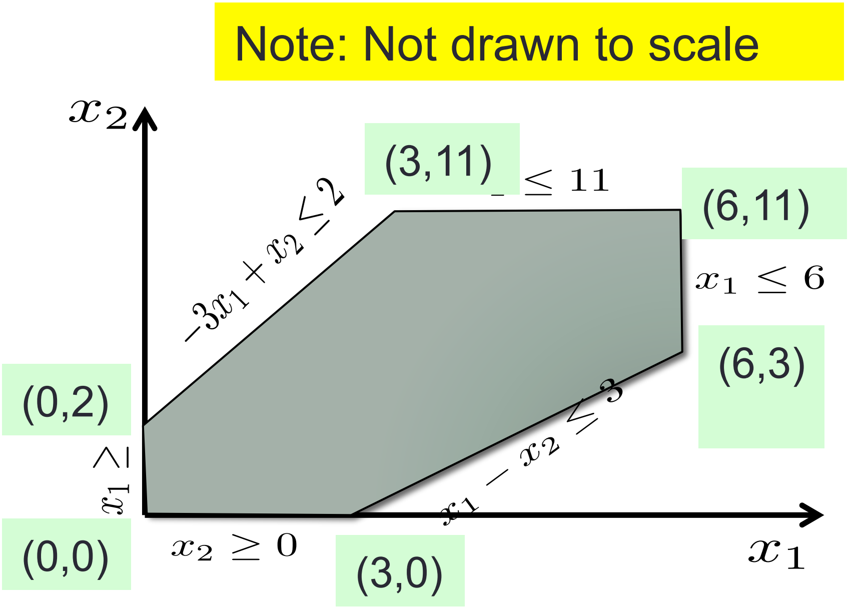
INTERIOR POINT METHODS: BASIC INTRODUCTION

Linear Programming Problem

$$\begin{array}{rrrcl} \text{max.} & x_1 & +2x_2 & & \\ \text{s.t.} & -3x_1 & +x_2 & \leq & 2 \\ & & +x_2 & \leq & 11 \\ & x_1 & -x_2 & \leq & 3 \\ & x_1 & & \leq & 6 \\ & x_1, & x_2 & \geq & 0 \end{array}$$



Karush-Kuhn-Tucker Conditions

- Very important for many optimization problems.

$$\begin{array}{lrcl} \max & \mathbf{c}^\intercal \mathbf{x} & & \\ & A\,\mathbf{x} + \mathbf{x_s} & = & \mathbf{b} \\ & \mathbf{x}, \mathbf{x_s} & \geq & \mathbf{0} \end{array} \qquad \begin{array}{lrcl} \min & \mathbf{b}^\intercal \mathbf{y} & & \\ & A^\intercal\,\mathbf{y} - \mathbf{y_s} & = & \mathbf{c} \\ & \mathbf{y}, \mathbf{y_s} & \geq & \mathbf{0} \end{array}$$

Primal

Dual

Necessary and Sufficient Conditions for optimal solution

$$(\mathbf{x}, \mathbf{x_s}, \mathbf{y}, \mathbf{y_s})$$

KKT conditions for Linear Programs

The primal-dual solution $(\mathbf{x}, \mathbf{x_s}, \mathbf{y}, \mathbf{y_s})$ is optimal iff it satisfies the following conditions:

$$\begin{array}{rcl} A\,\mathbf{x} + \mathbf{x_s} & = & \mathbf{b} \\ \mathbf{x}, \mathbf{x_s} & \geq & \mathbf{0} \end{array}$$

(x,x_s) is primal feasible

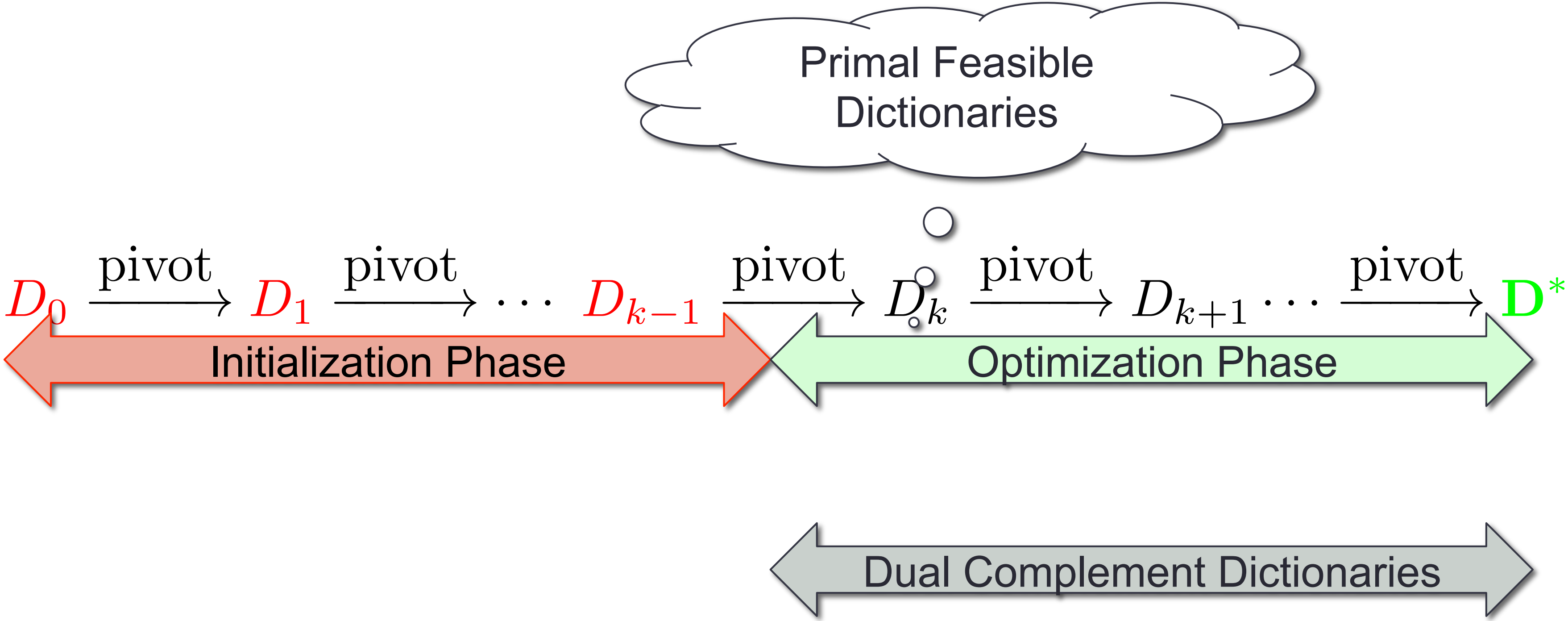
$$\begin{array}{rcl} A^\intercal\,\mathbf{y} - \mathbf{y_s} & = & \mathbf{c} \\ \mathbf{y}, \mathbf{y_s} & \geq & \mathbf{0} \end{array}$$

(y,y_s) is dual feasible

$$x_j y_{s,j} = 0$$
$$y_j x_{s,j} = 0$$

Product of complementary pairs is zero.

Simplex Method: Overview



Simplex Method

- Sequence of primal dual solutions (dictionaries)

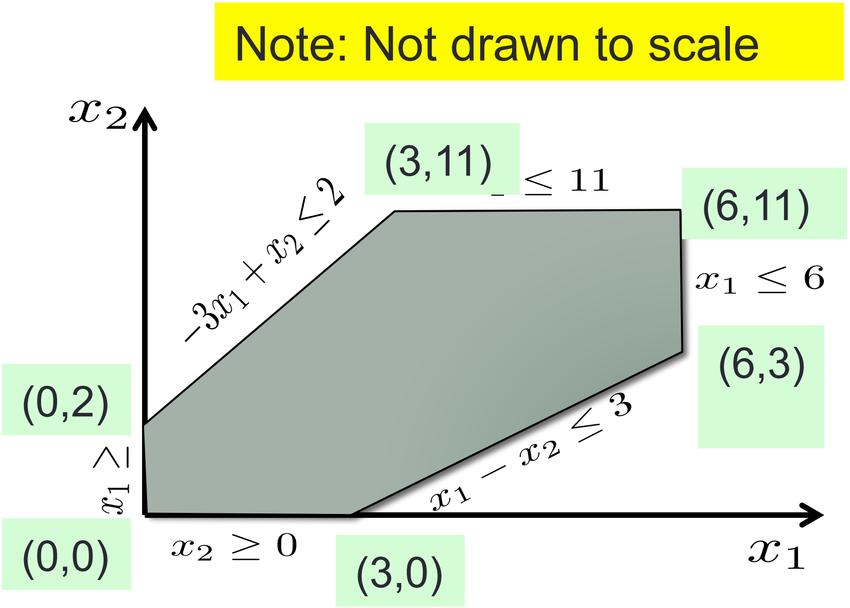
$$\boxed{(\mathbf{x}_0, \mathbf{y}_0) \rightarrow (\mathbf{x}_1, \mathbf{y}_1) \rightarrow \cdots \rightarrow (\mathbf{x}^*, \mathbf{y}^*)}$$

- Maintain Primal Feasibility.
- Maintain Complementarity Conditions.
- Solutions are vertices of primal/dual feasible regions.
- Dual Feasibility achieved only at the very end.

Interior Point Methods

A class of methods.

- Central Path methods
- Affine Scaling Method
- Active Set
- …



Converges to solution vs. Find the precise answer.

Interior Point Methods

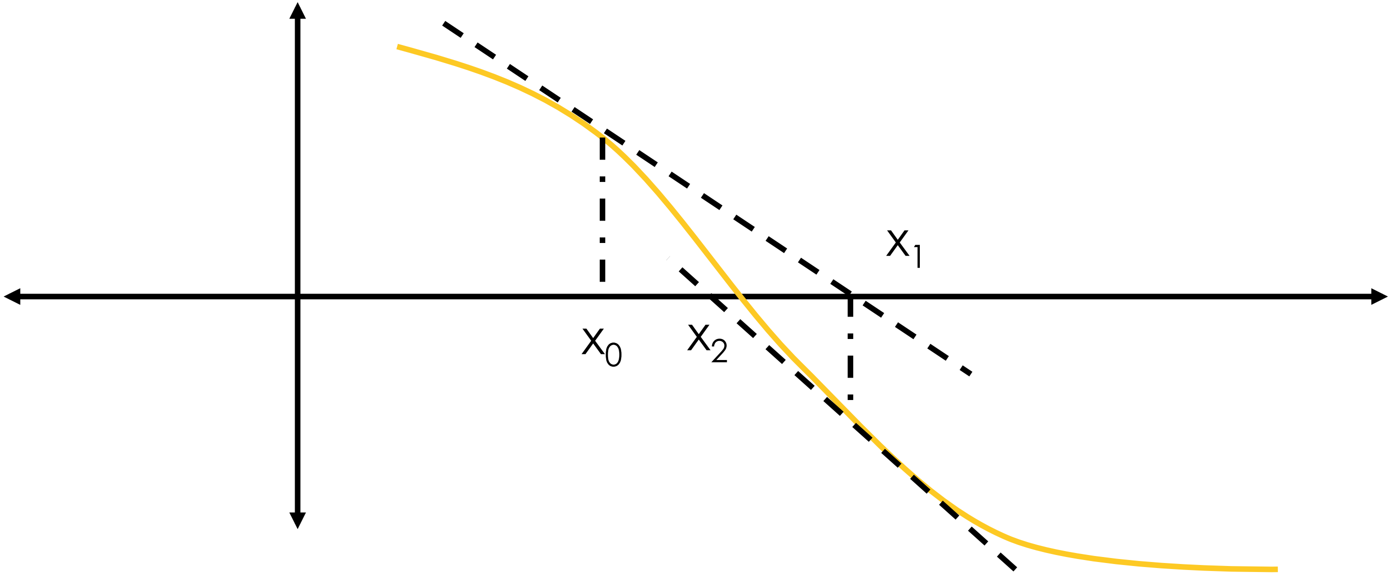
- We will consider a simple central path method.
- Our presentation sequence:
  - Newton’s Method for Solving Equations.
  - Relaxed (mu) complementarity conditions.
  - Central Path.
  - Computing the Newton Step.
  - Adjusting Step Size.
  - Some experiments.

NEWTON'S METHOD

Basic Goal

Solve the (system) of equations: $F(\mathbf{x}) = 0$

$F(\mathbf{x})$ assumed continuous and differentiable (smooth)



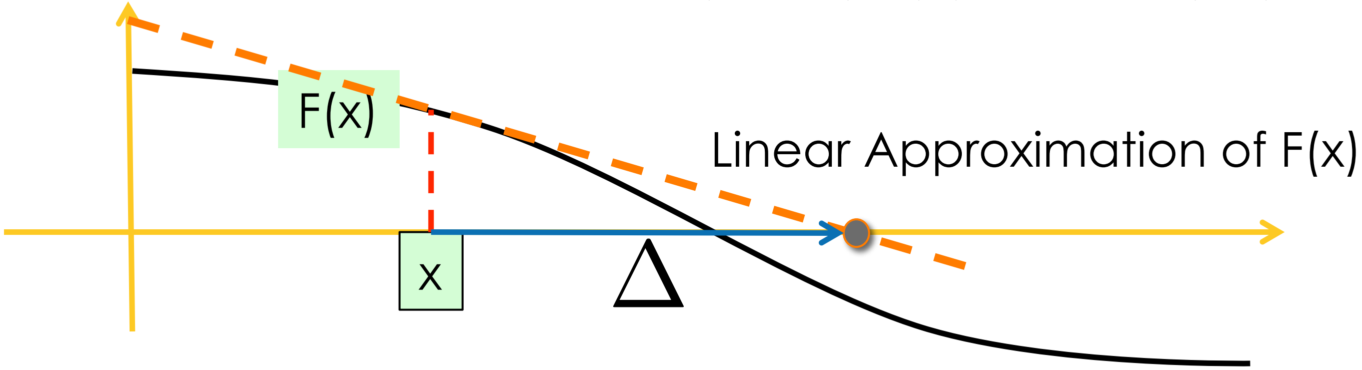
Newton Step

Currently, $F(\mathbf{x}) \neq 0$

Linear Approximation: $F(\mathbf{x}+\Delta) \simeq \; F(\mathbf{x}) + F'(\mathbf{x})\Delta$

$$F(\mathbf{x}) + F'(\mathbf{x})\Delta = \mathbf{0}$$

$$\Delta = -(F'(\mathbf{x}))^{-1}F(\mathbf{x})$$



Newton Step (n-dimensions)

$$F : \mathbb{R}^n \rightarrow \mathbb{R}^n$$

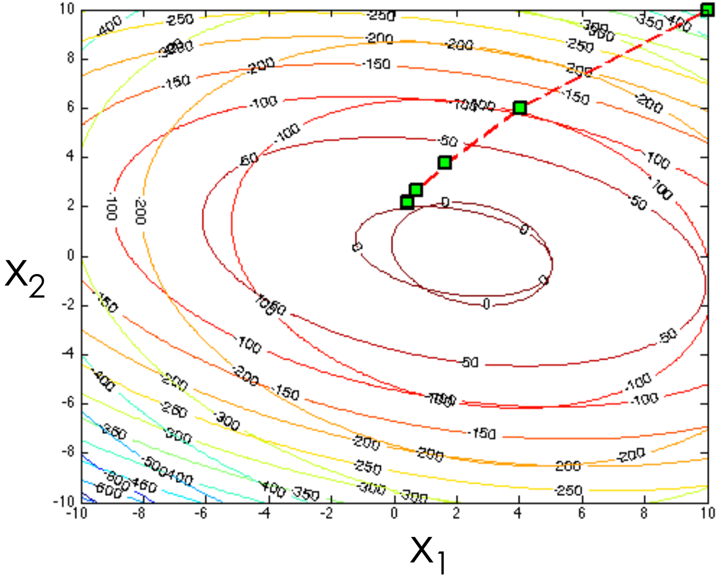
$$F(x) = \begin{bmatrix} F_1(\mathbf{x}) \\ \vdots \\ F_n(\mathbf{x}) \end{bmatrix}$$

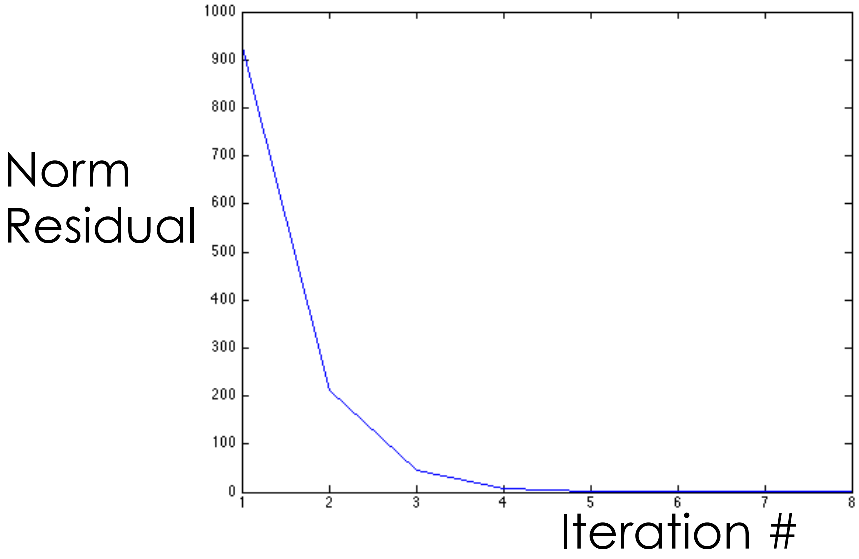
$$F'(\mathbf{x}) = \begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \frac{\partial F_1}{\partial x_2} & \cdots & \frac{\partial F_1}{\partial x_n} \\ \frac{\partial F_2}{\partial x_1} & \frac{\partial F_2}{\partial x_2} & \cdots & \frac{\partial F_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial x_1} & \frac{\partial F_n}{\partial x_2} & \cdots & \frac{\partial F_n}{\partial x_n} \end{bmatrix}$$

$$\Delta = -(F'(\mathbf{x}))^{-1} F(\mathbf{x})$$

Newton's method example

$$f(x_1, x_2) = \begin{pmatrix} -x_1^2 - 3x_2^2 - x_1x_2 + 3x_2 + 4x_1 + 5 \\ -2x_1^2 - 3x_2^2 - x_1x_2 + 10x_1 + 3x_2 \end{pmatrix} = 0$$





Newton Method Convergence

- If method converges, it does so to a root of F(x).
- Convergence is not guaranteed.
  - Only if starting point in the "basin of attraction" of the root.
  - F'(x) should not vanish at the root ("simple" root).
- Convergence is quadratic.
  - Often very fast when it does converge.

Complexity of each newton step.

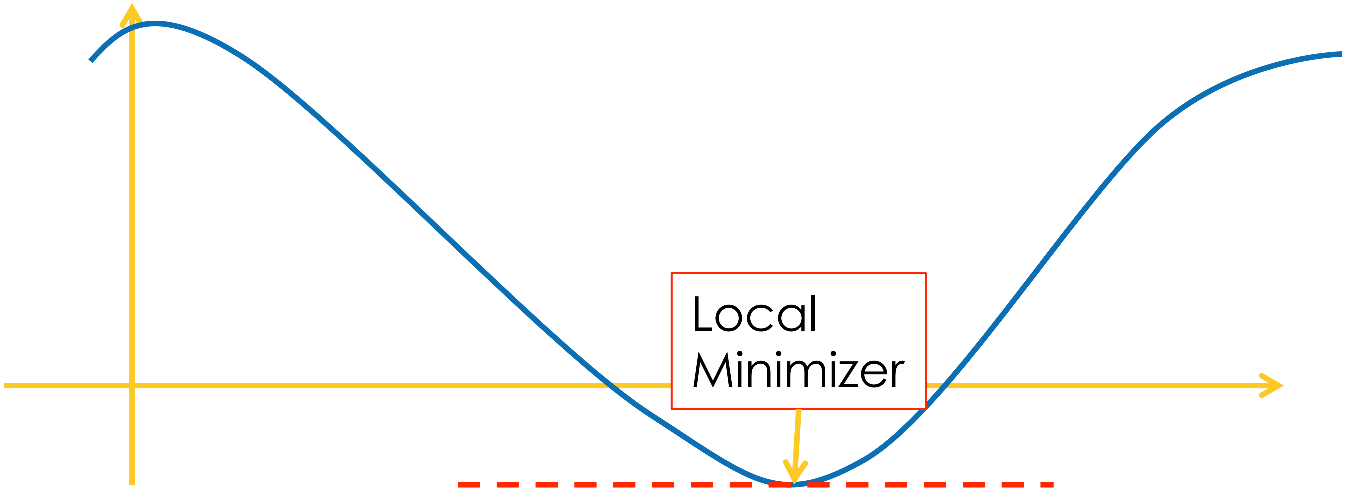
- Goal: Compute root of F(x) using Newton's method.

- Compute the Jacobian F'(x)

- Newton Step:
  - Invert the Jacobian and multiply with value of function.

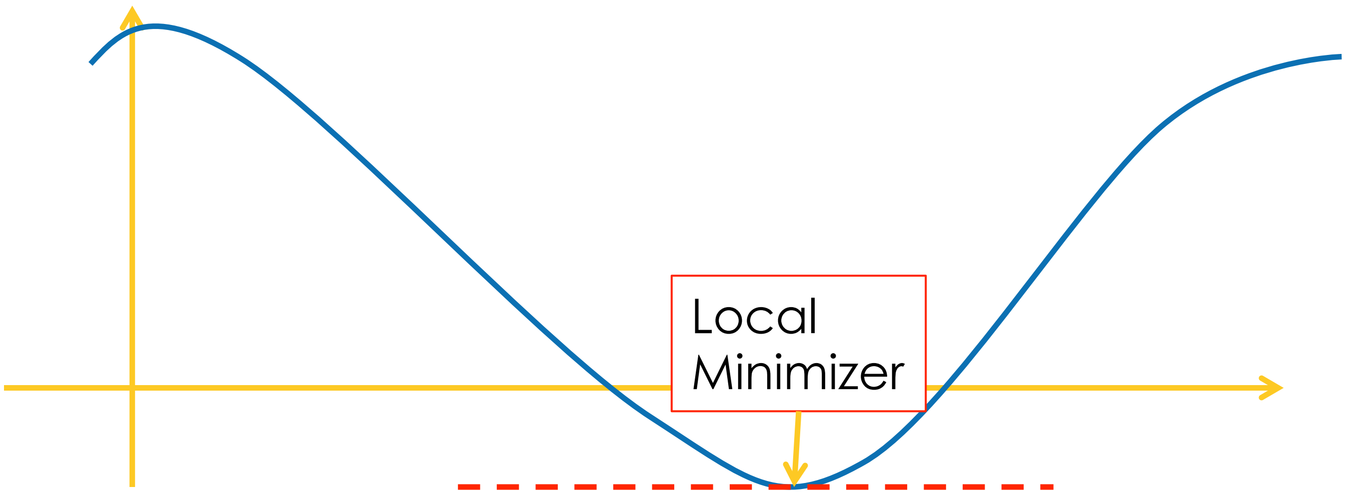
Newton Method for Optimization

- Goal: minimize function F(x) for all x.
- Unconstrained minimization problem.



Newton Method for Optimization

- Goal: minimize function F(x) for all x.
- Unconstrained minimization problem.



Minimization of smooth function.

$$F : \mathbb{R}^n \rightarrow \mathbb{R}$$

- F is a C^2 function.
- Continuous, first and second derivatives.

If $\mathbf{x} \in \mathbb{R}^n$ is a local minimizer of F then $\nabla F = 0$

First-Order Necessary Conditions

If $\nabla F(\mathbf{x}) = 0$ and $\nabla^2 F$ is positive definite at $\mathbf{x}$, then $\mathbf{x}$ is an isolated local minimum of F.

Second-Order Sufficient Condition

Newton method for finding minima

$$F : \mathbb{R}^n \rightarrow \mathbb{R}$$

$$\boxed{\nabla F(\mathbf{x}) = \begin{pmatrix} \frac{\partial F}{\partial x_1} \\ \vdots \\ \frac{\partial F}{\partial x_n} \end{pmatrix} = 0}$$

Solve

$$\text{Newton Step: } \Delta = \boxed{-(\nabla^2 F)^{-1}}(\nabla F)$$

Hessian Matrix

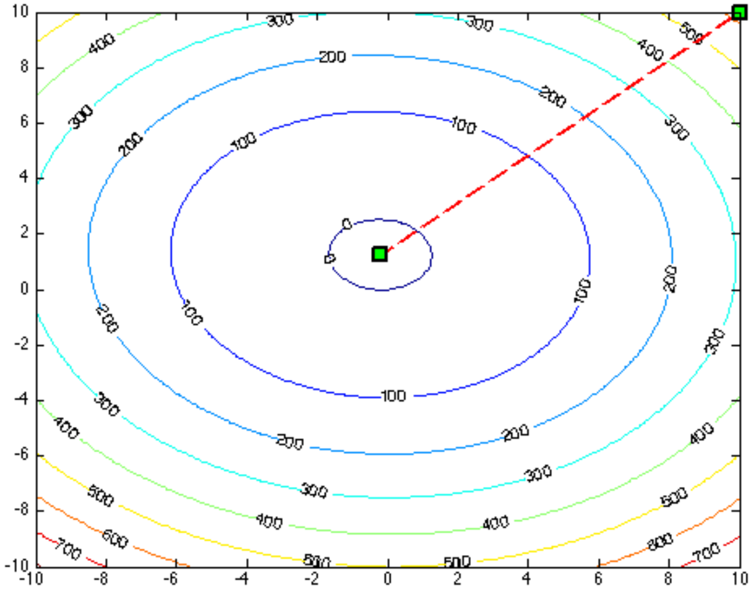
Hessian Matrix

$$\nabla^2 F = \begin{bmatrix} \frac{\partial^2 F}{\partial \ x_1^2} & \frac{\partial^2 F}{\partial \ x_1 x_2} & \cdots & \frac{\partial^2 F}{\partial \ x_1 x_n} \\ \frac{\partial^2 F}{\partial \ x_2 x_1} & \frac{\partial^2 F}{\partial \ x_2^2} & \cdots & \frac{\partial^2 F}{\partial \ x_2 x_n} \\ \vdots & & & \vdots \\ \frac{\partial^2 F}{\partial \ x_n x_1} & \frac{\partial^2 F}{\partial \ x_n x_2} & \cdots & \frac{\partial^2 F}{\partial \ x_n^2} \end{bmatrix}$$

Does inverse always exist?

$$\text{Newton Step: } \Delta = -(\nabla^2 F)^{-1}(\nabla F)$$

Newton's Method Example



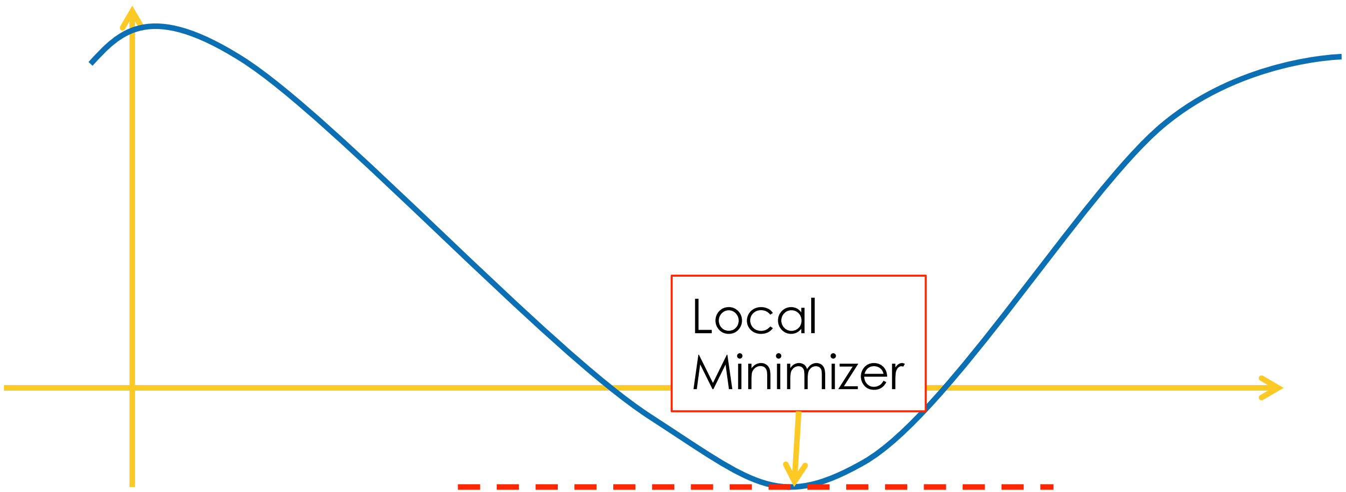
$$\min_{(x,y)}(3x^2 + 4y^2 + 0.2xy + x - 10y)$$

EQUALITY CONSTRAINED OPTIMIZATION

Lagrange Multiplier Method

Unconstrained Optimization

- Goal: minimize function F(x) for all x.
- Unconstrained minimization problem.



Equality Constrained Optimization

$$
\begin{aligned}
\min \quad & f(\mathbf{x}) \\
\text{s.t.} \quad & g_1(\mathbf{x}) = 0 \\
& g_2(\mathbf{x}) = 0 \\
& \vdots \\
& g_m(\mathbf{x}) = 0
\end{aligned}
$$

$$L(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) + \sum_{i=1}^{m} y_i g_i(\mathbf{x})$$

$$
\boxed{
\begin{aligned}
\nabla_{\mathbf{x}} L(\mathbf{x}, \mathbf{y}) &= 0 \\
\nabla_{\mathbf{y}} L(\mathbf{x}, \mathbf{y}) &= 0
\end{aligned}
}
$$

First order Necessary Conditions

Lagrange Multiplier Method

$$L(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) + \sum_{i=1}^{m} y_i g_i(\mathbf{x})$$

$$\boxed{\begin{aligned} \nabla_{\mathbf{x}} L(\mathbf{x}, \mathbf{y}) &= 0 \\ \nabla_{\mathbf{y}} L(\mathbf{x}, \mathbf{y}) &= 0 \end{aligned}}$$

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & g_1(\mathbf{x}) = 0 \\ & g_2(\mathbf{x}) = 0 \\ & \vdots \\ & g_m(\mathbf{x}) = 0 \end{aligned}$$

$$\frac{\partial f}{\partial x_j} + \sum_{i=1}^{m} y_i \frac{\partial g_i}{\partial x_j} = 0$$

$$g_i(\mathbf{x}) = 0$$

Solve using Newton's method

Example

$$\min \ \sin(x) + \cos(y) + z^2$$
$$\text{s.t. } x^2 + y^2 + z^2 = 1$$

$$L(x, y, z, \lambda) = \sin(x) + \cos(y) + z^2 + \lambda(x^2 + y^2 + z^2 - 1)$$

$$\begin{aligned} \cos(x) + 2\lambda x &= 0 \\ -\sin(y) + 2\lambda y &= 0 \\ 2z + 2\lambda z &= 0 \\ x^2 + y^2 + z^2 &= 1 \end{aligned}$$

Solve using Newton's method

LOG BARRIER METHOD.

Linear Programming Formulation

$$\begin{aligned} \max \quad & \mathbf{c}^{\intercal}\ \mathbf{x} & & \\ & A\mathbf{x} & \leq & \ \mathbf{b} \\ & \mathbf{x} & \geq & \ 0 \end{aligned}$$

$$\begin{aligned} \max \quad & \mathbf{c}^{\intercal}\ \mathbf{x} & & \\ & A\mathbf{x} + \mathbf{x}_s & = & \ \mathbf{b} \\ & \mathbf{x}, \mathbf{x_s} & \geq & \ 0 \end{aligned}$$

Primal Standard form with Slack Variables

Log Barrier Trick

Inequality constrained optimization:

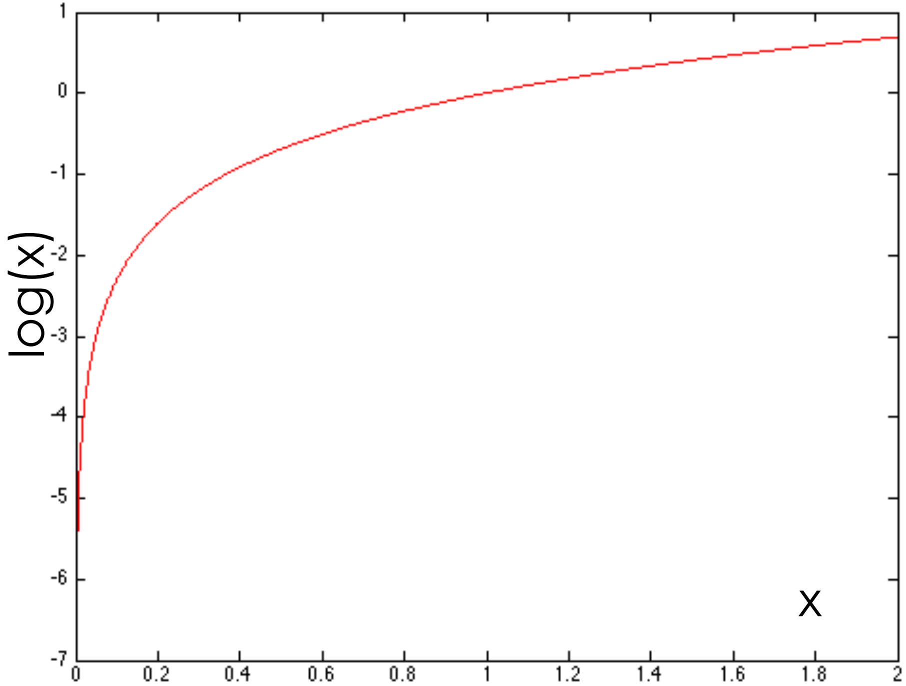
$$\max \; f(x) \text{ s.t. } g(x) \geq 0$$

Log Barrier Transformation of Inequality:

$$\max f(x) + \mu(\log(g(x)))$$

Log Barrier Trick (Log Function)

- log(x) is –Inf if x <= 0
- Adding log(x) to objective forbids x <= 0



Log Barrier Trick

$$\max f(x) + \mu(\log(g(x)))$$

A $\mu \to 0$, we converge to solution of original problem.

- Solve log-barrier problem for initial mu (start with g(x) > 0)
- Gradually decrease mu ($\mu \to 0$).
- Stopping criterion: Change in x is below tolerance.

Linear Programming Formulation

$$\begin{aligned} \max \quad & \mathbf{c}^\top \mathbf{x} \\ A\mathbf{x} & \leq \mathbf{b} \\ \mathbf{x} & \geq 0 \end{aligned}$$

$$\begin{aligned} \max \quad & \mathbf{c}^\top \mathbf{x} \\ A\mathbf{x} + \mathbf{x}_s & = \mathbf{b} \\ \mathbf{x}, \mathbf{x_s} & \geq 0 \end{aligned}$$

Primal Standard form with Slack Variables

Log Barrier Formulation

$$\begin{array}{lrcl} \max & \mathbf{c}^\intercal \ \mathbf{x} & & \\ & A\mathbf{x} + \mathbf{x}_s & = & \mathbf{b} \\ & \mathbf{x}, \mathbf{x_s} & \geq & 0 \end{array}$$

$$\max \ \mathbf{c}^\intercal \ \mathbf{x} + \mu \textstyle\sum_{j=1}^{n} \log(x_j) + \mu \sum_{i=1}^{m} \log(x_{s,i})$$

$$\text{s.t. } A\mathbf{x} + \mathbf{x_s} = \mathbf{b}$$

Equality Constrained Optimization

$$\max \ \mathbf{c}^\intercal \ \mathbf{x} + \mu \sum_{j=1}^{n} \log(x_j) + \mu \sum_{i=1}^{m} \log(x_{s,i})$$
$$\text{s.t. } A\mathbf{x} + \mathbf{x_s} = \mathbf{b}$$

$$\boxed{L(\mathbf{x}, \mathbf{y}) = \left(\begin{array}{c} \mathbf{c}^\intercal \mathbf{x} + \mu \ \sum_{j=1}^{n} \log(x_j) + \mu \sum_{i=1}^{m} \log(x_{s,i}) \\ +\mathbf{y}^\intercal \ (A\mathbf{x} + \mathbf{x_s} - \mathbf{b}) \end{array} \right)}$$

$$\frac{\partial L}{\partial x_j} = c_j + \frac{\mu}{x_j} + \mathbf{y}^\intercal A_{:,j}$$

MU-COMPLEMENTARITY CONDITIONS

Overview

$$\begin{array}{lccc} \max & \mathbf{c}^\intercal \ \mathbf{x} & & \\ & A\mathbf{x} + \mathbf{x}_s & = & \mathbf{b} \\ & \mathbf{x}, \mathbf{x_s} & \geq & 0 \end{array}$$

Primal Problem

Log Barrier Trick

$$\max \ \mathbf{c}^\intercal \ \mathbf{x} + \mu \sum_{j=1}^{n} \log(x_j) + \mu \sum_{i=1}^{m} \log(x_{s,i})$$
$$\text{s.t. } A\mathbf{x} + \mathbf{x_s} = \mathbf{b}$$

A $\mu \rightarrow 0$, we converge to solution of original problem.

Lagrange Multiplier Method

Lagrange Multiplier Method

$$L(\mathbf{x},\mathbf{y}) = \left(\begin{array}{c} \mathbf{c}^\intercal \mathbf{x} + \mu \sum_{j=1}^{n} \log(x_j) + \mu \sum_{i=1}^{m} \log(x_{s,i}) \\ +\mathbf{y}^\intercal \left(A\mathbf{x} + \mathbf{x_s} - \mathbf{b} \right) \end{array} \right)$$

$$\begin{aligned} \nabla_{\mathbf{x}} L(\mathbf{x},\mathbf{y}) &= 0 \\ \nabla_{\mathbf{y}} L(\mathbf{x},\mathbf{y}) &= 0 \end{aligned}$$

First Order Necessary Conditions.

Mu KKT conditions

$$
\boxed{
\begin{aligned}
A\mathbf{x} + \mathbf{x_s} &= \mathbf{b} && \text{Primal} \\
A^\intercal \mathbf{y} - \mathbf{y_s} &= \mathbf{c} && \text{Dual} \\
XY_s\mathbf{e} &= \mu\mathbf{e} && \text{Mu-Complementarity} \\
X_sY\mathbf{e} &= \mu\mathbf{e} &&
\end{aligned}
}
$$

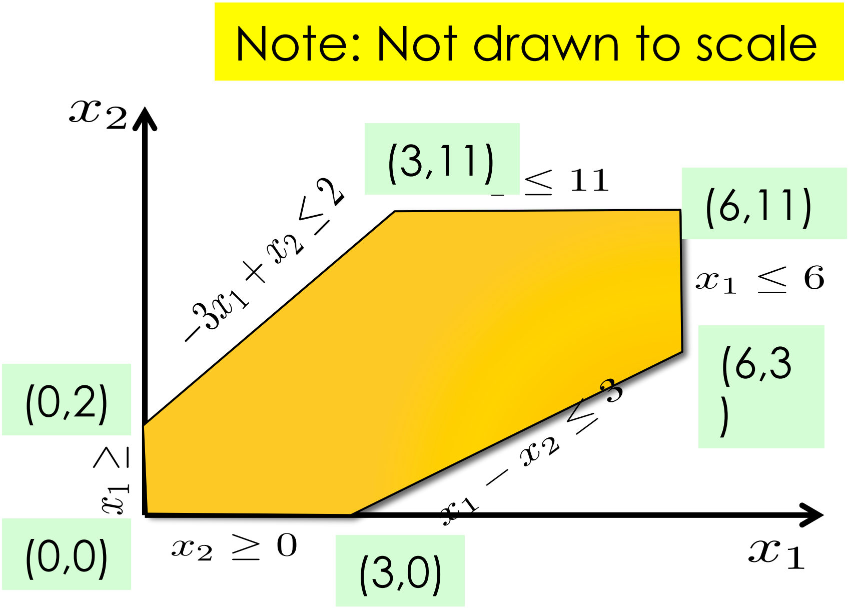
$$
\begin{aligned}
X &= \text{diag}(\mathbf{x}) \\
X_s &= \text{diag}(\mathbf{x_s}) \\
Y &= \text{diag}(\mathbf{y}) \\
Y_s &= \text{diag}(\mathbf{y_s})
\end{aligned}
$$

As mu approaches 0,
we obtain
KKT conditions!!

THE CENTRAL PATH

Linear Programming Problem

$$
\begin{array}{rrrcr}
\text{max.} & x_1 & +2x_2 & & \\
\text{s.t.} & -3x_1 & +x_2 & \leq & 2 \\
 & & +x_2 & \leq & 11 \\
 & x_1 & -x_2 & \leq & 3 \\
 & x_1 & & \leq & 6 \\
 & x_1, & x_2 & \geq & 0
\end{array}
$$



Overview

$$\begin{array}{lccc} \max & \mathbf{c}^\intercal \ \mathbf{x} & & \\ & A\mathbf{x} + \mathbf{x}_s & = & \mathbf{b} \\ & \mathbf{x}, \mathbf{x_s} & \geq & 0 \end{array}$$

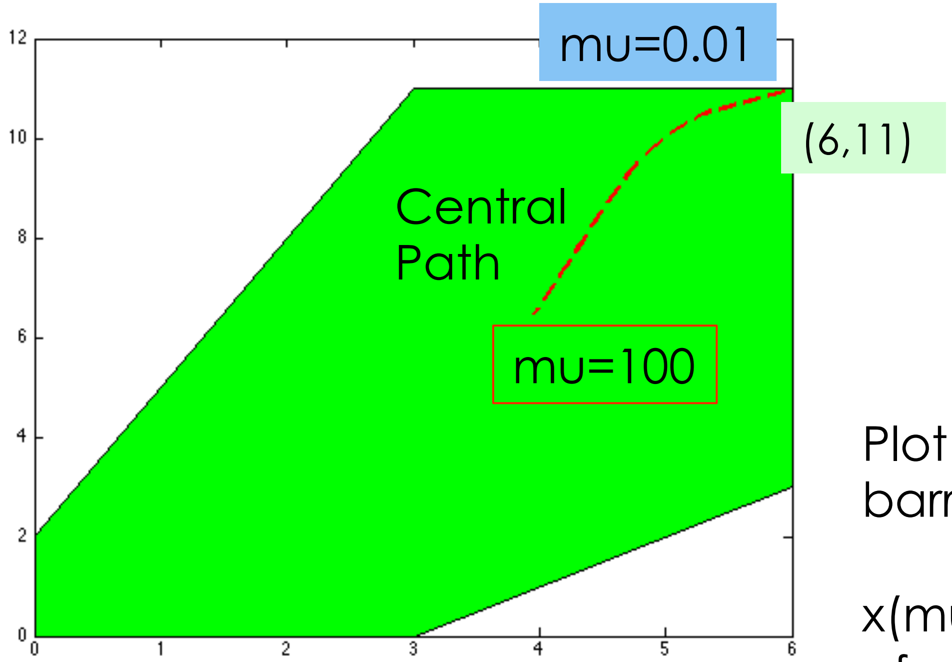
Primal Problem

Log Barrier Trick

$$\max \ \mathbf{c}^\intercal \ \mathbf{x} + \mu \sum_{j=1}^{n} \log(x_j) + \mu \sum_{i=1}^{m} \log(x_{s,i})$$
$$\text{s.t. } A\mathbf{x} + \mathbf{x_s} = \mathbf{b}$$

A $\mu \rightarrow 0$, we converge to solution of original problem.

Central Path



Plot optimal solutions for barrier problem:

x(mu) as a function of mu.

Solving Linear Programs

- Start with a large value of mu.
  - Use Newton’s method to solve for mu-KKT conditions.

- As we iterate, gradually reduce mu.
  - mu’ = 0.1 * mu

- Stop when value of primal infeasibility, dual infeasibility, and mu are small enough.

APPLYING NEWTON METHOD TO SOLVING LPS

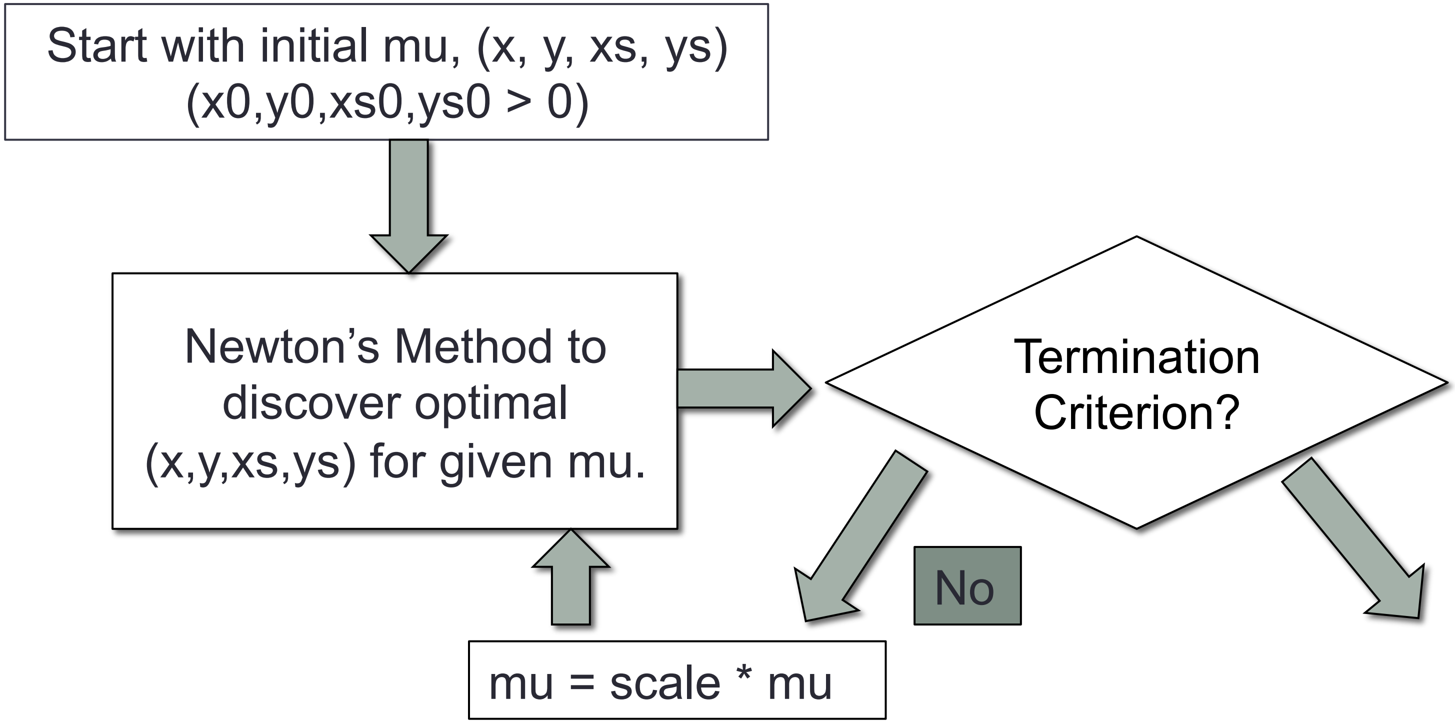
Mu KKT conditions

$$\boxed{\begin{aligned} A\mathbf{x} + \mathbf{x_s} &= \mathbf{b} && \text{Primal} \\ A^{\intercal}\mathbf{y} - \mathbf{y_s} &= \mathbf{c} && \text{Dual} \\ XY_s\mathbf{e} &= \mu\mathbf{e} && \text{Mu-Complementarity} \\ X_sY\mathbf{e} &= \mu\mathbf{e} && \end{aligned}}$$

$$\begin{aligned} X &= \text{diag}(\mathbf{x}) \\ X_s &= \text{diag}(\mathbf{x_s}) \\ Y &= \text{diag}(\mathbf{y}) \\ Y_s &= \text{diag}(\mathbf{y_s}) \end{aligned}$$

As mu approaches 0,
we obtain
KKT conditions!!

Overall Algorithm



Newton Step

$$A\mathbf{x} + \mathbf{x_s} = \mathbf{b} \qquad \text{Primal}$$

$$A^\intercal \mathbf{y} - \mathbf{y_s} = \mathbf{c} \qquad \text{Dual}$$

$$XY_s\mathbf{e} = \mu\mathbf{e} \qquad \text{Mu-Complementarity}$$

$$X_sY\mathbf{e} = \mu\mathbf{e}$$

Solve for $F(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s) = \mathbf{0}$

$$\boxed{F(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y_s}) = \begin{bmatrix} A\mathbf{x} + \mathbf{x}_s - \mathbf{b} \\ A^\intercal \mathbf{y} = \mathbf{y_s} - \mathbf{c} \\ XY_s\mathbf{e} - \mu\mathbf{e} \\ X_sY\mathbf{e} - \mu\mathbf{e} \end{bmatrix}}$$

Calculating Newton Step -1

$$F(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y_s}) = \begin{bmatrix} A\mathbf{x} + \mathbf{x}_s - \mathbf{b} \\ A^\intercal \mathbf{y} = \mathbf{y_s} - \mathbf{c} \\ XY_s\mathbf{e} - \mu\mathbf{e} \\ X_sY\mathbf{e} - \mu\mathbf{e} \end{bmatrix}$$

$$\nabla F = \begin{bmatrix} A & I_{m\times m} & 0_{m\times m} & 0_{m\times n} \\ 0_{n\times m} & 0_{n\times m} & A^\intercal & -I_{n\times n} \\ Y_s & 0_{n\times m} & 0_{n\times m} & X \\ 0_{m\times n} & Y & X_s & 0_{m\times n} \end{bmatrix}$$

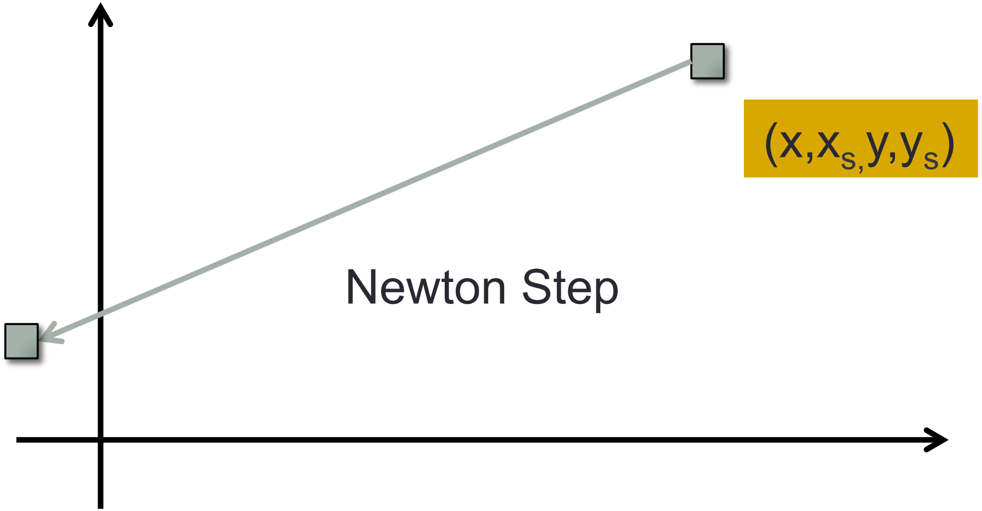
$$\Delta(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s) = -(\nabla F)^{-1} \times F(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s)$$

Calculating Newton Step -2

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} := \begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} + \Delta(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s)$$

Warning: This can violate the non-negativity of (x, x_s, y, y_s)

Sizing the Newton Step



Newton step can make some components of x, x_s, y, y_s negative.

Applying the Newton Step

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} := \begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} + \Delta(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s)$$

WRONG!!

Use a scale factor λ (∈ 0,1)

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} := \begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} + \boxed{\lambda} * \Delta(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s)$$

Finding Scale Factor

- Find (*largest*)λsuch that

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} := \begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} + \lambda * \Delta(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s)$$

guarantees that

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} > 0$$

Implementation Detail: We use a smaller value of λ than the largest possible

APPLYING NEWTON METHOD TO SOLVING LPS

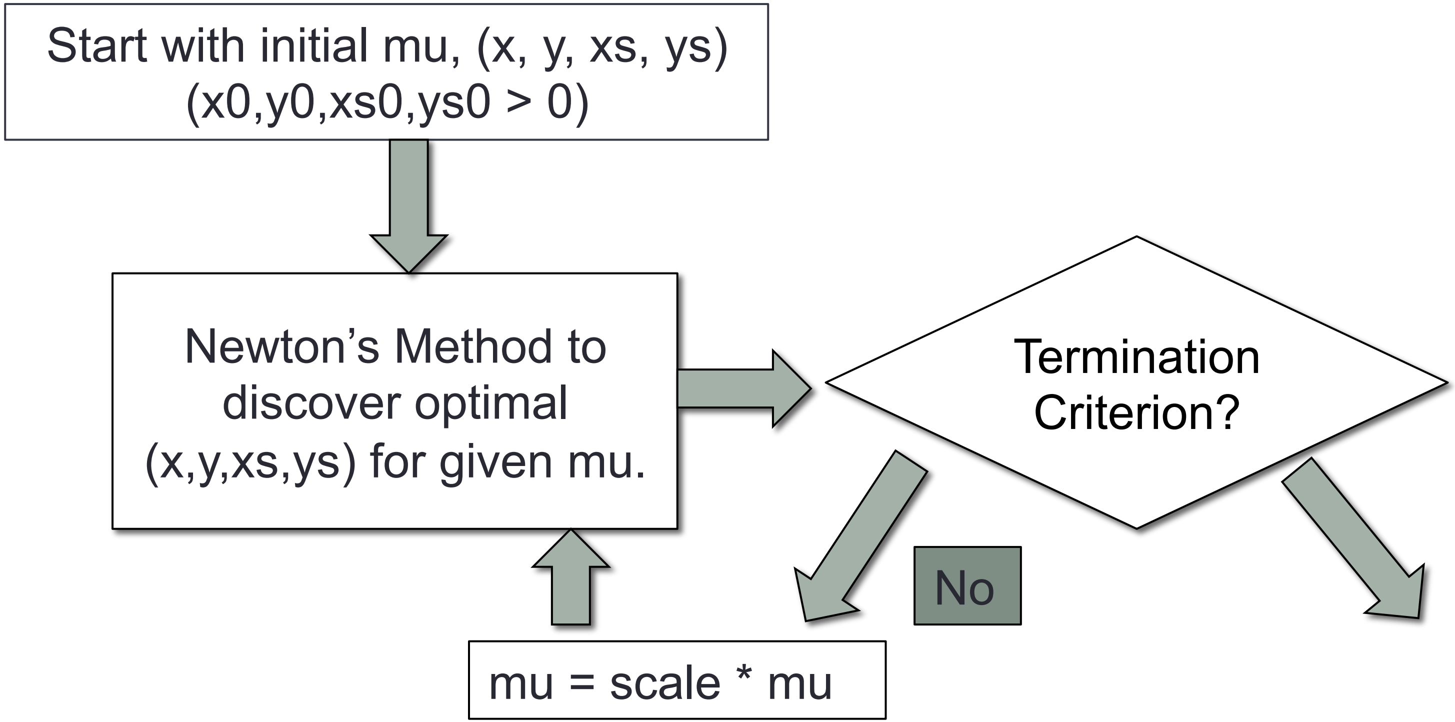
Mu KKT conditions

$$\boxed{\begin{array}{lr} A\mathbf{x} + \mathbf{x_s} = \mathbf{b} & \text{Primal} \\ A^\intercal\mathbf{y} - \mathbf{y_s} = \mathbf{c} & \text{Dual} \\ XY_s\mathbf{e} = \mu\mathbf{e} & \text{Mu-Complementarity} \\ X_sY\mathbf{e} = \mu\mathbf{e} & \end{array}}$$

$$X = \text{diag}(\mathbf{x})$$
$$X_s = \text{diag}(\mathbf{x_s})$$
$$Y = \text{diag}(\mathbf{y})$$
$$Y_s = \text{diag}(\mathbf{y_s})$$

As mu approaches 0,
we obtain
KKT conditions!!

Overall Algorithm



Newton Step

$$A\mathbf{x} + \mathbf{x_s} = \mathbf{b} \qquad \text{Primal}$$

$$A^\intercal \mathbf{y} - \mathbf{y_s} = \mathbf{c} \qquad \text{Dual}$$

$$XY_s\mathbf{e} = \mu\mathbf{e} \qquad \text{Mu-Complementarity}$$

$$X_sY\mathbf{e} = \mu\mathbf{e}$$

Solve for $F(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s) = \mathbf{0}$

$$\boxed{F(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y_s}) = \begin{bmatrix} A\mathbf{x} + \mathbf{x}_s - \mathbf{b} \\ A^\intercal \mathbf{y} = \mathbf{y_s} - \mathbf{c} \\ XY_s\mathbf{e} - \mu\mathbf{e} \\ X_sY\mathbf{e} - \mu\mathbf{e} \end{bmatrix}}$$

Calculating Newton Step -1

$$F(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y_s}) = \begin{bmatrix} A\mathbf{x} + \mathbf{x}_s - \mathbf{b} \\ A^\intercal \mathbf{y} = \mathbf{y_s} - \mathbf{c} \\ XY_s\mathbf{e} - \mu\mathbf{e} \\ X_sY\mathbf{e} - \mu\mathbf{e} \end{bmatrix}$$

$$\nabla F = \begin{bmatrix} A & I_{m\times m} & 0_{m\times m} & 0_{m\times n} \\ 0_{n\times m} & 0_{n\times m} & A^\intercal & -I_{n\times n} \\ Y_s & 0_{n\times m} & 0_{n\times m} & X \\ 0_{m\times n} & Y & X_s & 0_{m\times n} \end{bmatrix}$$

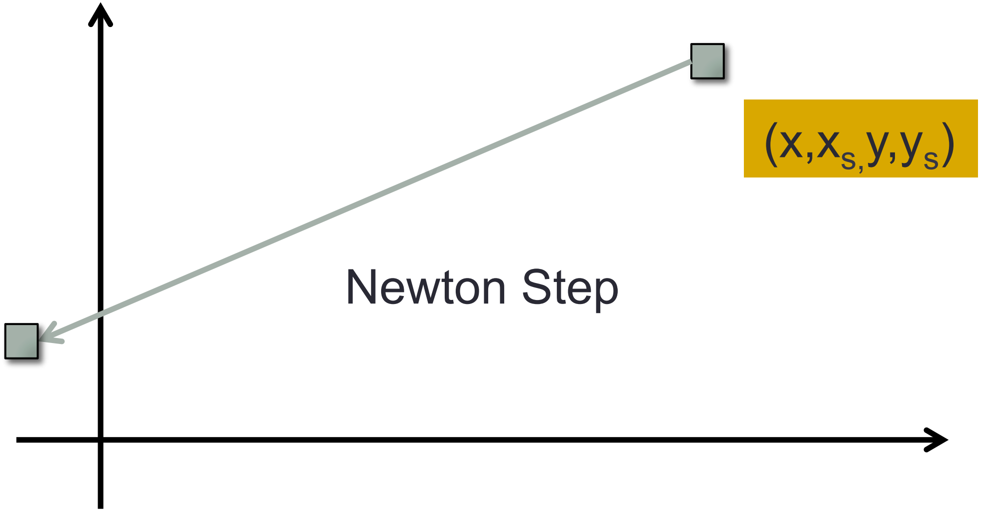
$$\Delta(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s) = -(\nabla F)^{-1} \times F(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s)$$

Calculating Newton Step -2

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} := \begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} + \Delta(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s)$$

Warning: This can violate the non-negativity of $(x, x_{s,} y, y_s)$

Sizing the Newton Step



Newton step can make some components of x,x_s,y,y_s negative.

Applying the Newton Step

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} := \begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} + \Delta(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s)$$

WRONG!!

Use a scale factor λ $\in (0,1)$

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} := \begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} + \lambda * \Delta(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s)$$

Finding Scale Factor

- Find (*<u>largest)</u>*λsuch that

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} := \begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} + \lambda * \Delta(\mathbf{x}, \mathbf{x}_s, \mathbf{y}, \mathbf{y}_s)$$

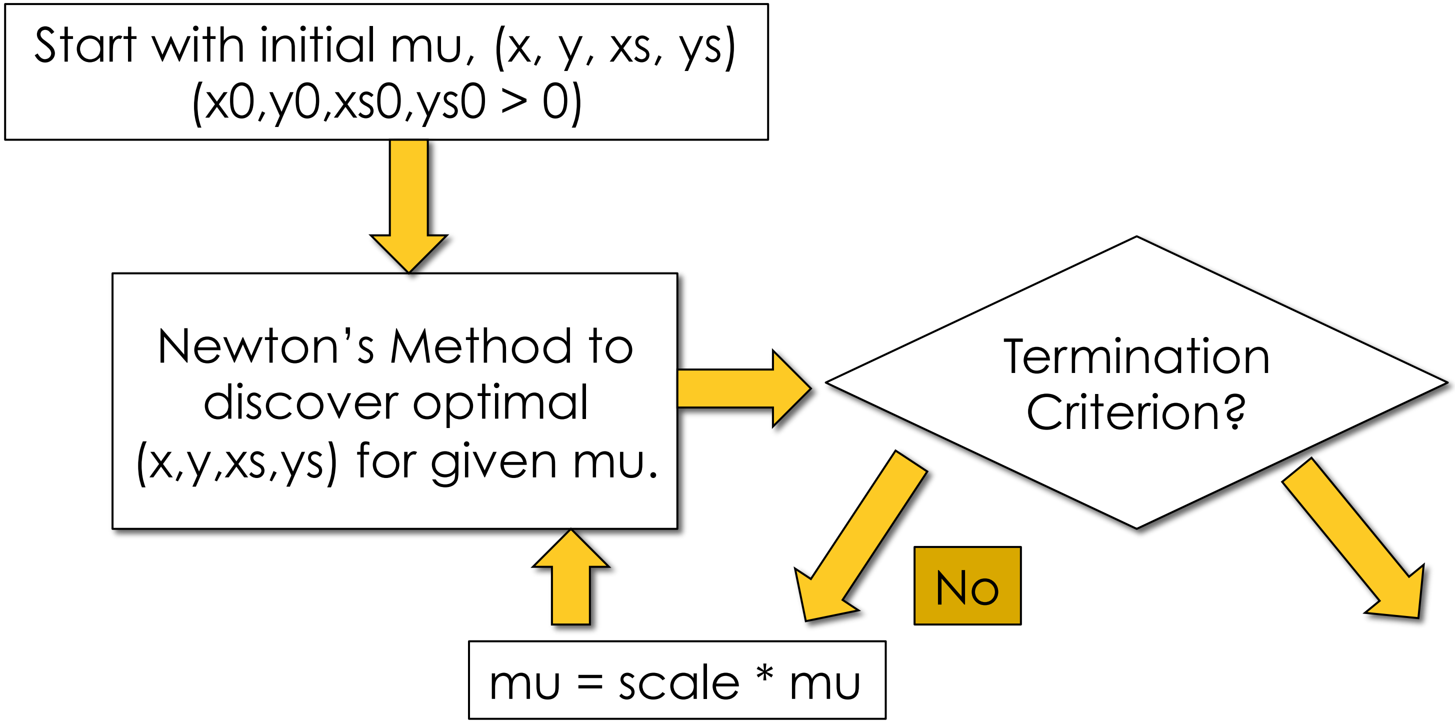
guarantees that

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{x}_s \\ \mathbf{y} \\ \mathbf{y}_s \end{pmatrix} > 0$$

Implementation Detail: We use a smaller value of λ than the largest possible

IMPLEMENTING A SIMPLE INTERIOR POINT SOLVER

Overall Algorithm



Implementation Details

Solve Log Barrier.
(A,b,c, x0,xs0,y0,ys0, mu)

Implement Newton's Method.

Solve to a tolerance or fixed number of iterations.

Main Solver Routine
(A,b,c)

Implement Termination Criterion.

Analyze the solutions to determine final result.

Termination Criterion

- Primal-Dual Gap: $\mathbf{c}^\intercal \mathbf{x} - \mathbf{b}^\intercal \mathbf{y}$
- Primal Infeasibility Gap: $||A\mathbf{x} + \mathbf{x}_s - \mathbf{b}||$
- Dual Infeasibility Gap: $||A^\intercal \mathbf{y} - \mathbf{y}_s - \mathbf{c}||$
- Value of mu
- Change in solution across iterations.
- Iteration Limit.

Termination with Optimal Value

- The primal-dual gap, primal/dual infeasibilities converge to values less than a tolerance.

```
Optimal solution found with objective value: 1406.699737
Number of iterations to converge: 30
Primal Feasibility Gap: 0.000000
Dual Feasibility Gap: 0.000000
Primal-Dual Gap: 0.000000
KKT-residual: 0.000000  (mu = 0.000000)
Number of Iterations: 30 (LIMIT: 30 )
```

Termination with Primal Unbounded

Dual is infeasible, large values of primal may be seen.

- Primal infeasibility gap may converge to 0, but dual does not.
- Often, Hessian faces condition number issues.
  - Inverting Hessian leads to numerical instabilities.
- Primal-Dual gap does not converge to 0.