

16.410/413
Principles of Autonomy and Decision Making
Lecture 17: The Simplex Method

Emilio Frazzoli

Aeronautics and Astronautics
Massachusetts Institute of Technology

November 10, 2010

Assignments

Readings

- [IOR] Chapter 4.

Outline

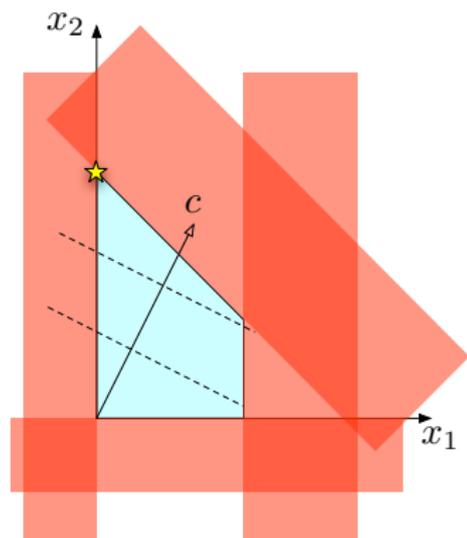
- 1 Geometric Interpretation
 - Geometric Interpretation
- 2 Algebraic Procedure
- 3 Intro to interior point methods

Geometric Interpretation

- Consider the following simple LP:

$$\begin{aligned} \max \quad & z = x_1 + 2x_2 = (1, 2) \cdot (x_1, x_2), \\ \text{s.t.} \quad & x_1 \leq 3, \\ & x_1 + x_2 \leq 5, \\ & x_1, x_2 \geq 0. \end{aligned}$$

- Each inequality constraint defines a hyperplane, and a **feasible** half-space.
- The intersection of all feasible half spaces is called the **feasible region**.
- The feasible region is a (possibly **unbounded**) polyhedron.
- The feasible region could be the empty set: in such case the problem is said **unfeasible**.



Geometric Interpretation (2)

- Consider the following simple LP:

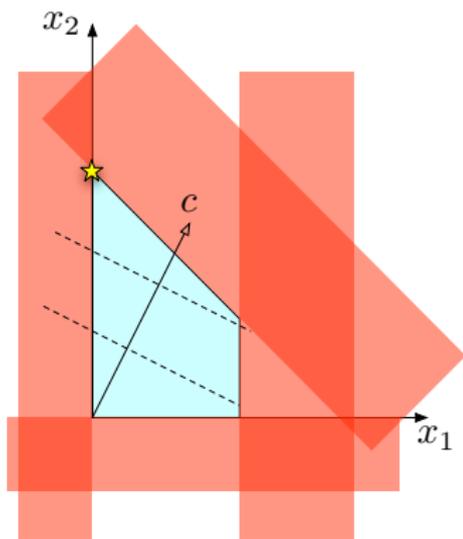
$$\max \quad z = x_1 + 2x_2 = (1, 2) \cdot (x_1, x_2),$$

$$\text{s.t.} \quad x_1 \leq 3,$$

$$x_1 + x_2 \leq 5,$$

$$x_1, x_2 \geq 0.$$

- The “ c ” vector defines the gradient of the cost.
- Constant-cost loci are planes normal to c .
- Most often, the optimal point is located at a vertex (corner) of the feasible region.
 - If there is a single optimum, it must be a corner of the feasible region.
 - If there are more than one, two of them must be adjacent corners.
 - If a corner does not have any adjacent corner that provides a better solution, then that corner is in fact the optimum.



A naïve algorithm (1)

- Recall the standard form:

$$\begin{array}{ll} \min & z = c^T x \\ \text{s.t.} & Ax = b, \\ & x \geq 0. \end{array}$$

- Corners of the feasible regions (also called **basic feasible solutions**) are solutions of $Ax = b$ (m equations in n unknowns, $n > m$), obtained setting $n - m$ variables to zero, and solving for the others (basic variables), ensuring that all variables are non-negative.

A naïve algorithm (1)

- Recall the standard form:

$$\begin{array}{ll} \min & z = c^T x \\ \text{s.t.:} & Ax = b, \\ & x \geq 0. \end{array} \quad \left(\text{or, really:} \quad \begin{array}{ll} \min & z = c_y^T y + c_s^T s \\ \text{s.t.:} & A_y y + I s = b, \\ & y, s \geq 0. \end{array} \right)$$

- Corners of the feasible regions (also called **basic feasible solutions**) are solutions of $Ax = b$ (m equations in n unknowns, $n > m$), obtained setting $n - m$ variables to zero, and solving for the others (basic variables), ensuring that all variables are non-negative.
- This amounts to:
 - picking n_y inequality constraints, (notice that $n = n_y + n_s = n_y + m$).
 - making them active (or binding),
 - finding the (unique) point where all these hyperplanes meet.
 - If all the variables are non-negative, this point is in fact a vertex of the feasible region.

A naïve algorithm (2)

- One could possibly generate all basic feasible solutions, and then check the value of the cost function, finding the optimum by enumeration.
- **Problem:** how many candidates?

$$\binom{n}{n-m} = \frac{n!}{m!(n-m)!}.$$

- for a “small” problem with $n = 10$, $m = 3$, we get 120 candidates.
- this number grows very quickly, the typical size of realistic LPs is such that n, m are often in the range of several hundreds, or even thousands.

Duality polyhedra—graphs

- Each vertex has at least n neighbors.
- One can draw a graph of neighboring vertices.
- Ideally, one could search this graph for the vertex that maximizes the objective.
- Remember that if a corner does not have any adjacent corner that provides a better solution, then that corner is in fact the optimum.
- Use an informed (“greedy”) algorithm to search for the max.

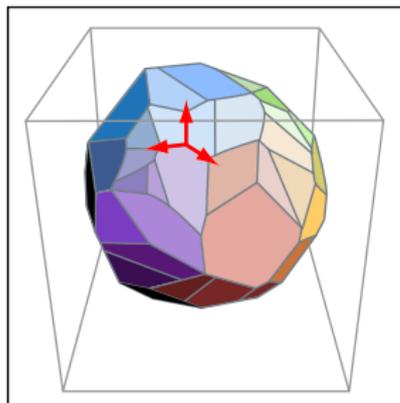
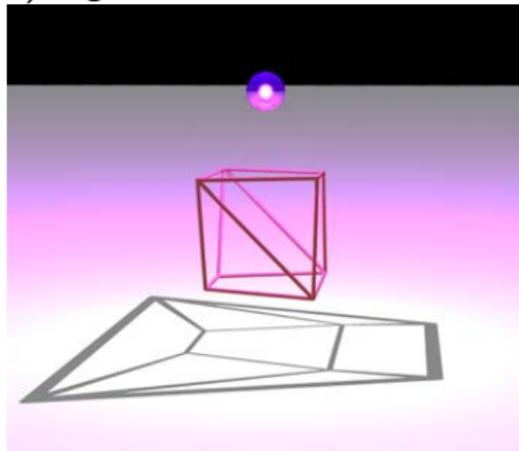


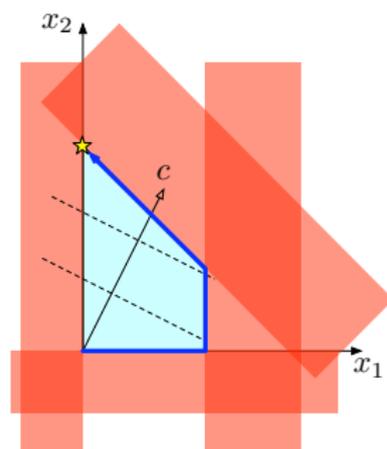
Image by MIT OpenCourseWare.



© David Eppstein. CC by-sa.
This content is excluded
from our Creative Commons
license. For more information,
<http://ocw.mit.edu/fairuse>.

Idea of the Simplex algorithm

- Start at a corner of the feasible region, e.g., the origin.
- While there is an adjacent corner that is a better solution, move to that corner.
- For “most” instances, the algorithm terminates (in a finite number of steps) at an optimal solution.



Finding adjacent vertices

- A vertex is identified by $n - m$ variables set to zero. The other variables can be computed uniquely; these are called **basic variables**.
- Two vertices are adjacent if they share all their (non-)basic variables, except one.
- So: in order to find neighbors to a vertex, remove one of the (non-)basic variables and add another one.

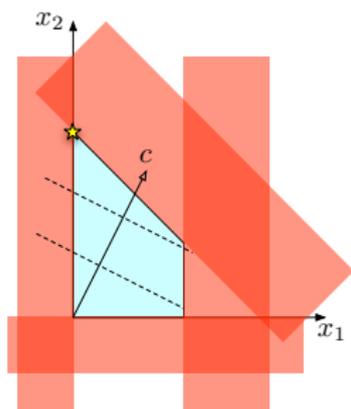
Finding adjacent vertices

- A vertex is identified by $n - m$ variables set to zero. The other variables can be computed uniquely; these are called **basic variables**.
- Non-basic variables correspond to the n_y inequality constraints chosen to be active.
- Two vertices are adjacent if they share all their (non-)basic variables, except one.
- Two vertices are adjacent if they activate the same constraints, except one.
- So: in order to find neighbors to a vertex, remove one of the (non-)basic variables and add another one.
- Make one of the active constraints inactive, and activate a formerly inactive one.

A simple example

- Consider the usual simple LP, in standard form:

$$\begin{aligned} \max \quad & z = x_1 + 2x_2 = (1, 2) \cdot (x_1, x_2), \\ \text{s.t.} \quad & x_1 + x_3 = 3, \\ & x_1 + x_2 + x_4 = 5, \\ & x_1, x_2, x_3, x_4 \geq 0. \end{aligned}$$



- Choose the origin as the initial corner/vertex. The basic variables coincide with the slack variables:

$$x_1 = x_2 = 0 \Rightarrow x_3 = 3, x_4 = 5 \Rightarrow z = 0.$$

- We have two choices to choose a neighbor:

- Shall we make x_1 or x_2 basic?

The cost depends more heavily on x_2 , so let's choose this one.

- Shall we make x_3 or x_4 non-basic?

If we increase x_2 , x_4 will get to zero first (in fact, x_3 will not change at all), so choose x_4 .

$$x_1 = x_4 = 0 \Rightarrow x_2 = 5, x_3 = 3 \Rightarrow z = 10.$$

Outline

1 Geometric Interpretation

2 Algebraic Procedure

- The tableau
- Algorithm
- Degeneracy
- Finding a feasible initial solution
- Analysis

3 Intro to interior point methods

The simplex method: tableau

- Consider the LP

$$\begin{aligned} \max \quad & z = 3x_1 + 5x_2 \\ \text{s.t.} \quad & x_1 \leq 4 \\ & 2x_2 \leq 12 \\ & 3x_1 + 2x_2 \leq 18 \\ & x_1, x_2 \geq 0. \end{aligned}$$

In standard form:

$$\begin{aligned} \max z, \quad & z - 3x_1 - 5x_2 = 0 \\ \text{s.t.} \quad & x_1 + x_3 = 4 \\ & 2x_2 + x_4 = 12 \\ & 3x_1 + 2x_2 + x_5 = 18 \\ & x_1, x_2, x_3, x_4, x_5 \geq 0. \end{aligned}$$

In tableau form:

	x_1	x_2	x_3	x_4	x_5	b
z	-3	-5				0
r_1	1		1			4
r_2		2		1		12
r_3	3	2			1	18

Simplex method by example

- As a first corner, let's choose the origin for the original variables, and compute the corresponding values for the slack variables.
- We get easily that $x_3 = 4$, $x_4 = 12$, and $x_5 = 18$.
- What is the “best next corner”?
(we need to set to zero one of the non-zero—or basic—variables, and release one of the variables previously set to zero)
- In order to figure out which variable should enter the basis:
 - Look at the objective function: $\max z, z - 3x_1 - 5x_2 = 0$.
 - Increasing x_2 is likely to increase z faster... let's go with that.
- To figure out which variable should leave the basis:
 - Release the constraints that limits the increase in x_2 the most.
 - Constraint 2 limits x_2 to $12/2=6$, constraint 3 limits x_2 to $18/2=9$: eliminate Constraint 2 (i.e., the slack variable x_4)

Simplex method by example

- Recalling the tableau form:
 - choose the column with the “most negative” entry in row zero (z).
 - choose the row that limits the most the increase in the corresponding variable.
(smallest positive ratio test.)
- Then **PIVOT** on the intersection, that is, clear everything in the yellow-shaded column, leaving only a 1 in the orange-shaded cell, by manipulating the matrix (without changing the solution of the equation!)

	x_1	x_2	x_3	x_4	x_5	b
z	-3	-5				0
r_1	1		1			4
r_2		2		1		12
r_3	3	2			1	18

	x_1	x_2	x_3	x_4	x_5	b
z	-3	0		$5/2$		30
r_1	1		1			4
r_2		1		$1/2$		6
r_3	3	0		-1	1	6

Simplex method by example (2)

- Now pick the x_1 column, and the r_3 row, and pivot...
- Note that now we cannot further increase z , since x_4 and x_5 must be non-negative. We have the optimal solution: just read it off the tableau (remember $x_4 = x_5 = 0$), we get: $x_1 = 2$, $x_2 = 6$, and $x_3 = 2$.
- The positive slack x_3 indicates that the first constraint is not active.

	x_1	x_2	x_3	x_4	x_5	b
z	-3	0		$5/2$		30
r_1	1		1			4
r_2		1		$1/2$		6
r_3	3	0		-1	1	6

	x_1	x_2	x_3	x_4	x_5	b
z	0	0		$3/2$	1	36
r_1	0		1	$1/3$	$-1/3$	2
r_2		1		$1/2$		6
r_3	1	0		$-1/3$	$1/3$	2

Simplex method: degeneracy

- Question: Do the above steps always result in a finite termination?
- Answer: Almost always yes, and terminate at the optimal solution.
- In some cases, (very rare in practice), the simplex algorithm as stated may result in infinite cycling. (two or more neighboring corners have the same cost)

	x_1	x_2	x_3	x_4	x_5	b
z				-8		20
r_1			1	$2/3$	$1/3$	4
r_2		1		-1		0
r_3	1	0		$-1/3$	$1/3$	2



	x_1	x_2	x_3	x_4	x_5	b
z		-8				20
r_1		$-2/3$	1	0	$1/3$	4
r_2		-1		1		0
r_3	1	$-1/3$		0	$1/3$	2

Simplex method: degeneracy

- Question: Do the above steps always result in a finite termination?
- Answer: Almost always yes, and terminate at the optimal solution.
- In some cases, (very rare in practice), the simplex algorithm as stated may result in infinite cycling. (two or more neighboring corners have the same cost)

	x_1	x_2	x_3	x_4	x_5	b
z				-8		20
r_1			1	$2/3$	$1/3$	4
r_2		1		-1		0
r_3	1	0		$-1/3$	$1/3$	2



	x_1	x_2	x_3	x_4	x_5	b
z		-8				20
r_1		$-2/3$	1	0	$1/3$	4
r_2		-1		1		0
r_3	1	$-1/3$		0	$1/3$	2

Degeneracy elimination methods

Charnes method: perturb the b vector $b \rightarrow b + [\varepsilon, \varepsilon^2, \dots]^T$.

Simplex method: finding a feasible initial solution

- The simplex method as presented previously assumed that the origin is a feasible solution (i.e., it satisfies all constraints).
- What if the origin is not feasible? We need to find a feasible solution.
- One possible (“big M”) approach is to
 - Introduce new artificial “constraint violation” variables to the constraints that are not satisfied at the origin.
 - Assign a very large penalty to such artificial variables.
 - Initialize the tableau to make the artificial variable basic (non-zero).
 - Solve the problem as usual.

Example: finding an initial feasible solution

- Consider the following LP:

$$\begin{aligned} \min \quad & z = 3x_1 + 5x_2 \\ \text{s.t.} \quad & x_1 \leq 4 \\ & 2x_2 \leq 12 \\ & 3x_1 + 2x_2 \geq 1 \\ & x_1, x_2 \geq 0. \end{aligned}$$

- In standard/tableau form:

$$\begin{aligned} \max z, \quad & z + 3x_1 + 5x_2 = 0 \\ \text{s.t.:} \quad & x_1 + x_3 = 4 \\ & 2x_2 + x_4 = 12 \\ & 3x_1 + 2x_2 - x_5 = 1 \\ & x_1, x_2, x_3, x_4, x_5 \geq 0. \end{aligned}$$

	x_1	x_2	x_3	x_4	x_5	b
z	3	5				0
r_1	1		1			4
r_2		2		1		12
r_3	3	2			-1	1

Example, cont'd

- Clearly, $(x_1, x_2) = (0, 0)$ is not a feasible solution (x_5 would be -1).
- So, modify the problem in this way (M stands for “very large number”):

$$\begin{aligned} \min \quad & z = 3x_1 + 5x_2 + Mx_6 \\ \text{s.t.:} \quad & x_1 \leq 4 \\ & 2x_2 \leq 12 \\ & 3x_1 + 2x_2 + x_6 \geq 1 \\ & x_1, x_2 \geq 0. \end{aligned}$$

- In standard/tableau form:

$$\begin{aligned} \max z, \quad & z + 3x_1 + 5x_2 + Mx_6 = 0 \\ \text{s.t.:} \quad & x_1 + x_3 = 4 \\ & 2x_2 + x_4 = 12 \\ & 3x_1 + 2x_2 - x_5 + x_6 = 1 \\ & x_1, x_2, x_3, x_4, x_5, x_6 \geq 0. \end{aligned}$$

	x_1	x_2	x_3	x_4	x_5	x_6	b
z	3	5				M	0
r_1	1		1				4
r_2		2		1			12
r_3	3	2			-1	1	1

Example, cont'd

- Since we want the artificial variable(s) to be non-zero—to ensure feasibility—we initialize the tableau to get 0's in the artificial variable columns in the objective function:

	x_1	x_2	x_3	x_4	x_5	x_6	b
z	3 -3M	5-2M			M		-M
r_1	1		1				4
r_2		2		1			12
r_3	3	2			-1	1	1

- Now proceed in the usual way—remembering that M stands for “very large number”

	x_1	x_2	x_3	x_4	x_5	x_6	b
z	3 -3M	5-2M			M		-M
r_1	1		1				4
r_2		2		1			12
r_3	3	2			-1	1	1

Example, cont'd

- Proceeding...

	x_1	x_2	x_3	x_4	x_5	x_6	b
z	0	3			1	$-1+M$	-1
r_1	0	$-2/3$	1		$1/3$	$-1/3$	4
r_2		2		1			12
r_3	1	$2/3$			$-1/3$	$1/3$	$1/3$

- We now have all positive elements in the objective row, and can read the solution:

$$x_1 = 1/3, \quad x_2 = 0, \quad x_3 = 4, \quad x_4 = 12, \quad x_5 = 0, \quad x_6 = 0$$

- The objective value is $z = -1$ (which corresponds to $z = 1$ in the original minimization problem).

Simplex method: summary

- Convert to standard form
- Pivot
 - choose most negative reduced cost (identifies new basic variable)
 - choose lowest positive ratio in ratio test
 - eliminate new basic variable from objective and all rows except lowest positive ratio in ratio test row
- Stop if optimal and read solution. Otherwise, pivot again.
- In some (degenerate) cases, the pivot step may not result in a strictly better solution, even though the solution is not optimal. In such cases (almost never occurring in practice) the simplex algorithm may not terminate.

There are methods to avoid infinite cycles.

- In the worst case, the simplex method might require visiting all feasible corners (exponential in n). In practice however, the method works very well.
- Recent advances (interior point methods) can solve LPs in polynomial time.

Simplex algorithm: analysis

- If there are no degenerate bases (true in random instances of LP), then it is impossible to return to a basis that has already been visited, because the objective function must decrease at every step.
- If there are degenerate bases, it is possible to cycle, which would mean that the same series of bases repeats as you pivot, so one could pivot forever.
- Smart rules about disambiguating ties in the lowest cost coefficient test and minimum positive ratio test can rule out possibility of infinite cycling
- Number of pivot steps \leq Number of corner point solutions

Simplex algorithm: analysis

- With n variables, each bounded in $[0, 1]$, there are 2^n corner point feasible solutions
- Pathological examples force examination of all of these
- Each pivot (Gaussian elimination) takes $O(mn)$ time
- Simplex algorithm is exponential in worst-case analysis, yet is remarkably successful in practice
- In fact, linear programming can be solved in polynomial time by interior point methods.
- Recently these have become competitive in practice with simplex.

Outline

- 1 Geometric Interpretation
- 2 Algebraic Procedure
- 3 Intro to interior point methods

Introduction to Interior Point methods

- Interior point methods solve **convex problems** of the form

$$\begin{aligned} \min \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x) \leq 0, \quad i = 1, \dots, m \\ & Ax = b, \end{aligned}$$

where the scalar functions f_0, f_1, \dots, f_m are convex and twice differentiable.

- A scalar function $f : \mathbb{R} \rightarrow \mathbb{R}$ is convex if the line between any two points on the graph of the function lies above (or on) the graph, or more formally if

$$\lambda f(x_1) + (1 - \lambda)f(x_2) \geq f(\lambda x_1 + (1 - \lambda)x_2)$$

for any $x_1, x_2 \in \mathbb{R}$, and $\lambda \in [0, 1]$.

The barrier interior-point method: main idea

- Convert the problem into an minimization with equality constraints by adding **barrier functions** in the following way:

$$\begin{aligned} \min \quad & f_0(x) - \frac{1}{t} \sum_{i=1}^m \log(-f_i(x)) \\ \text{s.t.} \quad & Ax = b, \end{aligned} \quad (*)$$

- Notice that the objective is only defined within the feasible set defined by the (strict) inequalities $f_i(x) < 0$.
- As $t \rightarrow +\infty$, the objective function is very close to f_0 , but diverges very steeply at the boundary of the feasible set.
- Idea: solve the problem iteratively, for $t = \mu^k t_0$, with $t_0 > 0$, $\mu > 1$.

The barrier interior-point method: physical interpretation

- Assume for simplicity that there are no equality constraints (or, alternatively, work directly in the subspace such that $Ax = b$).
- Imagine that each constraint generate an electrical field with potential $-\log(-f_i(x))$;
- A unit charge at position x will be subject to the (repulsive) forces

$$F_i(x) = \frac{1}{f_i(x)} \nabla f_i(x), \quad i = 1, \dots, m$$

- In addition to the repulsive forces from the constraints, the unit charge is “pushed” by an additional force

$$F_0(x) = -t \nabla f_0(x)$$

- For each t , the charge will come to rest at some point in the feasible region.
- As $t \rightarrow +\infty$, the charge will be pushed toward the optimal point.

The barrier interior-point method

Input: x strictly feasible (i.e., $f_i(x) < 0$, $i = 1, \dots, m$); $t > 0$; $\mu > 1$;
 $\epsilon > 0$.

repeat

$t \leftarrow \mu t$;

Let x^* be the solution of (\star) , computed using x as initial guess;

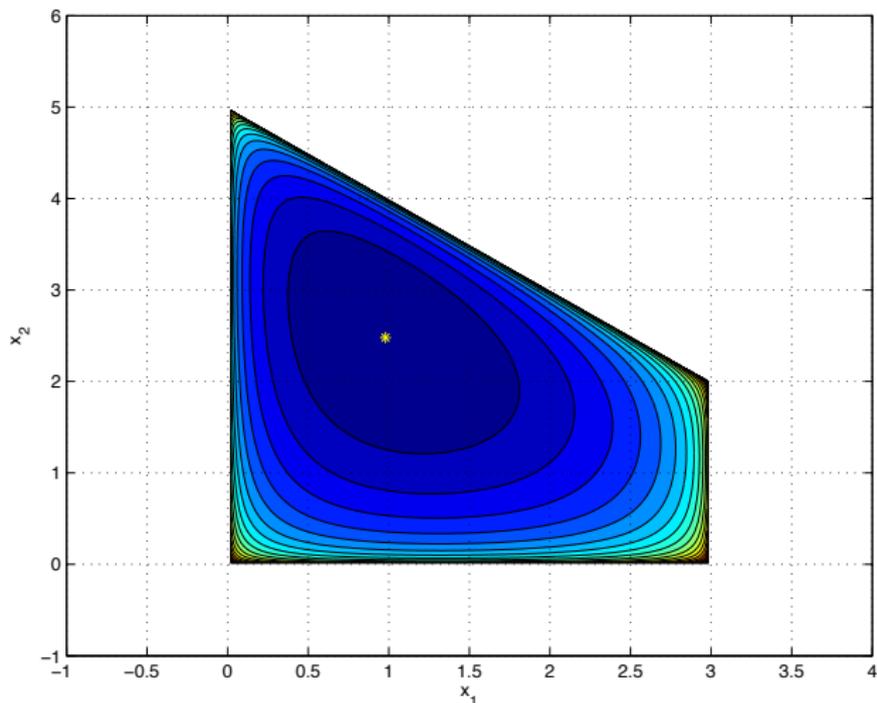
// e.g., use a gradient descent or Newton method)

$x \leftarrow x^*$;

until $t > m/\epsilon$;

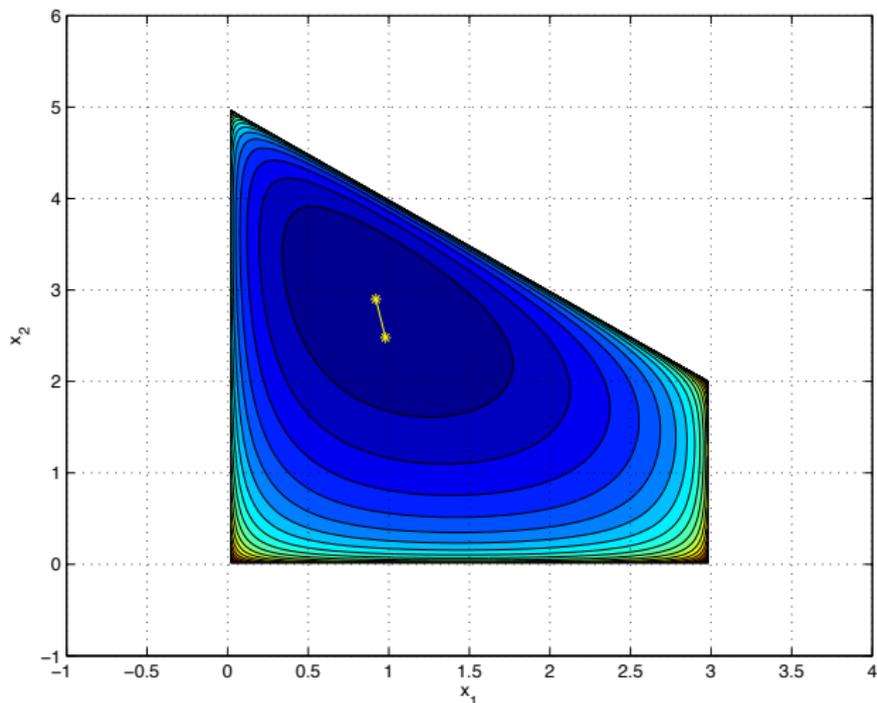
- The sequence of intermediate optimal solutions (i.e., the x^* 's) forms what is known as the **central path**, which is always contained in the interior of the feasible set.
- This algorithm terminates within $O(\sqrt{m})$ time, yielding a solution that is at most ϵ -suboptimal.

The barrier interior-point method: example



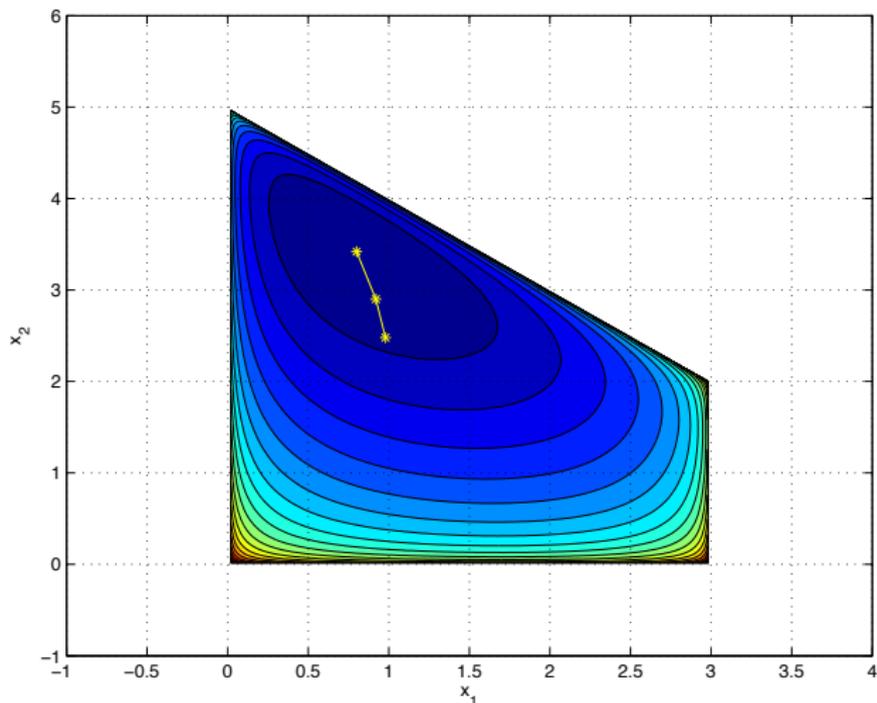
$$t = 1/8$$

The barrier interior-point method: example



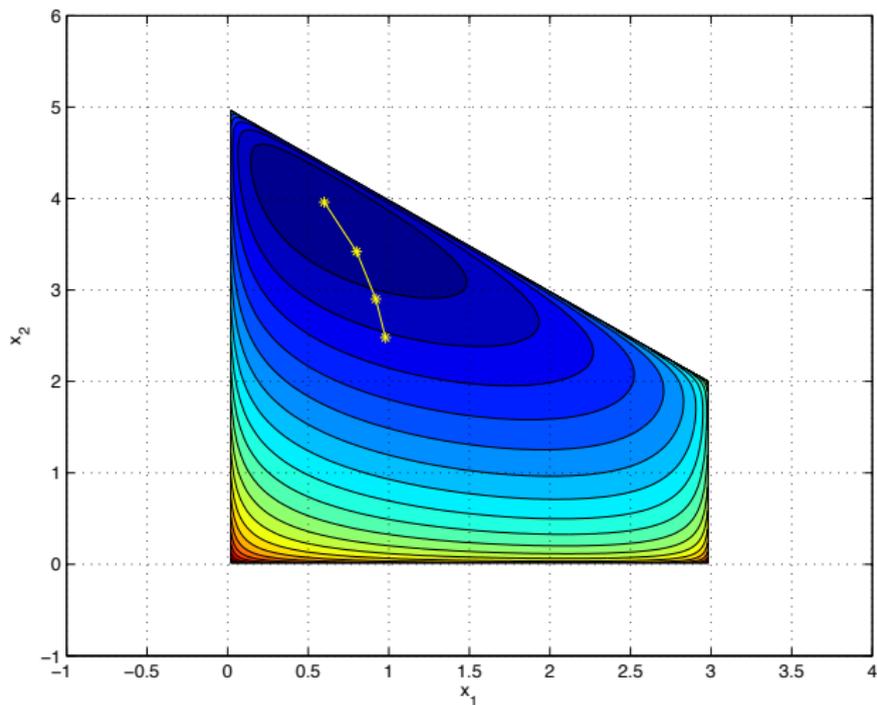
$$t = 1/4$$

The barrier interior-point method: example



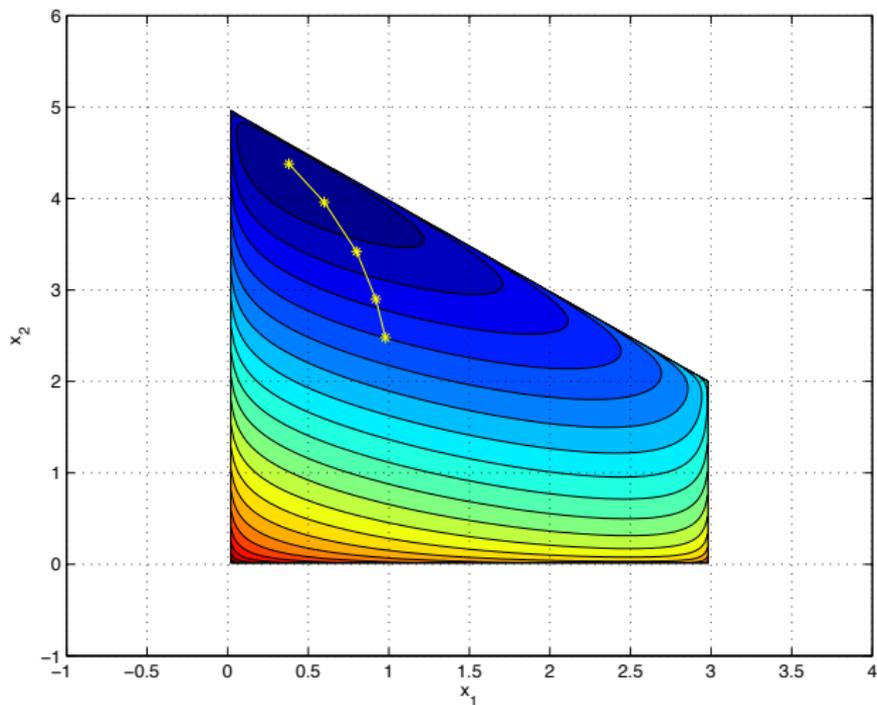
$$t = 1/2$$

The barrier interior-point method: example



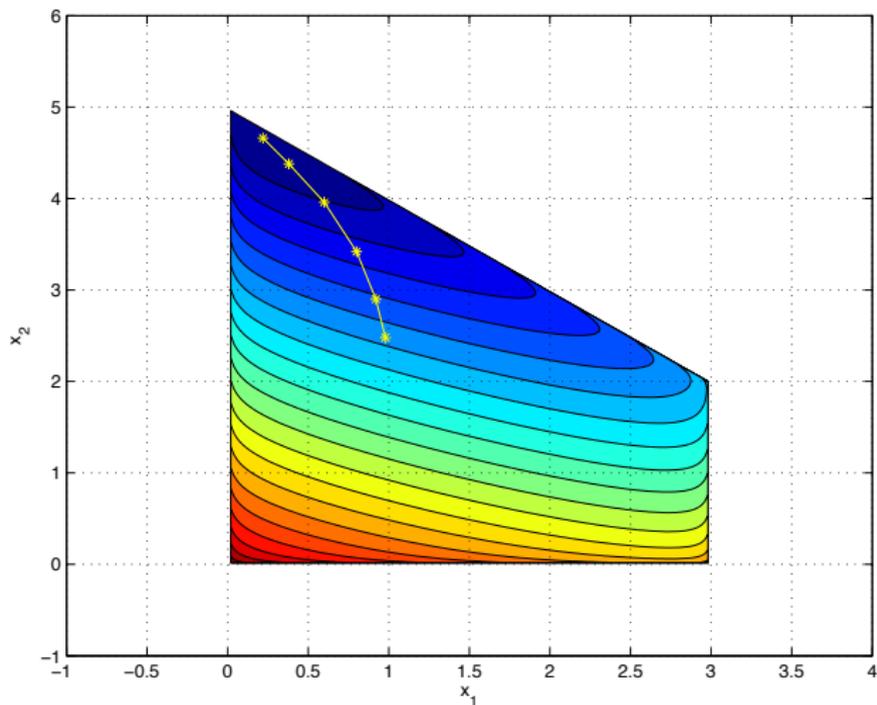
$t = 1$

The barrier interior-point method: example



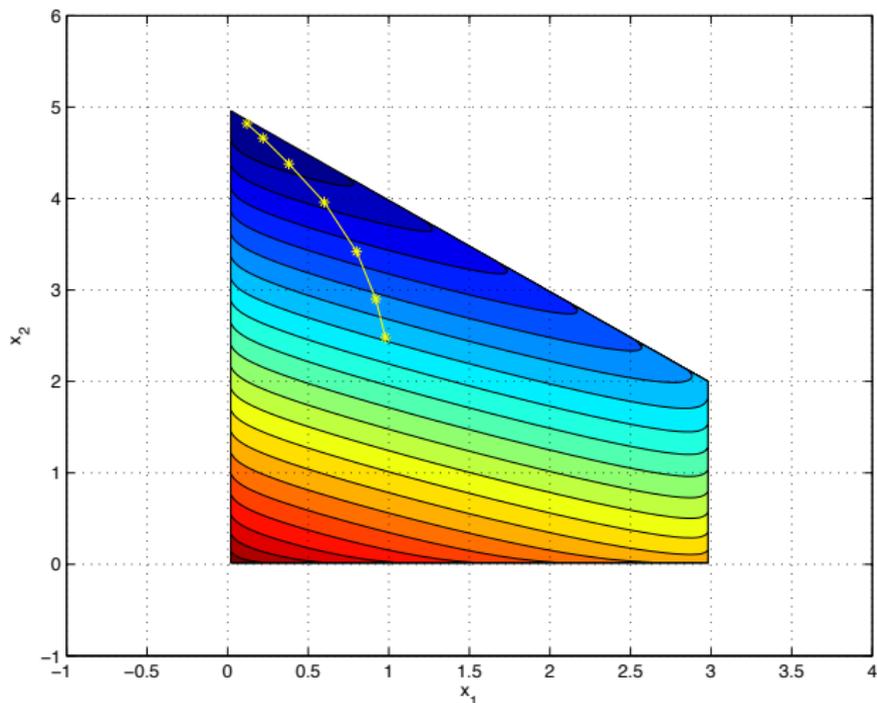
$t = 2$

The barrier interior-point method: example



$t = 4$

The barrier interior-point method: example



$t = 8$

Summary

- Interior point methods are widely applicable to **convex optimization problems**
(linear programs are just an example)
- (Well-implemented) interior point methods converge in polynomial time
(in the number of constraints m , and in the tolerance ϵ)
- Convex problems are in general very easy to solve numerically
- Open-source software for convex optimization:
 - SeDuMi: Primal-dual interior-point method
<http://sedumi.ie.lehigh.edu/>
 - Yalmip: User-friendly MATLAB interface
<http://control.ee.ethz.ch/~joloef/wiki/pmwiki.php>
- Non-convex optimization is a different story... stay tuned.

MIT OpenCourseWare
<http://ocw.mit.edu>

16.410 / 16.413 Principles of Autonomy and Decision Making

Fall 2010

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms> .