Solving Convex Programs by Random Walks

Dimitris Bertsimas
Sloan School of Management and
Operations Research Center, MIT
Cambridge, MA 02139.
dbertsim@mit.edu

Santosh Vempala
Department of Mathematics and
Laboratory for Computer Science, MIT
Cambridge, MA 02139.
vempala@math.mit.edu

Abstract

In breakthrough developments about two decades ago, L. G. Khachiyan [14] showed that the Ellipsoid method solves linear programs in polynomial time, while M. Grötschel, L. Lovász and A. Schrijver [4, 5] extended this to the problem of minimizing a convex function over any convex set specified by a separation oracle. In 1996, P. M. Vaidya [21] improved the running time via a more sophisticated algorithm. We present a simple new algorithm for convex optimization based on sampling by a random walk; it also solves for a natural generalization of the problem.

1 Introduction

The problem of minimizing a convex function over a convex set in $\mathbb{R}^n$ is a common generalization of well-known geometric optimization problems such as linear programming as well as a variety of combinatorial optimization problems including matchings, flows and matroid intersection, all of which have polynomial-time algorithms. As such, it represents a frontier of polynomial-time solvability and occupies a central place in the theory of algorithms.

In his groundbreaking work, Khachiyan [14] showed that the Ellipsoid method [22] solves linear programs in polynomial time. Shortly thereafter, Karp and Papadimitriou [13], Padberg and Rao [18], and Grötschel, Lovász and Schrijver [4] independently discovered the wide applicability of the ellipsoid method to combinatorial optimization problems. This culminated in the book by the last set of authors [5], in which it is shown that the Ellipsoid method solves the problem of minimizing a convex function over a convex set in $\mathbb{R}^n$ specified by a *separation oracle*, i.e., a procedure which given a point x either reports that the set contains x or returns a half-space that separates the set from x. For the special case of linear programming, the oracle simply checks if the query point satisfies all the constraints of the linear program, and if not, reports a violated constraint; another well-known special case is *semi-definite programming*. Vaidya [21] later improved the running time via a more sophisticated algorithm.

In this paper, we present a simple new algorithm for the problem, based on random sampling. Our algorithm also solves a natural generalization that was previously not known to be solvable in polynomial-time.

The key component of the algorithm is sampling a convex set by a random walk. Random walks have long been studied for their mathematical appeal, but of late they have also played a crucial role in the discovery of polynomial-time algorithms. Notable applications include estimating the volume of a convex set [3] and computing the permanent of a non-negative matrix [7]. They have also been used in machine learning and online algorithms [8]. Our algorithm is a new application of random walks to the field of optimization.

In the description below, we assume that the convex set K is contained in the axis-aligned cube of width R centered at the origin; further if K is non-empty then it contains a cube of width r (see e.g. [1, 5] for a justification). The choice of cubes here is somewhat arbitrary; we could instead use balls, for example. The parameter L is equal to $\log \frac{R}{r}$.

Algorithm.
Input: A separation oracle for a convex set K and a number L.
Output: A point in K or a guarantee that K is empty.

1. Let P be the axis-aligned cube of width R with center $z = 0$.
2. Check if z is in K. If so then report z and stop. If not, then let $a^T x \leq b$ be the half-space containing K reported by the oracle. Set $H = \{x \mid a^T x \leq a^T z\}$.
3. Set $P = P \cap H$. Pick N random points $y_1, y_2, \ldots, y_N$ from P. Set z to be their average: $z = \frac{1}{N} \sum_{i=1}^{N} y_i$.
4. Repeat steps 2 and 3 at most $2nL$ times. Report K is empty.

Roughly speaking, the algorithm is computing an approximate centroid in each iteration[1]. The number of samples required in each iteration, N, is $O(n)$ for an arbitrary convex set in $\mathbb{R}^n$ and $O(\log^2 m)$ if K is a polyhedron with m inequalities (i.e. a linear program). Note that optimization

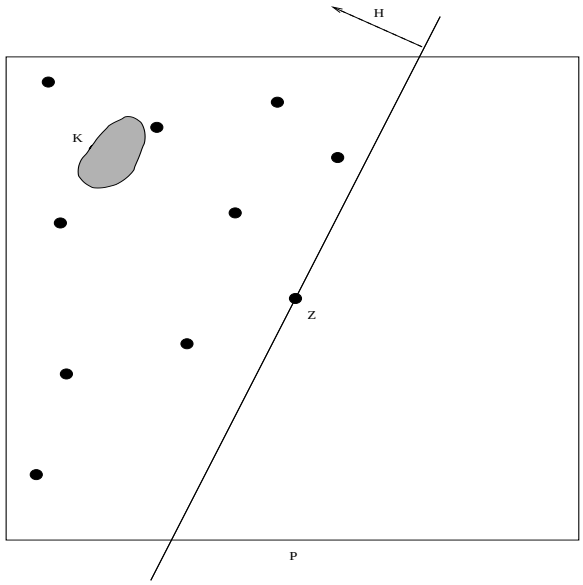

Figure 1: An illustration of the algorithm.

is easily reduced to the *feasibility* problem solved above: to (say) minimize a convex function $f(x)$ we simply add the constraint $f(x) \leq t$ and search (in a binary fashion) for the optimal t.

The idea behind the algorithm is that the volume of the enclosing polytope P is likely to drop by a constant factor in each iteration. We prove this in Section 2 and derive as a consequence that if

[1]The idea of an algorithm based on computing the *exact* centroid was suggested in 1965 by Y. Levin [15], but is computationally intractable.

the algorithm does not stop in $2nL$ iterations, then K must be empty (with high probability). Thus, the total number of calls to the separation oracle is at most $2nL$. This matches the (asymptotic) bound for Vaidya's algorithm, and is in general the best possible [22]. The ellipsoid algorithm, in contrast, makes up to $O(n^2L)$ iterations and as many calls to the oracle.

In the discussion so far, we have assumed that the convex set of interest has an efficient separation oracle. A natural question is whether optimization can be solved using a significantly weaker oracle, namely a *membership* oracle (which reports whether a query point is in the set or not, but provides no other information). One of the main results in [5] is that a linear function can be optimized over a convex set K given only by a membership oracle, provided K is "centered", i.e., we are also given a point y and a guarantee that a ball of radius r around y is contained in K. The algorithm is intricate and involves a sophisticated variant of the Ellipsoid method, called the *shallow-cut* Ellipsoid. Our algorithm provides a simple solution to this problem. In fact, as we show in Section 4, it solves the following generalization: let K be the intersection of two convex sets K_1 and K_2, where K_1 is a "centered" convex set with a membership oracle and K_2 is a convex set with a separation oracle; find a point in K if one exists. The generalization includes the special case of minimizing a convex function over a centered convex set given by a membersip oracle. This problem (and its special case mentioned above) are not known to be solvable using the Ellipsoid method or Vaidya's algorithm.

Each iteration of our algorithm needs random samples from the current polytope P. The time per iteration depends on how quickly we can draw random samples from P. The problem of sampling from a convex set has received much attention in recent years [3, 17, 11, 16, 9], in part because it is the only known way of efficiently estimating the volume of a convex set. The general idea is to take a random walk in the set. There are many ways to walk randomly; of these, the *ball* walk (go to a random point within a small distance) [9] and *hit-and-run* (go to a random point along a random direction) [16] have the best known bounds on the number of steps needed to draw a random sample. The bounds on the number of steps depend on how "round" the convex set is. For a set that is close to *isotropic position* (see Section 2), $O(n^3)$ steps are enough for one random point. In our case, the initial convex set, the cube, is indeed in isotropic position. However this might not be the case after some iterations. As we describe in Section 3, this problem can be tackled by computing an affine transformation that keeps the current polytope P into near-isotropic position. We also propose an alternative (in Section 3.1) that avoids this computation and instead incorporates the information about isotropy *implicitly* in the steps of a random walk by maintaining a set of points. The time per iteration is bounded by $O^*(n^4)$ steps of a random walk, giving an overall running time of $O^*(n^5L)$.

As the reader might notice, the current bound on the worst-case time per iteration of our algorithm is higher than both the Ellipsoid method and Vaidya's algorithm. However, in practice drawing random samples from convex sets might be much faster than the known worst-case bounds; also, sampling convex sets is an active research area and there might well be faster sampling methods in the future.

2 Number of iterations

The following definitions will be useful throughout the analysis. We assume that all our convex sets are closed and bounded.

Definition 1. *A convex set K in $\mathbb{R}^n$ is said to be in* isotropic position *if its center of gravity is the origin, i.e., for a random point x in K,*

$$E_K(x) = 0$$

and its variance-covariance matrix is the identity, i.e.,

$$E_K(xx^T) = I.$$

Equivalently, for any unit vector v ($||v|| = 1$)

$$\frac{1}{vol(K)} \int_K (v^T x)^2 dx = 1.$$

In other words, for a set K in isotropic position, the average squared length in any direction is 1. In particular, this implies that

$$\frac{1}{vol(K)} \int_K ||x||^2 dx = n.$$

A convex set can be brought into isotropic position by an affine transformation. To bring a convex set K with center of gravity z into isotropic position, let $A = E_K((x-z)(x-z)^T)$ and consider the transformation $K' = \{y : y = A^{-\frac{1}{2}}(x-z),\ x \in K\}$. Then $E_{K'}(y) = 0$ and $E_{K'}(yy^T) = I$.

Our proof utilizes log-concave functions which we introduce next.

Definition 2. *A function $f : \mathbb{R}^n \to \mathbb{R}_+$ is* log-concave *for any two points $a, b \in \mathbb{R}^n$ and any $\lambda \in (0,1)$,*

$$f(\lambda a + (1-\lambda)b) \geq f(a)^\lambda f(b)^{1-\lambda}.$$

In other words, a nonnegative function f is log-concave if its support is convex and $\log f$ is concave. For example, a function that is constant over a bounded convex set and zero outside the set is log-concave. Another example is a Gaussian density function. It can be easily verified from the definition above that the product of two log-concave functions is also log-concave (but their sum is not).

The next theorem, whose proof uses Lemma 3, is the key to bounding the number of iterations. In this section, by high probability we mean probability higher than any desired constant.

Theorem 1. *The volume of P drops by a factor of $\frac{2}{3}$ with high probability in each iteration.*

Proof. We will prove that for any convex set K, if z is the average of sufficiently many random samples from K, then any half-space through z cuts off a constant fraction of the volume of K. To this end we can assume without loss of generality that K is in isotropic position. This is because of two facts: (i) as shown earlier, any convex set can be brought into isotropic position by an affine transformation and (ii) on applying an affine transformation A to K, the volume scales by $det(A)$, i.e. $vol(AK) = det(A)vol(K)$; so affine transformations preserve ratios of volumes.

Let $y_1, y_2, \ldots, y_N$ be the N samples drawn uniformly from K. Let

$$z = \frac{1}{N} \sum_{i=1}^{N} y_i.$$

Then, using the isotropy of K,

$$E(y_i) = 0 \quad \text{and} \quad Var(||y_i||) = E(||y_i||^2) = n.$$

Thus,

$$E(z) = 0 \quad \text{and} \quad Var(||z||) = E(||z||^2) = \frac{n}{N}.$$

We next consider a unit vector $a \in \mathbb{R}^n$ ($||a|| = 1$) and define the one-dimensional marginal distribution f

$$f(y) = \frac{1}{vol(K)} \int_{x \in K, a^T x = y} dx.$$

In other words $f(y)$ is the $(n-1)$-dimensional volume of K intersected with the hyperplane $a^T x = y$. By Lemma 3, $\max_{y\in\mathbb{R}} f(y) < 1$.

Another fact that we will use is that any hyperplane through the center of gravity of a convex set has at least a $\frac{1}{e}$ fraction of its volume on either side [6]:

$$\int_{-\infty}^{0} f(y)dy \geq \frac{1}{e} \quad \text{and} \quad \int_{0}^{\infty} f(y)dy \geq \frac{1}{e}.$$

Assume without loss of generality that $a^T z \leq 0$. The fraction of the volume of K that is cut off by a half-space through z is at least

$$\int_{-\infty}^{a^T z} f(y)dy = \int_{-\infty}^{0} f(y)dy - \int_{a^T z}^{0} f(y)dy \geq \frac{1}{e} - |a^T z|.$$

Since $E(||z||^2) = \frac{n}{N}$ by choosing $N = O(n)$ we can have $||z||$ and hence $|a^T z|$ smaller than any constant. We choose N so that $\frac{1}{e} - |a^T z| \geq \frac{1}{3}$ with high probability. That is, any half-space through z cuts off at least $\frac{1}{3}$ of the volume of K. □

We remark that in the proof above we only need the random samples to be pairwise independent. Also note that Theorem 1 remains true even if the samples are only nearly uniform, i.e., the probability density everywhere is at most $(1+\epsilon)$ times the uniform density, for a sufficiently small ϵ.

Corollary 2. *If the algorithm does not find a feasible point in $2nL$ iterations, then with high probability, the given convex set is empty.*

Proof. The initial volume of P is R^n. If the set K is nonempty, then it has volume at least r^n. Thus the number of iterations in which the volume drops by $2/3$ is at most

$$\log_{\frac{3}{2}} \left(\frac{R^n}{r^n}\right) = n \log_{\frac{3}{2}} \frac{R}{r}$$

and thus with high probability, the total number of iterations is at most $2nL$. □

2.1 The maximum cross-sectional area

Lemma 3. *For an isotropic convex set K and any unit vector $a \in \mathbb{R}^n$, define a function $f : \mathbb{R} \to \mathbb{R}_+$ as*

$$f(y) = \frac{1}{vol(K)} \int_{x\in K, a^T x = y} dx.$$

Then

$$\max_{y\in\mathbb{R}} f(y) < 1.$$

Proof. Fix a unit vector $a \in \mathbb{R}^n$. Let f_K be the one-dimensional marginal distribution

$$f_K(y) = \frac{1}{vol(K)} \int_{x\in K, a^T x = y} dx.$$

In other words, $f_K(y)$ is the $(n-1)$-dimensional volume of K intersected with the hyperplane $a^T x = y$. Define

$$\bar{y}_K = \int_{\mathbb{R}} y f(y)\, dy \quad \text{and} \quad I(K) = \int_{\mathbb{R}} (y - \bar{y}_K)^2 f_K(y) dy.$$

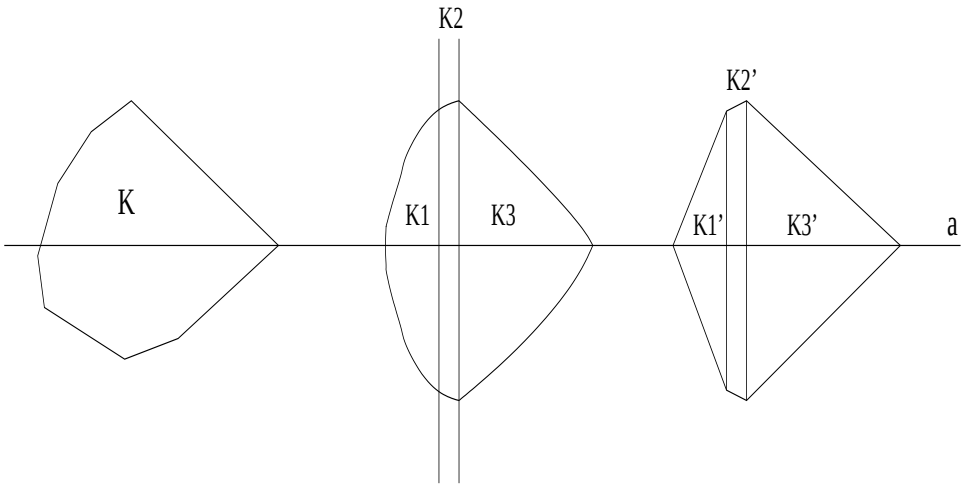

Figure 2: The proof of Lemma 3.

We will in fact prove a slightly stronger statement. For any convex body K for which f_K defined as above satisfies

$$\int_y y f(y)\,dy = 0 \quad \text{and} \quad \int_y y^2 f(y)\,dy = 1,$$

$\max_y f_K(y) < 1$.

To this end let K be such a convex set for which $\max_y f_K(y)$ is the maximum possible. We will show that K can be assumed to be an isotropic cone. To see this first consider the set K' obtained by replacing each cross-section $K \cap \{x | a^T x = y\}$ by an $(n-1)$-dimensional ball of area $f_K(y)$. Clearly K' has the same volume as K. Further, by the Brunn-Minkowski principle of symmetrization, K' is also convex (in other words, f_K is log-concave [19]). So without loss of generality, we can assume that K is symmetric about the vector a; let us also translate K so that $\bar{y}_K = 0$. Also, the conditions of f_K continue to be satisfied.

If K is a cone, we are done. If not, let y^* be a point where f_K is maximum and suppose that $y^* \geq 0$ (the other case is symmetric). Divide K into three parts:

$$K_1 = K \cap \{x | a^T x < 0\}, \quad K_2 = K \cap \{x | 0 \leq a^T x \leq y^*\}, \quad K_3 = K \cap \{x | a^T x > y^*\}.$$

We will now use the observation that moving mass away from the center of gravity can only increase the moment of inertia, and make the following operations. Replace K_1 by a cone K_1' of the same volume with base area equal to $f_K(0)$. Replace K_2 by a truncated cone K_2' of height y^*, top area $f_K(0)$ and base area $f_K(y^*)$. Finally replace K_3 by a cone K_3' of base area $f(y^*)$ and $vol(K_3') = vol(K_3) + vol(K_2) - vol(K_2')$. Let $K' = K_1' \cup K_2' \cup K_3'$ be the new convex set.

After the first step, the new center of gravity along a can only move to the left. Thus the next two steps also move mass away from the center of gravity. So at the end,

$$I(K') \geq I(K), \quad vol(K) = vol(K'), \quad \max f_K(y) = \max f_{K'}(y).$$

Now a simple computation shows that a cone of the same volume as K' and base area $f_K(y^*)$ has a larger moment of inertia than K', unless K' itself is a cone. However, if $I(K') > I(K)$, then we can scale down K' along a and scale it up perpendicular to a, in such a way as to maintain the volume, achieve $I(K') = I(K)$, and have $\max f_{K'}(y) > \max f_K(y)$. This contradicts the choice of K.

An isotropic cone has height $n+1\sqrt{\frac{n+2}{n}}$ and so $\max_y f(y) \leq \frac{n}{n+1}\sqrt{\frac{n}{n+2}}$. □

2.2 A better bound for linear programming

For the case of linear programming, when the target convex set K is a polyhedron defined by m linear inequalities, a smaller number of samples can be used in each iteration. The next lemma is the basis of the improved sample complexity.

Lemma 4. *Let K be a convex set and z be the average of N random samples from K. Let a be a fixed unit vector. Then the probability that the halfspace $a^T x \leq a^T z$ cuts off at least $\frac{1}{3}$ of K is at least $1 - 2^{-c\sqrt{N}}$ for some constant c.*

Proof. Assume without loss of generality that P is in isotropic position. Let $Y_i = a^T y_i$ for $i = 1, \ldots, N$. Then, Y_i's are independent random variables with distribution given by the log-concave function

$$f(y) = \frac{1}{vol(K)} \int_{x \in K, a^T x = y} dx.$$

We have

$$E(Y_i) = 0 \quad \text{and} \quad E(Y_i^2) = 1.$$

Let

$$Z = a^T z = \frac{1}{N} \sum_{i=1}^{N} Y_i.$$

Then,

$$E(Z) = 0 \quad \text{and} \quad E(Z^2) = \frac{1}{N}.$$

Now the distribution of Z is the convolution of the distributions of the Y_i's and is also log-concave [19]. Thus, there are constants $C > 1, D$ such that for any $t \geq C$,

$$P\left(|Z| > \frac{t}{\sqrt{N}}\right) \leq e^{-Dt}.$$

As a consequence, for $N > C^2/c^2$,

$$Pr(|a^T z| > c) \leq e^{-cD\sqrt{N}}.$$

By choosing c small enough, we get that the half-space $a^T x \leq a^T z$ cuts off at least $\frac{1}{3}$ of the volume of P with probability at least $1 - e^{-c'\sqrt{N}}$ for some absolute constant c'. □

Corollary 5. *Let the target convex set be the intersection of m half-spaces. Then with $N = O(\log^2 m)$ the volume of P drops by a factor of $\frac{2}{3}$ in each iteration with high probability.*

Proof. As in the proof of Theorem 1, let $y_1, \ldots, y_N$ be random variables denoting the samples from the current polytope P and let z be their average. In Theorem 1, we showed that for *any* unit vector a, the hyperplane normal to a and passing through z is likely to cut off at least $\frac{1}{3}$ of the volume of P. Suppose that the target set K is the intersection of m half-spaces defined by hyperplanes with normal vectors $a_1, \ldots, a_m$ (with $||a_i|| = 1$). Then we only need to show that each of the hyperplanes $a_i^T x = a_i^T z$ cuts off at least $\frac{1}{3}$ of P. This is because the separation oracle for the target convex set (which simply checks the m half-spaces and reports one that is violated) will return a hyperplane parallel to one of these.

By Lemma 4, any single hyperplane $a^T x = a^T z$ cuts off $\frac{1}{3}$ of P with probability at least $1 - 2^{-c\sqrt{N}}$ for some constant c. Setting $N = O(\log^2 m)$ implies that with high probability any one of m half-spaces will cut off a constant fraction of P. □

3 Sampling and Isotropy

In each iteration, we need to sample the current polytope. For this we take a random walk. There are many ways to walk randomly but the two ways with the best bounds on the *mixing time* are the ball walk and hit-and-run. They are both easy to describe.

Ball walk

1. Choose y uniformly at random from the ball of radius δ centered at the current point x.
2. If y is in the convex set then move to y; if not, try again.

Hit-and-run

1. Choose a line ℓ through the current point x uniformly at random.
2. Move to a point y chosen uniformly from $K \cap \ell$.

The mixing time of the walk depends on how close the convex set is to being in isotropic position. We say that a convex set K is in near-isotropic position if for any unit vector v,

$$\frac{1}{2} \leq \frac{1}{vol(K)} \int_K (v^T(x - \bar{x}))^2 dx \leq 2.$$

Equivalently, the variance-covariance matrix (also called the matrix of inertia) of the uniform distribution over K has eigenvalues between $\frac{1}{2}$ and 2. In addition to isotropy, the starting point (or distribution) of a random walk plays a role in the bounds on the mixing time. The best bounds available are for a *warm start* – a starting distribution with the property that the density at any point in K is at most a constant times the uniform density, π. The following result is paraphrased from [16].

Theorem 6. *[16] Let K be a near-isotropic convex set, σ be any distribution on it with the property that $\sigma(x) \leq 10\pi(x)$ and σ' be the distribution obtained after $O(\frac{n^3}{\epsilon^2} \log \frac{1}{\epsilon})$ steps of hit-and-run starting at σ. Then σ' has total variation distance less than ϵ from the uniform distribution.*

A similar statement holds for ball walk with step size $\delta = \Theta(\frac{1}{\sqrt{n}})$ (see [9]). One advantage of hit-and-run is that there is no need to choose the "step size" δ.

The problem with applying this directly in our algorithm to the current polytope P is that after some iterations P may not be in near-isotropic position. One way to maintain isotropic position is by using random samples to calculate an affine transformation. It was proven in [11] that for any convex set, $O(n^2)$ samples allow us to find an affine transformation that brings the set into near-isotropic position. This was subsequently improved to $O(n \log^2 n)$ [2, 20]. The procedure for a general convex set K is straightforward:

1. Let $y_1, y_2, \ldots, y_N$ be random samples from K.
2. Compute
$$\bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i \quad \text{and} \quad Y = \frac{1}{N} \sum_{i=1}^{N} (y_i - \bar{y})(y_i - \bar{y})^T.$$

The analysis of this procedure rests on the following consequence of a more general lemma due to Rudelson [20].

Lemma 7. *[20] Let K be a convex body in $\mathbb{R}^n$ in isotropic position and $\eta > 0$. Let $y_1, \ldots, y_m$ be independent random points uniformly distributed in K, with*

$$m \geq C\frac{n}{\eta^2}\log\frac{n}{\eta^2}(p + \log n).$$

for an absolute constant C and any positive integer p between 1 and n. Then

$$E||\frac{1}{m}\sum_{i=1^m} y_i y_i^T - I||^p \leq \eta^p$$

Corollary 8. *There is an absolute constant C such that such that for any integer $t \geq 1$, and $N \geq Ctn\log^2 n$, with probability at least $1 - \frac{1}{n^t}$, the set $K' = Y^{-\frac{1}{2}}(K - \bar{y})$ is in near-isotropic position.*

Proof. Without loss of generality, we can assume that K is in isotropic position. Let

$$Y = \frac{1}{m}\sum_{i=1^m} y_i y_i^T - I.$$

Then applying Lemma 7 we have

$$E||Y - I||^p \leq \eta^p.$$

Hence,

$$\begin{aligned} Pr[||Y - I|| > 2\eta] &= Pr[||Y - I||^p > (2\eta)^p] \\ &\leq \frac{E[||Y - I||^p]}{(2\eta)^p} \\ &\leq \frac{1}{2^p} \end{aligned}$$

Using $p = t\log n$ and $\eta = \frac{1}{4}$, we get that with $N \geq Ctn\log^2 n$ points (this is a different constant C from that in Lemma 7),

$$Pr[||Y - I|| > \frac{1}{2}] \leq \frac{1}{n^t}.$$

Next we evaluate $\max v^T Y v$ and $\min v^T Y v$ over unit vectors v.

$$\begin{aligned} v^T Y v &= v^T I v + v^T(Y - I)v \\ &\leq 1 + ||Y - I|| \end{aligned}$$

$$\begin{aligned} v^T Y v &= v^T I v + v^T(Y - I)v \\ &\geq 1 - ||Y - I|| \end{aligned}$$

Thus with probability $1 - \frac{1}{n^t}$, $v^T Y v$ is between $\frac{1}{2}$ and $\frac{3}{2}$ for any unit vector v, and so K' is in near isotropic position. □

In fact, using $N = O^*(n)$ points, the probability of failure in each iteration can be made so low (inverse polynomial) that the probability of failure over the entire course of the algorithm is smaller than than any fixed constant. In order to keep the sampling efficient, we calculate such an affine transformation in each iteration, using samples from the previous iteration. However we do not need to apply the transformation to the current polytope P. Instead, we could keep P as it is and use the following modified random walk: From a point x,

1. Choose a point y uniformly at random from $x + \delta Y^{\frac{1}{2}} B_n$.

2. If y is in the convex set then move to y; if not, try again.

A similar modification (choose the line ℓ from $Y^{\frac{1}{2}} B_n$ rather than B_n) can be made to hit-and-run also.

Theorem 9. *Each iteration of the algorithm can be implemented in $O^*(n^4)$ steps of a random walk in P.*

Proof. Our initial convex set, the cube, is in isotropic position and it is easy to sample from it. Given a warm start in a current polytope P in near-isotropic position, we take $O^*(n^3 N)$ steps of a random walk to get $2N$ nearly random samples (using Theorem 6). We compute the average z of N of these, and if z is not in the target convex set, then we refine P with a new half-space H to get P'. Of the remaining N points, at least a constant fraction (say $\frac{1}{4}$) are in P' with high probability. Further, their distribution is nearly uniform over P'. Using these points, we estimate $Y^{\frac{1}{2}}$ by the formula of Corollary 8. With high probability this is a nonsingular matrix. By Corollary 8, we only need $N = O^*(n)$ samples in each iteration. The reason for discarding the subset of samples used in computing the average is to avoid correlations with future samples. □

3.1 An isotropic variant

Here we describe an alternative method which implicitly maintains isotropy. It has the advantage of completely avoiding the computation of the linear transformation $Y^{\frac{1}{2}}$.

Instead of walking with a single point, we maintain m points $v_1, v_2, \ldots, v_m$.
For each $j = 1, \ldots, m$,

1. Choose a direction $\ell = \sum_{i=1}^{m} \alpha_i v_i$ where $\alpha_1, \ldots, \alpha_m$ are drawn independently from the standard normal distribution.

2. Move v_j to a random point in K along ℓ.

Theorem 10. *Suppose the multi-point walk is started with $m = \Omega(n \log^2 n)$ points drawn from a distribution σ that satisfies $\sigma(x) \leq 10\pi(x)$. Then the distribution of each point after a total of $O(m \frac{n^3}{\epsilon^2} \log \frac{1}{\epsilon})$ steps has total variation distance less than ϵ from the stationary distribution.*

Proof. The random walk described above is invariant under affine transformations, meaning there is a one-to-one map betweeen a walk in K and a walk in an affine transformation of K; hence we can assume that K is isotropic. Now consider a slightly different walk where we keep m points, pick one and make a standard hit-and-run step, i.e. the direction is chosen uniformly. The corresponding Markov chain has states for each m-tuple of points from K, and since each point is walking independently, it follows from Theorem 6 that the chain has the bound described above.

Compare this to the Markov chain for our random walk. The stationary distribution is the same (our Markov chain is symmetric). The main observation is that for $v_1, \ldots, v_m$ picked at random and $m = \Omega(n \log^2 n)$, the matrix of inertia is very likely to have eigenvalues bounded by constants from above and below. Using Corollary 8, we can choose m so that this probability is $1 - \frac{1}{n^{10}}$. Thus, for all but a $\frac{1}{n^{10}}$ fraction of states, the probability of each single transition of our Markov chain is within a constant factor of the first Markov chain. So the conductance of any subset of probability greater than $\frac{1}{n^8}$ (say) is within a constant factor of its conductance in the first chain. Since we start out with a warm distribution (i.e., a large subset), this implies the bound on the mixing time. □

Thus the entire convex programming algorithm now consists of the following: We maintain $2N$ random points in the current polytope P. We use N of them to generate the query point and refine P. Among the rest, we keep those that remain in P, and continue to walk randomly as described above till we have $2N$ random points again.

4 A generalization

In this section we consider the problem of optimization over a convex set given by a weaker oracle, namely a membership oracle. In [5], it is shown that this is indeed possible using a sophisticated variant of the ellipsoid algorithm, provided we have a *centered* convex set K. That is, in addition to the membership oracle, we are given a point $y \in K$, and a guarantee that a ball of radius r around y is contained in K. The algorithm and proof there are quite nice but intricate[2].

Our algorithm provides a solution for a more general problem: K is the intersection of two convex sets K_1 and K_2. We are given a separation oracle for K_1, a membership oracle for K_2 and a point in K_2. The problem is to minimize a convex function over K.

Theorem 11. *Let $K = K_1 \cap K_2$ where K_1 is given by a separation oracle and K_2 by a membership oracle. Let R be the radius of an origin-centered ball containing K. Then, given a point $y \in K_2$ such that the ball of radius r around y is contained in K_2 (i.e., K_2 is centered), the optimization problem over K can be solved in time polynomial in n and $\log \frac{R}{r}$.*

Proof. It is clear that optimization is reducible to feasibility, i.e., we just need to find a point x in K or declare that K is empty. We start with P as K_2 intersected with the ball of radius $2R$ with center y and apply our algorithm. The key procedure, a random walk in P, needs a point in P and a membership oracle for P, both of which we have initially and at each iteration. Using $O^*(n)$ samples we can bring P into near-isotropic position. Also, each subsequent query point z is guaranteed to be in K_2, since it is the average of points from K_2. Thus the algorithm needs at most $O(n \log \frac{R}{r})$ iterations and $O(n \log \frac{R}{r})$ calls to the separation oracle for K_1. Each iteration (after the first) makes $O^*(n^4)$ calls to the membership oracle for K_2.

To bring the initial set P into isotropic position, we can adapt the "chain of bodies" procedure of [11]. Since K_2 is centered, it contains a ball of radius r around y. Scale up so that $y + rB_n$ is in isotropic position and call it Q (just for convenience; this has the effect of scaling up R also). Now consider $Q' = P \cap y + 2^{\frac{1}{n}} rB_n$, i.e., K_2 intersected with the ball of radius $2^{1/n}r$ around y. The volume of Q' is at most twice the volume of Q and so a uniform sample from Q provides a warm start for sampling from Q'. Sample from Q', drawing each sample in $poly(n)$ time and use the samples to calculate an affine transformation that would put Q' in isotropic position. Apply the transformation to P. Reset Q to be the transformed Q', y to be its center of gravity and r to be the radius of the ball around y containing Q. Again let Q' be the intersection of P with the ball of radius $2^{1/n}r$ around y. After at most $O(n \log \frac{R}{r})$ such phases, P will be in near-isotropic position. □

5 Extensions

The methods of this paper suggest the following *interior point* algorithm for optimization. Suppose we want to maximize $c^T x$ over a full-dimensional convex set K. We start with a feasible solution z_1 in K. Then add the constraint $c^T x \geq c^T z_1$ and let $K := K \cap \{x \mid c^T x \geq c^T z_1\}$. Next we generate

[2]It is not known whether Vaidya's algorithm can be extended to solve this problem.

N random points y_i in the set K and let z_2 be their average. If $c^T z_2 - c^T z_1 < \epsilon$ we stop; otherwise we set $K := K \cap \{x \mid c^T x > c^T z_2\}$, and continue.

The algorithms seems to merit empirical evaluation; in practice, it might be possible to sample more efficiently than the known worst-case bounds. In any case, the algorithm presents a strong motivation to find geometric random walks that mix faster.

Acknowledgement. We thank Adam Kalai, Ravi Kannan, Laci Lovász and Mike Sipser for useful discussions and their encouragement.

References

[1] D. Bertsimas and J. Tsitsiklis, Introduction to Linear Optimization. *Athena Scientific*, 1997.

[2] J. Bourgain, "Random points in isotropic convex sets," Convex geometric analysis, 53–58, *Math. Sci. Res. Inst. Publ.*, 34, Cambridge Univ. Press, Cambridge, 1999.

[3] M. Dyer, A. Frieze and R. Kannan. "A random polynomial time algorithm for estimating the volumes of convex bodies," *Journal of the ACM*, 38, 1991.

[4] M. Grötschel, L. Lovász, and A. Schrijver, "The ellipsoid method and its consequences in combinatorial optimization," *Combinatorica* 1, 169–197, 1981.

[5] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Springer, 1988.

[6] B. Grunbaum. "Partitions of mass-distributions and convex bodies by hyperplanes," *Pacific J. Math.* 10, 1257–1261, 1960.

[7] M. Jerrum, A. Sinclair, E. Vigoda "A polynomial-time approximation algorithm for the permanent of a matrix with non-negative entries," Proc. STOC, 712-721, 2001.

[8] A. Kalai and S. Vempala, "Efficient algorithms for universal portfolios," *Proc. 41st Annual IEEE Symp. on the Foundations of Computer Science*, 486–491, 2000.

[9] R. Kannan and L. Lovász, "Faster mixing via average conductance," Proc. STOC, 1995.

[10] R. Kannan, L. Lovász and M. Simonovits. "Isoperimetric problems for convex bodies and a localization lemma," *Discrete Computational Geometry* 13, 541–559, 1995.

[11] R. Kannan, L. Lovász and M. Simonovits. "Random walks and and an $O^*(n^5)$ volume algorithm for convex sets," *Random Structures and Algorithms* 11, 1–50, 1997.

[12] R. Kannan, J. Mount, S. Tayur, "A randomized algorithm to optimize over certain convex sets," Math. Oper. Res. 20, 3, 529–549, 1995.

[13] R. M. Karp and C. H. Papadimitriou, "On linear characterization of combinatorial optimization problems," Siam J. Comp., 11, 620–632, 1982.

[14] L. G. Khachiyan, "A polynomial algorithm in linear programming," (in Russian), *Doklady Akedamii Nauk SSSR*, 244, 1093–1096, 1979 (English translation: *Soviet Mathematics Doklady*, 20, 191–194, 1979).

[15] A. Yu. Levin, "On an algorithm for the minimization of convex functions," (in Russian), *Doklady Akedemii Nauk SSSR*, 160, 1244–1247, 1965 (English translation: *Soviet Mathematics Doklady*, 6, 286-290, 1965).

[16] L. Lovász, "Hit-and-run mixes fast," *Mathematical Programming* 86, 443-461, 1998.

[17] L. Lovász and M. Simonovits, "Random walks in convex bodies and an improved volume algorithm," *Random Structures and Algorithms* 4, 259-412, 1993.

[18] M. W. Padberg and M. R. Rao, "The Russian method for linear programming III: Bounded integer programming," Research Report 81-39, New York University, Graduate School of Business Administration, New York, 1981.

[19] A. Prekopa, "On logarithmic concave measures and functions," *Acta Sci. Math. Szeged*, 34, 335-343, 1973.

[20] M. Rudelson, "Random vectors in the isotropic position," J. Funct. Anal. 164, 60–72, 1999.

[21] P. M. Vaidya, "A new algorithm for minimizing convex functions over convex sets," *Mathematical Programming*, 291–341, 1996.

[22] D. B. Yudin and A. S. Nemirovski, "Evaluation of the information complexity of mathematical programming problems," (in Russian), Ekonomika i Matematicheskie Metody 12, 128-142, 1976 (English translation: Matekon 13, 2, 3-45, 1976).