ORF522 – Linear and Nonlinear Optimization

11. Interior-point methods implementation

Bartolomeo Stellato — Fall 2020

Ed forum

- We also said that we didn’t’ want to be exactly on the central path but remain in a neighborhood. **What happens when we are directly on the central path?** From the picture it looked like there would still be a newton step and a centering step so we’d be ok?
- What will happen to Newton's method if we get to a corner?
- What is the main advantage of using methods like primal dual path following vs simply **taking very small positive tau**, say 1e-6, and solving the problem? In such as case, do we still have the problem of potentially getting stuck in the corner?
- Newton's method relies on differentiability of the function that we want to set to zero. What can we do if a **function is continuous but nondifferentiable**?
- When we take steps that are "mixtures" of Newton's direction and Central Path direction ($\sigma<1$), **how can we guarantee that there exists $\alpha>0$** such that we can make $y+\alpha\triangle y>0$?
- **Can the initialization of the central path method lead to non-convergence** or can we just find any point in the interior of the feasible set?

Recap

(Sparse) Cholesky factorization

Every positive definite matrix A can be factored as

$$A = PLL^T P^T \longrightarrow P^T AP = LL^T$$

P permutation, L lower triangular

(Sparse) Cholesky factorization

Every positive definite matrix A can be factored as

$$A = PLL^T P^T \longrightarrow P^T AP = LL^T$$

P permutation, L lower triangular

Permutations

- Reorder rows/cols of A with P to (heuristically) get **sparser** L
- P depends only on sparsity pattern of A (unlike LU factorization)
- If A is dense, we can set $P = I$

(Sparse) Cholesky factorization

Every positive definite matrix A can be factored as

$$A = PLL^T P^T \longrightarrow P^T AP = LL^T$$

P permutation, L lower triangular

Permutations

- Reorder rows/cols of A with P to (heuristically) get **sparser** L
- P depends only on sparsity pattern of A (unlike LU factorization)
- If A is dense, we can set $P = I$

Cost

- If A dense, typically $O(n^3)$ but usually much less
- It depends on the number of nonzeros in A, sparsity pattern, etc.
- Typically 50% faster than LU (need to find only one matrix)

Symmetric primal-dual problems

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

Symmetric primal-dual problems

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

Symmetric primal-dual problems

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \end{array} \longrightarrow$$

Primal

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

Dual

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

Optimality conditions

$$\begin{aligned} Ax + s - b &= 0 \\ A^T y + c &= 0 \\ s_i y_i &= 0 \\ s, y &\geq 0 \end{aligned}$$

Strict complementarity

Primal

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

Dual

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

Theorem

If the two problems have feasible solutions, then there exist feasible s and y with a **strict complementary sparsity** pattern:

$$y_i > 0, s_i = 0 \quad \text{or} \quad y_i = 0, s_i > 0$$

In other words, $s_i + y_i > 0$

Proof (left as exercise)

Details in [Theorem 10.6, Vanderbei]

Main idea

Optimality conditions

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

$$s, y \geq 0$$

$$S = \mathbf{diag}(s)$$
$$Y = \mathbf{diag}(y)$$

- 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"

Algorithm step

Linear system

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

Duality measure

$$\mu = \frac{s^T y}{m}$$

Centering parameter

$\sigma \in [0, 1]$

$\sigma = 0 \quad \Rightarrow \quad$ Newton step

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

Line search to enforce $x, s > 0$

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

Primal-dual path-following algorithm

Initialization

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

Iterations

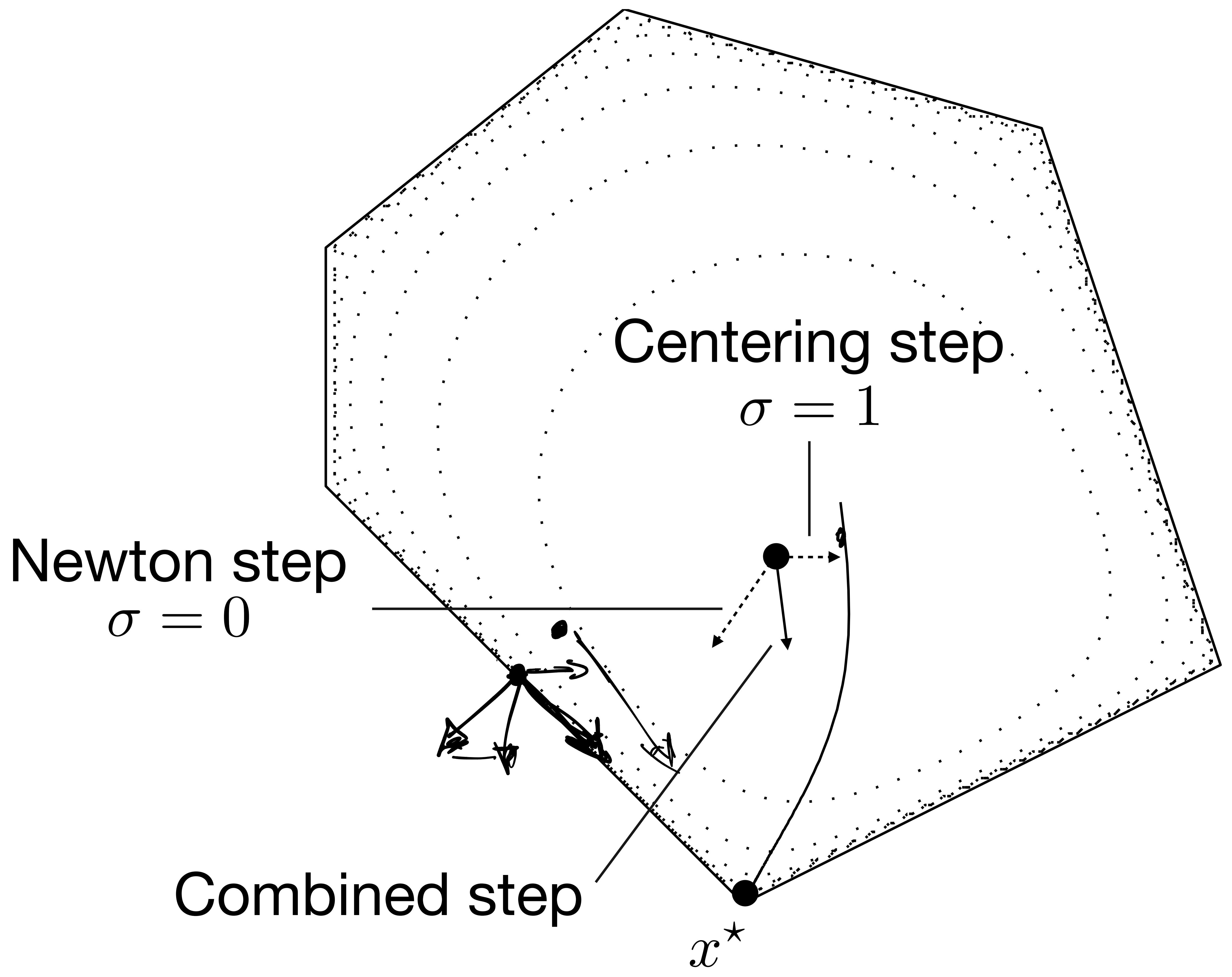
1. Choose $\sigma \in [0, 1]$

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

3. Find maximum α such that $y + \alpha\Delta y > 0$ and $s + \alpha\Delta s > 0$

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

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

Today's lecture

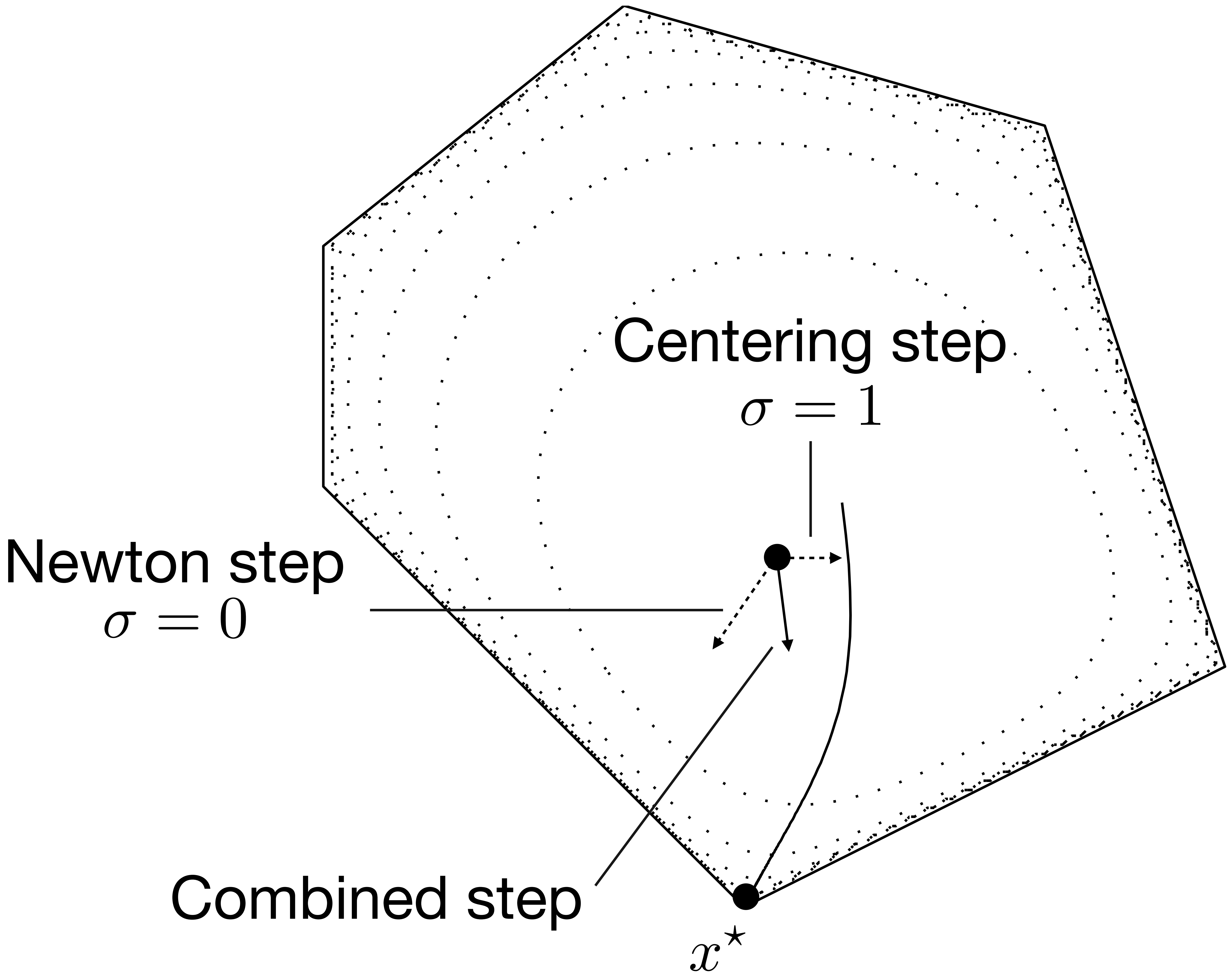
[Chapter 14, Nocedal and Wright][Chapter 22, Vanderbei]

- Mehrotra predictor-corrector algorithm
- Implementation details
- Homogeneous self-dual embedding
- Interior-point vs simplex

Predictor-corrector algorithm

Main idea:

Predict and select centering parameter

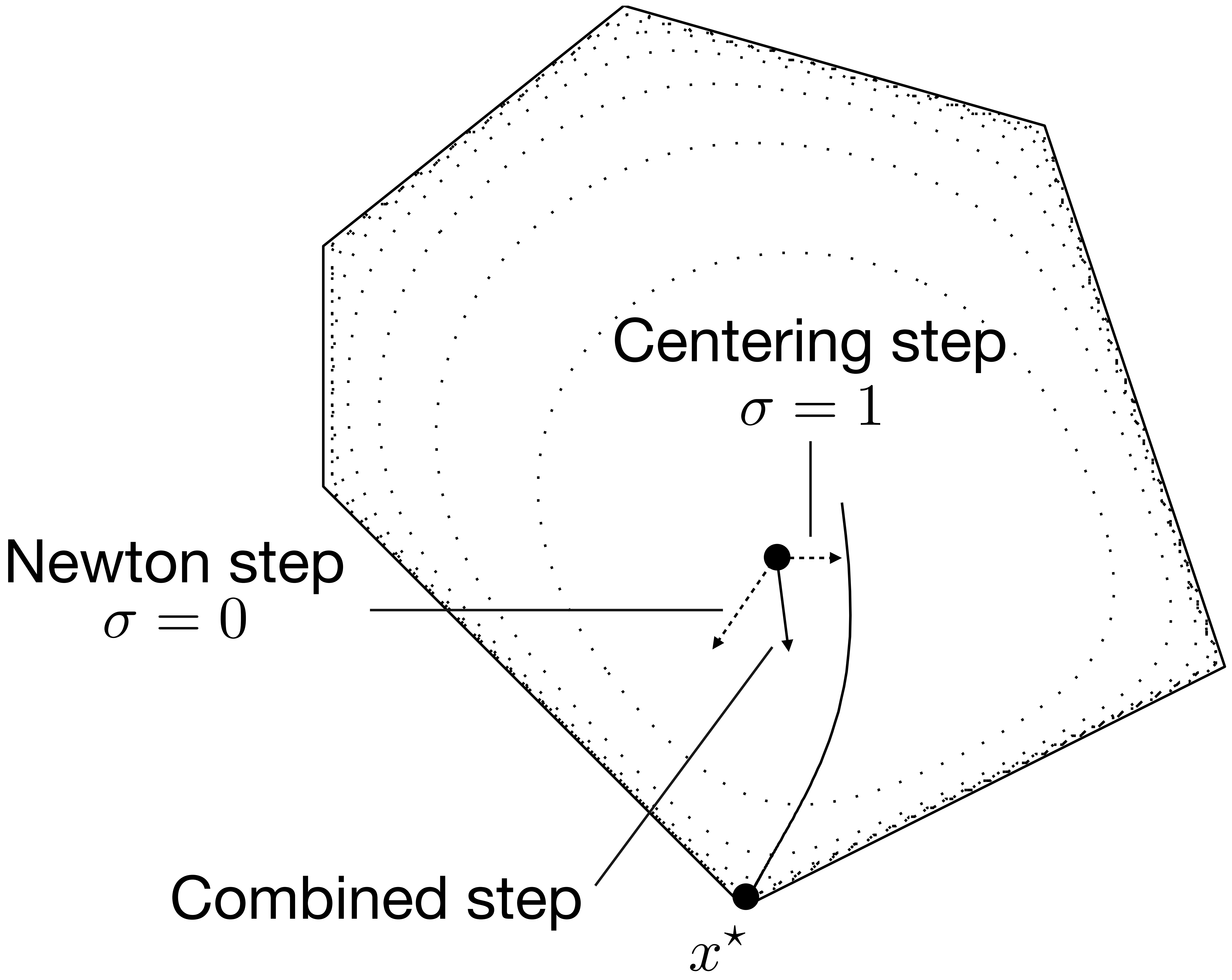


Predict

Compute Newton direction

Main idea:

Predict and select centering parameter



Predict

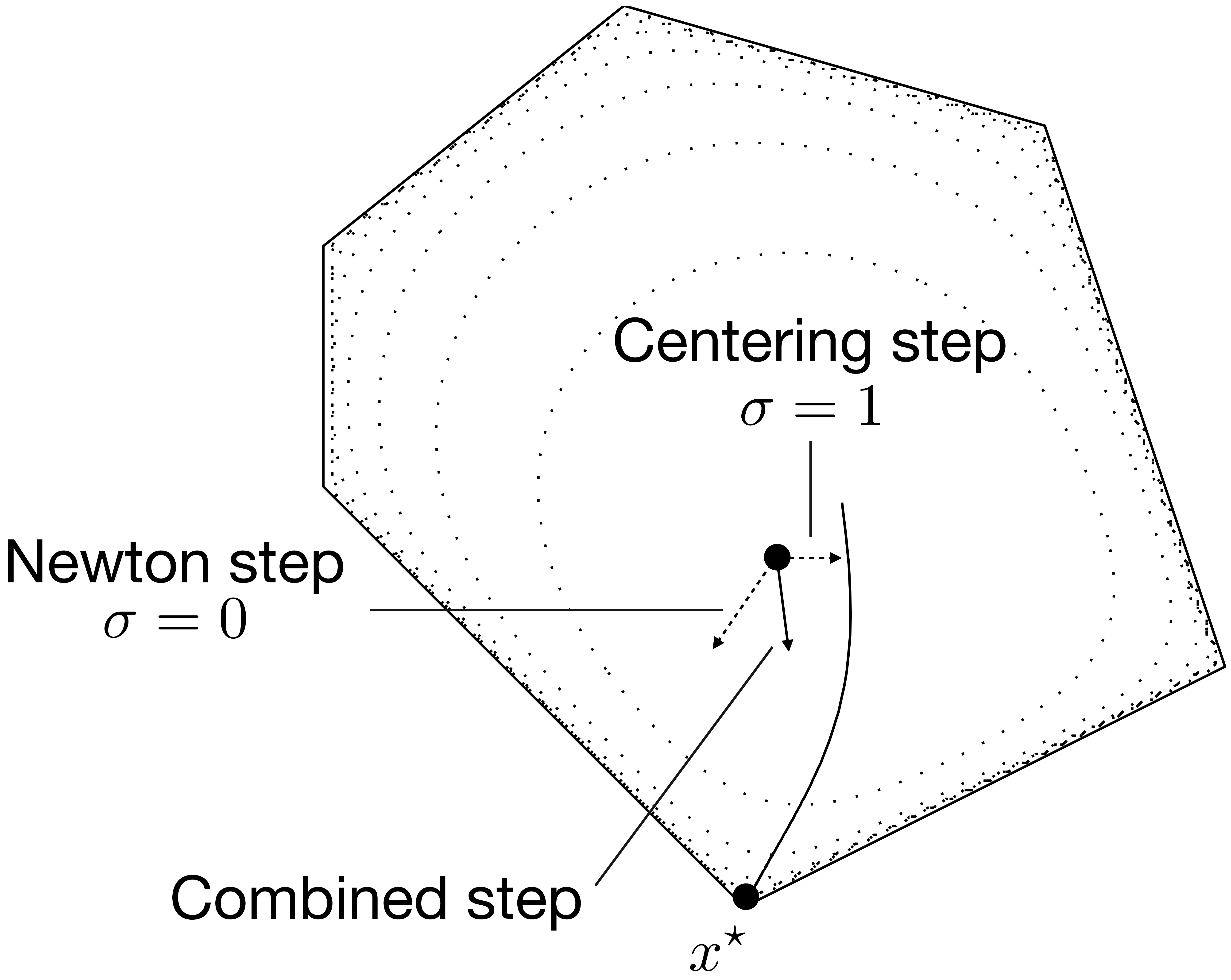
Compute Newton direction

Estimate

How good is the Newton step?
(how much can μ decrease?)

Main idea:

Predict and select centering parameter



Predict

Compute Newton direction

Estimate

How good is the Newton step?
(how much can μ decrease?)

Select centering parameter

Very roughly:
Pick $\sigma \approx 0$ if Newton step is good
Pick $\sigma \approx 1$ if Newton step is bad

Select centering parameter

Newton step

$$(\Delta x_a, \Delta s_a, \Delta y_a)$$

Maximum step-size

$$\alpha_p = \max\{\alpha \in [0,1] \mid s + \alpha \Delta s_a \geq 0\}$$
$$\alpha_d = \max\{\alpha \in [0,1] \mid y + \alpha \Delta y_a \geq 0\}$$

Select centering parameter

Newton step

$$(\Delta x_a, \Delta s_a, \Delta y_a)$$

Maximum step-size

$$\alpha_p = \max\{\alpha \in [0,1] \mid s + \alpha \Delta s_a \geq 0\}$$
$$\alpha_d = \max\{\alpha \in [0,1] \mid y + \alpha \Delta y_a \geq 0\}$$

Duality measure candidate
(after Newton step)

$$\mu_a = \frac{(s + \alpha_p \Delta s_a)^T (y + \alpha_d \Delta y_a)}{m}$$

Select centering parameter

Newton step

$$(\Delta x_a, \Delta s_a, \Delta y_a)$$

Maximum step-size

$$\alpha_p = \max\{\alpha \in [0,1] \mid s + \alpha \Delta s_a \geq 0\}$$
$$\alpha_d = \max\{\alpha \in [0,1] \mid y + \alpha \Delta y_a \geq 0\}$$

Duality measure candidate
(after Newton step)

$$\mu_a = \frac{(s + \alpha_p \Delta s_a)^T (y + \alpha_d \Delta y_a)}{m}$$

$\longrightarrow$

Centering parameter heuristic σ

$$\sigma = \left(\frac{\mu_a}{\mu}\right)^3$$

Mehrotra correction

Newton step

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y_a \\ \Delta x_a \\ \Delta s_a \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} \end{bmatrix}$$

Mehrotra correction

Newton step

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y_a \\ \Delta x_a \\ \Delta s_a \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} \end{bmatrix} \longrightarrow \quad s_i(\Delta y_a)_i + y_i(\Delta s_a)_i + s_i y_i = 0$$

Mehrotra correction

Newton step

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y_a \\ \Delta x_a \\ \Delta s_a \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} \end{bmatrix} \longrightarrow \quad s_i(\Delta y_a)_i + y_i(\Delta s_a)_i + s_i y_i = 0$$

Full step

$$(s_i + (\Delta s_a)_i)(y_i + (\Delta y_a)_i) = (\Delta s_a)_i (\Delta y_a)_i \neq 0$$

Complementarity violation depends on step length

Mehrotra correction

Newton step

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y_a \\ \Delta x_a \\ \Delta s_a \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} \end{bmatrix} \longrightarrow \quad s_i(\Delta y_a)_i + y_i(\Delta s_a)_i + s_i y_i = 0$$

Full step

$$(s_i + (\Delta s_a)_i)(y_i + (\Delta y_a)_i) = (\Delta s_a)_i(\Delta y_a)_i \neq 0$$

Complementarity violation depends on step length

Corrected direction

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

$$\Delta S_a = \mathbf{diag}(\Delta s_a)$$
$$\Delta Y_a = \mathbf{diag}(\Delta y_a)$$

Mehrotra predictor-corrector algorithm

Initialization

Given (x, s, y) such that $s, y > 0$

1. **Termination conditions**

   $r_p = Ax + s - b, \quad r_d = A^T y + c, \quad \mu = (s^T y)/m$

   If $\|r_p\|, \|r_d\|, \mu$ are small, **break** Optimal solution $(x^\star, s^\star, y^\star)$

2. **Newton step** (affine scaling)

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y_a \\ \Delta x_a \\ \Delta s_a \end{bmatrix} = \begin{bmatrix} -r_p \\ -r_d \\ -SY\mathbf{1} \end{bmatrix}$$

Mehrotra predictor-corrector algorithm

3. **Barrier parameter**

$$\alpha_p = \max\{\alpha \in [0,1] \mid s + \alpha \Delta s_a \geq 0\}$$
$$\alpha_d = \max\{\alpha \in [0,1] \mid y + \alpha \Delta y_a \geq 0\}$$
$$\mu_a = \frac{(s + \alpha_p \Delta s_a)^T (y + \alpha_d \Delta y_a)}{m}$$
$$\sigma = \left(\frac{\mu_a}{\mu}\right)^3$$

4. **Corrected direction**

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

Mehrotra predictor-corrector algorithm

5. **Update iterates**

$$\alpha_p = \max\{\alpha \geq 0 \mid s + \alpha \Delta s \geq 0\}$$
$$\alpha_d = \max\{\alpha \geq 0 \mid y + \alpha \Delta y \geq 0\}$$

$$(x, s) = (x, s) + \min\{1, \eta\alpha_p\}(\Delta x, \Delta s)$$
$$y = y + \min\{1, \eta\alpha_d\}\Delta y$$

Avoid corners

$$\eta = 1 - \epsilon \approx 0.99$$

Implementation details

Search equations

Step 2 (**Newton**) and 4 (**Corrected direction**) solve equations of the form

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} b_y \\ b_x \\ b_s \end{bmatrix}$$

Search equations

Step 2 (**Newton**) and 4 (**Corrected direction**) solve equations of the form

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} b_y \\ b_x \\ b_s \end{bmatrix}$$

Substitute last equation, $\Delta s = Y^{-1}(b_s - S\Delta y)$, into first

$$\begin{bmatrix} -Y^{-1}S & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \end{bmatrix} = \begin{bmatrix} b_y - Y^{-1}b_s \\ b_x \end{bmatrix}$$

Search equations

Step 2 (**Newton**) and 4 (**Corrected direction**) solve equations of the form

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ S & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} b_y \\ b_x \\ b_s \end{bmatrix}$$

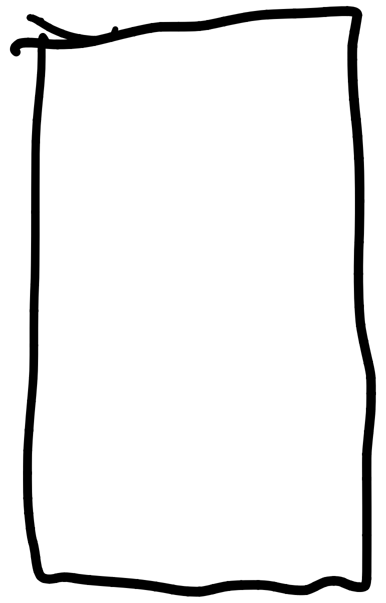
Substitute last equation, $\Delta s = Y^{-1}(b_s - S\Delta y)$, into first

$$\begin{bmatrix} -Y^{-1}S & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \end{bmatrix} = \begin{bmatrix} b_y - Y^{-1}b_s \\ b_x \end{bmatrix}$$

Substitute first equation, $\Delta y = S^{-1}Y(A\Delta x - b_y + Y^{-1}b_s)$, into second

$$\boxed{A^T S^{-1} Y A}\Delta x = \underline{b_x + A^T S^{-1} Y b_y - A^T S^{-1} b_s}$$

Simplified linear system



Coefficient matrix

$$B = A^T S^{-1} Y A$$

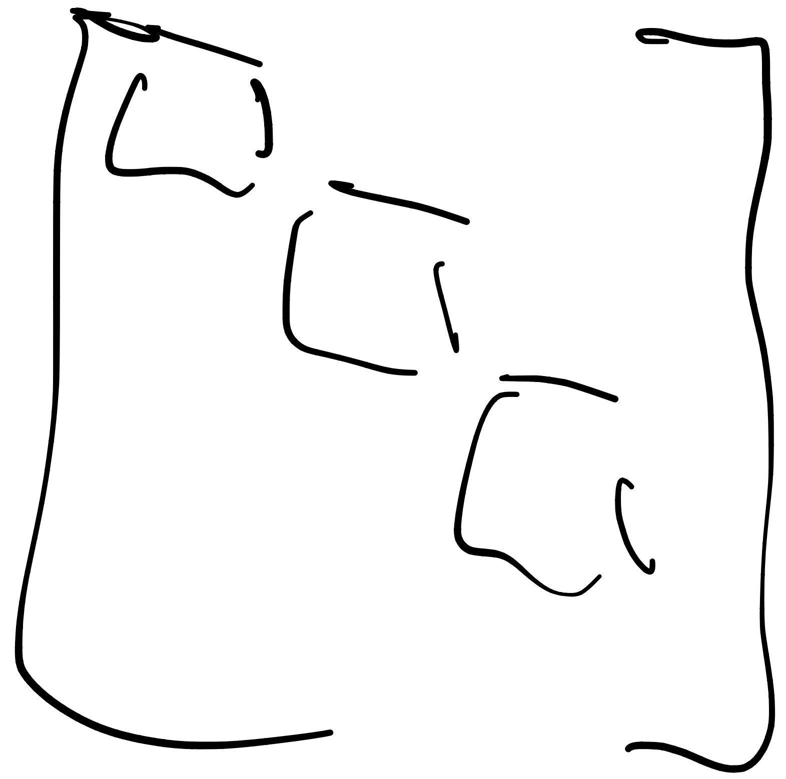
$s_i, y_i > 0$

y_i / s_i

Characteristics

- A is **large** and **sparse**
- $S^{-1}\cancel{Y}$ is **positive** and **diagonal**, different at each iteration
- B is **positive definite** if $\mathrm{rank}(A) = n$
- Sparsity pattern of B is the **pattern** of $A^T A$ (independent of $S^{-1}\cancel{Y}$)

Simplified linear system



Coefficient matrix

$$B = A^T S^{-1} Y A$$

Characteristics

- A is **large** and **sparse**
- $S^{-1}Y$ is **positive** and **diagonal**, different at each iteration
- B is **positive definite** if $\mathrm{rank}(A) = n$
- Sparsity pattern of B is the **pattern** of $A^T A$ (independent of $S^{-1}Y$)

Cholesky factorizations

$$B = PLL^T P^T$$

- Reordering only once to get P
- One numerical factorizaton per interior-point iteration $O(n^3)$
- Forward/backward substitution twice per iteration $O(n^2)$

Simplified linear system

Coefficient matrix

$$B = A^T S^{-1} Y A$$

Characteristics

- A is **large** and **sparse**
- $S^{-1}\cancel{Z}$ is **positive** and **diagonal**, different at each iteration
- B is **positive definite** if $\mathrm{rank}(A) = n$
- Sparsity pattern of B is the **pattern** of $A^T A$ (independent of $S^{-1}\cancel{Z}$)

Cholesky factorizations

$$B = PLL^T P^T$$

- Reordering only once to get P
- One numerical factorizaton per interior-point iteration $O(n^3)$
- Forward/backward substitution twice per iteration $O(n^2)$

$\longrightarrow$ **Per-iteration complexity** $O(n^3)$

Convergence

Mehrotra's algorithm

No convergence theory $\longrightarrow$ Examples where it **diverges** (rare!)

Convergence

Mehrotra's algorithm

No convergence theory $\longrightarrow$ Examples where it **diverges** (rare!)

Fantastic convergence **in practice** $\longrightarrow$ Less than 30 iterations

Convergence

Mehrotra's algorithm

No convergence theory $\longrightarrow$ Examples where it **diverges** (rare!)

Fantastic convergence **in practice** $\longrightarrow$ Less than 30 iterations

Theoretical iteration complexity

Alternative versions (slower than Mehrotra) converge in $O(\sqrt{n})$ iterations

Convergence

Mehrotra's algorithm

No convergence theory $\longrightarrow$ Examples where it **diverges** (rare!)

Fantastic convergence **in practice** $\longrightarrow$ Less than 30 iterations

Theoretical iteration complexity

Alternative versions (slower than Mehrotra) converge in $O(\sqrt{n})$ iterations

Average iteration complexity

Average iterations complexity is $O(\log n)$

Convergence

Mehrotra's algorithm

No convergence theory ⟶ Examples where it **diverges** (rare!)

Fantastic convergence **in practice** ⟶ Less than 30 iterations

Theoretical iteration complexity | **Operations**
Alternative versions (slower than Mehrotra) converge in $O(\sqrt{n})$ iterations ⟶ $O(n^{3.5})$

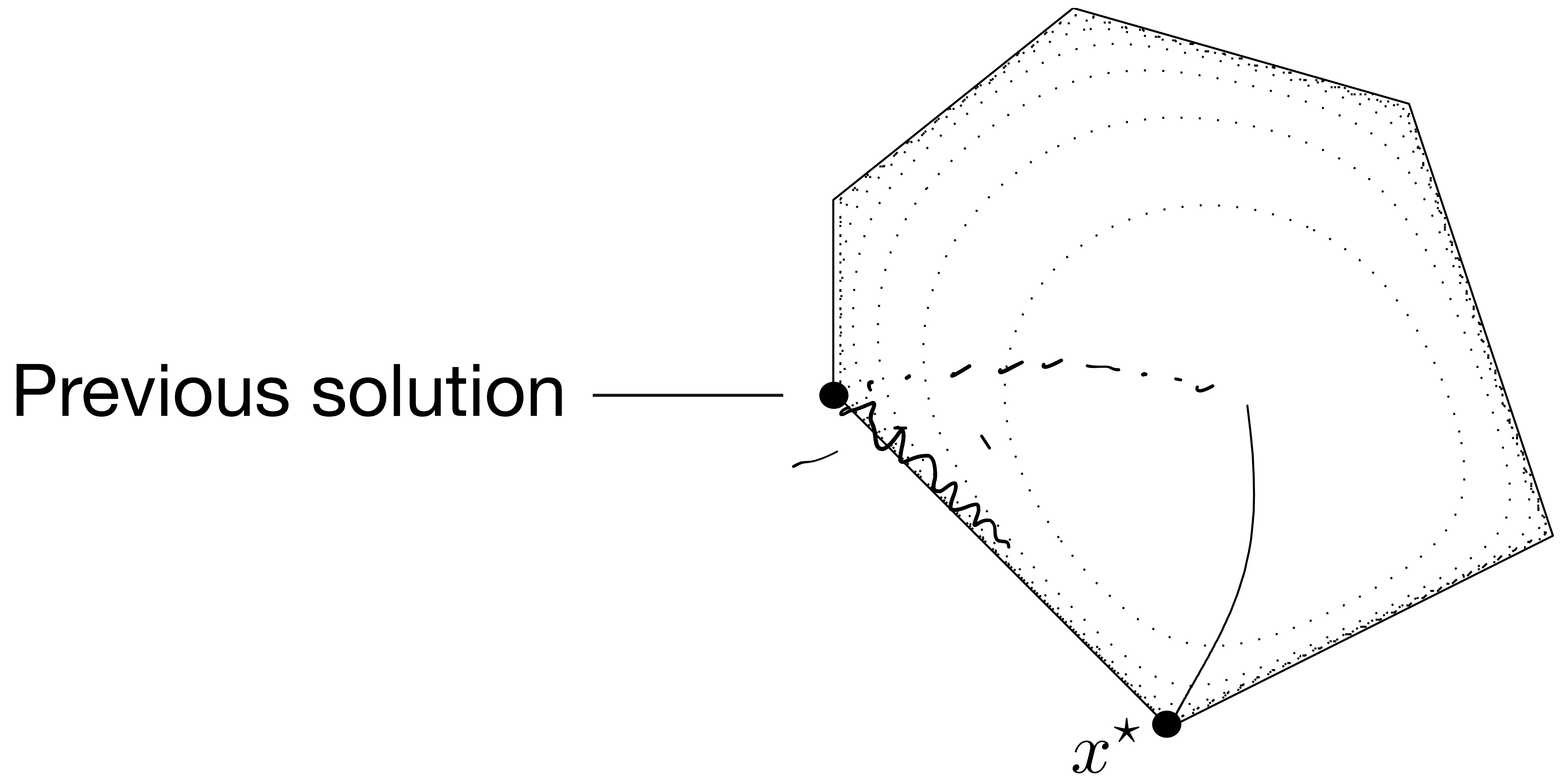
Average iteration complexity
Average iterations complexity is $O(\log n)$ ⟶ $O(n^3 \log n)$

Warm-starting

Interior-point methods are **difficult to warm-start**

Warm-starting

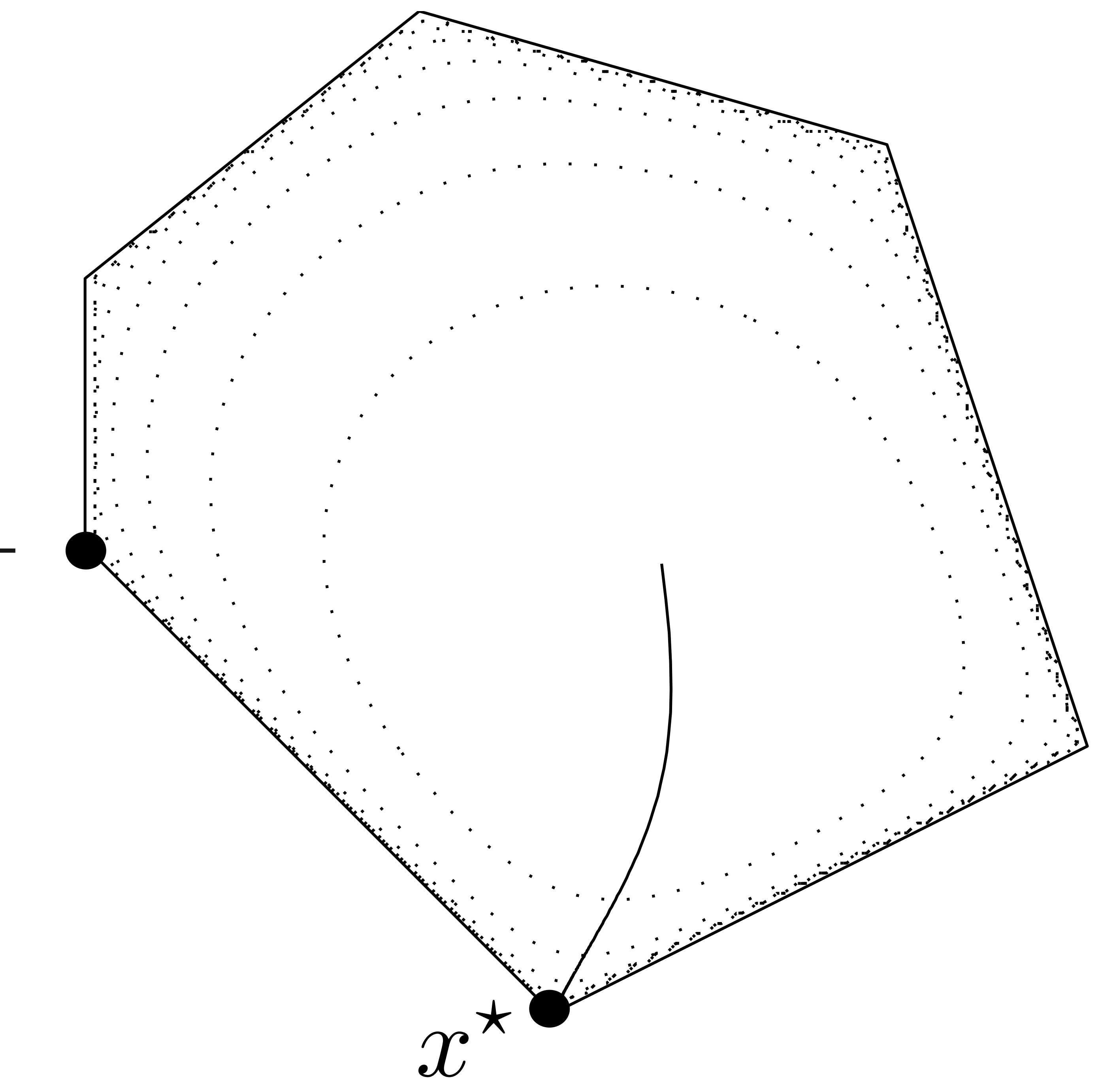
Interior-point methods are **difficult to warm-start**



Warm-starting

Interior-point methods are **difficult to warm-start**

Previous solution



Badly centered
initial point

↓

Hard to make progress
with long steps

Homogeneous self-dual embedding

Optimality conditions

Primal

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

Dual

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

Optimality conditions

$$\begin{bmatrix} 0 \\ s \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & A^T \\ -A & 0 \\ c^T & b^T \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c \\ b \\ 0 \end{bmatrix}$$

$$s, y \geq 0$$

Any $(x^\star, s^\star, y^\star)$ satisfying these conditions is **optimal**

Optimality conditions

Primal

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

Dual

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

Optimality conditions

$$
\begin{bmatrix} 0 \\ s \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & A^T \\ -A & 0 \\ c^T & b^T \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c \\ b \\ 0 \end{bmatrix}
$$

$$
s, y \geq 0
$$

Any $(x^\star, s^\star, y^\star)$ satisfying these conditions is **optimal**

What happens if the problem is infeasible?

How do you detect infeasibility/unboundedness?

Primal

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

Dual

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

Alternatives (Farkas lemma) Write feasibility problem and dualize…

- **primal feasible:** $Ax + s = b, \quad s \geq 0$
- **primal infeasible:** $A^T y = 0, \quad b^T y < 0, \quad y \geq 0$

How do you detect infeasibility/unboundedness?

Primal

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

Dual

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

Alternatives (Farkas lemma) Write feasibility problem and dualize…

- **primal feasible:** $Ax + s = b, \quad s \geq 0$
- **primal infeasible:** $A^T y = 0, \quad b^T y < 0, \quad y \geq 0$ (primal infeasibility certificate)

How do you detect infeasibility/unboundedness?

Primal

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

Dual

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

Alternatives (Farkas lemma) Write feasibility problem and dualize…

- **primal feasible:** $Ax + s = b, \quad s \geq 0$
- **primal infeasible:** $A^T y = 0, \quad b^T y < 0, \quad y \geq 0$ (primal infeasibility certificate)

- **dual feasible:** $A^T y + c = 0, \quad y \geq 0$
- **dual infeasible:** $Ax \leq 0, \quad c^T x < 0$

How do you detect infeasibility/unboundedness?

Primal

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

Dual

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

Alternatives (Farkas lemma) Write feasibility problem and dualize…

- **primal feasible:** $Ax + s = b, \quad s \geq 0$
- **primal infeasible:** $A^T y = 0, \quad b^T y < 0, \quad y \geq 0$ (primal infeasibility certificate)

- **dual feasible:** $A^T y + c = 0, \quad y \geq 0$
- **dual infeasible:** $Ax \leq 0, \quad c^T x < 0$ (dual infeasibility certificate)

The homogeneous self-dual embedding
Derivation

Introduce two new variables $\kappa, \tau \geq 0$

Homogeneous self-dual embedding

$$\begin{bmatrix} 0 \\ s \\ \kappa \end{bmatrix} = \begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ \tau \end{bmatrix}$$

$$s, y, \kappa, \tau \geq 0$$

The homogeneous self-dual embedding

Derivation

Introduce two new variables $\kappa, \tau \geq 0$

Homogeneous self-dual embedding

$$\begin{bmatrix} 0 \\ s \\ \kappa \end{bmatrix} = \begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ \tau \end{bmatrix}$$

$$s, y, \kappa, \tau \geq 0$$

$$\longrightarrow$$

$$Qu = v$$
$$u, v \geq 0$$

$$Q = \begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix}$$

$$u = (x, y, \tau)$$
$$v = (0, s, \kappa)$$

The homogeneous self-dual embedding

Properties

$$Qu = v$$
$$u, v \geq 0$$

$$Q = \begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix}$$

$$u = (x, y, \tau)$$
$$v = (0, s, \kappa)$$

The homogeneous self-dual embedding

Properties

$$Qu = v$$
$$u, v \geq 0$$

$$Q = \begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix}$$

$$u = (x, y, \tau)$$
$$v = (0, s, \kappa)$$

Matrix

- Q is **skew-symmetric**: $Q^T = -Q \quad \Rightarrow \quad u^T Q u = 0$
- $u \perp v$ **proof** $Qu - v = 0 \quad \Rightarrow \quad u^T Q u - u^T v = 0 \quad \Rightarrow \quad u^T v = 0 \quad \blacksquare$

The homogeneous self-dual embedding

Properties

$$Qu = v$$
$$u, v \geq 0$$

$$Q = \begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix}$$

$$u = (x, y, \tau)$$
$$v = (0, s, \kappa)$$

Matrix

- Q is **skew-symmetric**: $Q^T = -Q \quad \Rightarrow \quad u^T Q u = 0$
- $u \perp v$ **proof** $Qu - v = 0 \quad \Rightarrow \quad u^T Q u - u^T v = 0 \quad \Rightarrow \quad u^T v = 0 \quad \blacksquare$

Homogeneous

(u, v) satisfy $Qu = v,\ (v, u) \geq 0 \quad \Rightarrow \quad \alpha(u, v)$ with $\alpha \geq 0$ feasible

The homogeneous self-dual embedding

Properties

$$Qu = v$$
$$u, v \geq 0$$

$$Q = \begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix}$$

$$u = (x, y, \tau)$$
$$v = (0, s, \kappa)$$

Matrix

- Q is **skew-symmetric**: $Q^T = -Q \quad \Rightarrow \quad u^T Q u = 0$
- $u \perp v$ **proof** $Qu - v = 0 \quad \Rightarrow \quad u^T Q u - u^T v = 0 \quad \Rightarrow \quad u^T v = 0 \quad \blacksquare$

Homogeneous

(u, v) satisfy $Qu = v,\ (v, u) \geq 0 \quad \Rightarrow \quad \alpha(u, v)$ with $\alpha \geq 0$ feasible

Always feasible

$\alpha = 0 \quad \Rightarrow \quad (0, 0)$ is feasible

The homogeneous self-dual embedding

Outcomes

$$\begin{bmatrix} 0 \\ s \\ \kappa \end{bmatrix} = \begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ \tau \end{bmatrix}$$

$$s, y, \kappa, \tau \geq 0$$

Feasibility

$\tau > 0, \kappa = 0$ $\longrightarrow$ $(\hat{x}, \hat{s}, \hat{y}) = (x^\star/\tau, s^\star/\tau, y^\star/\tau)$ is a **solution** to the original problem

The homogeneous self-dual embedding

Outcomes

$$\begin{bmatrix} 0 \\ s \\ \kappa \end{bmatrix} = \begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ \tau \end{bmatrix}$$

$$s, y, \kappa, \tau \geq 0$$

Feasibility

$\tau > 0, \kappa = 0 \longrightarrow (\hat{x}, \hat{s}, \hat{y}) = (x^\star/\tau, s^\star/\tau, y^\star/\tau)$ is a **solution** to the original problem

Infeasibility

$\tau = 0, \kappa > 0 \longrightarrow c^T x + b^T y < 0$ (**impossible**). Must have infeasibility

The homogeneous self-dual embedding

Outcomes

$$\begin{bmatrix} 0 \\ s \\ \kappa \end{bmatrix} = \begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ \tau \end{bmatrix}$$

$$s, y, \kappa, \tau \geq 0$$

Feasibility

$\tau > 0, \kappa = 0 \longrightarrow (\hat{x}, \hat{s}, \hat{y}) = (x^\star/\tau, s^\star/\tau, y^\star/\tau)$ is a **solution** to the original problem

Infeasibility

$\tau = 0, \kappa > 0 \longrightarrow c^T x + b^T y < 0$ (**impossible**). Must have infeasibility

If $b^T y < 0$ then $\hat{y} = y/(-b^T y)$ is a **certificate of primal infeasibility**

$$A^T \hat{y} = 0, \quad b^T \hat{y} = -1 < 0, \quad \hat{y} \geq 0$$

The homogeneous self-dual embedding

Outcomes

$$\begin{bmatrix} 0 \\ s \\ \kappa \end{bmatrix} = \begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ \tau \end{bmatrix}$$

STRICT COMPLEMENTARITY

$\tau + \kappa > 0$

$$s, y, \kappa, \tau \geq 0$$

Feasibility

$\tau > 0, \kappa = 0 \longrightarrow (\hat{x}, \hat{s}, \hat{y}) = (x^\star/\tau, s^\star/\tau, y^\star/\tau)$ is a **solution** to the original problem

Infeasibility

$\tau = 0, \kappa > 0 \longrightarrow c^T x + b^T y < 0$ (**impossible**). Must have infeasibility

If $b^T y < 0$ then $\hat{y} = y/(-b^T y)$ is a **certificate of primal infeasibility**

$$A^T \hat{y} = 0, \quad b^T \hat{y} = -1 < 0, \quad \hat{y} \geq 0$$

If $c^T x < 0$ then $\hat{x} = x/(-c^T y)$ is a **certificate of dual infeasibility**

$$A\hat{x} \leq 0, \quad c^T \hat{x} = -1 < 0$$

Self-dual problem

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & Qu = v \\ & u, v \geq 0 \end{array}$$

Q skew-symmetric: $Q^T = -Q$

The dual is identical

Self-dual problem

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & Qu = v \\ & u, v \geq 0 \end{array}$$

Q skew-symmetric: $Q^T = -Q$

The dual is identical

Proof

$$g(\nu, \lambda, \mu) = \underset{u,v}{\text{minimize}}\ \mathcal{L}(u, v, \nu, \lambda, \mu) = \nu^T(Qu - v) - \lambda^T u - \mu^T v$$

Self-dual problem

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & Qu = v \\ & u, v \geq 0 \end{array}$$

Q skew-symmetric: $Q^T = -Q$

The dual is identical

Proof

$$g(\nu, \lambda, \mu) = \underset{u,v}{\text{minimize}}\ \mathcal{L}(u, v, \nu, \lambda, \mu) = \nu^T(Qu - v) - \lambda^T u - \mu^T v$$

$$\frac{\partial \mathcal{L}}{\partial u} = Q^T \nu - \lambda = 0$$

$$\frac{\partial \mathcal{L}}{\partial v} = -\nu - \mu = 0 \quad \Rightarrow \quad \nu = -\mu$$

Self-dual problem

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & Qu = v \\ & u, v \geq 0 \end{array}$$

Q skew-symmetric: $Q^T = -Q$

The dual is identical

Proof

$$g(\nu, \lambda, \mu) = \underset{u,v}{\text{minimize}}\ \mathcal{L}(u, v, \nu, \lambda, \mu) = \nu^T(Qu - v) - \lambda^T u - \mu^T v$$

$$\frac{\partial \mathcal{L}}{\partial u} = Q^T \nu - \lambda = 0$$

$$\frac{\partial \mathcal{L}}{\partial v} = -\nu^T - \mu = 0 \quad \Rightarrow \quad \nu = -\mu$$

$\longrightarrow$

Dual

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & Q\mu = \lambda \\ & \mu, \lambda \geq 0 \end{array}$$

$\blacksquare$

Interior-point method for homogeneous self-dual embedding

Complementarity problem

$$
\begin{aligned}
&Qu = v \\
&u^T v = 0 \\
&u, v \geq 0
\end{aligned}
$$

Equations

$$
h(u, v) = \begin{bmatrix} Qu - v \\ UV\mathbf{1} \end{bmatrix} = 0
$$

$$
u, v \geq 0
$$

Interior-point method for homogeneous self-dual embedding

Complementarity problem

$$
\begin{aligned}
&Qu = v \\
&u^T v = 0 \\
&u, v \geq 0
\end{aligned}
$$

Equations

$$
h(u, v) = \begin{bmatrix} Qu - v \\ UV\mathbf{1} \end{bmatrix} = 0
$$

$$
u, v \geq 0
$$

Directions

$$
\begin{bmatrix} Q & -I \\ U & V \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} = \begin{bmatrix} -r_e \\ -UV\mathbf{1} + \sigma\mu\mathbf{1} \end{bmatrix}
$$

$$
\begin{aligned}
r_e &= Qu - v \\
\mu &= (u^T v)/d
\end{aligned}
$$

Interior-point method for homogeneous self-dual embedding

Complementarity problem

$$
\begin{aligned}
&Qu = v \\
&u^T v = 0 \\
&u, v \geq 0
\end{aligned}
$$

Equations

$$
h(u, v) = \begin{bmatrix} Qu - v \\ UV\mathbf{1} \end{bmatrix} = 0
$$

$$
u, v \geq 0
$$

Directions

$$
\begin{bmatrix} Q & -I \\ U & V \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} = \begin{bmatrix} -r_e \\ -UV\mathbf{1} + \sigma\mu\mathbf{1} \end{bmatrix} \qquad \begin{aligned} r_e &= Qu - v \\ \mu &= (u^T v)/d \end{aligned}
$$

Line search to enforce $u, v > 0$

$$
(u, v) \leftarrow (u, v) + \alpha(\Delta u, \Delta v)
$$

Interior-point method for homogeneous self-dual embedding

Complementarity problem

$$
\begin{aligned}
Qu &= v \\
u^T v &= 0 \\
u, v &\geq 0
\end{aligned}
$$

Equations

$$
h(u, v) = \begin{bmatrix} Qu - v \\ UV\mathbf{1} \end{bmatrix} = 0
$$

$$
u, v \geq 0
$$

Directions

$$
\begin{bmatrix} Q & -I \\ U & V \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} = \begin{bmatrix} -r_e \\ -UV\mathbf{1} + \sigma\mu\mathbf{1} \end{bmatrix} \qquad \begin{aligned} r_e &= Qu - v \\ \mu &= (u^T v)/d \end{aligned}
$$

Line search to enforce $u, v > 0$

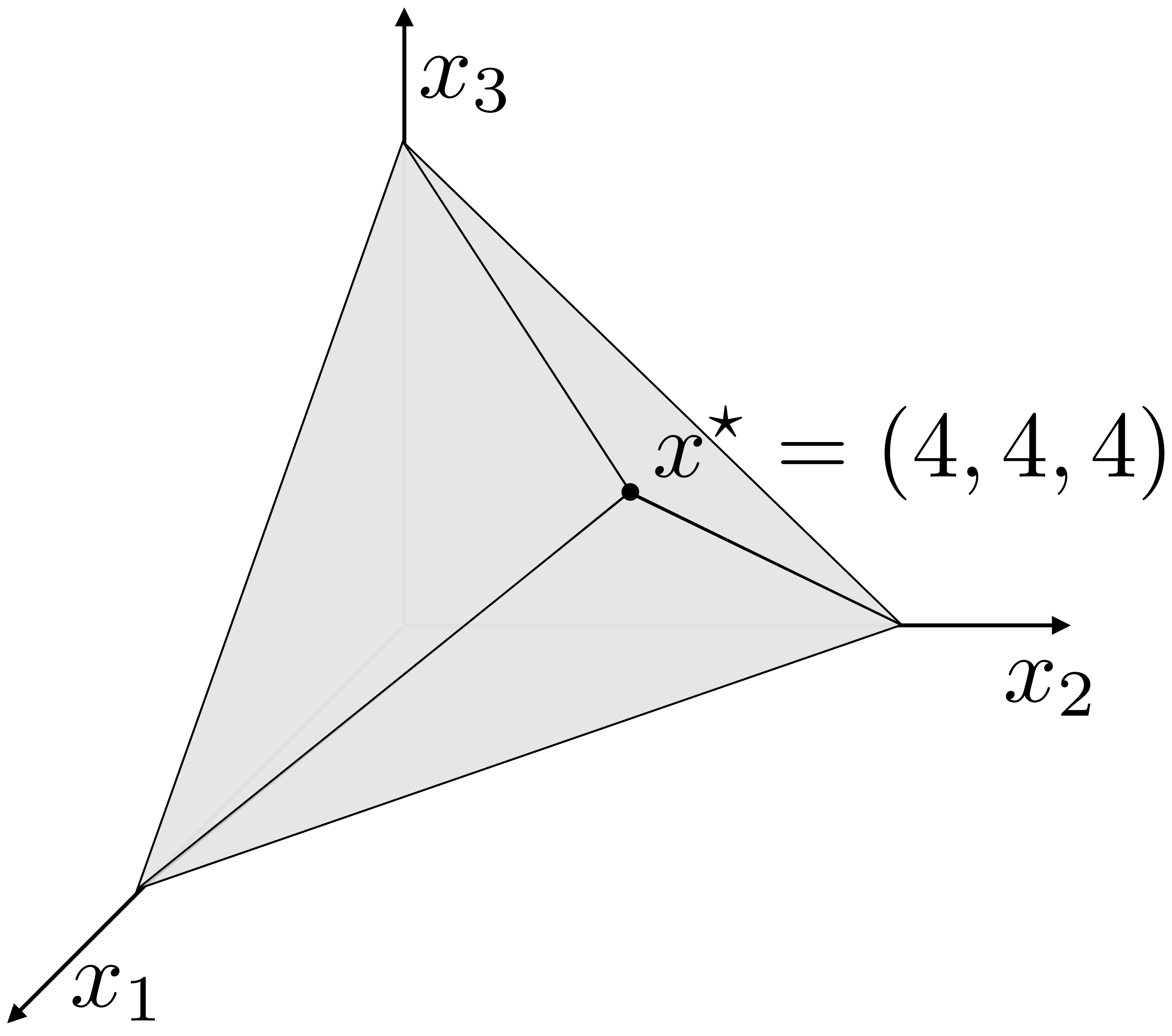
$$
(u, v) \leftarrow (u, v) + \alpha(\Delta u, \Delta v)
$$

Interior-point methods can solve **linear complementarity problems**

Interior-point vs simplex

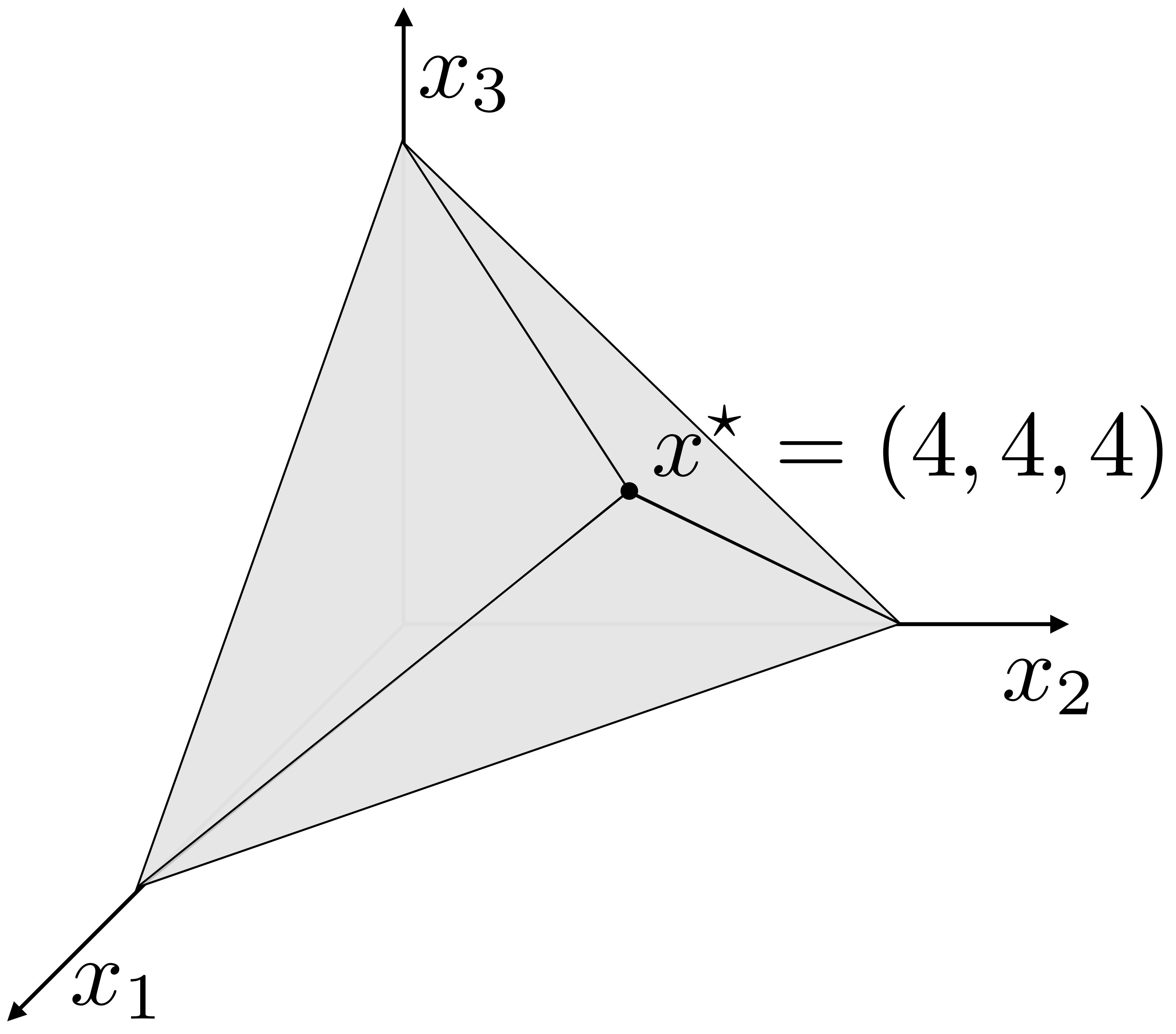
Example

$$
\begin{array}{ll}
\text{minimize} & -10x_1 - 12x_2 - 12x_3 \\
\text{subject to} & x_1 + 2x_2 + 2x_3 \leq 20 \\
& 2x_1 + x_2 + x_3 \leq 20 \\
& 2x_1 + 2x_2 + x_3 \leq 20 \\
& x_1, x_2, x_3 \geq 0
\end{array}
$$



Example

$$\begin{array}{ll} \text{minimize} & -10x_1 - 12x_2 - 12x_3 \\ \text{subject to} & x_1 + 2x_2 + 2x_3 \leq 20 \\ & 2x_1 + x_2 + x_3 \leq 20 \\ & 2x_1 + 2x_2 + x_3 \leq 20 \\ & x_1, x_2, x_3 \geq 0 \end{array}$$



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

$$c = (-10, -12, -12)$$

$$A = \begin{bmatrix} 1 & 2 & 2 \\ 2 & 1 & 2 \\ 2 & 2 & 1 \end{bmatrix}$$

$$b = (20, 20, 20)$$

Example with real solver

CVXOPT (open-source)

$\frac{\kappa}{\tau}$

Code

```
import numpy as np
import cvxpy as cp

c = np.array([-10, -12, -12])
A = np.array([[1, 2, 2],
              [2, 1, 2],
              [2, 2, 1]])
b = np.array([20, 20, 20])
n = len(c)

x = cp.Variable(n)
problem = cp.Problem(cp.Minimize(c @ x),
                     [A @ x <= b, x >= 0])
problem.solve(solver=cp.CVXOPT, verbose=True)
```

Output

```
     pcost       dcost       gap    pres   dres   k/t
 0: -1.3077e+02 -2.3692e+02  2e+01  1e-16  6e-01  1e+00
 1: -1.3522e+02 -1.4089e+02  1e+00  2e-16  3e-02  4e-02
 2: -1.3599e+02 -1.3605e+02  1e-02  2e-16  3e-04  4e-04
 3: -1.3600e+02 -1.3600e+02  1e-04  1e-16  3e-06  4e-06
 4: -1.3600e+02 -1.3600e+02  1e-06  1e-16  3e-08  4e-08
Optimal solution found.
```

Solution

```
In [3]: x.value
Out[3]: array([3.99999999, 4.        , 4.        ])
```

NOT BFS !

[The CVXOPT linear and quadratic cone program solvers, L. Vandenberghe 2010]

Average interior-point complexity

Random LPs

minimize $c^T x$

subject to $Ax \leq b$

n variables

$3n$ constraints

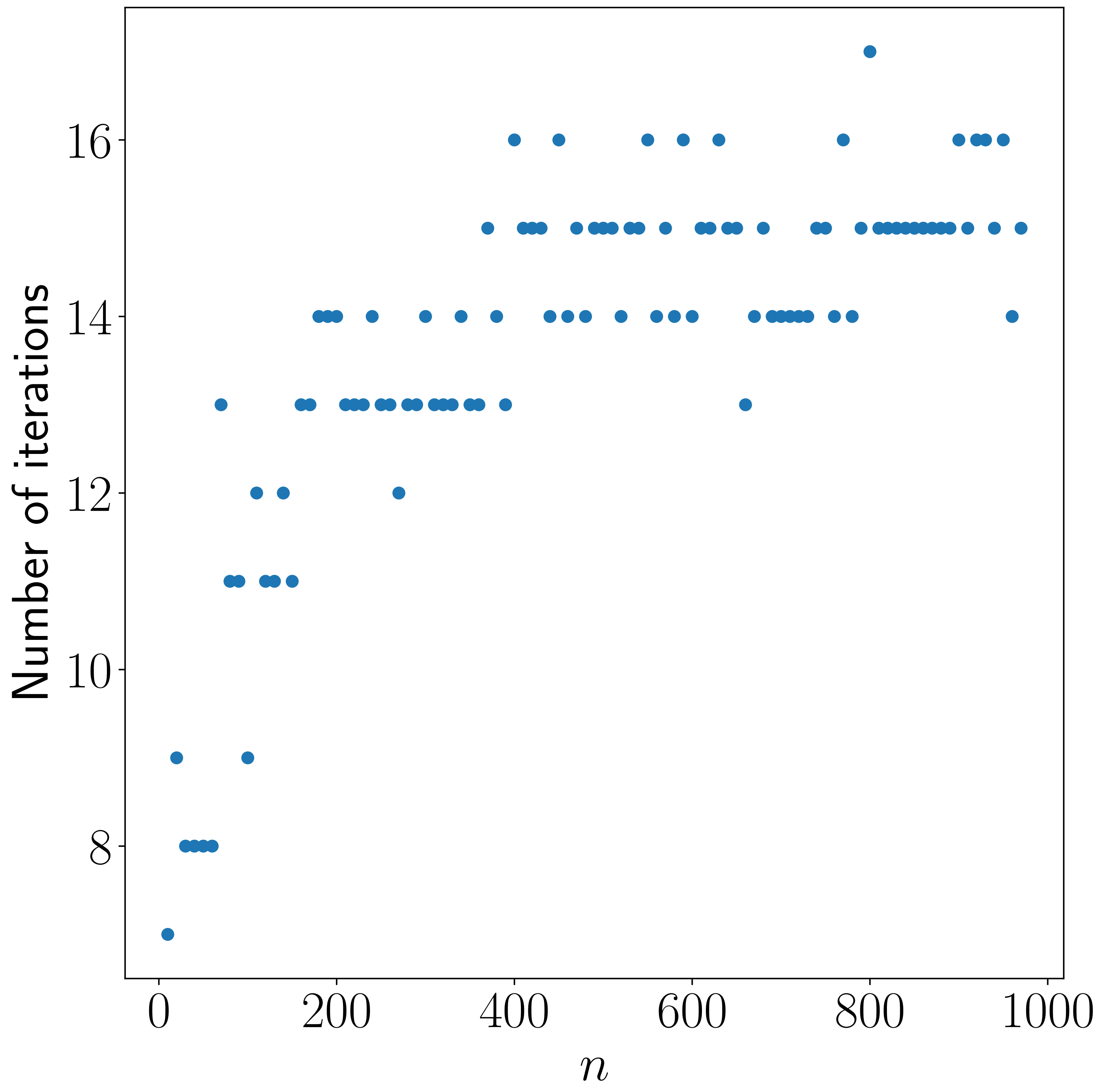
Average interior-point complexity

Random LPs

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

n variables
$3n$ constraints

Iterations: $O(\log n)$



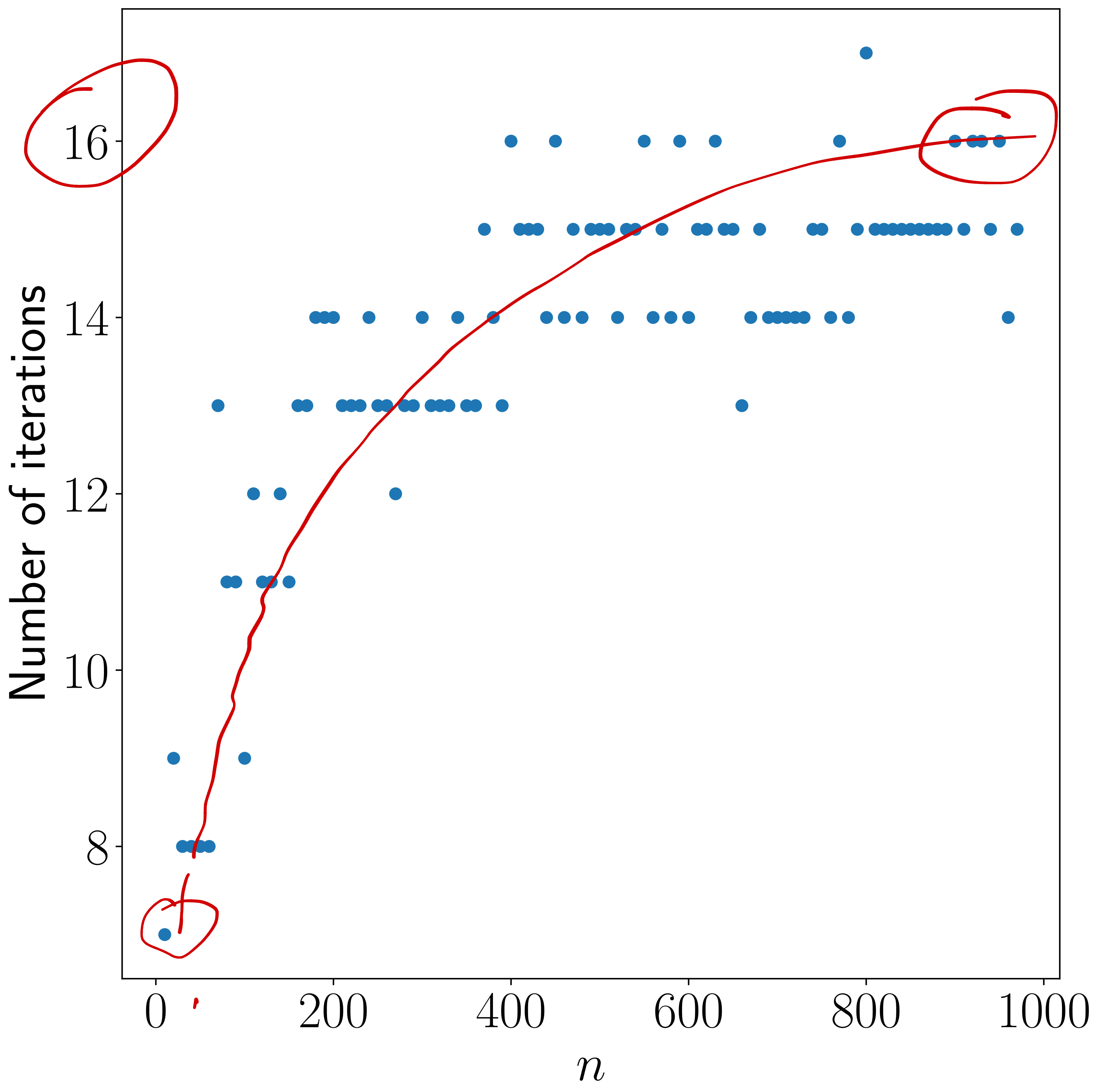
Average interior-point complexity

Random LPs

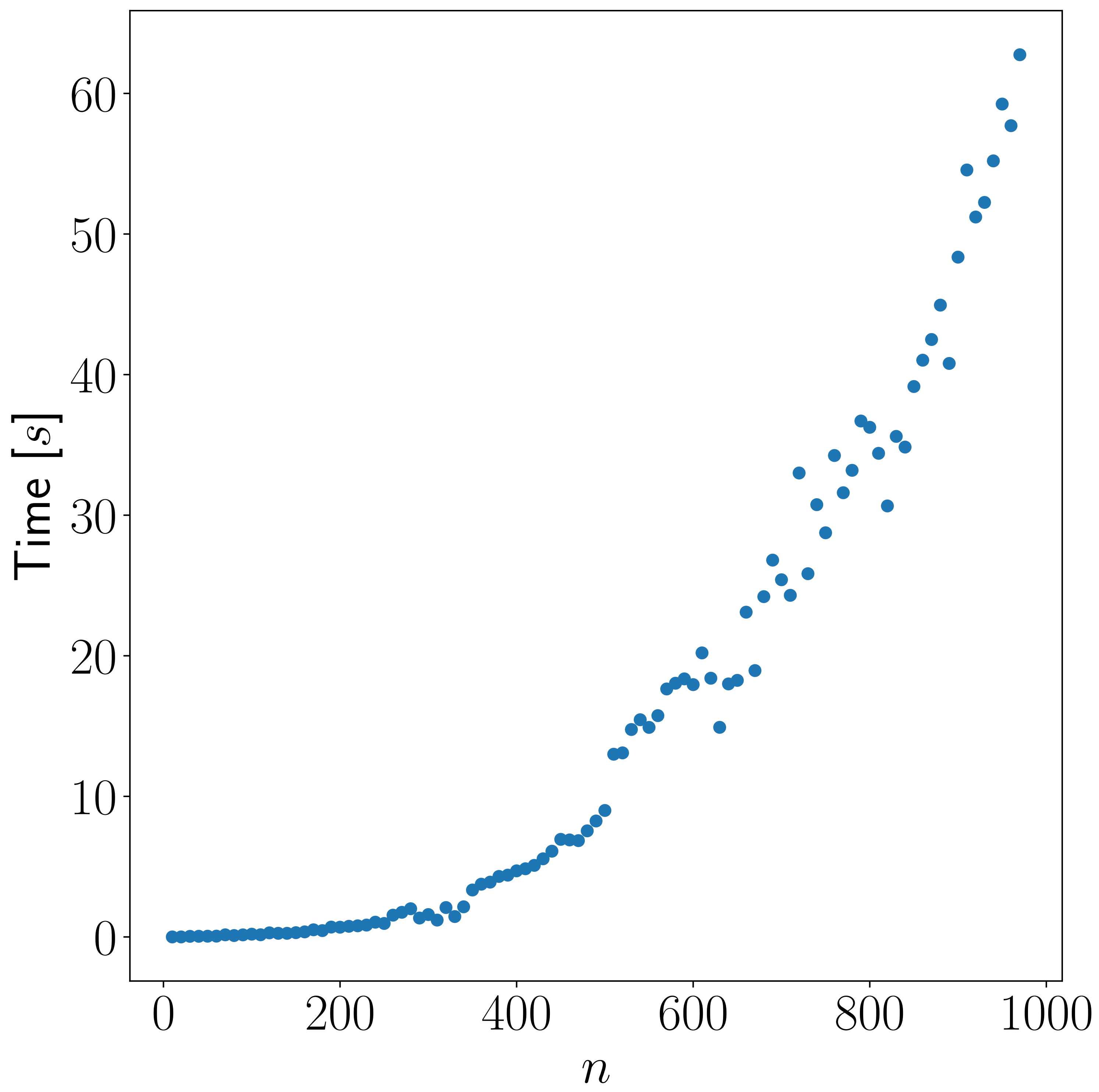
$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \end{array}$$

n variables
$3n$ constraints

Iterations: $O(\log n)$



Time: $O(n^3 \log n)$



Comparison between interior-point method and simplex

Primal simplex

- Primal feasibility
- Zero duality gap

↓

Dual feasibility

Dual simplex

- Dual feasibility
- Zero duality gap

↓

Primal feasibility

Primal-dual interior-point

- Interior condition

↓

- Primal feasibility
- Dual feasibility
- Zero duality gap

Comparison between interior-point method and simplex

Primal simplex

- Primal feasibility
- Zero duality gap

↓

Dual feasibility

Dual simplex

- Dual feasibility
- Zero duality gap

↓

Primal feasibility

Exponential worst-case complexity

Primal-dual interior-point

- Interior condition

↓

- Primal feasibility
- Dual feasibility
- Zero duality gap

Polynomial worst-case complexity

Comparison between interior-point method and simplex

Primal simplex

- Primal feasibility
- Zero duality gap

↓

Dual feasibility

Dual simplex

- Dual feasibility
- Zero duality gap

↓

Primal feasibility

Primal-dual interior-point

- Interior condition

↓

- Primal feasibility
- Dual feasibility
- Zero duality gap

Exponential worst-case complexity

Requires feasible point

Polynomial worst-case complexity

Allows infeasible start

Comparison between interior-point method and simplex

Primal simplex

- Primal feasibility
- Zero duality gap

↓

Dual feasibility

Dual simplex

- Dual feasibility
- Zero duality gap

↓

Primal feasibility

Primal-dual interior-point

- Interior condition

↓

- Primal feasibility
- Dual feasibility
- Zero duality gap

Simplex	Primal-dual interior-point
Exponential worst-case complexity	**Polynomial worst-case complexity**
Requires feasible point	**Allows infeasible start**
Can be warm-started	**Cannot be warm-started**

Which algorithm should I use?

Dual simplex

- Small-to-medium problems
- Repeated solves with varying data

Interior-point (barrier)

- Medium-to-large problems
- Sparse structured problems

Which algorithm should I use?

Dual simplex

- Small-to-medium problems
- Repeated solves with varying data

Interior-point (barrier)

- Medium-to-large problems
- Sparse structured problems

How do solvers with multiple options decide?

<u>Concurrent Optimization</u>

Which algorithm should I use?

Dual simplex

- Small-to-medium problems
- Repeated solves with varying data

Interior-point (barrier)

- Medium-to-large problems
- Sparse structured problems

How do solvers with multiple options decide?

<u>Concurrent Optimization</u>

MOSEK, KNITRO

Why not both? (crossover)

Interior-point $\longrightarrow$ Few simplex steps

Interior-point methods implementation

Today, we learned to:

- **Apply** Mehrotra predictor-corrector algorithm
- **Exploit** linear algebra to speedup computations
- **Detect** infeasibility/unboundedness with homogeneous self-dual embedding
- **Analyze** empirical complexity
- **Compare** interior-point and simplex methods

Next lecture

- Introduction to nonlinear optimization