MIT 6.7220/15.084 — Nonlinear Optimization Thu, Feb 15th 2024

Lecture 4A
Feasibility, optimization, and separation

Instructor: Prof. Gabriele Farina (✉ gfarina@mit.edu)*

In Lecture 1, we discussed how nonlinear optimization problems can be generally computationally intractable. In Lecture 3, we introduced a class of functions—called *convex* functions—for which first-order optimality conditions are both necessary and sufficient (assuming a convex feasible set Ω). In this lecture, we continue our study of convex optimization, showing that *in many cases* the solution to a convex optimization problem

$$\begin{aligned} \min_x \quad & f(x) \text{ convex} \\ \text{s.t.} \quad & x \in \Omega \text{ convex} \end{aligned}$$

can be found in polynomial time. However, it is also important to keep in mind that *not all convex optimization problems* can be solved in polynomial time. For example, optimization over the *copositive cone* (a convex set) is known to be intractable; we will talk more about that in a later class.

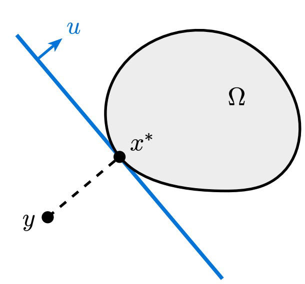
1 Separating a point from a closed convex set

An important property of any convex set Ω is that whenever a point y is not in Ω, then we can *separate* y from Ω using a *hyperplane*. In other words, *flat* separation surfaces are enough for certifying that a point $y \notin \Omega$.

Theorem 1.1. Let $\Omega \subseteq \mathbb{R}^n$ be a nonempty, closed, and convex set, and let $y \in \mathbb{R}^n$ be a point. If $y \notin \Omega$, then there exist $u \in \mathbb{R}^n, v \in \mathbb{R}$ such that

$$\langle u, y \rangle < v, \qquad \text{and} \qquad \langle u, x \rangle \geq v \quad \forall x \in \Omega.$$

Proof. The proof of the result rests on a very simple idea: the direction of the halfspace will be made orthogonal to the line that connects y to its projection x^* onto Ω, and the halfspace boundary will be set so that it passes through x^*. We now make the argument formal.



First, since Ω is nonempty and closed, a Euclidean projection x^* of y onto Ω exists,[1] as we discussed in Lecture 1. In other words, the nonlinear optimization problem

$$\begin{aligned} \min_x \quad & \tfrac{1}{2}\|x - y\|_2^2 \\ \text{s.t.} \quad & x \in \Omega \end{aligned}$$

*These notes are class material that has not undergone formal peer review. The TAs and I are grateful for any reports of typos.

must have at least a solution $x^* \in \Omega$. Furthermore, since the objective function is differentiable and Ω is convex, from the first-order optimality conditions (see Lecture 2) we know that

$$\langle x^* - y, x - x^* \rangle \geq 0 \quad \forall x \in \Omega. \tag{1}$$

Let now

$$\begin{aligned} u &:= x^* - y, && [\triangleright \text{ this is the direction that connects } y \text{ to } x^*] \\ \text{and } v &:= \langle u, x^* \rangle. && [\triangleright \text{ so that the halfspace boundary passes through } x^*] \end{aligned}$$

Note that $u \neq 0$, since $x^* \in \Omega$ but $y \notin \Omega$. So, $\|u\| > 0$ and therefore

$$\langle u, y \rangle = \langle u, x^* - u \rangle = v - \|u\|_2^2 < v.$$

Thus, to complete the proof, we now need to show that $\langle u, x \rangle \geq v$ for all $x \in \Omega$. But this is exactly what (1) guarantees, since $u = x^* - y$ and $v = \langle u, x^* \rangle$ by definition. □

The result above might not seem like much. After all, the proof is pretty straightforward, and the geometric intuition strong enough that one might be tempted to just take it for granted. Instead, the consequences of the result are deep, far-reaching, and intimately tied to some of the most significant breakthroughs in mathematical optimization theory.

1.1 Separation oracles

The result established in Theorem 1.1 justifies the following definition.

Definition 1.1 ((Strong) separation oracle). Let $\Omega \subseteq \mathbb{R}^n$ be convex and compact. A *strong separation oracle* for Ω is an algorithm that, given any point $y \in \mathbb{R}^n$, correctly outputs one of the following:

- "$y \in \Omega$", or
- "$(y \notin \Omega, u, v)$", where the pair $(u, v) \in \mathbb{R}^n \times \mathbb{R}$ is such that

$$\langle u, y \rangle < v, \quad \text{and} \quad \langle u, x \rangle \geq v \quad \forall x \in \Omega.$$

1.2 Finding separating hyperplanes in practice

Theorem 1.1 guarantees the *existence* of a separating hyperplane. In many problems of interest, *constructing* a separation oracle is simple.

Example 1.1 (Separation oracle for a convex polytope). Let Ω be a convex polytope, that is, the convex set defined by the intersection of a finite number of halfspaces (linear inequalities)

$$\Omega := \{x \in \mathbb{R}^n : Ax \leq b\}, \quad \text{where} \quad A = \begin{pmatrix} — \ a_1^\top \ — \\ \vdots \\ — \ a_m^\top \ — \end{pmatrix} \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m.$$

Then, given a point $y \in \mathbb{R}^m$, we can implement a separation oracle as follows:

- if $Ay \leq b$, return "$y \in \Omega$";

[1]In fact, it is easy to prove that the projection is unique (see Homework 1, Exercise 4). However, we do not need uniqueness for the argument that follows.

- else, at least one of the inequalities $a_j^\top y \le b_j$, $j \in \{1, ..., m\}$ is violated. In other words, there exists j such that $a_j^\top y > b_j$, while by definition of Ω, $a_j^\top x \le b_j$ for all x. This shows that the response "$(y \notin \Omega, -a_j, -b_j)$" is a valid response.

Remark 1.1. Example 1.1 shows that whenever we have a finite number m of inequalities, a separation oracle for the polytope defined by those inequalities can be implemented in time that depends linearly on m and the dimension of the embedding space. This result establishes a *blanket* guarantee, but in some cases, one can do better: depending on the structure of the inequalities, sometimes one can get away with sublinear complexity in m. In some cases, one might be able to construct an efficient separation oracle even for polytopes that have an infinite number of inequalities!

We proceed with another classic example of a feasible set that admits a simple separation oracle.

Example 1.2 (Separation oracle for the semidefinite cone). Let $\Omega = \{M \in \mathbb{R}^{n\times n} : M \succcurlyeq 0\}$ be the set of semidefinite matrices, that is, all symmetric matrices such that $v^\top M v \ge 0$ for all $v \in \mathbb{R}^n$—or, equivalently, such that all of M's eigenvalues are nonnegative.

Then, given a point $Y \in \mathbb{R}^{n\times n}$, we can implement a separation oracle as follows:
- if Y is *not* symmetric, then there exist $i, j \in \{1, ..., n\}$ such that $Y_{ij} < Y_{ji}$; return "$(Y \notin \Omega, E_{ij} - E_{ji}, 0)$", where E_{ij} is the matrix of all zeros, except in position i, j where it has a 1.
- else, if Y is symmetric, we can compute all of its eigenvalues and eigenvectors. If one eigenvalue is negative, then the corresponding eigenvector w must be such that $w^\top Y w = \langle Y, ww^\top \rangle < 0$. Hence, return "$(Y \notin \Omega, ww^\top, 0)$".
- otherwise, return "$Y \in \Omega$".

As we show next, a fundamental result in optimization theory reveals that under mild hypotheses, if the feasible set admits an efficient separation oracle and the objective function is convex, then the solution can be computed efficiently.

2 Optimization via separation

In a major breakthrough in mathematical optimization, Khachiyan, L. G. [Kha80] proposed a polynomial-time algorithm for using separation oracles to find the minimum of a linear function. The algorithm, which goes under the name of *ellipsoid method* is actually more general, and applies to general convex objectives on sets for which separation oracles are available. The result builds on top of previous work by Šor, N. Z. [Šor77] and Yudin, D. B., & Nemirovskii, A. S. [YN76].

In particular, Khachiyan's result was the first to show that linear programming problems can be solved in polynomial time. This was an unexpected result at the time, and in fact, the complexity of linear programming solvers was conjectured to be *not* polynomial (more on this in the next section). The result of Khachiyan stirred so much enthusiasm in the research community that the New York Times even advertised it on its first page.

Despite the enthusiasm, the ellipsoid method turned out to be very impractical. Still, it is a great theoretical idea, and its consequences are pervasive.

"All the News That's Fit to Print"

The New York Times

LATE CITY EDITION

VOL. CXXIX . . No. 44,394 — NEW YORK, WEDNESDAY, NOVEMBER 7, 1979 — 25 CENTS

IRAN'S CIVIL GOVERNMENT OUT; HOSTAGES FACE DEATH THREAT; OIL EXPORTS BELIEVED HALTED

STUDENTS WARN U.S.

Similar Transportation Bond Issues Are Passed in New York and Jersey

CARTER SAID TO ASK FOR LOYALTY CHECK OF AIDES ON POLITICS

F.A.A. Seeks Fine of $1.5 Million For 'Unairworthy' Braniff Flights

Main Iran Oil Port Reported Closed And Prices of Spot Petroleum Soar

Interior Dept. Assailed On Missing a Deadline For Species Protection

Energy Dept. Accuses 9 Refiners Of $1.18 Billion Oil Overcharges

President Carter after announcing that Robert S. Strauss, right, would head his campaign. At left are Secretary of State Cyrus R. Vance and Sol M. Linowitz, who is succeeding Mr. Strauss as Mideast negotiator.

A Soviet Discovery Rocks World of Mathematics

By MALCOLM W. BROWNE

A surprise discovery by an obscure Soviet mathematician has rocked the world of mathematics and computer analysis, and experts have begun exploring its practical applications.

Mathematicians describe the discovery by L.G. Khachian as a method by which computers can find guaranteed solutions to a class of very difficult problems that have hitherto been tackled on a kind of hit-or-miss basis.

Apart from its profound theoretical interest, the discovery may be applicable in weather prediction, complicated industrial processes, petroleum refining, the scheduling of workers at large factories, secret codes and many other things.

"I have been deluged with calls from virtually every department of government for an interpretation of the significance of this," a leading expert on computer methods, Dr. George B. Dantzig of Stanford University, said in an interview.

The solution of mathematical problems by computer must be broken down into a series of steps. One class of problem sometimes involves so many steps that it could take billions of years to compute.

The Russian discovery offers a way by which the number of steps in a solution can be dramatically reduced. It also offers the mathematician a way of learning quickly whether a problem has a solution or not, without having to complete the entire immense computation that may be required.

According to the American journal Sci-

Continued on Page A20, Column 3

Figure 1: https://timesmachine.nytimes.com/timesmachine/1979/11/07/issue.html

2.1 The intuition behind the ellipsoid method

Formalizing the details of the ellipsoid method is rather complex. A major source of difficulty is the fact that the algorithm needs to approximate square roots using fractions to be implementable on a finite-precision machine, and that causes all sorts of tricky analyses that the approximation error can indeed be kept under control. These details are certainly important, but are notoriously tedious, and fundamentally they are just that, *details*. If you are curious to read a formal account, I recommend the authoritative book by Grötschel, M., Lovász, L., & Schrijver, A. [GLS93]. For this lecture, we just focus on the *idea* behind the ellipsoid method.

The idea behind the ellipsoid method is rather elegant. At its core, it is a generalization of *binary search* from one dimension to multiple dimensions. At every iteration of the algorithm, the space is "cut" by using a separating hyperplane.

Feasibility. To build intuition, ignore for now the objective function, and consider the following problem: given a separation oracle for Ω (closed and convex), either find $x \in \Omega$, or determine that Ω is empty. You are given two radiuses:

- the radius $R > 0$ guarantees that if Ω is not empty, then $\Omega \cap \mathbb{B}_R(0) \neq \emptyset$;
- the radius $r > 0$ guarantees that if Ω is not empty, then it contains a ball of radius r in its interior.

If this problem were one-dimensional, then Ω would be either empty or an interval, and a separation oracle would be an algorithm that, given any $y \in \mathbb{R}$, would return whether $y \in \Omega$, or one of the statements "y is too small" / "y is too large". Solving the problem now appears easy: start from the interval $[-R, R]$, and perform a binary search using the separation oracle to guide the search. Once the size of the search interval drops below r, we know that Ω is empty.

The ellipsoid method generalizes this idea to multiple dimensions. At every iteration, it keeps track of a "search space" (the generalization of the search interval above). Then, it queries the separation oracle for the center of this search space. If the point does not belong to Ω, and the separation oracle returns the separating hyperplane $\langle u, x\rangle = v$, then the search space is cut by considering now only the subset of the search space that intersects $\{x \in \mathbb{R}^n : \langle u, x\rangle \geq v\}$. The process continues until the volume of the search space becomes smaller than the radius r. The reason why this method is called the "ellipsoid method" is that the search space in the multi-dimensional case is not kept in the form of an interval, but rather as an ellipsoid. This is mostly for computational reasons, since we need to have an internal way of representing the search domain that is convenient to use.

Incorporating the objective. The above idea can be extended to incorporate an objective function $f(x)$. To do that, we will need to start cutting not only the search ellipsoid, but also the feasible set to make sure we end up at the optimum. In other words, you can think of this extended ellipsoid method as having "two modes": while it has not found a feasible point in Ω, it cuts the search ellipsoid; then, once feasible points are found, it cuts the feasible set to exclude all values above the current value.

- Initialize at time $t = 1$ with the starting point $y_1 \coloneqq 0 \in \mathbb{R}^n$, starting ellipsoid $\mathcal{E}_1 \coloneqq \mathbb{B}_R(0)$, and starting feasible set $\Omega_1 \coloneqq \Omega$.
- At each time t, we ask a separation oracle for Ω_t whether the center $c_t \in \mathbb{R}^n$ of the search ellipsoid $\mathcal{E}_t$ belongs to Ω_t or not.[2] There are only two cases:
  - If the center c_t is *not* feasible, then set $\Omega_{t+1} \coloneqq \Omega_t$, and cut the search space by setting $\mathcal{E}_{t+1}$ to an ellipsoid that contains the intersection between $\mathcal{E}_t$ and the halfspace containing Ω_t returned by the separation oracle.
  - If the center c_t is feasible, then we know for sure that all points $x \in \Omega_t$ such that $\langle \nabla f(c_t), x - c_t\rangle \geq 0$ are such that $f(x) \geq f(c_t)$. This follows trivially from the linear lower bound property of convex functions (Theorem 1.1 of Lecture 3):

$$\langle \nabla f(c_t), x - c_t \rangle \geq 0 \qquad \Longrightarrow \qquad f(x) \geq f(c_t) + \langle \nabla f(c_t), x - c_t \rangle \geq f(c_t).$$

Hence, we can cut *both* the search ellipsoid $\mathcal{E}_t$ and the feasible set Ω_t by considering their intersection with the halfspace $H_t \coloneqq \{x \in \mathbb{R}^n : \langle \nabla f(c_t), x - c_t \rangle \leq 0\}$. In particular, we set $\Omega_{t+1} \coloneqq \Omega_t \cap H_t$, and set $\mathcal{E}_{t+1}$ to a smaller ellipsoid that contains $\mathcal{E}_t \cap H_t$.

- Finally, after the volume of the search ellipsoid has gotten sufficiently small (this happens after $T = O(n^2)\log(R/r)$ iterations), we output the following:
  - If we never encountered a center c_t that was feasible, then we report that Ω was infeasible.
  - Else, we output the c_t that minimizes f, out of those that were feasible.

Assuming that we can ignore all sorts of tedious rounding issues, the following guarantee can be shown [Gup20].

Theorem 2.1. Let R and r be as above, and let the range of the function f on Ω be bounded by $[-B, B]$. Then, the ellipsoid method described above run for $T \geq 2n^2 \log(R/r)$ steps either correctly reports that $\Omega = \emptyset$, or produces a point x^* such that

$$f(x^*) \leq f(x) + \frac{2BR}{r} \exp\left(-\frac{T}{2n(n+1)}\right) \qquad \forall x \in \Omega.$$

2.2 Takeaway message: Separation implies optimization

If you squint your eyes, what the ellipsoid method proves constructively is the following: if we know how to construct a separation oracle for a set Ω, then we can optimize over Ω. Of course, this is a bit of a simplification (and there are all sorts of little conditions here and there as we have seen above), but nonetheless it is a good first approximation of the general message.

The opposite direction is also known to be true, even when by "optimization" we simply mean optimization of linear objective functions.

Further readings and bibliography

If you want to read more about the ellipsoid method, the book by Grötschel, M., Lovász, L., & Schrijver, A. [GLS93] is a standard and accessible reference on the topic. The bound on the approximation error incurred by the ellipsoid method was taken from Gupta, A. [Gup20].

[Kha80] L. G. Khachiyan, "Polynomial algorithms in linear programming," *USSR Computational Mathematics and Mathematical Physics*, vol. 20, no. 1, pp. 53–72, 1980.

[Šor77] N. Z. Šor, "Cut-off method with space extension in convex programming problems," *Cybernetics*, vol. 13, no. 1, pp. 94–96, 1977.

[YN76] D. B. Yudin and A. S. Nemirovskii, "Informational complexity and efficient methods for the solution of convex extremal problems," *Matekon*, vol. 13, no. 2, pp. 22–45, 1976.

[GLS93] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*. Berlin, Germany: Springer, 1993. [Online]. Available: `https://link.springer.com/book/10.1007/978-3-642-78240-4`

[2] There is a caveat here: technically, we are assuming as given a separation oracle for Ω, *not* Ω_t. Yet, because Ω_t is obtained from Ω by intersecting with halfspaces, it is easy to see that one separation oracle for Ω_t can be constructed efficiently starting from that for Ω and the description of the intersected hyperplanes. Try working out the details!

[Gup20] A. Gupta, "The Centroid and Ellipsoid Algorithms." [Online]. Available: https://www.cs.cmu.edu/~15850/notes/lec21.pdf