

ORF522 – Linear and Nonlinear Optimization

10. Interior-point methods for linear optimization

Midterm

- **Date/Time:** Thursday October 13, 11:00am – 12:20pm (80 min)
- **Where:** in class
- **Material allowed:** single sheet of paper, double-sided, hand-written or typed.
- **Topics:** linear optimization

Ed Forum

- Since introducing a new constraint to a primal problem is equivalent to introducing a new variable to the dual, is there a difference between choosing to solve one problem rather than another?
- How efficient is differentiable optimization? In neural network, people use ReLU because it is simple in evaluating both the forward pass values and backward differentiations. However, it seems like we need to solve a LP in every node in forward pass and to solve a linear system in backward pass.
- In lecture 9, we have seen that if we perturb the constraints a bit by some vector u , the optimal value behaves as a piecewise linear function in u . In case $c^T x$ and Ax are replaced by any suitable convex functions in x , is such a sensitivity analysis still possible?

Recap

Adding new variables

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \longrightarrow \begin{array}{ll} \text{minimize} & c^T x + c_{n+1} x_{n+1} \\ \text{subject to} & Ax + A_{n+1} x_{n+1} = b \\ & x, x_{n+1} \geq 0 \end{array}$$

Solution x^*, y^*

Solution $(x^*, 0), y^*$ **optimal** for the new problem?

Adding new variables

Optimality conditions

minimize $c^T x + c_{n+1}x_{n+1}$

subject to $Ax + A_{n+1}x_{n+1} = b \longrightarrow$ Solution $(x^*, 0)$ is still **primal feasible**

$$x, x_{n+1} \geq 0$$

Is y^* still **dual feasible**?

$$A_{n+1}^T y^* + c_{n+1} \geq 0$$

Yes

$(x^*, 0)$ still **optimal** for new problem

Otherwise

Primal simplex

Adding new constraints

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

Solution x^*, y^*

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & a_{m+1}^T x = b_{m+1} \\ & x \geq 0 \end{array}$$

Dual

$$\begin{array}{ll} \text{maximize} & -b^T y \\ \text{subject to} & A^T y + a_{m+1} y_{m+1} + c \geq 0 \end{array}$$

Solution $x^*, (y^*, 0)$ **optimal** for the new problem?

Adding new constraints

Optimality conditions

maximize $-b^T y$

subject to $A^T y + a_{m+1}y_{m+1} + c \geq 0$ \longrightarrow Solution $(y^*, 0)$ is still **dual feasible**

Is x^* still **primal feasible**?

$$Ax = b$$

$$a_{m+1}^T x = b_{m+1}$$

$$x \geq 0$$

Yes

x^* still **optimal** for new problem

Otherwise

Dual simplex

Today's lecture

[Chapter 14, NO][Chapters 17/18, LP]

- History
- Newton's method
- Central path
- Primal-dual path following method
- Logarithmic barrier functions
- Convergence

History

Ellipsoid method

Khachian (1979)

Answer to major question
Is worst-case LP complexity polynomial? Yes!

Drawbacks

Very inefficient. Much slower than simplex!

Benefits

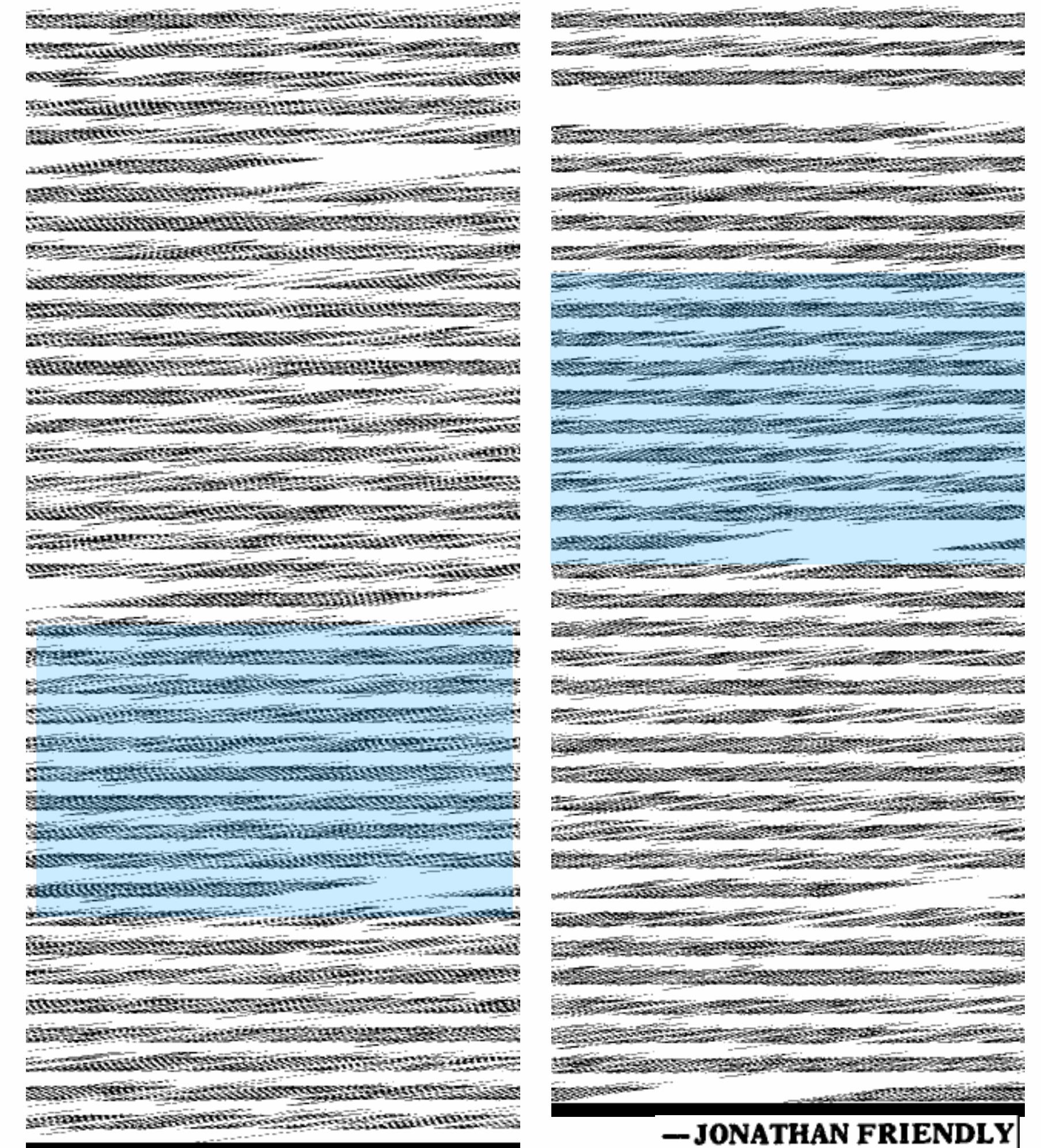
Motivated new research directions

Shazam! A Shortcut for Computers

A garment manufacturer has three kinds of dresses — A, B and C. On kind he has 17 bolts of one cloth and another, as well as 200 buttons. 25 of each. He has three cutters, 10 and 75 bakers and one finisher. Dress A, on which he makes a profit of \$1.25 a unit, requires one combination of unit, material, accessories and work; the material with a \$1.50 profit, takes a B dress, a combination, and the \$2.25 different cost. A third set of requirements for dress C has yet to be scheduled. How should he schedule his production to make the most money? That is an easy example of a kind of problem that is eminently practical, but difficult becomes computationally variable because of the number of factors and constraints that must be handled to get a best solution. As the number of variables and constraints grows — as, for instance, in model of the national economy or in scheduling of production at any refinery — the difficulty must increase.

most powerful computers might have to run for hours to tell a plant manager how to handle a small change in, say, the amount of crude oil being delivered to his tanks. And adding one new restriction can substantially increase the number of possible answers and thus the time required to check them for an optimum solution.

Last week, intrigued mathematicians were trying to sort out the meaning of what looked like a stun-



— JONATHAN FRIENDLY

The New York Times

Published: November 11, 1979

Copyright © The New York Times

Interior-point methods

1950s-1960s: nonlinear convex optimization

- Sequential unconstrained optimization (Fiacco & McCormick), Logarithmic barrier method (Frish), affine scaling method (Dikin), etc.
 - No worst-case complexity theory but often good practical performance

1980s-1990s: interior point methods

- Karmarkar's algorithm (1984)
 - Competitive with simplex, often faster for larger problems

plicated systems, often with thousands of variables. They arise in a variety of commercial and government applications, ranging from allocating time on a communications satellite to routing millions of telephone calls over long distances. Linear programming is particularly useful whenever a limited, expensive resource must be spread most efficiently among competing users. And investment companies use the approach in creating portfolios with the best mix of stocks and bonds.

Faster Solutions Seen

The Bell Labs mathematician, Dr. Narendra Karmarkar, has devised a radically new procedure that may aid the routine handling of such problems by businesses and Government agencies and also make it possible to tackle problems that are now far beyond reach.

This is a path-breaking result," Dr. Ronald L. Graham, director of mathematical sciences for Bell Labs in Murray Hill, N.J. "Science has its mo-



Newton's method

Newton's method for nonlinear equations

Goal: solve
$$h(x) = 0$$

Derivative

$$Dh = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \dots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_m}{\partial x_1} & \dots & \frac{\partial h_m}{\partial x_n} \end{bmatrix}$$

First-order approximation

$$h(x) \approx h(\bar{x}) + Dh(\bar{x})(x - \bar{x}) \longrightarrow$$

Iteratively set to zero

$$h(x^k) + Dh(x^k)(x^{k+1} - x^k) = 0$$

Δx

Iterations

- Solve $Dh(x^k)\Delta x = -h(x^k)$
- $x^{k+1} \leftarrow x^k + \Delta x$

Newton method

Convergence

Iterations

- Solve $Dh(x^k)\Delta x = -h(x^k)$
- $x^{k+1} \leftarrow x^k + \Delta x$

Remarks

- Iterations can be **expensive** (linear system solution)
- **Fast (quadratic) convergence** close to the solution x^*

Optimality conditions

	→	Primal
minimize $c^T x$		minimize $c^T x$
subject to $Ax \leq b$		subject to $Ax + s = b$
		$s \geq 0$
		Dual
		maximize $-b^T y$
		subject to $A^T y + c = 0$
		$y \geq 0$

Optimality conditions

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$$s_i y_i = 0$$

$$s, y \geq 0$$

Main idea

Optimality conditions

$$h(x, s, y) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} \end{bmatrix} = 0$$
$$s, y \geq 0$$
$$S = \text{diag}(s)$$
$$Y = \text{diag}(y)$$

- Apply variants of Newton's method to solve $h(x, s, y) = 0$
- Enforce $s, y > 0$ (strictly) at every iteration
- **Motivation** avoid getting stuck in “corners”

Newton's method for optimality conditions

Root-finding equation

$$h(x, s, y) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} \end{bmatrix} = 0$$

Linear system

Dh

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -h \\ -r_p \\ -r_d \\ -SY\mathbf{1} \end{bmatrix}$$

Residuals

$$r_p = Ax + s - b$$

$$r_d = A^T y + c$$

Issue

Line search to enforce $s, y > 0$
 $(x, s, y) \leftarrow (x, s, y) + \alpha(\Delta x, \Delta s, \Delta y)$

Newton's step does not allow significant progress towards both $h(x, s, y) = 0$ and $s, y \geq 0$.

Central path

Smoothed optimality conditions

Optimality conditions

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$s_i y_i = \tau$ ← Same τ for every pair

$$s, y \geq 0$$

Same optimality conditions for a “smoothed” version of our problem

Duality gap

$$s^T y = (b - Ax)^T y = b^T x - x^T A^T y = b^T y + c^T x$$

Newton's method for smoothed optimality conditions

Smoothed optimality conditions

$$h_\tau(x, s, y) = \begin{bmatrix} Ax + s - b \\ A^T y + c \\ SY\mathbf{1} - \tau\mathbf{1} \end{bmatrix} = 0$$
$$s, y \geq 0$$

Linear system

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY + \tau\mathbf{1} \end{bmatrix}$$

Line search to enforce $s, y > 0$
 $(x, s, y) \leftarrow (x, s, y) + \alpha(\Delta x, \Delta s, \Delta y)$

The path parameter

Duality measure

$\mu = \frac{s^T y}{m}$ (average value of the pairs $s_i y_i$)

Linear system

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} + \sigma\mu\mathbf{1} \end{bmatrix}$$

Centering parameter

$$\sigma \in [0, 1]$$

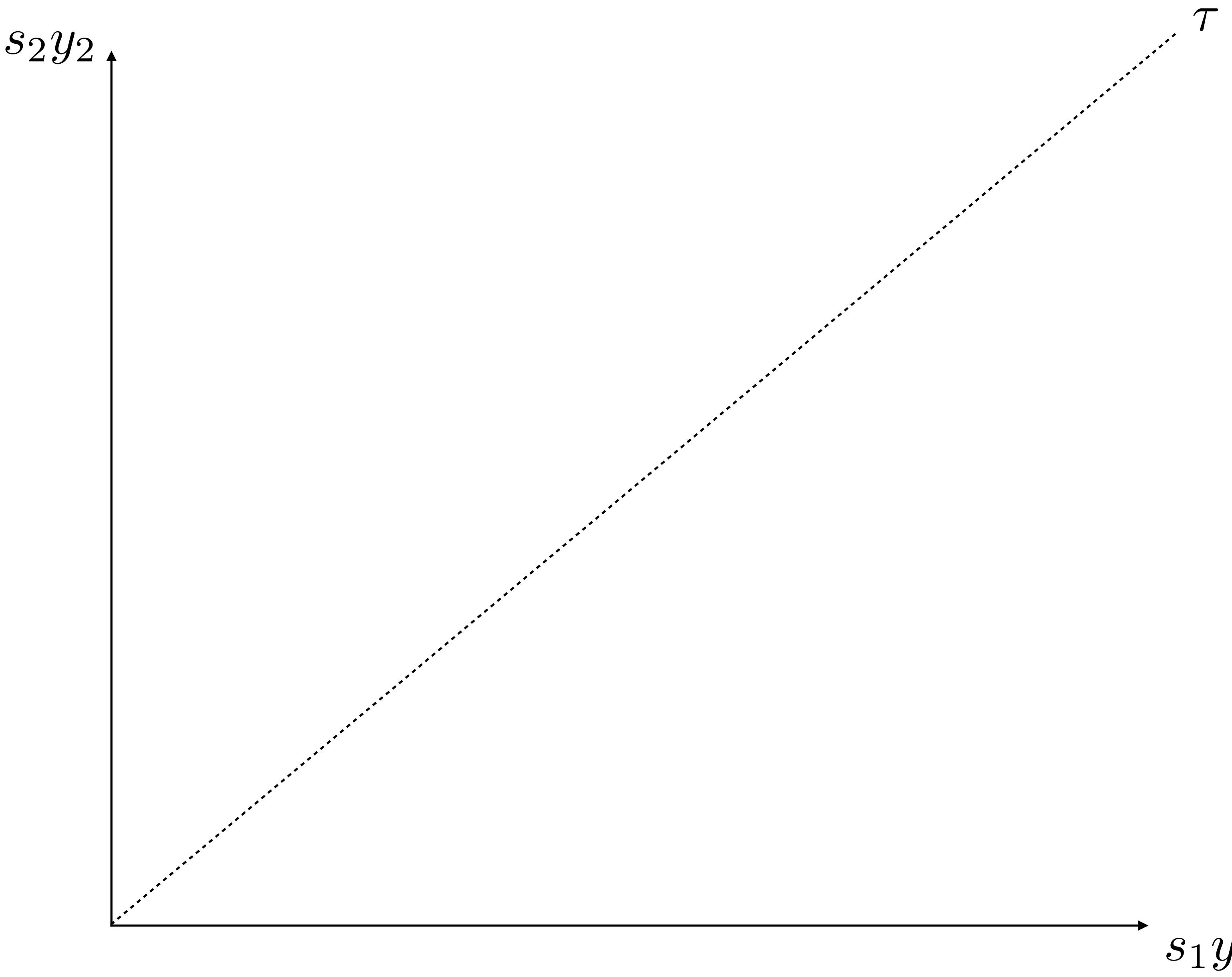
$\sigma = 0 \Rightarrow$ Newton step

$\sigma = 1 \Rightarrow$ Centering step towards $(y^*(\mu), x^*(\mu), s^*(\mu))$

Line search to enforce $s, y > 0$

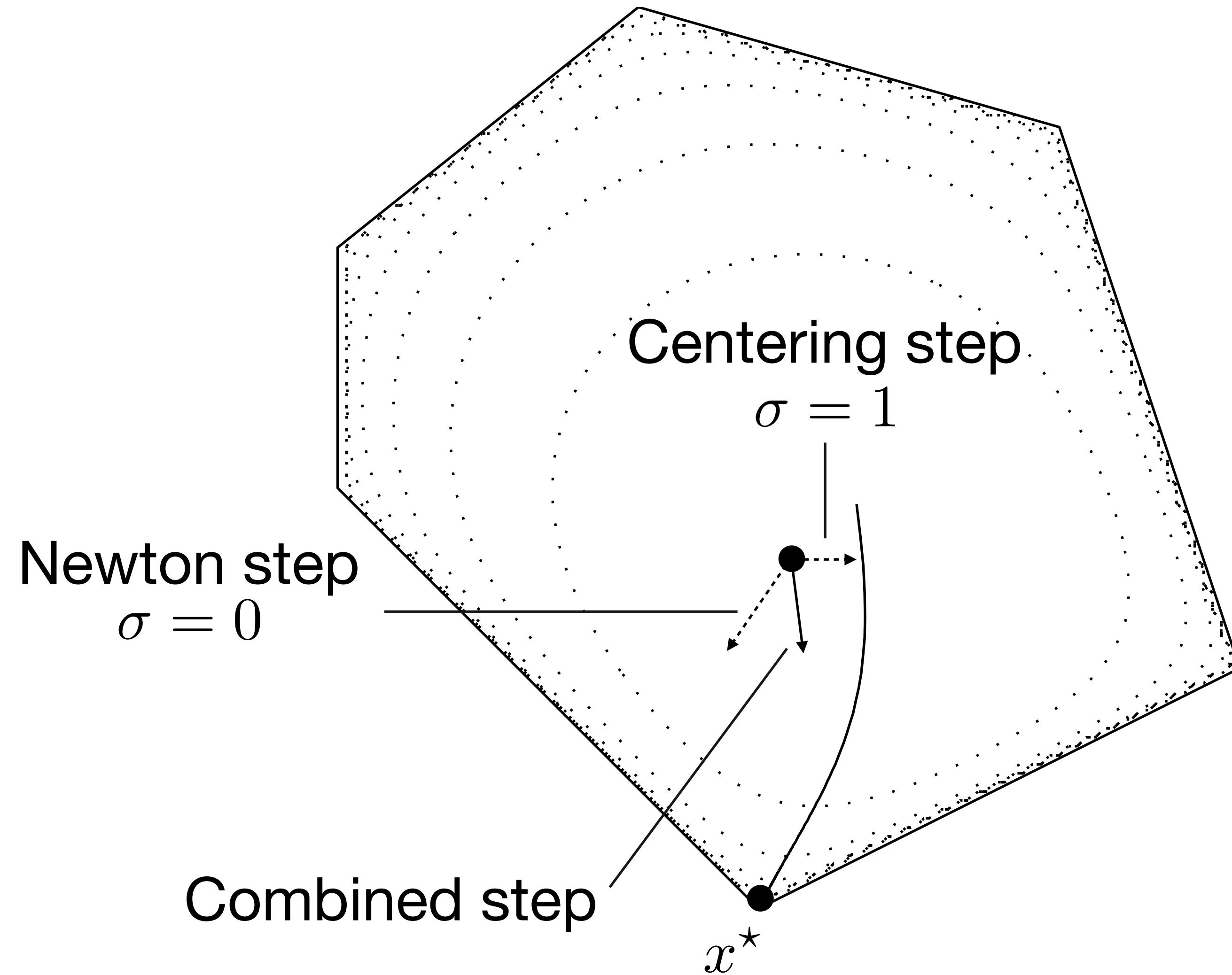
$$(y, x, s) \leftarrow (y, x, s) + \alpha(\Delta y, \Delta x, \Delta s)$$

The central path



Primal-dual path-following method

Path-following algorithm idea



Centering step

It brings towards the **central path** and is usually biased towards $s, y > 0$.

No progress on duality measure μ

Newton step

It brings towards the **zero duality measure** μ . Quickly violates $s, y > 0$.

Combined step

Best of both worlds with longer steps

Primal-dual path-following algorithm

Initialization

- Given (x_0, s_0, y_0) such that $s_0, y_0 > 0$

Iterations

- Choose $\sigma \in [0, 1]$

2. Solve
$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} + \sigma\mu\mathbf{1} \end{bmatrix}$$
 where $\mu = s^T y/m$

- Find maximum α such that $y + \alpha\Delta y > 0$ and $s + \alpha\Delta s > 0$
- Update $(x, s, y) \leftarrow (x, s, y) + \alpha(\Delta x, \Delta s, \Delta y)$

Working towards optimality conditions

Optimality conditions satisfied **only at convergence**

Primal residual

$$r_p = Ax + s - b \rightarrow 0$$

Stopping criteria

$$\|r_p\| \leq \epsilon_{\text{pri}}$$

Dual residual

$$r_d = A^T y + c \rightarrow 0$$

$$\|r_d\| \leq \epsilon_{\text{dua}}$$

Complementary slackness

$$s^T y \rightarrow 0$$

$$s^T y \leq \epsilon_{\text{gap}}$$

Logarithmic Barrier Functions

Smoothed optimality conditions

Optimality conditions

$$Ax + s - b = 0$$

$$A^T y + c = 0$$

$s_i y_i = \tau$ ← Same τ for every pair

$$s, y \geq 0$$

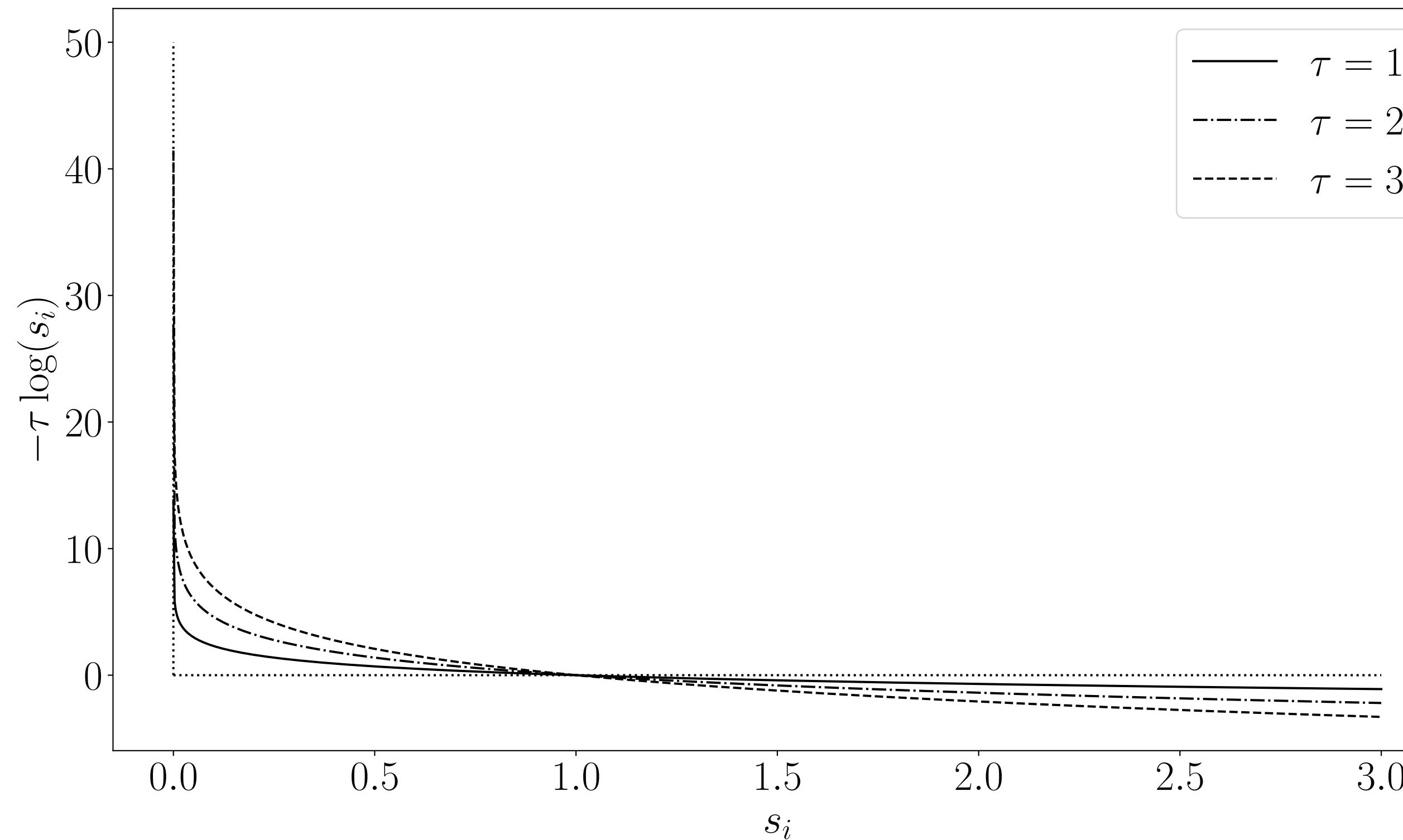
Same optimality conditions for a “smoothed” version of our problem

Do solutions actually exist?

What do they represent?

Logarithmic barrier

$$\phi(s) = -\tau \sum_{i=1}^m \log(s_i) \quad \text{on domain} \quad s_i > 0$$



As $\tau \rightarrow 0$ it approximates

$$\mathcal{I}_{s_i \geq 0} = \begin{cases} 0 & \text{if } s_i \geq 0 \\ \infty & \text{otherwise} \end{cases}$$

Smoothed problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax + s = b \\ & s \geq 0 \end{array} \longrightarrow \begin{array}{ll} \text{minimize} & c^T x + \phi(x) = c^T x - \tau \sum_{i=1}^m \log(s_i) \\ \text{subject to} & Ax + s = b \end{array}$$

Dual cost

$$g(y) = \underset{x,s}{\text{minimize}} \mathcal{L}(x, s, y) = c^T x + \phi(s) + y^T (Ax + s - b)$$

$$\frac{\partial \mathcal{L}}{\partial x} = A^T y + c = 0$$

$$\frac{\partial \mathcal{L}}{\partial s_i} = -\tau \frac{1}{s_i} + y_i = 0 \implies s_i y_i = \tau$$

Central path

$$\begin{array}{ll}\text{minimize} & c^T x - \tau \sum_{i=1}^m \log(s_i) \\ \text{subject to} & Ax + s = b\end{array}$$

Set of points $(x^*(\tau), s^*(\tau), y^*(\tau))$
with $\tau > 0$ such that

$$Ax + s - b = 0$$

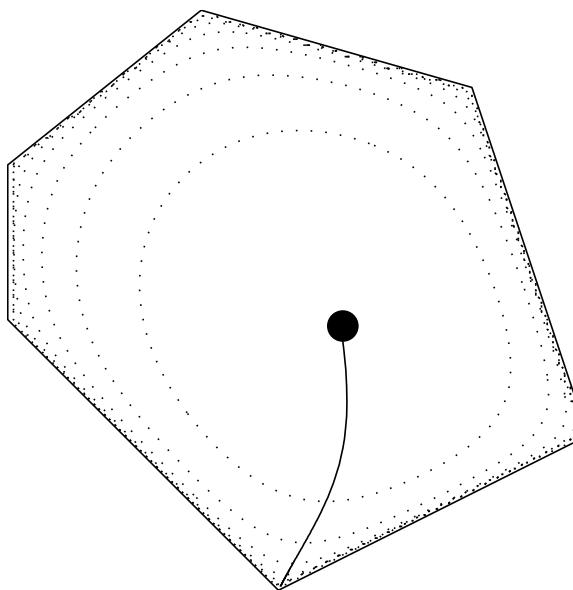
$$A^T y + c = 0$$

$$s_i y_i = \tau$$

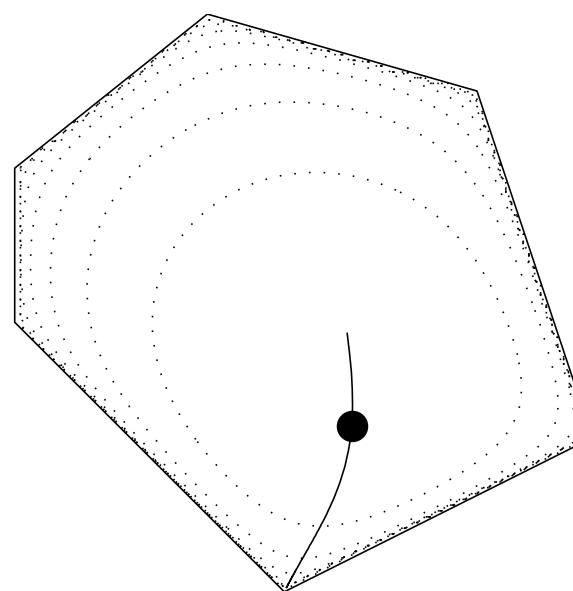
$$s, y \geq 0$$

Analytic
Center

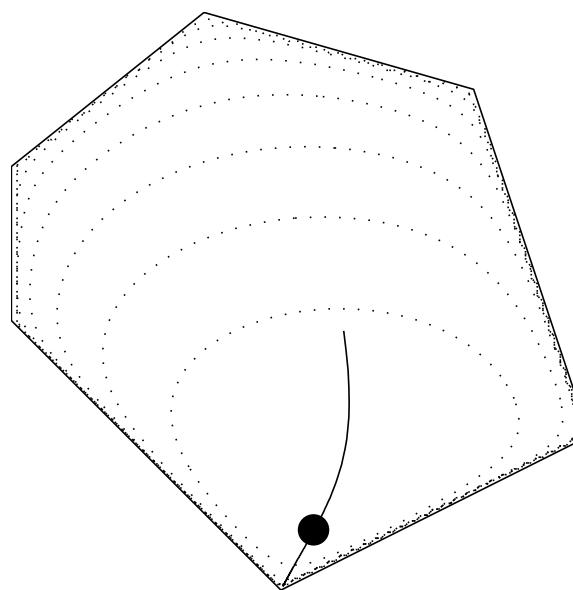
$$\tau \rightarrow \infty$$



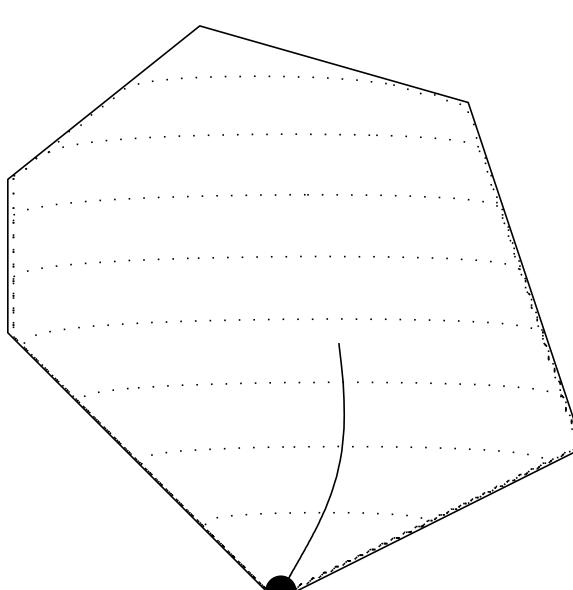
1000



1



1/5



1/100

τ

Main idea

Follow central path as $\tau \rightarrow 0$

32

Convergence

Definitions

Primal-dual strictly feasible set

$$\mathcal{F}^\circ = \{(x, s, y) \mid Ax + s = b, A^T y + c = 0, s, y > 0\}$$

Central path neighborhood

$$\mathcal{N}(\gamma) = \{(x, s, y) \in \mathcal{F}^\circ \mid s_i y_i \geq \gamma \mu\} \quad \text{with } \gamma \in (0, 1] \quad (\text{almost all the feasible region})$$

Theorem

[Page 402-406, NO]

[Theorem 14.3] Smallest decrement

$$\mu_{k+1} \leq (1 - \delta/n) \mu_k \quad \text{with constant } \delta > 0$$

Iteration complexity

Given $(x_0, s_0, y_0) \in \mathcal{N}(\gamma)$, there exists $K = O(n \log(1/\epsilon))$ such that

$$\mu_k \leq \epsilon \mu_0 \quad \text{for all } k \geq K$$

Remark Modified versions achieve $O(\sqrt{n} \log(1/\epsilon))$

Iteration complexity proof

[Page 402-406, NO]

$$\mu_{k+1} \leq (1 - \delta/n) \mu_k$$

(take logarithm)

$$\log \mu_{k+1} \leq \log(1 - \delta/n) + \log \mu_k$$

(apply iteratively)

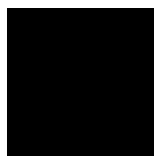
$$\log \mu_k \leq k \log(1 - \delta/n) + \log \mu_0$$

Since $\log(1 + \beta) \leq \beta$, $\forall \beta > -1$ \longrightarrow $\log(\mu_k / \mu_0) \leq k(-\delta/n)$

If $k(-\delta/n) \leq \log(\epsilon)$

then $\log(\mu_k / \mu_0) \leq \log(\epsilon)$. Therefore, $\mu_k / \mu_0 \leq \epsilon$

Rewriting the inequality: $k \geq (n/\delta) \log(1/\epsilon)$



Interior-point methods for linear optimization

Today, we learned to:

- **Apply** Newton's method to solve optimality conditions
- **Analyze** the central path and the smoothed optimality conditions
- **Develop** a prototype primal-dual path-following algorithm

Next lecture

- Practical interior-point method (Mehrotra predictor-corrector algorithm)
- Linear algebra implementation details
- Linear optimization recap