set $\{u \in N_v : f_u^e \in S\}$. Fixing $f \in \{B_v, Z_v\}$, $S \subseteq F$, and τ, let us denote by $\mathrm{In}_f^S(\tau)$ the candidate set of originating states, and by $\mathcal{U}_f^S(\tau)$ their common set of uncolored vertices. Then, for any $f \in \{B_v, Z_v\}$ and any $S \subseteq F$,

$$\gamma^S(f) = \max_{\tau\in\Omega}\sum_{\sigma\in\mathrm{In}_f^S(\tau)}\rho_f(\sigma,\tau) \ . \qquad (23)$$

To bound $\rho_f(\sigma, \tau)$ in (23), we recall that RECOLOR assigns to each colored vertex $u \in N_v$ a random color from $R_u^v(\sigma)$ and invoke Lemma IV.2 to derive the first equality in (24). For the second equality we observe that for every $u \in N_v$, the set R_u^v is determined by the colors of the vertices in $V \setminus N_v$. Since RECOLOR only changes the color of vertices in N_v, it follows that $R_u^v(\sigma) = R_u^v(\tau)$, yielding

$$\rho_f(\sigma,\tau) = \frac{1}{\prod_{u \in N_v \setminus \mathcal{U}_f^S(\tau)} |R_u^v(\sigma)|} = \frac{1}{\prod_{u \in N_v \setminus \mathcal{U}_f^S(\tau)} |R_u^v(\tau)|} =: \frac{1}{\Lambda_f^S(\tau)} \ . \tag{24}$$

Next we bound $|\mathrm{In}_f^S(\tau)|$ as follows. First we observe that if $\sigma \in \mathrm{In}_f^S(\tau)$, then $\sigma(u) \neq \tau(u)$ implies $u \in N_v \setminus \mathcal{U}_f^S(\tau)$ and, therefore, $\sigma(u) \in R_u^v(\tau)$ since $\sigma(u) \in L_u(\sigma) \subseteq R_u^v(\sigma) = R_u^v(\tau)$. Thus, the set of τ-mutations that result from recoloring each vertex in $N_v \setminus \mathcal{U}_f^S(\tau)$ with a color from $R_u^v(\tau)$ so that the resulting state belongs to f is a superset of $\mathrm{In}_f^S(\tau)$. Denoting this last set by $\mathrm{Viol}(f,\tau)$, we conclude that

$$\gamma^S(f) = \max_{\tau \in \Omega} \frac{|\mathrm{In}_f^S(\tau)|}{\Lambda_f^S(\tau)} \leq \max_{\tau \in \Omega} \frac{|\mathrm{Viol}(f,\tau)|}{\Lambda_f^S(\tau)} = \max_{\tau \in \Omega} \Pr[\textsc{recolor}(v,\tau) \in f] \ , \tag{25}$$

where for the last equality we use the definition of $\Lambda_f^S(\tau)$.

Remark IV.1. *We note that expressing the sum of the transition probabilities into a state in terms of a random experiment, as we do in* (25)*, was the key technical idea of [35] in order to apply the entropy compression method. It is also the one that breaks down if we allow our algorithm to go through improper colorings.*

To conclude the proof of Lemma IV.3 we prove the following bounds in Appendix C via fairly routine calculations.

Lemma IV.4. *For each vertex v and $\sigma \in \Omega$:*

(a) $\Pr[\textsc{recolor}(v,\sigma) \in B_v] \leq 2\mathrm{e}^{-\frac{L}{6}}$.
(b) $\Pr[\textsc{recolor}(v,\sigma) \in Z_v] \leq 3q\mathrm{e}^{-\frac{L}{60}}$.

■

V. Applications to Backtracking Algorithms

An important class of algorithms naturally devoid of "collateral fixes" are *backtracking* algorithms. In particular, consider a Constraint Satisfaction Problem (CSP) over a set of variables $V = \{v_1, v_2 \ldots, v_n\}$, each variable v_i taking values in a domain $\mathcal{D}_i$, with a set of constraints $\mathcal{C} = \{c_1, c_2, \ldots, c_m\}$ over these variables. The backtracking algorithms we consider operate as follows. (Note that in Step 1, we can always take θ to be the distribution under which all variables are unassigned; this does not affect the convergence condition (6) but may have a mild effect on the running time.)

Generic Backtracking Algorithm

1: Sample a partial non-violating assignment σ_0 according to a distribution θ and set $i = 0$
2: **while** unassigned variables exist **do**
3: Let v be the lowest indexed unassigned variable in σ_i
4: Choose a new value for v according to a state-dependent probability distribution
5: **if** one or more constraints are violated **then**
6: Remove the values from enough variables so that no constraint is violated
7: Let σ_{i+1} be the resulting assignment
8: $i \leftarrow i + 1$

Let Ω be the set of partial assignments to V that do not violate any constraint in $\mathcal{C}$. For each variable $v_i \in V$, let flaw $f_i \subseteq \Omega$ comprise the partial assignments in which v_i is unassigned. Clearly, each flaw f_i can only be removed by addressing it, as addressing any other flaw can only unassign v_i. Thus, every flaw is primary and a flawless state is a complete satisfying assignment.

In this section we present applications of our main theorem to analyze backtracking search algorithms. First, we prove a useful corollary of Theorem II.4 that holds in the so-called *variable setting*. Second, we analyze a backtracking algorithm of Esperet and Parreau [20] for acyclic edge coloring that lies outside the variable setting; in particular, we recover their 4Δ bound on the acyclic chromatic index and, further, we show how it can make constructive an existential result of Bernshteyn [13]. We emphasize that these analyses follow very easily from our framework.

A. The variable setting

In this section we show how we can use Theorem II.4 to employ backtracking algorithms in order to capture applications in the variable setting, i.e., the setting considered by Moser and Tardos. In particular, we consider a product measure over variables V and define a bad event for each constraint $c \in \mathcal{C}$ being violated. We will prove the following corollary of Theorem II.4.

Theorem V.1. *Let P be any product measure over a set of variables V and let A_c be the event that constraint c is violated. If there exist positive real numbers $\{\psi_v\}_{v \in V}$ such that for every variable $v \in V$,*

$$\frac{1}{\psi_v}\left(1 + \sum_{c \ni v} P(A_c) \prod_{u \in c} \psi_u\right) < 1 \ , \tag{26}$$

then there exists a backtracking algorithm that finds a satisfying assignment after an expected number of $O\left(\log(P_{\min}^{-1}) + |V| \log_2\left(\frac{1+\psi_{\max}}{\psi_{\min}}\right)\right)$ *steps.*

Before proving Theorem V.1, we first use it to capture a well-known application of the Lovász Local Lemma to sparse k-SAT formulas when P is the uniform measure. For a k-SAT formula Φ, we denote its maximum degree by $\Delta \equiv \Delta(\Phi)$, i.e., each variable of Φ is contained in at most Δ clauses.

Theorem V.2. *Every k-SAT formula Φ with maximum degree $\Delta < \frac{2^k}{\mathrm{e}k}$ is satisfiable. Moreover, there exists a backtracking algorithm that finds a satisfying assignment of Φ efficiently.*

Proof: Setting $\psi_v = \psi = 2\alpha > 0$ we see that it suffices to find a value $\alpha > 0$ such that

$$\frac{1}{\psi} + \frac{1}{2^k}\Delta\psi^{k-1} = \frac{1}{2\alpha} + \frac{1}{2}\Delta\alpha^{k-1} < 1 \ ,$$

which is feasible whenever

$$\Delta < \max_{\alpha>0} \frac{2\alpha - 1}{\alpha^k} = \frac{2^k}{k} \cdot \left(1 - \frac{1}{k}\right)^{k-1} \leq \frac{2^k}{\mathrm{e}k} \ .$$

■

Remark V.1. *In [21] it is shown that using a non-uniform product measure P one can improve the bound of Theorem V.2 to $\Delta < \frac{2^{k+1}}{\mathrm{e}(k+1)}$ and that this is asymptotically tight. We note that we can achieve the same bound using Theorem V.1 with the same P, but since this is a rather involved LLL application we will not explicitly present it here.*

Proof of Theorem V.1. We consider the following very simple backtracking algorithm. Start with each variable unassigned. Then, in each state σ, choose the lowest indexed unassigned variable v and sample a value for it according to the product measure P. If one or more constraints become violated, remove the value from every variable in the lowest indexed violated constraint.

Let Ω be the set of partial non-violating assignments. Let $\mu : \Omega \to \mathbb{R}$ be the probability measure that assigns to each state $\sigma \in \Omega$ the value $\mu(\sigma) \propto \prod_{\substack{v \in V \\ v \notin U(\sigma)}} P(\sigma(v))$, where for brevity we abuse notation by letting $P(\sigma(v))$ denote the event that variable v is assigned value $\sigma(v)$.

Theorem V.1 will follow immediately from the following lemma. (For brevity, we will index flaws with variables instead of integers.)

Lemma V.3. *For each vertex v and set of variables $S \neq \emptyset$,*

$$\gamma_v^S = \begin{cases} 1 \ \textit{if } S = \emptyset; \\ P(A_c) \ \textit{if } S = c, \textit{ where } c \ni v \textit{ is a constraint;} \\ 0 \ \textit{otherwise.} \end{cases}$$

Proof: Notice that the actions related to flaw f_v can only remove the value from sets of variables that correspond to constraints that contain v. Thus, $\gamma_v^S = 0$ for every set $S \neq \emptyset$ that does not correspond to a constraint containing v. Recalling the definition of charges and $U(\sigma)$, we have

$$\gamma_v^S = \max_{\tau\in\Omega} \sum_{\substack{\sigma \in f_v \\ S = U(\tau)\setminus(U(\sigma)\setminus\{v\})}} \frac{\mu(\sigma)}{\mu(\tau)}\rho_v(\sigma,\tau) \ . \tag{27}$$

To see the claim for the case of the empty set, notice that, given a state τ, there exists at most one state σ such that $\rho_v(\sigma,\tau) > 0$ and that $U(\tau) \setminus (U(\sigma) \setminus \{v\}) = \emptyset$. This is because we can uniquely reconstruct σ from τ by removing the value from v at τ. Then we have

$$\begin{aligned} \frac{\mu(\sigma)}{\mu(\tau)}\rho_v(\sigma,\tau) &= \frac{\prod_{u\in V\setminus U(\sigma)} P(\sigma(u))}{\prod_{u\in V\setminus U(\tau)} P(\tau(u))} P(\tau(v)) \\ &= \frac{1}{P(\tau(v))} P(\tau(v)) = 1 \ . \end{aligned}$$

To see the claim for the case where $S = c$, consider the set $\mathrm{viol}(c)$ consisting of the set of assignments of the variables of c that violate c. Notice now that for every state $\tau \in \Omega$ there is an injection from the set of states σ such that that $\rho_v(\sigma,\tau) > 0$ and $S = U(\tau) \setminus (U(\sigma) \setminus \{v\})$ to $\mathrm{viol}(c)$. This is because c should be violated at each such state σ, and hence σ should be of the form $\sigma = \tau_\alpha$ for $\alpha \in \mathrm{viol}(c)$, where τ_α is the state induced by τ when assigning α to the variables of c. Observe further that, for every state of the form $\tau_\alpha, \alpha \in \mathrm{viol}(c)$, we have

$$\begin{aligned} \frac{\mu(\tau_\alpha)}{\mu(\tau)}\rho_v(\tau_\alpha,\tau) &= \Big(\prod_{u\in c\setminus\{v\}} P\left(X_u = \tau_\alpha(u)\right)\Big) \\ &\times P(X_v = \tau(v)) = P\left(A_c^\alpha\right) \ , \end{aligned} \tag{28}$$

where $P(A_c^\alpha)$ is the probability of the event that the variables of c receive assignment α. Combining (28) with (27) and the fact that $P(A_c) = \sum_{\alpha\in\mathrm{viol}(c)} P(A_c^\alpha)$ concludes the proof of Lemma V.3. ■

Finally, plugging Lemma V.3 into Theorem II.4 concludes the proof of Theorem V.1.

B. Acyclic edge coloring

An edge-coloring of a graph is *proper* if all edges incident to each vertex have distinct colors. A proper edge coloring is *acyclic* if it has no bichromatic cycles, i.e., no cycle receives exactly two (alternating) colors. The smallest number of colors for which a graph G has an acyclic edge-coloring is denoted by $\chi'_a(G)$.

Acyclic Edge Coloring was originally motivated by the work of Coleman et al. [16, 17] on the efficient computation of Hessians and, since then, there has been a series of papers [6, 20, 26, 32, 36, 40] that upper bound $\chi'_a(G)$ for graphs with bounded degree. The current best result was given recently by Giotis *et* *al.* [22] who showed that $\chi'_a(G) \leq 3.74\Delta$ in graphs with maximum degree Δ.

A simple backtracking algorithm. We show how one can apply Theorem II.4 to recover the main application of the framework of Esperet and Parreau [20] with a much simpler proof. This is a canonical application of the entropy compression method to the analysis of backtracking algorithms and has inspired many other results in the area. (Indeed, the authors in [20] already give several applications of their techniques to other problems besides acyclic edge coloring.)

Let G be a graph with m edges $E = \{e_1, \ldots, e_m\}$ and suppose we have q available colors.

Definition V.4. *Given a graph $G = (V, E)$ and a (possibly partial) edge-coloring of G, say that color c is* 4-forbidden for *$e \in E$ if assigning c to e would result in either a violation of proper edge-coloration, or a bichromatic 4-cycle containing e. Say that c is* 4-available *if it is not 4-forbidden.*

Lemma V.5 ([20])**.** *In any proper edge-coloring of G, at most $2(\Delta - 1)$ colors are 4-forbidden for any $e \in E$.*

Proof: The 4-forbidden colors for $e = \{u, v\}$ can be enumerated as: (i) the colors on edges adjacent to u; and (ii) for each edge e_v adjacent to v, either the color of e_v (if no edge with that color is adjacent to u), or the color of some edge e' which together with e, e_v and an edge adjacent to u form a cycle of length 4. ∎

Consider the following backtracking algorithm for Acyclic Edge Coloring with $q = 2(\Delta - 1) + Q$ colors. At each step, choose the lowest indexed uncolored edge e and attempt to color it choosing uniformly at random among the 4-available colors for e. If one or more bichromatic cycles are created, then choose the lowest indexed one of them, say $C = \{e_{i_1}, e_{i_2}, \ldots, e_{i_{2\ell}} = e\}$, and remove the colors from all its edges except for e_{i_1} and e_{i_2}.

The main result of [20] states that every graph G admits an acyclic edge coloring with $q > 4(\Delta - 1)$ colors. Moreover, such a coloring can be found in $O\left(|E||V|\Delta^2 \ln \Delta\right)$ time with high probability.

We prove the following theorem, which achieves the same bound on q and improves the running time bound when the graph is dense.

Theorem V.6. *Every graph G admits an acyclic edge coloring with $q > 4(\Delta - 1)$ colors. Moreover, such a coloring can be found in $O(|E||V|\Delta)$ time with high probability.*

Proof: Let Ω be the set of partial acyclic edge colorings of G. For each edge e, let f_e be the subset (flaw) of Ω that contains the partial acyclic edge colorings of G in which e is uncolored. We will apply Theorem II.4 using the $\|\cdot\|_1$ norm and $M = \mathrm{diag}(\mu(\sigma))$, where μ is the uniform distribution over Ω.

We first compute the charges γ_e^S for each edge e and set of edges S. Notice that for γ_e^S to be non-zero, it should either be that $S = \emptyset$, or that S contains e and there exists a cycle $C = \{e_{i_1}, e_{i_2}\} \cup S$ so that, when a recoloring of e makes C bichromatic, the backtracking step uncolors precisely the edges in S. With that in mind, for each edge e and each set S that contains e, let $\mathcal{C}_e(S)$ denote the set of cycles with the latter property.

Lemma V.7. *For each edge e, let*

$$\gamma_e^S = \begin{cases} \frac{1}{Q} & \text{if } S = \emptyset \ ; \\ \frac{|\mathcal{C}_e(S)|}{Q} & \text{if } e \in S \ ; \\ 0 & \text{otherwise} \ . \end{cases}$$

Proof: Notice that

$$\begin{aligned} \gamma_e^S &= \max_{\tau \in \Omega} \sum_{\substack{\sigma \in f_e \\ S = U(\tau) \setminus (U(\sigma) \setminus \{e\})}} \rho_e(\sigma, \tau) \\ &\leq \max_{\tau \in \Omega} \sum_{\substack{\sigma \in f_e \\ S = U(\tau) \setminus (U(\sigma) \setminus \{e\})}} \frac{1}{Q} \ , \end{aligned}$$

since, according to Lemma V.5, $\rho_e(\sigma, \tau) \leq \frac{1}{Q}$ for each pair $(\sigma, \tau) \in f_e \times \Omega$. The proof follows by observing that, for each state τ:

- If $S = \emptyset$, then there exists at most one state σ such that $\rho_e(\sigma, \tau) > 0$ and $U(\tau) \setminus (U(\sigma) \setminus \{\mathrm{e}\}) = \emptyset$ (we can reconstruct σ from τ by uncoloring e).
- If $S \ni e$ and $|S| = 2\ell - 2$, then there exist at most $|\mathcal{C}_e(S)|$ states such that $\rho_e(\sigma, \tau) > 0$ and $S = U(\tau) \setminus (U(\sigma) \setminus \{\mathrm{e}\})$. Given a cycle $C = S \cup \{e_{i_1}, e_{i_2}\}$ we reconstruct σ from τ by finding the colors of edges in $S \setminus \{e\}$ from $\tau(e_{i_1}), \tau(e_{i_2})$, exploiting the facts that the backtracking step corresponds to an uncoloring of a bichromatic cycle; e is uncolored; and every other edge has the same color as in τ.
- For all other S, there exists no state σ such that $\rho(\sigma, \tau) > 0$ and $S = U(\tau) \setminus (U(\sigma) \setminus \{\mathrm{e}\})$.

∎

Observe that there are at most $(\Delta - 1)^{2\ell - 2}$ cycles of length 2ℓ containing a specific edge e. In other words, there exist at most $(\Delta - 1)^{2\ell - 3}$ sets of edges S of size $2\ell - 2$ that contain e and such that $\gamma_e^S > 0$ and, in addition, note that we always have $|\mathcal{C}_e(S)| \leq \Delta - 1$.

Thus, if $Q = c(\Delta - 1)$ for some constant c, setting $\psi_e = \alpha\gamma_\emptyset^e = \frac{\alpha}{Q}$, where $\alpha \in (1, c)$ is a constant, Lemma V.7 implies

$$
\begin{aligned}
&\frac{1}{\psi_e}\left(\sum_{S\subseteq E}\gamma_e^S\prod_{e\in S}\psi_j\right)\\
&\leq \min_{\alpha\in(1,c)}\left(\frac{1}{\alpha}+\sum_{i=3}^{\infty}\left(\frac{\Delta-1}{Q}\right)^{2i-2}\alpha^{2i-3}\right)\\
&\leq \min_{\alpha\in(1,c)}\left(\frac{1}{\alpha}+\frac{1}{c}\sum_{i=3}^{\infty}\left(\frac{\alpha}{c}\right)^{2i-3}\right)\\
&= \min_{\alpha\in(1,c)}\left(\frac{1}{\alpha}+\frac{\alpha^3}{c^2(c^2-\alpha^2)}\right)=\frac{2}{c}\ ,
\end{aligned}
$$

for $\alpha^* = c\left(\frac{\sqrt{5}-1}{2}\right)$. Thus, if $c > 2$ the probability that the algorithm fails to find an acyclic edge coloring within $\frac{T_0+s}{\delta}$ steps is 2^{-s}, where $\delta = 1-\frac{2}{c}$, and, according to Remark II.4,

$$T_0 = \log_2|\Omega| = O(|E|)\ .$$

The proof is concluded by observing that each step can be performed in time $O(|V|\Delta)$ (the time it takes to find a 2-colored cycle containing a given edge, if such a cycle exists, in a graph with a proper edge-coloring). ■

An application of the local cut lemma. In [13], Bernshteyn introduced a non-constructive generalized LLL condition, called the "Local Cut Lemma", with the aim of drawing connections between the LLL and the entropy compression method. He later applied it in [11] to the problem of Acyclic Edge Coloring giving improved bounds assuming further constraints on the graph besides sparsity. In particular, he proved the following.

Theorem V.8 ([11])**.** *Let G be a graph with maximum degree Δ and let H be a fixed bipartite graph. If G does not contain H as a subgraph, then there exists an acyclic edge coloring of G using at most $3(\Delta+o(1))$ colors.*

We now show how to use our framework to give a constructive proof of Theorem V.8. This will follow immediately from the following structural lemma in [11].

Lemma V.9 ([11])**.** *There exist positive constants γ,δ such that the following holds. Let G be a graph with maximum degree Δ that does not contain H as a subgraph. Then, for any edge $e\in E(G)$ and for any integer $k\geq 4$, the number of cycles of length k in G that contain e is at most $\gamma\Delta^{k-2-\delta}$.*

Constructive Proof of Theorem V.8: Notice that in this case, making almost identical calculations to those in the proof of Theorem V.6, invoking Lemma V.9 to upper bound the number of cycles that contain e, and setting $\alpha = \frac{c}{\beta}$, we obtain

$$
\begin{aligned}
\frac{1}{\psi_e}\left(\sum_{S\subseteq E}\gamma_e^S\prod_{h\in S}\psi_h\right) &\leq \min_{\alpha\in(1,c)}\left(\frac{1}{\alpha}+\frac{(\alpha)^3\gamma\Delta^{-\delta}}{c^2(c^2-\alpha^2)}\right)\\
&= \frac{1}{c}\min_{\beta>1}\left(\beta+\frac{\beta\gamma\Delta^{-\delta}}{\beta(\beta^2-1)}\right)\ .
\end{aligned}
$$

Thus, as Δ grows, the value of c required for the algorithm to terminate approaches 1, concluding the proof. ■

VI. Acknowledgements

We are grateful to Paris Syminelakis for various insightful remarks and comments. We also thank David Harris for helpful discussions.

References

[1] Dimitris Achlioptas and Amin Coja-Oghlan. Algorithmic barriers from phase transitions. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 793–802. IEEE Computer Society, 2008.

[2] Dimitris Achlioptas and Fotis Iliopoulos. Random walks that find perfect objects and the Lovász local lemma. *J. ACM*, 63(3):22:1–22:29, 2016.

[3] Dimitris Achlioptas, Fotis Iliopoulos, and Vladimir Kolmogorov. A local lemma for focused stochastic algorithms. To appear in *SIAM Journal on Computing*. Preprint at *arXiv:1805.02026*.

[4] Dimitris Achlioptas, Fotis Iliopoulos, and Alistair Sinclair. Beyond the Lovász local lemma: Point to set correlations and their algorithmic applications. 2019. Preprint at *arXiv:1805.02026*.

[5] Dimitris Achlioptas and Assaf Naor. The two possible values of the chromatic number of a random graph. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, pages 587–593. ACM, 2004.

[6] Noga Alon. A parallel algorithmic version of the local lemma. *Random Structures & Algorithms*, 2(4):367–378, 1991.

[7] Noga Alon, Michael Krivelevich, and Benny Sudakov. Coloring graphs with sparse neighborhoods. *Journal of Combinatorial Theory, Series B*, 77(1):73–82, 1999.

[8] Noga Alon and Joel H. Spencer. *The Probabilistic Method*. Wiley Publishing, 4th edition, 2016.

[9] Adrian Balint and Uwe Schöning. Choosing probability distributions for stochastic local search and the role of make versus break. In *Proceedings of the International Conference on Theory and Applications of Satisfiability Testing*, pages 16–29. Springer, 2012.

[10] Adrian Balint and Uwe Schöning. Engineering a lightweight and efficient local search SAT solver. In Lasse Kliemann and Peter Sanders, editors, *Algorithm Engineering: Selected Results and Surveys*, volume 9220 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2016.

[11] Anton Bernshteyn. New bounds for the acyclic chromatic index. *Discrete Mathematics*, 339(10):2543–2552, 2016.

[12] Anton Bernshteyn. The Johansson–Molloy theorem for DP-coloring. *arXiv preprint arXiv:1708.03843*, 2017.

[13] Anton Bernshteyn. The local cut lemma. *European Journal of Combinatorics*, 63:95–114, 2017.

[14] Rodrigo Bissacot and Luís Doin. Entropy compression method and legitimate colorings in projective planes. *arXiv preprint arXiv:1710.06981*, 2017.

[15] Karthekeyan Chandrasekaran, Navin Goyal, and Bernhard Haeupler. Deterministic algorithms for the Lovász local lemma. *SIAM J. Comput.*, 42(6):2132–2155, 2013.

[16] Thomas F Coleman and Jin Yi Cai. The cyclic coloring problem and estimation of spare Hessian matrices. *SIAM J. Algebraic Discrete Methods*, 7(2):221–235, 1986.

[17] Thomas F. Coleman and Jorge J. Moré. Estimation of sparse Hessian matrices and graph coloring problems. *Mathematical Programming*, 28(3):243–270, 1984.

[18] Vida Dujmović, Gwenaël Joret, Jakub Kozik, and David R Wood. Nonrepetitive colouring via entropy compression. *Combinatorica*, 36(6):661–686, 2016.

[19] Paul Erdős and László Lovász. Problems and results on 3-chromatic hypergraphs and some related questions. In *Infinite and finite sets (Colloq. Keszthely, 1973; dedicated to P. Erdős on his 60th birthday), Vol. II*, pages 609–627. Colloq. Math. Soc. János Bolyai, Vol. 10. North-Holland, Amsterdam, 1975.

[20] Louis Esperet and Aline Parreau. Acyclic edge-coloring using entropy compression. *European Journal of Combinatorics*, 34(6):1019–1027, 2013.

[21] Heidi Gebauer, Tibor Szabó, and Gábor Tardos. The local lemma is tight for SAT. In *Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 664–674. Society for Industrial and Applied Mathematics, 2011.

[22] Ioannis Giotis, Lefteris M. Kirousis, Kostas I. Psaromiligkos, and Dimitrios M. Thilikos. Acyclic edge coloring through the Lovász local lemma. *Theoretical Computer Science*, 665:40–50, 2017.

[23] Adam Gkagol, Gwenaël Joret, Jakub Kozik, and Piotr Micek. Pathwidth and nonrepetitive list coloring. *arXiv preprint arXiv:1601.01886*, 2016.

[24] Daniel Gonçalves, Mickaël Montassier, and Alexandre Pinlou. Entropy compression method applied to graph colorings. *arXiv preprint arXiv:1406.4380*, 2014.

[25] Jarosław Grytczuk, Jakub Kozik, and Piotr Micek. New approach to nonrepetitive sequences. *Random Structures & Algorithms*, 42(2):214–225, 2013.

[26] Bernhard Haeupler, Barna Saha, and Aravind Srinivasan. New constructive aspects of the Lovász local lemma. *J. ACM*, 58(6):28:1–28:28, 2011.

[27] David G. Harris and Aravind Srinivasan. A constructive algorithm for the Lovász local lemma on permutations. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 907–925. Society for Industrial and Applied Mathematics, 2014.

[28] Nicholas J. A. Harvey and Jan Vondrák. An algorithmic proof of the Lovász local lemma via resampling oracles. In *Proceedings of the 56th Annual IEEE Symposium on Foundations of Computer Science*, pages 1327–1346. IEEE Computer Society, 2015.

[29] Fotis Iliopoulos. Commutative algorithms approximate the LLL-distribution. In *Proceedings of APPROX/RANDOM*, pages 44:1–44:20. Schloß Dagstuhl–Leibniz-Zentrum für Informatik, 2018.

[30] Fotis Iliopoulos and Alistair Sinclair. Efficiently list-edge coloring multigraphs asymptotically optimally. To appear in *Proceedings of the 31st Annual ACM-SIAM Symposium on Discrete Algorithms,* 2020. Preprint at *arXiv:1812:10309*.

[31] A. Johansson. Asympotic choice number for triangle free graphs. *Unpublished manuscript*, 1996.

[32] Kashyap Kolipaka, Mario Szegedy, and Yixin Xu. A sharper local lemma with improved applications. In *Proceedings of APPROX/RANDOM*, pages 603–614. Springer Berlin Heidelberg, 2012.

[33] Kashyap Babu Rao Kolipaka and Mario Szegedy. Moser and Tardos meet Lovász. In *Proceedings of the 43rd Annual ACM Symposium on the Theory of Computing*, pages 235–244. ACM, 2011.

[34] Vladimir Kolmogorov. Commutativity in the algorithmic Lovász local lemma. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science*, pages 780–787. IEEE Computer Society, 2016.

[35] Michael Molloy. The list chromatic number of graphs with small clique number. *Journal of Combinatorial Theory, Series B*, 134:264–284, 2019.

[36] Michael Molloy and Bruce Reed. Further algorithmic aspects of the local lemma. In *Proceedings of the 30th Annual ACM Symposium on the Theory of Computing*, pages 524–529. ACM, 1998.

[37] Michael Molloy and Bruce Reed. *Graph Colouring and the Probabilistic Method*. Springer-Verlag, Berlin, 2002.

[38] Robin A. Moser. A constructive proof of the Lovász local lemma. In *Proceedings of the 41st Annual ACM Symposium on the Theory of Computing*, pages 343–350. ACM, 2009.

[39] Robin A. Moser and Gábor Tardos. A constructive proof of the general Lovász local lemma. *J. ACM*, 57(2):Art. 11, 15, 2010.

[40] Sokol Ndreca, Aldo Procacci, and Benedetto Scoppola. Improved bounds on coloring of graphs. *Eur. J. Comb.*, 33(4):592–609, 2012.

[41] Pascal Ochem and Alexandre Pinlou. Application of entropy compression in pattern avoidance. *arXiv preprint arXiv:1301.1873*, 2013.

[42] Wesley Pegden. An extension of the Moser-Tardos algorithmic local lemma. *SIAM Journal on Discrete Mathematics*, 28(2):911–917, 2014.

[43] Bart Selman, Henry A. Kautz, and Bram Cohen. Local search strategies for satisfiability testing. In *Proceedings of DIMACS Workshop on Cliques, Coloring and Satisfiability*, pages 521–532. DIMACS/AMS, 1993.

[44] Bart Selman, Henry A. Kautz, and Bram Cohen. Noise strategies for improving local search. In *Proceedings of the 12th National Conference on Artificial Intelligence, Volume 1*, pages 337–343. AAAI Press/MIT Press, 1994.

[45] Van H Vu. A general upper bound on the list chromatic number of locally sparse graphs. *Combinatorics, Probability and Computing*, 11(1):103–111, 2002.

APPENDIX

A. Matrices and norms

Let $\|\cdot\|$ be any norm over vectors in $\mathbb{R}^n$. The *dual* norm, also over vectors in $\mathbb{R}^n$, is defined as

$$\|z\|_* = \sup_{\|x\|=1} |z^\top x| \ .$$

For example, the dual norm of $\|\cdot\|_\infty$ is $\|\cdot\|_1$. It can be seen that $\|\cdot\|_{**} = \|\cdot\|$ and that for any vectors x, z,

$$z^\top x = \|x\| \left(\frac{z^\top x}{\|x\|} \right) \le \|z\|_* \|x\| \ . \tag{29}$$

The corresponding *operator norm*, over $n \times n$ real matrices, is defined as

$$\|A\| \equiv \sup_{\|x\|=1} \|Ax\| \ .$$

For example, if A is a matrix with non-negative entries then $\|A\|_\infty$ and $\|A\|_1$ can be seen to be the maximum *row* and *column* sum of A, respectively. Operator norms are submultiplicative, i.e., for every operator norm $\|\cdot\|$ and any two $n \times n$ matrices A, B,

$$\|AB\| \le \|A\| \|B\| \ . \tag{30}$$

Finally, for any vector norm $\|\cdot\|$, any row vector $x^\top$ and $n \times n$ matrix A, we have

$$\|x^\top A\|_* \le \|x^\top\|_* \|A\| \ . \tag{31}$$

B. The matrix norms framework in action

In this appendix we illustrate how the framework of matrix norms captures a variety of convergence arguments for local search algorithms, both LLL-inspired ones and others. Following the same notation as in Section I-B of the introduction, we let P denote the stochastic transition matrix of the search algorithm on Ω, and A its restriction to the flawed states Ω^*. Recall from our discussion in the introduction that our goal is to bound $\rho(A)$, the spectral radius of A, using an operator norm as a surrogate.

We begin with the classical potential function argument. Consider any function ϕ on Ω such that $\phi(\sigma) > 0$ for $\sigma \in \Omega^*$, while $\phi(\sigma) = 0$ for $\sigma \notin \Omega^*$. The potential argument asserts that eventually $\phi = 0$ (i.e., the particle escapes Ω^*) if ϕ is always reduced in expectation, i.e., if for every $\sigma \in \Omega^*$,

$$\sum_{\tau \in \Omega} P[\sigma, \tau] \phi(\tau) < \phi(\sigma) \ . \tag{32}$$

To express this argument via matrix norms, let $A' = MAM^{-1}$ where M is the diagonal $|\Omega^*| \times |\Omega^*|$ matrix $\mathrm{diag}(1/\phi(\sigma))$. Thus, $A'[\sigma, \tau] = A[\sigma, \tau]\phi(\tau)/\phi(\sigma)$. Recalling that $\|\cdot\|_\infty$ is the maximum row sum of a matrix, we see that condition (32) for the potential function is nothing other than $\|A'\|_\infty < 1$, implying $\rho(A) = \rho(A') \le \|A'\|_\infty < 1$.

Next we show how the same approach, using the dual matrix norm $\|\cdot\|_1$, captures the Moser-Tardos algorithm for k-SAT. Given a k-SAT formula on n variables with clauses c_i, let $\Omega = \{0,1\}^n$ denote the set of all assignments, and Ω^* the set of non-satisfying assignments. To simplify exposition, we assume that in each step, the algorithm picks the lowest-indexed unsatisfied clause c_i and resamples all variables in c_i. Thus, the state-evolution is a Markov chain, and if we denote its transition matrix by P, we are interested in the submatrix A that is the projection of P onto Ω^*. For each clause c_i, let A_i be the $|\Omega^*| \times |\Omega^*|$ submatrix of A comprising all rows (states) where the resampled clause is c_i. (All other rows of A_i are 0.) For $t \ge 1$, let $\mathcal{W}_t$ contain every t-sequence of (indices of) clauses that has non-zero probability of being the first t clauses resampled by the algorithm. In other words, $\mathcal{W}_t$ is the set of all t-sequences of indices from $[m]$ corresponding to non-vanishing t-products of matrices from $\{A_1, \dots, A_m\}$, i.e., $\mathcal{W}_t = \{W = (w_i) \in [m]^t : \prod_{i=1}^t A_{w_i} \ne 0\}$. For every operator norm $\|\cdot\|$ we get:

$$\begin{aligned}\rho(A)^t = \rho(A^t) \le \|A^t\| &= \left\| \left(\sum_{i \in [m]} A_i \right)^t \right\| \\ &= \left\| \sum_{W \in \mathcal{W}_t} \prod_{i=1}^t A_{w_i} \right\| \le \sum_{W \in \mathcal{W}_t} \left\| \prod_{i=1}^t A_{w_i} \right\| \\ &\le \sum_{W \in \mathcal{W}_t} \prod_{i=1}^t \|A_{w_i}\| . \end{aligned} \tag{33}$$

The first inequality here follows from the fact that $\rho(A) \le \|A\|$ for any operator norm $\|\cdot\|$, the second inequality is the triangle inequality, and the third follows by submultiplicativity of operator norms.

To get a favorable bound, we will apply (33) with the norm $\|\cdot\|_1$, i.e., the maximum column sum. We see that for all $j \in [m]$, every column of A_j has at most one non-zero entry, since $A_j(\sigma, \tau) > 0$ only if σ is the (unique) mutation of τ so that c_j is violated. Recalling that all non-zero entries of A equal 2^{-k}, we conclude that $\|A_j\|_1 = 2^{-k}$ for all $j \in [m]$. Therefore, $\|A^t\|_1 \le |\mathcal{W}_t| 2^{-kt}$. To bound $|\mathcal{W}_t|$ we use a simple necessary condition for membership in $\mathcal{W}_t$ which, by a standard counting argument, implies that if each clause shares variables with at most Δ other clauses then $|\mathcal{W}_t| \le 2^m (\mathrm{e}\Delta)^t$. Therefore $\rho(A)^t \le 2^m (\mathrm{e}\Delta 2^{-k})^t$, implying that if $\Delta < 2^k/\mathrm{e}$ then $1 > \|A\|_1 \ge \rho(A)$ and the algorithm terminates within $O(m)$ steps with high probability. This matches exactly the Moser-Tardos condition (which is tight).

A very similar argument can be used to capture even the most general existing versions of the algorithmic LLL [2, 28, 3], which are described by arbitrary flaws and, for each flaw f_i, an arbitrary corresponding transition matrix A_i for addressing the flaw. Note that (33) is in essence a weighted counting of witness sequences, the weight of each sequence being the product of the norms $\|A_{w_i}\|_1$. Observe also that in our k-SAT example above, the only probabilistic notion was the transition matrix A and we did not make any implicit or explicit reference to a probability measure μ. To cast general algorithmic LLL arguments in this same form, any measure μ is incorporated as a *change of basis* for the transition matrix A, i.e., we bound $\|A'\|_1 = \|MAM^{-1}\|_1$ as $\sum_{W \in \mathcal{W}_t} \prod_{i=1}^t \|MA_{w_i}M^{-1}\|_1$, where M is the diagonal $|\Omega^*| \times |\Omega^*|$ matrix $\mathrm{diag}(\mu(\sigma))$, similarly to the potential function argument. We thus see that the measure μ is nothing other than a tool for analyzing the progress of the algorithm.

C. Proof of Lemma IV.4

Our computations are similar to the ones in [35]. The following version of Chernoff Bounds will be useful:

Lemma A.1. *Suppose $\{X_i\}_{i=1}^m \in \{0,1\}$ are boolean variables, and set $Y_i = 1 - X_i$, $X = \sum_{i=1}^m X_i$. If $\{Y_i\}_{i=1}^m$ are negatively correlated, then for any $0 < t \le \mathbb{E}[X]$*

$$\Pr[|X - \mathbb{E}[X]| > t] < 2 \exp\left(-\frac{t^2}{3\mathbb{E}[X]} \right) .$$

Proof of part (a) of Lemma IV.4: Let $v \in V$ and $\sigma \in \Omega$ be arbitrary and let $\tau \in \Omega$ be the (random) state output by $\textsc{Recolor}(v, \sigma)$. For each color $c \in \mathcal{L}_v$, let $P_v^c = \{u \in N_v : c \in R_u^v(\sigma)\}$ and define

$$\rho(c) = \sum_{u \in P_v^c} \frac{1}{|R_u^v(\sigma)| - 1} .$$

Since $c \in R_u^v(\sigma)$ implies $|R_u^v(\sigma)| \ge 2$, and since $1 - 1/x > \exp(-1/(x-1))$ for $x \ge 2$, we see that

$$\begin{aligned}\mathbb{E}[|L_v(\tau)|] &= 1 + \sum_{c \in \mathcal{L}_v} \prod_{u \in P_v^c} \left(1 - \frac{1}{|R_u^v(\sigma)|} \right) \\ &> \sum_{c \in \mathcal{L}_v} \prod_{u \in P_v^c} \exp\left(-\frac{1}{|R_u^v(\sigma)| - 1} \right) \\ &= \sum_{c \in \mathcal{L}_v} \mathrm{e}^{-\rho(c)} . \end{aligned} \tag{34}$$

Also, since each $R_u^v(\sigma)$ has $|R_u^v(\sigma)| - 1$ non-Blank colors, we see that

$$\begin{aligned} Z_v &:= \sum_{c \in \mathcal{L}_v} \rho(c) \\ &\le \sum_{u \in N_v} \sum_{c \in R_u^v(\sigma) \setminus \text{Blank}} \frac{1}{|R_u^v(\sigma)| - 1} \le \Delta . \end{aligned} \tag{35}$$

The fact that e^{-x} is convex implies that the right hand side of (34) is at least $|\mathcal{L}_v|\exp(-Z_v/|\mathcal{L}_v|)$. Recalling that $|\mathcal{L}_v| = q = (1+\epsilon)\frac{\Delta}{\ln\sqrt{f}}$, and combining (34) with (35), yields

$$\begin{aligned}\mathbb{E}[|L_v(\tau)|] > q\mathrm{e}^{-Z_v/q} &\geq (1+\epsilon)\frac{\Delta}{\ln\sqrt{f}}\mathrm{e}^{-\Delta/q} \\ &= 2(1+\epsilon)\frac{\Delta}{\ln f}f^{-\frac{1}{2(1+\epsilon)}} = 2L \ . \qquad (36)\end{aligned}$$

Let X_c be the indicator variable that $c \in L_v(\tau)$, so that $|L_v(\tau)| = 1 + \sum_{c\in\mathcal{L}_v(\tau)} X_c$. It is not hard to see that the variables $Y_c = 1 - X_c$ are negatively correlated, so that applying Lemma A.1 with $t = \frac{1}{2}\mathbb{E}[|L_v(\tau)|] > L$ yields

$$\begin{aligned}&\Pr\left[|L_v(\tau)| < \tfrac{1}{2}\mathbb{E}\left[|L_v(\tau)|\right]\right] \\ &\qquad\qquad \leq 2\mathrm{e}^{-\mathbb{E}[|L_v(\tau)|]/12} < 2\mathrm{e}^{-L/6} \ .\end{aligned}$$

This concludes the proof. ∎

Proof of part (b) of Lemma IV.4: Let $\Psi = \{c \in L_v(\sigma) : \rho(c) \geq L/20\} \setminus \mathrm{Blank}$. The probability that $L_v(\tau)$ contains at least one color from Ψ is at most

$$\begin{aligned}\mathbb{E}\left[|L_v(\tau)\cap\Psi|\right] &= \sum_{c\in\Psi}\prod_{u\in P_v^c}\left(1 - \frac{1}{|R_u^v(\sigma)|}\right) \\ &< \sum_{c\in\Psi}\prod_{u\in P_v^c}\exp\left(-\frac{1}{2(|R_u^v(\sigma)|-1)}\right) < \sum_{c\in\Psi}\mathrm{e}^{-\rho(c)/2} \ ,\end{aligned}$$

where we used the fact that $c \in R_u^v(\sigma)$ implies $|R_u^v(\sigma)| \geq 2$, and that $1 - 1/x < \exp(-1/(2(x-1)))$ for $x \geq 2$. Finally note that $\sum_{c\in\Psi}\mathrm{e}^{-\rho(c)/2} \leq q\mathrm{e}^{-L/40}$ by the definition of the set Ψ.

Recall that $T_{v,c}(\tau) = \{u \in N_v : \tau(u) = \mathrm{Blank} \text{ and } c \in L_u(\tau)\}$. Since $L_u(\tau) \subseteq R_u(\tau) = R_u(\sigma)$, it follows that $T_{v,c}(\tau) \subseteq P_v^c$ and, therefore, $\mathbb{E}\left[|T_{v,c}(\tau)|\right] \leq \sum_{u\in P_v^c} 1/|R_u^v(\sigma)| \leq \rho(c)$. Since the vertices in P_v^c are colored (and thus become Blank) independently and since $\rho(c) < L/20$ for $c \notin \Psi$, applying Lemma A.1 with $t = L/20$ yields $\Pr\left[|T_{v,c}(\tau)| > \mathbb{E}\left[|T_{v,c}(\tau)|\right] + L/20\right] < 2\mathrm{e}^{-L/60}$. Applying the union bound over all q colors, we see that the probability there is at least one $c \notin \Psi$ for which $|T_{v,c}(\tau)| > L/10$ is at most $2q\mathrm{e}^{-L/60}$. Thus, with probability at least $1 - 3q\mathrm{e}^{-L/60}$,

$$\begin{aligned}\sum_{c\in L_v(\tau)\setminus\mathrm{Blank}}|T_{v,c}(\tau)| &= \sum_{c\in L_v(\tau)\setminus(\Psi\cup\mathrm{Blank})}|T_{v,c}(\tau)| \\ &< \frac{L}{10}|L_v(\tau)| \ .\end{aligned}$$

∎

D. Proof of Proposition II.1

We use the term "with high probability" to refer to probabilities that tend to 1 as n goes to infinity. Corollary II.1 follows in a straightforward way from the following lemma.

Lemma A.2. *For any $\delta \in (0,1)$ there exists a constant d_0 such that, for any $d \in \left(d_0\ln n, (n\ln n)^{\frac{1}{3}}\right)$, each vertex of the random graph $G = G(n, d/n)$ is contained in at most Δ^δ triangles with high probability, where Δ is the maximum degree of G.*

Proof of Corollary II.1: According to [5], for a graph $G \in G(n, d/n)$ we know that with high probability

$$\chi(G) = \frac{1}{2}\frac{d}{\ln d}(1+o(1)) \ . \qquad (37)$$

Fix $\zeta \in (0,1)$ and $\delta \in (0, \frac{2\zeta}{1+2\zeta})$. According to Lemma A.2, there exists a constant d_0 such that for any $d \in \left(d_0\ln n, (n\ln n)^{\frac{1}{3}}\right)$ each vertex of $G = G(n, d/n)$ is contained in at most Δ^δ triangles with probability that tends to 1 as n goes to infinity. Thus, we can apply Theorem II.5 with parameter $\zeta > 0$ since

$$f = \frac{\Delta^2}{\Delta^\delta} > \Delta^{2-\frac{2\zeta}{1+2\zeta}}(\ln\Delta)^2,$$

for large enough Δ. This yields an upper bound q on the chromatic number of G that is at most

$$\begin{aligned} q &= (1+\zeta)\frac{\Delta}{\ln\sqrt{f}} \\ &\leq (1+\zeta)\frac{\Delta}{\frac{1+\zeta}{1+2\zeta}\ln\Delta + \ln\ln\Delta} \\ &\leq (1+2\zeta)\frac{\Delta}{\ln\Delta} \ . \qquad (38)\end{aligned}$$

Moreover, since the expected degree of every vertex of G is d, and its distribution is binomial with parameter $\frac{d}{n}$, standard Chernoff bounds and the union bound imply that, for any $\eta \in (0,1)$, $\Delta \leq (1+\eta)d$ with high probability, for large enough d_0.

Combining the latter fact with (37) and (38), we deduce that we can find an arbitrarily small constant $\eta' \in (0,1)$ such that

$$q \leq (2+\eta')\chi(G)$$

by choosing ζ and η sufficiently small. Picking $\eta' = \frac{4\epsilon}{1-2\epsilon}$ we obtain $\chi(G) \geq \frac{q}{2+\eta'} \geq q(\frac{1}{2} - \epsilon)$, concluding the proof of Proposition II.1. ∎

Finally, we go back and prove Lemma A.2.

Proof of Lemma A.2: Let Δ_v be the random variable that equals the degree of vertex v of G. Observe that $\Delta_v \sim \mathrm{Binom}(n-1, \frac{d}{n})$ and, therefore, using a standard Chernoff bound and the fact that $d \geq d_0\log n$ we get that

$$\Pr\left[\Delta_v \notin (1 \pm \frac{1}{10})d\right] \leq \frac{1}{n^2} \ ,$$

for large enough d_0. Thus, by a union bound we get that $\Pr[\Delta \in (1 \pm \frac{1}{10})d] \leq \frac{1}{n}$.

Let T_v be the number of triangles that contain vertex v and B be the event that $\Delta \notin (1 \pm \frac{1}{10})d$. Then,

$$\begin{aligned}\Pr[T_v > \Delta^\delta] &\leq \Pr[T_v > \Delta^\delta \mid \overline{B}] + \Pr[B] \\ &\leq \frac{\Pr[\left(T_v > \Delta^\delta\right) \cap \overline{B}]}{1 - \frac{1}{n}} + \frac{1}{n} \quad .\end{aligned}$$

Observe that $T_v \sim \operatorname{Binom}\left(\binom{n-1}{2}, \left(\frac{d}{n}\right)^3\right)$ and $\mathbb{E}[T_v] \leq \frac{d^3}{2n}$. Thus, for any fixed value of $\Delta \in (1 \pm \frac{1}{10})d$, setting $1 + \beta = \frac{\Delta^\delta}{d^3/2n}$ and using a standard Chernoff bound we obtain

$$\Pr[T_v > \Delta^\delta] \leq \mathrm{e}^{-\frac{\beta^2 d^3/2n}{3}} \leq \frac{1}{n^2},$$

since

$$\begin{aligned}\beta &\geq \frac{\left((1 - \frac{1}{10})d\right)^\delta - d^3/2n}{d^3/2n} > 0 \quad , \\ \frac{1}{3}\beta^2 \frac{d^3}{2n} &\geq \frac{1}{3} \frac{\left(\left((1 - \frac{1}{10})d\right)^\delta - d^3/2n\right)^2}{d^3/2n} \geq 2 \ln n\end{aligned}$$

whenever $d \in [d_0 \ln n, (n \ln n)^{\frac{1}{3}}]$ and for large enough n and d_0. Taking a union bound over v concludes the proof of the lemma. ■