THEORY OF COMPUTING, Volume 20 (6), 2024, pp. 1–23
www.theoryofcomputing.org

On a Generalization of Iterated and Randomized Rounding

Nikhil Bansal*

Received April 24, 2019; Revised November 18, 2022; Published December 5, 2024

Abstract. We give a general method for rounding linear programs, that combines the commonly used iterated rounding and randomized rounding techniques. In particular, we show that whenever iterated rounding can be applied to a problem with some slack, there is a randomized procedure that returns an integral solution that satisfies the guarantees of iterated rounding and also has concentration properties. We use this to give new results for several classical problems such as rounding column-sparse LPs, makespan minimization on unrelated machines and degree-bounded spanning trees.

1 Introduction

A powerful approach in approximation algorithms is to formulate the problem at hand as a 0-1 integer program and consider some efficiently solvable relaxation for it. Then, given some fractional solution $x \in [0,1]^n$ to this relaxation, apply a suitable *rounding* procedure to x to

An extended abstract of this paper appeared in the Proceedings of the 51st Annual ACM Symposium on Theory of Computing (STOC'19) [7].

*Supported by the ERC Consolidator Grant 617951 and the NWO VICI grant 639.023.812.

ACM Classification: F.2.2, G.1.6

AMS Classification: 68W25, 68W20

Key words and phrases: approximation, randomized rounding, scheduling, discrepancy, semidefinite programming, random walks

 DOI: 10.4086/toc.2024.v020a006

obtain an integral 0-1 solution. Arguably, the two most basic and extensively studied techniques for rounding such relaxations are randomized rounding and iterated rounding.

Randomized rounding. Here, the fractional values $x_i \in [0, 1]$ are interpreted as probabilities, and used to round the variables independently to 0 or 1. A key property of this rounding is that each linear constraint is preserved in expectation, and its value is tightly concentrated around its mean as given by Chernoff bounds, or more generally Bernstein's inequality. Randomized rounding is well-suited to problems where the constraints do not have much structure, or when they are *soft* and some error can be tolerated. Sometimes these errors can be fixed by applying problem-specific alteration steps. We refer to [36, 35] for various applications of randomized rounding.

Iterated rounding. This technique is quite different from randomized rounding and is useful for problems that may have some *hard* combinatorial constraints that must be maintained, e. g., if the final solution must be a spanning tree or a matching. It is also useful for problems where the constraints may have some other interesting structural property such as column-sparsity that we may wish to exploit.

The rounding is based on linear algebra and it proceeds in several iterations $k = 1, 2, \ldots$, until all variables are eventually rounded to 0 or 1. More specifically, we start with $x^{(0)} = x$ initially, and let $x^{(k-1)} \in \mathbb{R}^n$ be the solution at the beginning of iteration k and n_k denote the number of fractional variables in $x^{(k-1)}$ (i. e., those strictly between 0 and 1). Then one cleverly chooses some collection of linear constraints on these n_k fractional variables, say specified by rows of the matrix $W^{(k)}$ with $\mathrm{rank}(W^{(k)}) \leq n_k - 1$, and updates the solution as $x^{(k)} = x^{(k-1)} + y^{(k)}$ by some non-zero vector $y^{(k)}$ satisfying $W^{(k)} y^{(k)} = \mathbf{0}$, so that some fractional variable reaches 0 or 1. Such a $y^{(k)}$ exists as the null space of $W^{(k)}$ has dimension $n_k - \mathrm{rank}(W^{(k)}) \geq 1$. Notice that once a variable reaches 0 or 1 it stays fixed.

Despite its simplicity, this method is extremely powerful and most fundamental results in combinatorial optimization such as the integrality of matroid, matroid-intersection and non-bipartite matching polytopes follow very cleanly using this approach. Similarly, several breakthrough results for problems such as degree-bounded spanning trees, survivable network design and rounding for column-sparse LPs were obtained by this method. An excellent reference is [23].

1.1 Need for combining the approaches

In many problem settings, however, one needs a rounding that combines the features of both randomized and iterated rounding [15, 16, 19, 3]. We give several examples in Section 1.3, but a typical scenario is where the problem involves finding an object with specific combinatorial constraints that cannot be violated, e. g., a spanning tree to connect nodes in a network, or a one-sided matching (assignment) of jobs to machines; and additionally a list of other soft side-constraints, e. g., a bound on the maximum degree of the spanning tree to prevent any particular node from being overloaded, or perhaps edges are of several types and we wish to purchase a certain minimum number of each type due to fairness considerations, or there may be multiple budget constraints for various subsets of edges.

As the soft constraints are typically arbitrary and lack structure, essentially the best one can hope for is to satisfy them fractionally and then apply randomized rounding. On the other hand, randomized rounding can be quite bad at satisfying the hard combinatorial constraints, and iterated rounding is the right approach to handle them. So given a problem with both hard and soft constraints, either technique by itself does not suffice and one would like a rounding that *simultaneously* does as well as iterated rounding on the hard constraints and as well as randomized rounding on the soft constraints.

Dependent rounding. Motivated by such problems, there has been extensive work on developing *dependent rounding techniques*. Roughly speaking, these techniques round the fractional solution in some random but correlated way to satisfy the hard constraints and also ensure some concentration properties for the soft constraints. Some examples of such methods include swap rounding [15, 16], randomized pipage rounding [2, 33, 19, 20], maximum-entropy sampling [3, 32, 4], rounding via discrepancy [26, 29, 11] and Gaussian random walks [28].

A key idea here is that the weaker property of *negative* dependence (instead of independence) also suffices to get concentration. There is a rich and deep theory of negative dependence and various notions such as negative correlation, negative cylinder dependence, negative association, strongly Rayleigh property and determinantal measures, that imply interesting concentration properties [27, 13, 17]. This insight has been extremely useful and for many general problems such as those involving assignment or matroid polytopes, one can exploit the underlying combinatorial structure to design rounding approaches that ensure negative dependence between all or some suitable collection of random variables.

Limitations. Even though these dependent rounding methods are powerful and ingenious, they are also limited by the fact that requiring negative dependence substantially restricts the kinds of rounding steps that can be designed, and the type of problems that they can be applied to. Moreover, even when such a rounding is possible, it typically requires a lot of creativity and careful understanding of the problem structure to come up with the rounding for the problem at hand.

1.2 Our results

Our main result is a new and general dependent rounding approach that we call *sub-isotropic rounding*. In particular, it combines iterated and randomized rounding in a completely generic way and significantly extends the scope of previous dependent rounding techniques. Before describing our result, we need some definitions.

Let $X_1, \ldots, X_n$ be independent 0-1 random variables with means $x_i = \mathbb{E}[X_i]$ and $a_1, \ldots, a_n$ be arbitrary reals (possibly negative). It is well known [14] that the sum $S = \sum_i a_i X_i$ satisfies the following tail bound for any $t \geq 0$:

$$\text{(Bernstein's inequality)} \qquad \Pr[S - \mathbb{E}[S] \geq t] \leq \exp\left(-\frac{t^2}{2\sum_i a_i^2(x_i - x_i^2) + 2Mt/3}\right) \tag{1.1}$$

where $M = \max_i |a_i|$. The lower tail follows by applying the above to $-a_i$, and the standard Chernoff bounds correspond to (1.1) when $a_i \in [0,1]$ for $i \in [n]$.

The following relaxation of Bernstein's inequality will be highly relevant for us.

Definition 1.1 (β-concentration)**.** Let $\beta \geq 1$. For a vector-valued random variable $X = (X_1, \ldots, X_n)$ where X_i are possibly dependent 0-1 random variables, we say that X is β-concentrated around its mean $x = (x_1, \ldots, x_n)$ where $x_i = \mathbb{E}[X_i]$, if for every $a \in \mathbb{R}^n$, the real-valued random variable $\langle a, X\rangle$ satisfies Bernstein's inequality up to a factor of β in the exponent, i. e.,

$$\Pr[\langle a, X\rangle - \mathbb{E}[\langle a, x\rangle] \geq t] \leq \exp\left(-\frac{t^2/\beta}{2\sum_i a_i^2(x_i - x_i^2) + 2Mt/3}\right) \tag{1.2}$$

where $M = \max_i |a_i|$.

Main result. We show that whenever iterated rounding can be applied to a problem such that in iteration k, there is some *slack* in the sense that $\mathrm{rank}(W^{(k)}) \leq (1-\delta)n_k$ for some fixed $\delta > 0$, then $O(1/\delta)$-concentration can be achieved for free. More precisely, we have the following result.

Theorem 1.2. *For any fixed $\delta \in (0,1)$, let us formalize an iterated rounding algorithm as follows. Given a starting solution x, initialize $x^{(0)} = x$. In each step k, for $k \geq 1$, the algorithm selects a matrix $W^{(k)}$ with* $\mathrm{rank}(W^{(k)}) \leq (1-\delta)n_k$*. Now it can pick* any *$y^{(k)}$ satisfying $W^{(k)}y^{(k)} = 0$ and set $x^{(k)} = x^{(k-1)} + y^{(k)}$, and iterate until $x^{(\mathit{final})}$ is in $\{0,1\}^n$. Let $V \subset \{0,1\}^n$ be the set of outcomes that can be reached by the iterated rounding algorithm.*

Then the sub-isotropic rounding algorithm outputs a random vector X satisfying

1. *$X \in V$ with probability 1, and*

2. *$\mathbb{E}[X] = x$ and X is β-concentrated around x with $\beta = 20/\delta$.*

Remark 1.3. A simple example shows that the dependence $\beta = \Omega(1/\delta)$ in Theorem 1.2 cannot be improved. Let $\delta = 1/t$ for some integer t, and consider n variables $x_1, \ldots, x_n$, partitioned into n/t blocks $B_1, \ldots, B_{n/t}$ where block $B_i = \{x_{(i-1)t+1}, \ldots, x_{it}\}$. For each B_i there are $t-1$ constraints $x_{(i-1)t+1} = x_{(i-1)t+2} = \cdots = x_{it}$, and hence there are $(t-1)(n/t) = n(1-\delta)$ constraints in total. Consider the starting solution with all $x_j = 1/2$. Now, no matter what random choices the algorithm makes, the variables within a block evolve identically and all reach the same value 0 or 1. So the linear function $S = x_1 + \cdots + x_n$ will only be $1/\delta$-concentrated.

The generality of Theorem 1.2 directly gives new results for several problems where iterated rounding gives useful guarantees. All one needs to show is that the original iterated rounding argument for the problem can be applied with some slack, which is often straightforward and only worsens the approximation guarantee slightly. In particular, note that Theorem 1.2 makes no assumption about the combinatorial structure of the problem and by working with the more relaxed notion of β-concentration, we can completely avoid the need for negative dependence.

1.3 Motivating problems and applications

We now describe several applications of our result and also briefly discuss why they seem beyond the reach of current dependent rounding methods.

1.3.1 Rounding for column-sparse LPs

Let $x \in [0,1]^n$ be some fractional solution satisfying $Ax = b$, where $A \in \mathbb{R}^{m\times n}$ is an $m \times n$ matrix. The celebrated Beck–Fiala algorithm [12] (see also [21] for a related result) uses iterated rounding to produce an integral solution X satisfying $\|A(X-x)\|_\infty < t$, where t is the maximum ℓ_1 norm of the columns of A. This is substantially better than randomized rounding for small t, where the error for any row is typically its ℓ_2 norm which can be substantially larger than t.

Many problems, however, involve both some column-sparse constraints that come from the underlying combinatorial problem, and some general arbitrary constraints which might not have much structure. This motivates the following natural question.

Question 1.4. Let M be a linear system with two sets of constraints given by matrices A and B, where A is column-sparse, while B is arbitrary. Given some fractional solution x, can we round it to get error $O(t)$ for the rows of A, while doing no worse than randomized rounding for the constraints in B?

Remark 1.5. Note that simply applying iterated rounding on the rows of A gives no control on the error for B. Similarly, just doing randomized rounding will not give $O(t)$ error for A. Also as A and B are arbitrary, negative dependence based techniques do not seem to apply.

We show that a direct modification of the Beck–Fiala argument gives slack δ, for any $\delta \in [0,1)$, while worsening the error bound slightly to $t/(1-\delta)$. Setting, say $\delta = 1/2$ and applying Theorem 1.2 gives $X \in \{-1,1\}^n$ that (i) has error at most $2t$ for rows of A, (ii) satisfies $\mathbb{E}[X_i] = x_i$ and is $O(1)$-concentrated, thus giving similar guarantees as randomized rounding for the rows of B. In fact, the solution produced by the algorithm will satisfy concentration for all linear constraints and not just for the rows of B.

We also consider an extension to the Komlós setting, where the error depends on the maximum ℓ_2 norm of columns of A. These results are described in Section 4.1.

1.3.2 Makespan minimization on unrelated machines

The classical problem of makespan minimization on unrelated machines is the following. Given n jobs and m machines, where each job $j \in [n]$ has arbitrary size p_{ij} on machine $i \in [m]$, assign the jobs to machines to minimize the maximum machine load. In a celebrated result, [24] gave a rounding method with additive error at most $p_{\max} := \max_{ij} p_{ij}$, i. e., it gives an assignment with makespan at most $\mathrm{Opt} + p_{\max}$ where Opt is the value of an optimum LP solution. In many practical problems, however, there are other soft resource constraints and side constraints that are added to the fractional formulation. So it is useful to find a rounding that satisfies these approximately but increases the makespan by only $O(p_{\max})$. This motivates the following natural problem.

Question 1.6. Given a fractional assignment x, find an integral assignment X with additive error $O(p_{\max})$ and that also satisfies $\mathbb{E}[X_{ij}] = x_{ij}$ and concentration for all linear functions of x_{ij}, i. e., for all $\{a_{ij}\}_{ij}$, with high probability it holds that $\sum_{ij} a_{ij} X_{ij} \approx \sum_{ij} a_{ij} x_{ij}$.

Questions related to finding a good assignment with some concentration properties have been studied before [19, 2, 16], and several methods such as randomized pipage rounding and swap rounding have been developed for this. However, these methods crucially rely on the underlying matching structure and round the variables alternately along cycles, which limits them in various ways: either they give partial assignments, or only get concentration for edges incident to a vertex.

We show that the iterated rounding proof of the result of [24] can be easily modified to work for any slack $\delta \in [0, 1/2)$ while giving additive error $p_{\max}/(1 - 2\delta)$. Theorem 1.2 (say, with $\delta = 1/4$), thus gives a solution that has additive error at most $2p_{\max}$ and satisfies $O(1)$-concentration. The result also extends naturally to the k-resource setting, where p_{ij} is a k-dimensional vector. These results are described in Section 4.2.

1.3.3 Degree-bounded spanning trees and thin trees

In the minimum cost degree-bounded spanning tree problem, we are given an undirected graph $G = (V, E)$ with edge costs $c_e \geq 0$ for $e \in E$, and integer degree bounds b_v for $v \in V$, and the goal is to find a minimum cost spanning tree satisfying the degree bounds. In a breakthrough result, Singh and Lau [31] gave an iterated rounding algorithm that given any fractional spanning tree x, finds a spanning tree T with cost at most $\langle c, x\rangle$ and a degree violation of plus one.

The celebrated thin-tree conjecture asks[1] if given a fractional spanning tree x, there is a spanning tree T satisfying $\Delta_T(S) \leq \beta \Delta_x(S)$ for every $S \subset V$, where $\beta = O(1)$. Here $\Delta_T(S)$ is the number of edges of T crossing S, and $\Delta_x(S)$ is the x-value crossing S. This conjecture has received a lot of attention recently, due to its connection to the asymmetric travelling salesman problem (ATSP) [3, 1]. Despite the recent breakthrough on ATSP [34], the thin-tree conjecture remains open.

If one only considers single vertex sets $S = \{v\}$, the result of [31] implies that $\Delta_T(v) \leq 2\Delta_x(v)$ for each vertex v (as $\Delta_x(v) \geq 1$ in any fractional spanning tree x). On the other hand for general sets S, the best known algorithmic results give $\beta = O(\log n/\log\log n)$ [3, 15, 32, 20]. These algorithms crucially rely on the negative dependence properties of spanning trees, which do

[1] Equivalently, any k-edge-connected graph G has a spanning tree satisfying $\Delta_T(S) = O(1/k)\Delta_G(S)$ for every $S \subset V$.

not give anything better for single vertex cuts (e. g., even if $b_v = 2$ for all v, by a balls-and-bins argument a random tree will have maximum degree $\Theta(\log n/\log\log n)$).

The motivates the following natural question as a first step toward the thin-tree conjecture.

Question 1.7. Can we find a spanning tree that achieves $\beta = O(1)$ for single vertex cuts and $\beta = O(\log n/\log\log n)$ for general cuts?

We show that the iterated rounding algorithm of [31] can be easily modified to create slack $\delta \in (0, 1/2)$ while violating the degree bounds additively by less than $2/(1-2\delta)$. Applying Theorem 1.2 with $\delta = 1/6$, the degree violation is strictly below 3 and this gives a distribution supported on trees with a degree violation of plus 2 and satisfies $O(1)$-concentration. By a standard cut counting argument [3], the concentration property implies $O(\log n/\log\log n)$-thinness for every cut. We describe these results in Section 4.3. In fact, we consider the more general setting of the minimum cost degree-bounded matroid-basis problem.

1.4 Overview of techniques

We now give a high level overview of our algorithm and analysis. The starting observation is that randomized rounding can be viewed as an iterative algorithm by doing a discrete version of the standard Brownian motion on the cube as follows. Given $x^{(0)}$ as the starting fractional solution, consider a random walk in the $[0, 1]^n$ cube starting at $x^{(0)}$, with tiny step size $\pm\gamma$ chosen independently for each coordinate, where upon reaching a face of the cube (i. e., some x_i reaches 0 or 1) the walk stays on that face. The process stops upon reaching some vertex $X = (X_1, \dots, X_n)$ of the cube. By the martingale property of random walks, the probability that $X_i = 1$ is exactly $x_i^{(0)}$ and as the walk in each coordinate is independent, X has the same distribution on $\{0, 1\}^n$ as under randomized rounding.

A first attempt. Now consider iterated rounding, and recall that here the update $y^{(k)}$ at iteration k must lie in the nullspace of $W^{(k)}$. So a natural first idea to combine this with randomized rounding, is to do a random walk in the null space of $W^{(k)}$ until some variable reaches 0 or 1. The slack condition $\mathrm{rank}(W^{(k)}) \le (1-\delta)n_k$ implies that the nullspace has at least δn_k dimensions, which could potentially give "enough randomness" to the random walk.

It turns out, however, that doing a standard random walk in the null space of $W^{(k)}$ does not work. The problem is that as the constraints defining $W^{(k)}$ can be completely arbitrary in our setting, the random walk can lead to very high correlations between certain subsets of coordinates causing the β-concentration property to fail. For example, suppose $\delta = 1/2$ and $W^{(0)}$ consists of the constraints $x_i = x_{i+1}$ for $i = 1, \dots, n/2 - 1$. Then the random walk will update $x_{n/2}, \dots, x_n$ independently, but for $x_1, \dots, x_{n/2}$ the updates must satisfy $\Delta x_1 = \dots = \Delta x_{n/2}$, and hence will be completely correlated. So the linear function $x_1 + \dots + x_{n/2}$ will not have any concentration (as all the variables will simultaneously rise by $-\delta$ or by $+\delta$).

A different random walk. To get around this problem, we design a different random walk in the null space of $W^{(k)}$, which looks similar to an independent walk in *every* direction even though the coordinates are correlated. More formally, consider a random vector $Y = (Y_1, \ldots, Y_n)$, where Y_i are mean-zero random variables. For a parameter $\eta \geq 1$, we say that Y is η-weakly pairwise independent if for every $a = (a_1, \ldots, a_n) \in \mathbb{R}^n$,

$$\mathbb{E}[\langle a, Y\rangle^2] = \mathbb{E}\Big[(\sum_i a_i Y_i)^2\Big] \leq \eta \sum_i a_i^2 \mathbb{E}[Y_i^2].$$

If $Y_1, \ldots, Y_n$ are pairwise independent, note that the above holds as equality with $\eta = 1$, and hence this can be viewed as a relaxation of pairwise independence. We show that whenever $\mathrm{rank}(W^{(k)}) \leq (1-\delta)n_k$, there exist η-weakly pairwise independent random updates $y^{(k)}$ that lie in the null space of $W^{(k)}$ (which has dimension at least δn_k) with $\eta \approx 1/\delta$. Moreover these updates can be found by solving a semidefinite program (SDP).

Next, using a variant of Freedman's martingale analysis [18], we show that applying these η-weakly pairwise independent random updates (with small increments) until all the variables reach 0-1, gives an integral solution that satisfies $O(\eta)$-concentration.

These techniques are motivated by our recent works on algorithmic discrepancy [8, 9]. While discrepancy is closely related to rounding [25, 29], a key difference in discrepancy is that the error for rounding a linear system $Ax = b$ depends on the ℓ_2 norms of the coefficients of the constraints and not on b. E. g., suppose $x \in [0,1]^n$ satisfies $x_1 + \ldots + x_n = \log n$, then the sum stays $O(\log n)$ upon randomized rounding with high probability, while using discrepancy methods directly gives $\Omega(\sqrt{n})$ error, which would be unacceptably large in this setting. So our results can be viewed as using techniques from discrepancy theory to obtain bounds that are sensitive to x. Recently, this direction was explored in [11] but their method gave much weaker results and applied to very limited settings.

2 Technical preliminaries

2.1 Tail bounds for supermartingales

We will need the following tail bound for supermartingales with a strong negative drift.

Theorem 2.1. *Let $\alpha \in (0,1)$. Let $\{Z_k : k = 0, 1, \ldots,\}$ be a sequence of random variables with $Y_k := Z_k - Z_{k-1}$, such that Z_0 is constant and $Y_k \leq 1$ for all $k \geq 1$. If*

$$\mathbb{E}_{k-1}[Y_k] \leq -\alpha\, \mathbb{E}_{k-1}[Y_k^2]$$

for all $k \geq 1$, where $\mathbb{E}_{k-1}[\,\cdot\,]$ denotes $\mathbb{E}[\,\cdot\,|Z_1, \ldots, Z_{k-1}]$, then for $t \geq 0$,

$$\Pr[Z_k - Z_0 \geq t] \leq \exp(-\alpha t).$$

Before proving Theorem 2.1, we first need a simple lemma.

Lemma 2.2. *Let X be a random variable satisfying $X \leq 1$. Then for any $\lambda > 0$,*

$$\mathbb{E}[e^{\lambda X}] \leq \exp\Big(\lambda \mathbb{E}[X] + (e^{\lambda} - \lambda - 1)\mathbb{E}[X^2]\Big).$$

Proof. Let $f(\lambda, x) = (\mathrm{e}^{\lambda x} - \lambda x - 1)/x^2$, where we set $f(\lambda, 0) = \lambda^2/2$. By standard integration, it is easily verified that $f(\lambda, x) = \int_0^\lambda \int_0^s \mathrm{e}^{tx} dt\, ds$. As e^{tx} is non-decreasing in x for all $t \geq 0$, this implies that $f(\lambda, x)$ is non-decreasing in x. In particular, $f(\lambda, x) \leq f(\lambda, 1)$ for any $x \leq 1$ and hence

$$\mathrm{e}^{\lambda x} = 1 + \lambda x + f(\lambda, x)x^2 \leq 1 + \lambda x + f(\lambda, 1)x^2 = 1 + \lambda x + (\mathrm{e}^{\lambda} - \lambda - 1)x^2.$$

Taking expectations and using that $1 + x \leq \mathrm{e}^x$ for all $x \in \mathbb{R}$ gives,

$$\mathbb{E}[\mathrm{e}^{\lambda X}] \leq 1 + \lambda \mathbb{E}[X] + (\mathrm{e}^{\lambda} - \lambda - 1)\mathbb{E}[X^2] \leq \exp\left(\lambda \mathbb{E}[X] + (\mathrm{e}^{\lambda} - \lambda - 1)\mathbb{E}[X^2]\right). \qquad \square$$

Proof of Theorem 2.1. By Markov's inequality,

$$\Pr[Z_k - Z_0 \geq t] = \Pr[\exp(\alpha(Z_k - Z_0)) \geq \exp(\alpha t)] \leq \frac{\mathbb{E}[\exp(\alpha(Z_k - Z_0))]}{\exp(\alpha t)}$$

so it suffices to show that $\mathbb{E}[\exp(\alpha(Z_k - Z_0))] \leq 1$. As Z_0 is deterministic, this is same as $\mathbb{E}[\exp(\alpha Z_k)] \leq \exp(\alpha Z_0)$. Now,

$$\begin{aligned}
& \mathbb{E}_{k-1}\left[\mathrm{e}^{\alpha Z_k}\right] = \mathrm{e}^{\alpha Z_{k-1}} \mathbb{E}_{k-1}\left[\mathrm{e}^{\alpha(Z_k - Z_{k-1})}\right] = \mathrm{e}^{\alpha Z_{k-1}} \mathbb{E}_{k-1}\left[\mathrm{e}^{\alpha Y_k}\right] \\
\leq\ & \mathrm{e}^{\alpha Z_{k-1}} \exp\left(\alpha \mathbb{E}_{k-1}[Y_k] + (\mathrm{e}^{\alpha} - \alpha - 1)\mathbb{E}_{k-1}\left[Y_k^2\right]\right) \qquad \text{(Lemma 2.2)} \\
\leq\ & \mathrm{e}^{\alpha Z_{k-1}} \exp\left((\mathrm{e}^{\alpha} - \alpha^2 - \alpha - 1)\mathbb{E}_{k-1}\left[Y_k^2\right]\right) \\
\leq\ & \mathrm{e}^{\alpha Z_{k-1}} \qquad (\text{as } \mathrm{e}^{\alpha} \leq 1 + \alpha + \alpha^2 \text{ for } 0 \leq \alpha \leq 1).
\end{aligned}$$

As this holds for all k, the result follows by the property of Iterated Expectations. $\square$

2.2 Semidefinite matrices

Let M_n denote the class of all symmetric $n \times n$ matrices with real entries. For two matrices $A, B \in \mathbb{R}^{n \times n}$, the trace inner product of A and B is defined as $\langle A, B \rangle = \mathrm{tr}(A^T B) = \sum_{i=1}^n \sum_{j=1}^n a_{ij} b_{ij}$. A matrix $U \in M_n$ is positive semidefinite (PSD) if all its eigenvalues are non-negative and we denote this by $U \succeq 0$. Equivalently, $U \succeq 0$ iff $a^T U a = \langle a a^T, U \rangle \geq 0$ for all $a \in \mathbb{R}^n$.

For, $U \succeq 0$ let $U^{1/2} = \sum_i \lambda_i^{1/2} u_i u_i^T$, where $U = \sum_i \lambda_i u_i u_i^T$ is the spectral decomposition of U with orthonormal eigenvectors u_i. Then $U^{1/2}$ is PSD and $U = V^T V$ for $V = U^{1/2}$. For $Y, Z \in M_n$, we say that $Y \preceq Z$ if $Z - Y \succeq 0$.

2.3 Approximate independence and sub-isotropic random variables

Let $Y = (Y_1, \ldots, Y_n)$ be a random vector with $Y_1, \ldots, Y_n$ possibly dependent.

Definition 2.3 ((ζ, η) sub-isotropic random vector)**.** For $\zeta \in (0, 1]$ and $\eta \geq 1$, we say that Y is (ζ, η) sub-isotropic if it satisfies the following conditions.

1. $\mathbb{E}[Y_i] = 0$ and $\mathbb{E}[Y_i^2] \leq 1$ for all $i \in [n]$, and $\sum_{i=1}^n \mathbb{E}[Y_i^2] \geq \zeta n$.

2. For all $c = (c_1, \ldots, c_n) \in \mathbb{R}^n$ it holds that

$$\mathbb{E}\Big[(\sum_{i=1}^n c_i Y_i)^2\Big] \leq \eta \sum_{i=1}^n c_i^2 \,\mathbb{E}[Y_i]^2. \tag{2.1}$$

Note that if $Y_1, \ldots, Y_n$ are pairwise independent then (2.1) holds with equality for $\eta = 1$.

Let $U \in M_n$ be the $n \times n$ covariance matrix of $Y_1, \ldots, Y_n$, i. e., $U_{ij} = \mathbb{E}[Y_i Y_j]$. Every covariance matrix is PSD as $c^T U c = \mathbb{E}[(\sum_i c_i Y_i)^2] \geq 0$ for all $c \in \mathbb{R}^n$. Let $\mathrm{diag}(U)$ denote the diagonal $n \times n$ matrix with entries U_{ii}, then (2.1) can be written as $c^T(\eta \,\mathrm{diag}(U) - U)c \geq 0$ for every $c \in \mathbb{R}^n$, and hence equivalently expressed as

$$U \preceq \eta \,\mathrm{diag}(U).$$

Generic construction. We will use the following generic way to produce (ζ, η) sub-isotropic vectors. Let U be a $n \times n$ PSD matrix satisfying: $U_{ii} \leq 1$, $\mathrm{tr}(U) \geq \zeta n$ and $U \preceq \eta \,\mathrm{diag}(U)$. Let $r \in \mathbb{R}^n$ be a random vector where each coordinate is independently and uniformly ± 1. Then the random vector $Y = U^{1/2} r$ has covariance $\mathbb{E}[YY^T] = U^{1/2}\mathbb{E}[rr^T](U^T)^{1/2} = U$, and the properties of U imply that Y is (ζ, η) sub-isotropic.

Remark 2.4. In other similar constructions, r is often chosen as a random Gaussian, but we prefer to choose it as random ± 1 as it is bounded, and this makes some technical arguments easier later on.

We will need the following result from [9], about finding sub-isotropic random vectors orthogonal to a subspace.

Theorem 2.5 ([9], Theorem 6)**.** *Let $G \subset \mathbb{R}^n$ be an arbitrary subspace with dimension $\dim(G) = \ell = \delta n$. Then for any $\zeta > 0$ and $\eta > 1$ satisfying $1/\eta + \zeta \leq \delta$, there is a $n \times n$ PSD matrix U, that is computable in polynomial time, and satisfies the following properties:*
(i) $\langle hh^T, U\rangle = 0$ for all $h \in G^\perp$, where $G^\perp$ is the subspace orthogonal to G.
(ii) $U_{ii} \leq 1$ for all $i \in [n]$.
(iii) $tr(U) \geq \zeta n$.
(iv) $U \preceq \eta \; diag(U)$.

The first condition gives that the range space of U is contained in the subspace G, as $\langle hh^T, U\rangle = 0$ implies that $\|U^{1/2}h\|^2 = 0$ and hence $h^T U^{1/2} = \mathbf{0}$. So, for any vector $r \in \mathbb{R}^n$ the vector $Y = U^{1/2} r$ lies in G (as for every $h \in G^\perp$, $h^T Y = h^T U^{1/2} r = \langle \mathbf{0}, r\rangle = 0$). So this gives a polynomial time algorithm to generate a (ζ, η) sub-isotropic random vector $Y \in G$.

3 The algorithm and analysis

Recall that by iterated rounding we refer to any procedure that given some starting fractional solution x, sets $x^{(0)} = x$, and applies a sequence of updates as follows. Given the vector $x^{(k-1)}$ at the beginning of iteration k, call a variable $i \in [n]$ frozen if $x_i^{(k-1)}$ is 0 or 1, and alive otherwise. Let n_k denote the number of alive variables. Based on $x^{(k-1)}$, the algorithm picks a set of constraints of rank at most $n_k - 1$, given by the rows of some matrix $W^{(k)}$, and finds some non-zero vector $y^{(k)}$ satisfying $W^{(k)}y^{(k)} = 0$ and $y_i^{(k)} = 0$ for i frozen. The solution is updated as $x^{(k)} = x^{(k-1)} + y^{(k)}$.

Let $\delta > 0$ be the slack parameter. We assume that the problem to be solved has an iterated rounding procedure where in each iteration k one can choose some matrix $W^{(k)}$ with $\mathrm{rank}(W^{(k)}) \le (1-\delta)n_k$. We now describe the rounding algorithm.

3.1 The algorithm

Initialize $x^{(0)} = x$, where x in the starting fractional solution given as input. For each iteration $k = 1, \ldots,$ repeat the following until all the variables reach 0 or 1.

Iteration k. Let $x^{(k-1)}$ be the solution at the beginning of iteration k, and let $A_k \subset [n]$ denote the set of *alive* coordinates i with $x_i^{(k-1)} \in (0,1)$. Only the variables in A_k will be updated henceforth. So for ease of notation we assume that $A_k = [n_k]$.

1. Apply Theorem 2.5, with $n = n_k$, G the nullspace of $W^{(k)}$, $\zeta = \delta/10$ and $\eta = 10/(9\delta)$ to find the covariance matrix U.

2. Let $\gamma = 1/(2n^{3/2})$. Let $r_k \in \mathbb{R}^{n_k}$ be a random vector with independent ± 1 entries. Set

$$x^{(k)} = x^{(k-1)} + y^{(k)} \quad \text{with} \quad y^{(k)} = \gamma_k U^{1/2} r_k,$$

where γ_k is the largest value such that (i) $\gamma_k \le \gamma$ and (ii) both $x^{(k-1)} + y^{(k)}$ and $x^{(k-1)} - y^{(k)}$ lie in $[0,1]^{n_k}$.

3.2 Analysis

Let $X = (X_1, \ldots, X_n)$ denote the final 0-1 solution returned by the algorithm. The property that $\mathbb{E}[X_i] = x_i$ follows directly as the update $y^{(k)}$ at each time step has mean zero in each coordinate. As the algorithm always moves in the nullspace of $W^{(k)}$, clearly it will also satisfy the iterated rounding guarantee with probability 1.

To analyze the running time, we note that whenever $\gamma_k < \gamma$, there is at least $1/2$ probability that some new variable will reach 0 or 1 after the update (as the new solution is either $x^{(k)} + y^{(k)}$ or $x^{(k)} - y^{(k)}$ with probability half each). So, in expectation there are at most $2n$ such steps. So let us focus on the iterations where $\gamma_k = \gamma$.

Let us define the energy of $x^{(k)}$ as $E^{(k)} := \sum_i (x_i^{(k)})^2$. During the update, conditioned on all the randomness up to time $k-1$, $E^{(k)}$ rises in expectation by at least $\gamma^2 n_k \zeta$ as,

$$\begin{aligned}\mathbb{E}_{k-1}\left[E^{(k)} - E^{(k-1)}\right] &= \gamma^2 \mathbb{E}_{k-1}\Big[\sum_i (2x_i^{(k-1)} y_i^{(k)} + (y_i^{(k)})^2)\Big] \\ &= \gamma^2 \sum_i \mathbb{E}_{k-1}\left[(y_i^{(k)})^2\right] = \gamma^2 \mathrm{tr}(U) \geq \gamma^2 \zeta n_k.\end{aligned}$$

where the second equality uses that $\mathbb{E}_{k-1}\left[y_i^{(k)}\right] = 0$ for each i. As the final energy can be at most n, standard arguments [6, 26] imply that, with constant probability, the algorithm terminates in $O(n \log n + \log n/\gamma^2)$ time steps.

It remains to show that the rounding satisfies the concentration property, which we do next.

3.2.1 Isotropic updates imply concentration

Let $X = (X_1, \ldots, X_n)$ denote the final 0-1 solution returned by the algorithm. Fix any $a = (a_1, a_2, \ldots, a_n) \in \mathbb{R}^n$. We will show that

$$\Pr[\langle a, x\rangle - \mathbb{E}[\langle a, x\rangle] \geq t] \leq \exp\left(-\frac{t^2/\beta}{2(\sum_i a_i^2(x_i - x_i^2) + Mt/3)}\right)$$

with $\beta = 18\eta$ which equals $20/\delta$ by our choice of the parameters, and $M = \max_i |a_i|$.

Proof. By scaling a_i's and t, we can assume that $M = 1$. Let us define the random variable

$$Z_k = \sum_i a_i x_i^{(k)} + \lambda \sum_i a_i^2 x_i^{(k)}(1 - x_i^{(k)}),$$

where the parameter $\lambda \leq 1$ will be optimized later.

It is useful to think of Z_k as $\mu_k + \lambda v_k$, where $\mu_k = \langle a, x^{(k)}\rangle$ and $v_k = \sum_i a_i^2 x_i^{(k)}(1 - x_i^{(k)})$ are the mean and variance if we would randomly round the coordinates of $x^{(k)}$ to 0-1. Initially, $Z_0 = \mu + \lambda v$ where $\mu = \sum_i a_i x_i$ and $v = \sum_i a_i^2(x_i - x_i^2)$.

We will show that Z_k satisfies the conditions of Theorem 2.1 for an appropriate α, and use it to show the required tail bound. Roughly speaking, as the algorithm proceeds, Z_k has a strong negative drift as the energy term v_k decreases in expectation and μ_k does not change in expectation. In the proof of Theorem 2.1, this negative drift offsets the positive terms that arise while bounding the exponential moment of Z_k.

We now give the details. We first compute $Y_k := Z_k - Z_{k-1}$ and show that it is bounded. Using $x^{(k)} = x^{(k-1)} + y^{(k)}$, we have

$$\begin{aligned}Y_k &= Z_k - Z_{k-1} \\ &= \sum_i a_i(x_i^{(k)} - x_i^{(k-1)}) + \sum_i \left(\lambda a_i^2 (x_i^{(k)}(1 - x_i^{(k)}) - x_i^{(k-1)}(1 - x_i^{(k-1)}))\right) \\ &= \sum_i a_i y_i^{(k)} + \sum_i \left(\lambda a_i^2 y_i^{(k)}(1 - 2x_i^{(k-1)} - y_i^{(k)})\right). \qquad (3.1)\end{aligned}$$

Claim 3.1. *For all k, the update $y^{(k)}$ satisfies $\|y^{(k)}\|_2 \le \gamma\, n = \frac{1}{2\sqrt{n}}$.*

Proof. Recall that $y^{(k)} = \gamma U^{1/2} r_k$. Let $U^{1/2}(i)$ denote the i-th column of $U^{1/2}$. As

$$\langle U^{1/2}(i), U^{1/2}(i)\rangle = U_{ii} \le 1,$$

the columns of $U^{1/2}$ have length at most 1. Let $r_k(i)$ denote the i-th entry of r_k. Applying the triangle inequality to the columns of $U^{1/2}$,

$$\|U^{1/2} r_k\|_2 \le \sum_i |r_k(i)| \|U^{1/2}(i)\|_2 \le \|r_k\|_1 \le n.$$

This gives that $\|y^{(k)}\|_2 \le \gamma n$. □

Claim 3.2. *For all k, $|Y_k| \le 1$.*

Proof. First we note that the second term in (3.1) is at most $\sum_i |a_i y_i^{(k)}|$. This follows as $|a_i|^2 \le |a_i|$ (as $M = 1$), $\lambda \le 1$ by our assumption, and $1 - 2x_i^{(k-1)} - y_i^{(k)} \in [-1, 1]$ (as $1 - x_i^{(k-1)} \in [0, 1]$ and $x_i^{(k-1)} + y_i^{(k)} = x_i^{(k)} \in [0, 1]$). As $\|a\|_\infty \le 1$ and using the bound on $\|y^{(k)}\|_2$ in Claim 3.1, we have

$$\begin{aligned} |Y_k| &\le \sum_i a_i y_i^{(k)} + \sum_i |a_i y_i^{(k)}| \le 2 \sum_i |y_i^{(k)}| \\ &= 2\|y^{(k)}\|_1 \le 2n^{1/2} \|y^{(k)}\|_2 \le 2\gamma n^{3/2} = 1. \end{aligned}$$ □

We now upper bound the negative drift of Z_k.

Claim 3.3. $\mathbb{E}_{k-1}[Y_k] \le -(\lambda/8\eta)\mathbb{E}_{k-1}\left[Y_k^2\right]$

Proof. As $\mathbb{E}_{k-1}\left[y_i^{(k)}\right] = 0$ for all i, and as $x_i^{(k-1)}$ is deterministic conditioned on the randomness until $k-1$, taking expectations $\mathbb{E}_{k-1}[\cdot]$ in (3.1) gives

$$\mathbb{E}_{k-1}[Y_k] = -\lambda \sum_i a_i^2 \mathbb{E}_{k-1}\left[(y_i^{(k)})^2\right]. \tag{3.2}$$

We now upper bound $\mathbb{E}_{k-1}[Y_k^2]$. Using $(a+b)^2 \le 2a^2 + 2b^2$ twice for the expression in (3.1),

$$\begin{aligned} Y_k^2 &\le 2(\sum_i a_i y_i^{(k)})^2 + 2\lambda^2 \left(\sum_i a_i^2 y_i^{(k)} (1 - 2x_i^{(k-1)} - y_i^{(k)})\right)^2 \\ &\le 2(\sum_i a_i y_i^{(k)})^2 + 4\lambda^2 \left(\left(\sum_i a_i^2 y^{(k)} (1 - 2x_i^{(k-1)})\right)^2 + \left(\sum_i a_i^2 (y_i^{(k)})^2\right)^2 \right) \end{aligned} \tag{3.3}$$

Taking expectations $\mathbb{E}_{k-1}[\cdot]$ in (3.3), we now upper bound the terms on the right. As $y^{(k)}$ is (ζ, η) sub-isotropic, by (2.1), the first term satisfies

$$\mathbb{E}_{k-1}\Big[(\sum_i a_i y_i^{(k)})^2\Big] \le \eta \sum_i a_i^2 \mathbb{E}_{k-1}\big[(y_i^{(k)})^2\big].$$

Similarly, by the sub-isotropic property, the second term satisfies

$$\mathbb{E}_{k-1}\Big[(\sum_i a_i^2 y_i^{(k)}(1-2x_i^{(k-1)}))^2\Big] \le \eta \sum_i a_i^4 (1-2x_i^{(k-1)})^2 \mathbb{E}_{k-1}\big[(y_i^{(k)})^2\big] \le \sum_i a_i^2 \mathbb{E}_{k-1}\big[(y_i^{(k)})^2\big],$$

where the last step uses that $|a_i| \le 1$ and $|1-2x_i^{(k-1)}| \le 1$.

Finally, as $|a_i| \le 1$ and $\sum_i (y_i^{(k)})^2 \le \gamma n \le 1/2$ by Claim 3.1, the third term can be bounded as

$$\left(\sum_i a_i^2 (y_i^{(k)})^2\right)^2 \le (1/2) \sum_i a_i^2 (y_i^{(k)})^2. \tag{3.4}$$

Plugging these bounds and using that $\lambda \le 1$, (3.3) gives that,

$$\mathbb{E}_{k-1}\big[Y_k^2\big] \le 8\eta \sum_i a_i^2 \mathbb{E}_{k-1}\big[(y_i^{(k)})^2\big] = -(8\eta/\lambda)\mathbb{E}[Y_k].$$

where the last equality uses (3.2). □

By Claim 3.3, we can apply Theorem 2.1 with $\alpha = \lambda/8\eta$, provided that the conditions for Theorem 2.1 hold. Indeed, $\alpha \le 1$ holds as $\lambda \le 1$ and $\eta \ge 1$, and $|Y_k| < 1$ holds by Claim 3.1.

Applying Theorem 2.1 now gives that $\Pr[Z_T - Z_0 \ge t'] \le \exp(-\lambda t'/8\eta)$. As $Z_0 = \mu + \lambda v$, this gives

$$\Pr[Z_T - \mu - \lambda v \ge t'] \le \exp(-t'\lambda/4\eta). \tag{3.5}$$

Setting $\lambda = t'/(t'+2v)$ (note that this satisfies our assumption $\lambda \le 1$), so that $\lambda v = t'v/(t'+2v) \le t'/2$, (3.5) implies that

$$\Pr[Z_T - \mu \ge 3t'/2] \le \exp(-t'\lambda/4\eta).$$

Setting $t = 3t'/2$ and plugging the value of λ gives the desired result that

$$\Pr[Z_T - \mu \ge t] \le \exp\left(-\frac{t^2/(18\eta)}{2(v+t/3)}\right). \qquad \square$$

4 Applications

4.1 Rounding column-sparse LPs

Let $x \in [0,1]^n$ be a fractional solution to $Ax = b$, where $A \in \mathbb{R}^{m\times n}$ is an arbitrary $m \times n$ matrix. Let $t = \max_{j\in[n]}(\sum_{i\in[m]} |a_{ij}|)$ be the maximum ℓ_1-norm of the columns of A. Beck and Fiala [12] gave a rounding method to find $X \in \{0,1\}^n$ so that the maximum rounding error for any row satisfies $\|AX - b\|_\infty = \|A(X-x)\|_\infty < t$.

Beck–Fiala rounding. We first recall the iterated rounding algorithm in [12]. The algorithm starts with $x^0 = x$ and proceeds in iterations. Consider some iteration k, and let A_k denote the matrix A restricted to the alive coordinates. Call a row *big* if its ℓ_1-norm in A_k is strictly more than t. The key point is that by an averaging argument, the number of big rows is strictly less than n_k as each column has ℓ_1-norm at most t and thus the total ℓ_1-norm of all entries A_k is at most tn_k. The algorithm chooses $W^{(k)}$ to be matrix consisting of the big rows of A_k and applies the iterated rounding update.

Let us now analyze the error. Fix some row i. As long as this row i is big, its rounding error is 0 during the update steps. Consider the first iteration when this row is no longer big. Then, no matter how the remaining alive variables are rounded in subsequent iterations, the error incurred will be (strictly) less than its ℓ_1-norm, which is at most t.

Introducing slack. To apply Theorem 1.2, we can easily introduce δ-slack for any $0 \le \delta < 1$, as follows. In iteration k, call a row big if its ℓ_1 norm exceeds $t/(1-\delta)$, and by the argument above the number of big rows is strictly less than $n_k(1-\delta)$. Theorem 1.2 now directly gives the following result.

Theorem 4.1. *Given a matrix A with maximum ℓ_1-norm of any column at most t, and any $x \in [0,1]^n$, then for any $0 \le \delta < 1$ the algorithm returns $X \in \{0,1\}^n$ such that $\|A(X-x)\|_\infty \le t/(1-\delta)$, $\mathbb{E}[X_i] = x_i$ and X satisfies $O(1/\delta)$-concentration.*

This implies the following useful corollary.

Corollary 4.2. *Let M be a matrix, and A be some subset of rows of M so that the columns of M restricted to A have ℓ_1-norm at most t. Setting $\delta = 1/2$, the rounding error is at most $2t$ for rows of A, while the other rows of M have error similar to that as under randomized rounding.*

Komlós setting. For a $m \times n$ matrix A, let $t = \max_{j \in [n]} (\sum_{i \in [m]} a_{ij}^2)^{1/2}$ denote the maximum ℓ_2-norm of the columns of A. The long-standing Komlós conjecture (together with a connection between hereditary discrepancy and rounding due to [25]) states that any $x \in [0,1]^n$ can be rounded to $X \in \{0,1\}^n$ so that $\|A(X-x)\|_\infty = O(t)$. Currently, the best known bound on this rounding error is $O(t\sqrt{\log m})$ [5, 8].

An argument similar to that for Theorem 4.1 gives the following result in this setting.

Theorem 4.3. *If A has maximum column ℓ_2-norm t, then given any $x \in [0,1]^n$, the algorithm returns an $X \in \{0,1\}^n$ satisfying $\|A(X-x)\|_\infty = O(t\sqrt{\log m})$ and the $O(1)$-concentration property.*

Proof. We will apply Theorem 1.2 with $\delta = 1/2$. During any iteration k, call row i big if its squared ℓ_2 norm in A_k exceeds $2t^2$. As the sum of squared entries of A_k is at most $t^2 n_k$, the number big rows is at most $n_k/2$ and we set $W^{(k)}$ to A_k restricted to the big rows.

The $O(1)$-concentration follows directly from Theorem 1.2. To bound the error for rows of A, we argue as follows. Fix a row i. Clearly, row i incurs zero error while it is big. Let k be the first iteration when row i is not big, and condition on the randomness up to this point. Call an (alive) coordinate j large if $|a_{ij}| \geq t/\sqrt{\log m}$, and let L_i denote the set of large coordinates in row i. Let $\tilde{a}_i$ denote the row a_i with the coordinates in L removed. As $\sum_j a_{ij}^2 \leq 2t$ we have $|L_i| \leq 2\log m$ and so the rounding error due to the coordinates in L_i can be at most

$$\sum_{j\in L_i} |a_{ij}| \leq |L_i|^{1/2}(\sum_{j\in L_i} |a_{ij}|^2)^{1/2} = O(t\sqrt{\log m}).$$

Applying the $O(1)$-concentration property of the rounded solution X, the error due to the entries of $\tilde{a}_i$ satisfies

$$\Pr\left[\sum_{j\notin L} a_{ij}(X_j - x_j^{(k)}) \geq ct\sqrt{\log m}\right] = \exp\left(-c'\frac{c^2t^2\log m}{\sum_{j\notin L} a_{ij}^2 + Mct\sqrt{\log m}}\right)$$

for some fixed constant c'.

As $\sum_{j\notin L} a_{ij}^2 \leq 2t^2$ and $M \leq t/\sqrt{\log m}$, the right hand side is $\exp(-\Omega(cc'\log m))$. Choosing c large enough so that this is $\ll 1/m$, the result follows by a union bound over the rows. ◻

4.2 Makespan minimization on unrelated machines

In the unrelated machine setting, there are r jobs and m machines, and each job $j \in [r]$ has size p_{ij} on a machine $i \in [m]$. The goal is to assign all the jobs to machines to minimize the maximum machine load.

LP formulation. The standard LP relaxation has fractional assignment variables $x_{ij} \in [0,1]$ for $j \in [r]$ and $i \in [m]$. Consider the smallest target makespan T for which the following LP is feasible.

$$\begin{aligned} \sum_{j\in[r]} p_{ij}x_{ij} &\leq T && \forall i \in [m] && \text{(load constraints)} \\ \sum_{i\in[m]} x_{ij} &= 1 && \forall j \in [r] && \text{(assignment constraints)} \\ x_{ij} &= 0 && \forall i,j \text{ such that } p_{ij} > T \end{aligned}$$

The last constraint is valid for any integral solution, and so we can assume that $p_{\max} := \max_{ij} p_{ij}$ is at most T. In a celebrated result, [24] gave a rounding procedure that produces an integral solution with makespan at most $T + p_{\max}$. We now sketch the iterated rounding based proof of this result from [23].

Iterated rounding proof. As always, we start with $x^{(0)} = x$ and fix the variables that get rounded to 0 or 1. Consider some iteration k. Let n_k denote the number of fractional variables, and let R_k denote the set of jobs that are still not integrally assigned to some machine. For a machine i, define the excess as

$$e_i := \sum_{j \in R_k : x_{ij}^{(k)} > 0} (1 - x_{ij}^{(k)}), \tag{4.1}$$

and note that e_i is simply the maximum (fractional) number of extra jobs that can be possibly assigned to i if all the non-zero variables are rounded to 1. An elegant counting argument in [23] shows that if $W^{(k)}$ consists of load constraints for machines with $e_i > 1$, and assignment constraints for jobs in R_k, then $\mathrm{rank}(W^{(k)}) < n_k$.

Introducing slack. We now extend the argument of [23] to introduce some slack so that we can apply Theorem 1.2. This will give the following result.

Theorem 4.4. *Given any $\delta \in [0, 1/2)$, and a fractional solution x to the problem, there is a rounding where the integral solution X increases the load on any machine by $p_{\max}/(1-2\delta)$, satisfies $\mathbb{E}[X_{ij}] = x_{ij}$ for all i, j and has $O(1/\delta)$-concentration.*

Proof. Consider some iteration k, and let n_k denote the number of fractional variables $x_{ij}^{(k)} \in (0, 1)$, and let R_k denote the jobs that are still not integrally assigned. Let $r_k = |R_k|$. For a machine i, we define the excess e_i as in (4.1). Let M_k denote the set of machines with $e_i > 1/(1-2\delta)$.

$W^{(k)}$ will consist of load constraints for machines in M_k and assignment constraints for jobs in R_k. More precisely, the update $y_{ij}^{(k)}$ will satisfy the following two conditions: (i) $\sum_j p_{ij} y_{ij}^{(k)} = 0$ for all $i \in M_k$ and (ii) $\sum_i y_{ij}^{(k)} = 0$, for all $j \in R_k$. We say that machine i is *protected* in iteration k if $i \in M_k$. For a protected machine, the fractional load does not change after an update. When a machine ceases to be protected for the first time, the definition of excess ensures that its extra load in subsequent iterations can be at most $p_{\max}/(1-2\delta)$.

It remains to show that $\mathrm{rank}(W_k) \le (1-\delta)n_k$. As each job in R_k contributes at least two fractional variables to n_k, we first note that

$$2r_k \le n_k. \tag{4.2}$$

Let $m_k = |M_k|$. Then we also have the following.

Claim 4.5. *$m_k \le (1-2\delta)(n_k - r_k)$.*

Proof. Clearly $m_k/(1-2\delta) \le \sum_{i \in M_k} e_i$ as each $i \in M_k$ has excess more than $1/(1-2\delta)$. Next,

$$\sum_{i \in M_k} e_i = \sum_{i \in M_k} \sum_{j \in R_k : x_{ij}^{(k)} > 0} (1 - x_{ij}^{(k)}) \le \sum_{i \in M} \sum_{j \in R_k : x_{ij}^{(k)} > 0} (1 - x_{ij}^{(k)}) = n_k - r_k,$$

where the first equality uses the definition of e_i and second uses the definition of n_k and that $\sum_{i \in M} x_{ij}^{(k)} = 1$ for each job $j \in R_k$. Together this gives $m_k \le (1-2\delta)(n_k - r_k)$. □

Multiplying (4.2) by δ and adding to the inequality in Claim 4.5 gives $m_k + r_k \leq (1-\delta)n_k$, which implies the result as $\mathrm{rank}(W_k) \leq r_k + m_k$. □

Remark 4.6. Setting $\delta = 0$ recovers the additive $p_{\max}$ result of [24]. Theorem 4.4 also generalizes directly to q resources, where job j has load vector $p_{ij} = (p_{ij}(1), \ldots, p_{ij}(q))$ on machine i, and the goal is to find an assignment A of jobs to machines to minimize $\max_{h,i}(\sum_{j:A(j)=i} p_{ij}(h))$. A direct modification of the proof above gives an additive $qp_{\max}/(1-2\delta)$ error and the $O(1/\delta)$-concentration property.

4.3 Minimum cost degree-bounded matroid basis

Instead of just the degree-bounded spanning tree problem, we consider the more general matroid setting as all the arguments apply directly without additional work.

Minimum cost degree-bounded matroid-basis problem (DegMat). The input is a matroid M defined on elements V with costs $c : V \to \mathbb{R}^+$ and m degree constraints specified by (S_j, b_j) for $j \in [m]$, where $S_j \subset V$ and $b_j \in \mathbb{Z}^+$. The goal is to find a minimum-cost base I in M satisfying the degree bounds, i. e., $|I \cap S_j| \leq b_j$ for all $j \in [m]$. The matroid M is given implicitly, by an independence oracle (which given a query I, returns whether I is an independent set or not).

Iterated rounding algorithm. The natural LP formulation for the problem has the variables $x_i \in [0,1]$ for each element $i \in V$ and the goal is to minimize the cost $\sum_i c_i x_i$, subject to the following constraints.

$$\begin{array}{rclll}
\sum_{i\in T} x_i & \leq & r(T) & \forall T \subset V & \text{(rank constraints)} \\
\sum_{i\in V} x_i & = & r(V) & & \text{(matroid base constraint)} \\
\sum_{i\in S_j} x_i & \leq & b_j & \forall j \in [m] & \text{(degree constraints)}
\end{array}$$

Here $r(\cdot)$ is the rank function of M.

Given a feasible LP solution with cost c^*, [22, 10] gave an iterated rounding algorithm that finds a solution with cost at most c^* and an additive degree violation of at most $q-1$. Here $q = \max_i |\{j : i \in S_j\}$ is the maximum number of sets that contain any element i. Note that $q = 2$ for the degree bounded spanning tree problem, as the elements here are edges and the sets S_j consist of edges incident to a vertex, so that each edges can lie in at most two such sets.

We briefly sketch the argument in [22, 10]. The algorithm starts with $x^{(0)} = x$ and applies iterated rounding as follows. Consider some iteration k. Let A_k denote the set of fractional variables and let $n_k = |A_k|$. For a set S_j, define the excess as

$$e_j := \sum_{i\in A_k\cap S_j} (1 - x_i^{(k)}), \tag{4.3}$$

the maximum degree violation for S_j even if all current fractional variables are rounded to 1.

Let D_k be the set of indices j of degree constraints with excess $e_j \geq q$. The algorithm chooses $W^{(k)}$ to consist of the degree constraints in D_k (call these protected constraints) and some basis for the tight matroid rank constraints. An elegant counting argument then shows that $\mathrm{rank}(W_k) \leq n_k - 1$. The correctness follows since if a degree constraint is no longer protected, then its excess is strictly below q, and by integrality of b_j and the final rounded solution, the degree violation can be at most $q - 1$.

Introducing slack. We will extend the argument above in a straightforward way to introduce some slack, and then apply Theorem 1.2 to obtain the following result.

Theorem 4.7. *For any $0 < \delta < 1$, there is an algorithm for the DegMat problem that produces a basis with additive degree violation strictly less than $q/(1-2\delta)$ and satisfies $O(1/\delta)$-concentration.*

Setting $\delta = 1/6$ so that $2/(1-2\delta) = 3$, and noting that the degree violation is strictly less than this bound, gives the following.

Corollary 4.8. *For the minimum cost degree bounded spanning tree problem, given a fractional solution x there is an algorithm to find a spanning tree with degree violation of plus two and satisfying $O(1)$-concentration.*

We now describe the argument. Consider iteration k. Let A_k be the set of fractional variables and $n_k = |A_k|$. We need to specify how to choose $W^{(k)}$ and show that $\mathrm{rank}(W^{(k)}) \leq (1-\delta)n_k$. Let D_k denote the set of indices j of degree constraints with excess $e_j \geq q/(1-2\delta)$. Let $\mathcal{F}$ denote the family of the tight matroid constraints that hold with equality, i. e., $\sum_{i\in S\cap A_k} x_i = r_k(S)$, where r_k is the rank function of the matroid M_k obtained from M by deleting elements with $x_i = 0$ and contracting those with $x_i = 1$. It is well-known [30] that there is a chain family of tight sets $\mathcal{C} = \{C_1, \dots, C_\ell\}$, with $C_1 \subset C_2 \subset \cdots \subset C_\ell$, such that the rank constraint of every $S \in \mathcal{F}$ lies in the linear span of the constraints for sets in $\mathcal{C}$. Let $c_k = |\mathcal{C}|$ and $d_k = |D_k|$. We set $W^{(k)}$ to be the degree constraints in D_k and the rank constraints in $\mathcal{C}$.

Claim 4.9. $\mathrm{rank}(W^{(k)}) \leq (1-\delta)n_k$.

Proof. It suffices to show that $c_k + d_k \leq (1-\delta)n_k$. As each x_i is fractional and as the ranks $r_k(C)$ are integral, it follows that any two sets in chain family differ by at least two elements, i. e., $|C_{i+1} \setminus C_i| \geq 2$. This implies that $c_k \leq n_k/2$. We also note that $r_k(C_1) < r_k(C_2) \cdots <$ and in particular the rank $r(C_{c_k})$ of the largest set in $\mathcal{C}$ is at least c_k. This gives that $\sum_{i\in A_k} x_i \geq c_k$.

Next, as $e_j \geq q/(1-2\delta)$ for each $j \in D_k$, we have that $qd_k \leq (1-2\delta)\sum_{j\in D_k} e_j$. Moreover, by definition of e_j,

$$\sum_{j\in D_k} e_j = \sum_{j\in D_k} \sum_{i\in A_k\cap S_j} (1-x_i) = \sum_{i\in A_k} q_i(1-x_i)$$

where $q_i = |\{j : i \in S_j, j \in D_k\}|$ is the number of tight degree constraints in D_k that contain element i. As $q_i \leq q$, the above is at most $q\sum_{i\in A_k}(1-x_i) \leq qn_k - qc_k$, where we use that $\sum_{i\in A_k} x_i \geq c_k$, and $\sum_{i\in A_k} 1 = |A_k| = n_k$.

Together this gives that $d_k \leq (1-2\delta)(n_k - c_k)$, and adding 2δ times the inequality $c_k \leq n_k/2$ to this gives that $d_k + c_k \leq (1-\delta)n_k$, which proves the desired claim. □

The degree violation property follows as before, since if a degree constraint is no longer protected, then its excess is strictly below $q/(1-2\delta)$.

Finally, we remark that as the underlying LP has exponential size, some care is needed in implementing the rounding algorithm, in particular in maintaining the chain family and in computing the step size of the walk. These issues are discussed in [11].

Acknowledgements

We thank the referees for their extremely thorough reading of the manuscript, catching various errors, and for several useful comments and suggestions that significantly improved the presentation.

References

[1] NIMA ANARI AND SHAYAN OVEIS GHARAN: Effective-resistance-reducing flows, spectrally thin trees, and asymmetric TSP. In *Proc. 56th FOCS*, pp. 20–39. IEEE Comp. Soc., 2015. [doi:10.1109/FOCS.2015.11] 6

[2] SANJEEV ARORA, ALAN M. FRIEZE, AND HAIM KAPLAN: A new rounding procedure for the assignment problem with applications to dense graph arrangement problems. *Math. Programming*, 92(1):1–36, 2002. [doi:10.1007/s101070100271] 3, 6

[3] ARASH ASADPOUR, MICHEL X. GOEMANS, ALEKSANDER MADRY, SHAYAN OVEIS GHARAN, AND AMIN SABERI: An $O(\log n/\log\log n)$-approximation algorithm for the asymmetric traveling salesman problem. *INFORMS*, 65(4):1043–1061, 2017. Preliminary version in SODA'10. [doi:10.1287/opre.2017.1603] 2, 3, 6, 7

[4] ARASH ASADPOUR AND AMIN SABERI: An approximation algorithm for max-min fair allocation of indivisible goods. *SIAM J. Comput.*, 39(7):2970–2989, 2010. [doi:10.1137/080723491] 3

[5] WOJCIECH BANASZCZYK: Balancing vectors and Gaussian measures of n-dimensional convex bodies. *Random Struct. Algor.*, 12(4):351–360, 1998. [doi:10.1002/(SICI)1098-2418(199807)12:4<351::AID-RSA3>3.0.CO;2-S] 15

[6] NIKHIL BANSAL: Constructive algorithms for discrepancy minimization. In *Proc. 51st FOCS*, pp. 3–10. IEEE Comp. Soc., 2010. [doi:10.1109/FOCS.2010.7] 12

[7] NIKHIL BANSAL: On a generalization of iterated and randomized rounding. In *Proc. 51st STOC*, pp. 1125–1135. ACM Press, 2019. [doi:10.1145/3313276.3316313] 1

[8] Nikhil Bansal, Daniel Dadush, and Shashwat Garg: An algorithm for Komlós Conjecture matching Banaszczyk's bound. *SIAM J. Comput.*, 48(2):534–553, 2019. [doi:10.1137/17M1126795] 8, 15

[9] Nikhil Bansal and Shashwat Garg: Algorithmic discrepancy beyond partial coloring. In *Proc. 49th STOC*, pp. 914–926. ACM Press, 2017. [doi:10.1145/3055399.3055490] 8, 10

[10] Nikhil Bansal, Rohit Khandekar, and Viswanath Nagarajan: Additive guarantees for degree-bounded directed network design. *SIAM J. Comput.*, 39(4):1413–1431, 2010. [doi:10.1137/080734340] 18

[11] Nikhil Bansal and Viswanath Nagarajan: Approximation-friendly discrepancy rounding. In *Proc. 18th Integer Prog. Combinat. Optim. (IPCO'16)*, volume 9682 of *LNCS*, pp. 375–386. Springer, 2016. [doi:10.1007/978-3-319-33461-5_31] 3, 8, 20

[12] József Beck and Tibor Fiala: "Integer-making" theorems. *Discr. Appl. Math.*, 3(1):1–8, 1981. [doi:10.1016/0166-218X(81)90022-6] 5, 14, 15

[13] Julius Borcea, Peter Brändén, and Thomas M. Liggett: Negative dependence and the geometry of polynomials. *J. AMS*, 22(2):521–567, 2009. [doi:10.1090/S0894-0347-08-00618-8] 3

[14] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart: *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford Univ. Press, 2013. [doi:10.1093/acprof:oso/9780199535255.001.0001] 3

[15] Chandra Chekuri, Jan Vondrák, and Rico Zenklusen: Dependent randomized rounding via exchange properties of combinatorial structures. In *Proc. 51st FOCS*, pp. 575–584. IEEE Comp. Soc., 2010. [doi:10.1109/FOCS.2010.60] 2, 3, 6

[16] Chandra Chekuri, Jan Vondrák, and Rico Zenklusen: Multi-budgeted matchings and matroid intersection via dependent rounding. In *Proc. 22nd Ann. ACM–SIAM Symp. on Discrete Algorithms (SODA'11)*, pp. 1080–1097. SIAM, 2011. [doi:10.1137/1.9781611973082.82] 2, 3, 6

[17] Devdatt P. Dubhashi and Alessandro Panconesi: *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge Univ. Press, 2009. [doi:10.1017/CBO9780511581274] 3

[18] David A. Freedman: On tail probabilities for martingales. *Ann. Probab.*, 3(1):100–118, 1975. [doi:10.1214/aop/1176996452] 8

[19] Rajiv Gandhi, Samir Khuller, Srinivasan Parthasarathy, and Aravind Srinivasan: Dependent rounding and its applications to approximation algorithms. *J. ACM*, 53(3):324–360, 2006. [doi:10.1145/1147954.1147956] 2, 3, 6

[20] Nicholas J. A. Harvey and Neil Olver: Pipage rounding, pessimistic estimators and matrix concentration. In *Proc. 25th Ann. ACM–SIAM Symp. on Discrete Algorithms (SODA'14)*, pp. 926–945. SIAM, 2014. [doi:10.1137/1.9781611973402.69] 3, 6

[21] RICHARD M. KARP, FRANK THOMSON LEIGHTON, RONALD L. RIVEST, CLARK D. THOMPSON, UMESH V. VAZIRANI, AND VIJAY V. VAZIRANI: Global wire routing in two-dimensional arrays. *Algorithmica*, 2:113–129, 1987. [doi:10.1007/BF01840353] 5

[22] TAMÁS KIRÁLY, LAP CHI LAU, AND MOHIT SINGH: Degree bounded matroids and submodular flows. *Combinatorica*, 32(6):703–720, 2012. [doi:10.1007/s00493-012-2760-6] 18

[23] LAP-CHI LAU, R. RAVI, AND MOHIT SINGH: *Iterative Methods in Combinatorial Optimization*. Cambridge Univ. Press, 2011. [doi:10.1017/CBO9780511977152] 2, 16, 17

[24] JAN KAREL LENSTRA, DAVID B. SHMOYS, AND ÉVA TARDOS: Approximation algorithms for scheduling unrelated parallel machines. *Math. Programming*, 46:259–271, 1990. [doi:10.1007/BF01585745] 6, 16, 18

[25] LÁSZLÓ LOVÁSZ, JOEL H. SPENCER, AND KATALIN VESZTERGOMBI: Discrepancy of set-systems and matrices. *Europ. J. Combinat.*, 7(2):151–160, 1986. [doi:10.1016/S0195-6698(86)80041-5] 8, 15

[26] SHACHAR LOVETT AND RAGHU MEKA: Constructive discrepancy minimization by walking on the edges. *SIAM J. Comput.*, 44(5):1573–1582, 2015. [doi:10.1137/130929400] 3, 12

[27] ROBIN PEMANTLE: Towards a theory of negative dependence. *J. Math. Phys.*, 41(3):1371–1390, 2000. [doi:10.1063/1.533200] 3

[28] YUVAL PERES, MOHIT SINGH, AND NISHEETH K. VISHNOI: Random walks in polytopes and negative dependence. In *Proc. 8th Innovations in Theoret. Comp. Sci. Conf. (ITCS'17)*, pp. 50:1–10. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017. [doi:10.4230/LIPIcs.ITCS.2017.50] 3

[29] THOMAS ROTHVOSS: The entropy rounding method in approximation algorithms. In *Proc. 23rd Ann. ACM–SIAM Symp. on Discrete Algorithms (SODA'12)*, pp. 356–372. SIAM, 2012. [doi:10.1137/1.9781611973099.32] 3, 8

[30] ALEXANDER SCHRIJVER: *Combinatorial Optimization: Polyhedra and Efficiency*. Volume B. Springer, 2003. Springer. 19

[31] MOHIT SINGH AND LAP CHI LAU: Approximating minimum bounded degree spanning trees to within one of optimal. *J. ACM*, 62(1):1:1–19, 2015. [doi:10.1145/2629366] 6, 7

[32] MOHIT SINGH AND NISHEETH K. VISHNOI: Entropy, optimization and counting. In *Proc. 46th STOC*, pp. 50–59. ACM Press, 2014. [doi:10.1145/2591796.2591803] 3, 6

[33] ARAVIND SRINIVASAN: Distributions on level-sets with applications to approximation algorithms. In *Proc. 42nd FOCS*, pp. 588–597. IEEE Comp. Soc., 2001. [doi:10.1109/SFCS.2001.959935] 3

[34] OLA SVENSSON, JAKUB TARNAWSKI, AND LÁSZLÓ A. VÉGH: A constant-factor approximation algorithm for the asymmetric traveling salesman problem. *J. ACM*, 67(6):37:1–53, 2020. [doi:10.1145/3424306] 6

[35] VIJAY V. VAZIRANI: *Approximation Algorithms*. Springer, 2001. ACM DL. 2

[36] DAVID P. WILLIAMSON AND DAVID B. SHMOYS: *The Design of Approximation Algorithms*. Cambridge Univ. Press, 2011. CUP. 2

AUTHOR

Nikhil Bansal
Department of Computer Science
University of Michigan, Ann Arbor
bansal@gmail.com
https://bansal.engin.umich.edu/

ABOUT THE AUTHOR

NIKHIL BANSAL is a professor in the Department of Computer Science at University of Michigan, Ann Arbor. He attended the Indian Institute of Technology, Mumbai for his B. Tech. degree, and received his Ph. D. from Carnegie Mellon University, Pittsburgh. He got fascinated by algorithms after taking an undergraduate class by Ajit A. Diwan, and has worked on various algorithmic questions since then. During his free time, he enjoys reading, hiking and doing Yoga.