Rounding Techniques in Approximation Algorithms

Lecture 15: Iterative Randomized Rounding and Isotropy

Lecturer: Nathan Klein

1 Overview

Over the past three units, we have learned how to *exert more control* over the ways in which we round a solution $x \in [0,1]^n$ to a linear program:

1. **Independent Randomized Rounding.** Here, we flip a coin independently for every variable i and round it to 1 with probability x_i and 0 otherwise. We have very little fine-grained control over the resulting solution x. On the other hand, using Chernoff bounds we can argue that the resulting solution has certain properties with high probability, e.g. getting congestion $O\left(\frac{\log n}{\log\log n}\right)$ for multi-commodity flows. Typically, to argue that a property holds *globally*, we need to lose a factor of approximately $\log n$, as we also saw for k-edge-connectivity.

   Overall: No guaranteed structure, strong concentration bounds.

2. **Dependent Randomized Rounding.** Here we showed that if $x \in P$ for some polytope P with *integral vertices*, then we can randomly round x to a vertex of P, thus imposing the constraints of P on our solution. In the case that P was a matroid polytope, we showed that our solution *also* obeys Chernoff bounds via randomized pipage rounding. In this sense, dependent randomized rounding was a generalization of independent randomized rounding where we can set $P = [0,1]^n$. So, here we are able to enjoy the nice properties of independent randomized rounding with the additional structure of a matroid. We used this to get a much better approximation algorithm for two k-edge-connectivity problems (compared to independent randomized rounding) and an $O\left(\frac{\log n}{\log\log n}\right)$ approximation for ATSP.

   While powerful, there are two major limitations to this technique. First, often there is not a natural integral polytope to use (other than the trivial $[0,1]^n$), in which case it's not clear how to start. Second, the polytopes we find, while integral, may not be matroids. In this case, we do not necessarily get to use Chernoff bounds: in fact, in some cases we provably cannot. In the approximation algorithm for Prize-Collecting TSP we studied, we decomposed x into a distribution over spanning trees, but we did not prove anything about the concentration properties of this distribution.

   Overall: Some amount of guaranteed structure, strong concentration bounds in some cases.

3. **Iterative Rounding and Relaxation.** In iterative rounding, we keep *all* of the structure: we simply increase variables to 1 (paying a penalty in the costs) and ensure all constraints are met. We used this to give a 2-approximation for the survivable network design problem. In iterative relaxation, we fix integral variables and iteratively drop (relax) constraints that are close to being satisfied by the variables that are already integral. We used this to prove the Beck-Fiala theorem and obtain bounded degree matroids.

In both cases, we have very fine-grained control over the structure of the solution. However, we get no concentration bounds at all! In other words, if we care about a constraint, we need to enforce it "manually" in the LP. This is problematic if there are many constraints we care about.

Overall: Lots of guaranteed structure, no concentration bounds.

While dependent randomized rounding was strictly stronger, in some sense, than independent randomized rounding, it seems incomparable to iterative rounding and relaxation and produces very different kinds of results. For example, it is not at all clear how iterative methods could be used to obtain an $O\left(\frac{\log n}{\log\log n}\right)$ approximation for ATSP, since there are exponentially many constraints we need to keep track of. Conversely, dependent randomized rounding seems to have little hope of obtaining bounded degree matroids.

Thus, a natural question is whether it is possible to get the best of both worlds: is there a procedure that integrates randomized rounding and iterative rounding (or relaxation) to produce solutions with lots of guaranteed structure *and* strong concentration bounds? A result you may want to prove with such a thing is as follows: given a point x in the spanning tree polytope with $x(\delta(v)) \leq O(1)$ for all $v \in V$, can we produce a spanning tree with maximum degree $O(1)$ which is $O\left(\frac{\log n}{\log\log n}\right)$ thin?

It turns out the answer is, perhaps surprisingly, yes. This is the subject of the next few lectures of this course. We will follow a work of Bansal [Ban19], following up on an earlier work of Bansal and Garg [BG17].

2 Iterative Randomized Rounding

We will start by slightly generalizing what it means for an algorithm to be following the iterative relaxation framework.

2.1 Iterative Relaxation

Suppose we are given an optimization problem asking to minimize $c^T x$ for $c \in \mathbb{R}^n$ over objects in $\mathbb{Z}^n$, and we have some set of constraints. Abstractly, an iterative relaxation algorithm does the following given some starting point $x_0 \in \mathbb{R}^n$:

At every step k, we have some point $x_k \in \mathbb{R}^n$. Some variables are fixed to integer values and the remaining $n_k \geq 1$ of them are fractional (if x_k is integral we terminate). The algorithm then chooses a collection of linear constraints, say given by the rows of a matrix $W^{(k)}$, with $\dim(W^{(k)}) < n_k$, and updates $x^{(k+1)} = x^{(k)} + y^{(k)}$ for some $y^{(k)} \neq \mathbf{0}$ obeying $W^{(k)} y^{(k)} = \mathbf{0}$ and minimizing $c^T y^{(k)}$ if the problem involves costs. Such a $y^{(k)}$ exists since $\dim(\ker(W^{(k)})) \geq 1$ by our assumption on $W^{(k)}$.

Thus, an iterative rounding algorithm only needs to decide what $W^{(k)}$ is given x_k.

2.2 Re-interpreting Beck-Fiala

Let's rephrase the Beck-Fiala algorithm in this language. We begin with say $x_0 = \mathbf{0}$, and we want to color every element -1 or 1 to minimize the discrepancy. We first drop every constraint with at most t entries. We showed that this means that the number of constraints is at most $n-1$. So, we choose $W^{(0)}$ to be the set of all remaining constraints. Since there is no cost function, we then arbitrarily choose any non-zero direction $d \in \ker(W^{(0)})$ and move along it until some variable is set to -1 or 1. We will keep doing this, setting $W^{(k)} = W^{(0)}$, until we reach an extreme point, i.e. until there is nothing in $\ker(W^{(k)})$. At this point, $\dim(W^{(k)}) = n_k$, violating the condition of iterative relaxation. So, we will update $W^{(k)}$ by dropping constraints that at x_k have at most t entries not set to -1 or 1. This process will continue until we output a coloring with discrepancy at most $2t-1$.

2.3 Moving Randomly in the Kernel

If we want to inject randomness in this process, we need to do so in our choice of $d \in \ker(W^{(k)})$.[1] Clearly, one possibility is to choose some $d \in \ker(W^{(k)})$ and either move to $x+d$ or $x-d$ with some probability so as to preserve the expectation of x. This is analogous to what we did with pipage rounding, which is in fact an example of iterative randomized rounding where $W^{(k)}$ is the set of tight constraints at x_k. And indeed, if we could always choose d to have exactly one -1 and one $+1$, this would all work out exactly as things did there: we would have negative correlation and thus concentration.

However, we don't a priori know anything about $\ker(W^{(k)})$. There is no guarantee that such a direction exists there. Indeed, it could be that $\dim(\ker(W^{(k)})) = 1$ and $n_k = n$. In this case, we're in trouble. What if the kernel is just a line from $\mathbf{0} \in \mathbb{R}^n$ to $\mathbf{1} \in \mathbb{R}^n$? Then we may not get an update to W until we hit $\mathbf{0}$ or $\mathbf{1}$. Say $x_k = (\frac{1}{2}, \ldots, \frac{1}{2})$. Then, we will either output all 0s or all 1s. This clearly does not obey Chernoff bounds since the expected number of coordinates set to 1 is $\frac{n}{2}$ but we get all 1s with probability 1/2.

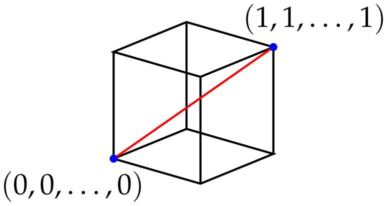

Figure 1: A problematic 1-dimensional kernel, where we imagine this is an n-dimensional cube.

It turns out it is enough to ensure that $\dim(\ker(W^{(k)})) \geq \delta n_k$ for some absolute constant $\delta > 0$ (and every possible $W^{(k)}$). Intuitively, this says that there are enough dimensions to move randomly without creating huge correlations. However, it is not so easy to prove.

One first guess might be to do a random walk in $\ker(W^{(k)})$. It turns out this does not work as we may still have some highly correlated directions (a nice exercise is to see why this fails). Instead, we will use a framework called sub-isotropic rounding. To get some context for this method, we will first learn about isotropic distributions.

[1] One could also try modifying the subspaces randomly, but this may result in the solution not having the desired structure.

2.4 Isotropy

In physics, an object is isotropic with respect to some measurement if that measurement is the same over many different orientations of that object.[2] For example, a sphere is isotropic. In math, it has a number of meanings, but we will focus on one for distributions.

Definition 2.1 (Isotropic Distribution). *Let μ be a distribution over vectors in $\mathbb{R}^n$. We say μ is isotropic if its covariance matrix is I_n, i.e. if $\mathbb{E}_{v\sim\mu}\left[vv^T\right] = I_n$.*

Generally we will also study centered distributions, so that $\mathbb{E}_{v\sim\mu}\left[v\right] = \mathbf{0}$: this translates to the desire that our updates do not change the expectation of x. Another definition applies to sets of vectors $v_1, \dots, v_k$, and a set is typically said to be in isotropic position if $\sum_{i=1}^k v_i v_i^T = I_n$. Let's look at some isotropic sets of vectors.[3] (see Fig. 2).

1. **Example 1.** $\{(-1,0),(1,0)\}$ is not isotropic since $\sum v_i v_i^T = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}$.
2. **Example 2.** $\{(0,1),(0,-1),(\frac{1}{\sqrt{2}},\frac{1}{\sqrt{2}}),(-\frac{1}{\sqrt{2}},-\frac{1}{\sqrt{2}})\}$. Here you can see intuitively that things are skewed towards some directions, and indeed $\sum v_i v_i^T = \begin{bmatrix} 1 & 1 \\ 1 & 3 \end{bmatrix}$.
3. **Example 3.** $\{(1,0),(0,1),(-1,0),(0,-1)\}$ is isotropic, as $\sum v_i v_i^T = I_2$.
4. **Example 4.** Say $v_1, \dots, v_k$ are isotropic and so are $v'_1, \dots, v'_k$. Then, for any c, c', we have:

$$\sum (cv_i)(cv_i^T) + \sum (c'v'_i)(c'v_i'^T) = c^2 I_n + c'^2 I_n = (c^2 + c'^2) I_n$$

So, I can set, for example, $c = \frac{1}{2}, c' = \sqrt{\frac{3}{4}}$, yielding the last example in the figure below, which is isotropic.

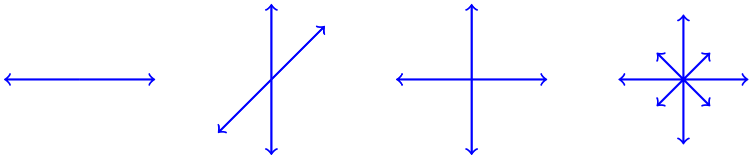

Figure 2: The left two sets are not isotropic, and the right two sets are.

Why are such distributions (or sets, as above) called isotropic? Well, following the definition in physics, we should probably measure these distributions in different directions and make sure they are the same.

How do you measure a distribution in some direction? We already assume that $\mathbb{E}_{v\sim\mu}\left[v\right] = \mathbf{0}$ (which we will for any update), so, $\mathbb{E}_{v\sim\mu}\left[\langle c, v\rangle\right] = 0$. This is not very interesting and is true for any centered distribution, including the uniform distribution over the two non-isotropic sets in Fig. 2.

[2] It comes from the Greek ísos, "equal," and trópos, "turn."

[3] Dan Spielman says "they look like jacks," see this great talk on the Kadison-Singer problem.

For an isotropic distribution, we say instead that the *variance* of the distribution $\langle c, v\rangle$ should not depend on the direction of c. This is $\mathbb{E}\left[\langle c, v\rangle^2\right] - \mathbb{E}\left[\langle c, v\rangle\right]^2 = \mathbb{E}\left[\langle c, v\rangle^2\right]$, and we would want:

$$\mathbb{E}_{v\sim\mu}\left[\langle c, v\rangle^2\right] \propto \|c\|^2$$

And actually this is an equivalent notion of isotropy:

Fact 2.2. *μ is isotropic if and only if*

$$\mathbb{E}_{v\sim\mu}\left[\langle c, v\rangle^2\right] = \|c\|^2$$

Proof.

$$\mathbb{E}_{v\sim\mu}\left[(c^T v)^2\right] = \mathbb{E}_{v\sim\mu}\left[c^T v v^T c\right] = c^T \mathbb{E}_{v\sim\mu}\left[v v^T\right] c = \|c\|^2$$

As desired. □

Why do we care about isotropy? Well, the intuition here should be that if we can get our updates to be isotropic, maybe we will get concentration, since it means our updates are not correlated with any particular direction. Unfortunately, **there is no hope for this**. At step k, if $c \in W^{(k)}$ and y is our update, recall it is required by iterative relaxation that $W^{(k)}y = \mathbf{0}$. So in particular, if $c \in W^{(k)}$, i.e. the rowspace of this matrix, we must have $\langle c, v\rangle = 0 \neq \|c\|^2$, so we cannot make our distribution over updates isotropic.

To give a concrete example and type-check what's been going on so far, suppose

$$W^{(k)} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Then the kernel is generated by the vector $\begin{pmatrix} 0 \\ 1 \\ -1 \end{pmatrix}$, so a generic vector is given by $v = \begin{pmatrix} 0 \\ x \\ -x \end{pmatrix}$. Take anything in the rowspace of $W^{(k)}$, say $c = \begin{pmatrix} 6 \\ 3 \\ 3 \end{pmatrix}$. Then $\langle \begin{pmatrix} 6 \\ 3 \\ 3 \end{pmatrix}, \begin{pmatrix} 0 \\ x \\ -x \end{pmatrix} \rangle = 0$ as we claimed above.

In the next lecture, we will see a weaker definition of isotropy that is possible to obtain and still implies strong concentration.

References

[Ban19] Nikhil Bansal. “On a generalization of iterated and randomized rounding”. In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2019. Phoenix, AZ, USA: Association for Computing Machinery, 2019, 1125–1135. ISBN: 9781450367059. DOI: 10.1145/3313276.3316313 (cit. on p. 2).

[BG17] Nikhil Bansal and Shashwat Garg. “Algorithmic discrepancy beyond partial coloring”. In: *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2017. Montreal, Canada: Association for Computing Machinery, 2017, 914–926. ISBN: 9781450345286. DOI: 10.1145/3055399.3055490 (cit. on p. 2).