SI152: Numerical Optimization

Lecture 16: Barrier Methods

Hao Wang
Email: haw309@gmail.com

ShanghaiTech University

December 5, 2024

Outline

1. Algorithmic Development
2. Algorithmic Issues

Consider the constrained optimization problem:

$$\begin{aligned} \min_{x} \; & f(x) \\ \text{s.t. } & c_{\mathcal{E}}(x) = 0 \\ & c_{\mathcal{I}}(x) \leq 0. \end{aligned}$$

- If no problems were "large", then SQP would be hard to beat.
- But, problems are large and solving numerous QP subproblems is expensive.
- Interior-point methods provide an efficient alternative for large problems.
- Interior-point methods also attempt to make use of the power of Newton's method. (In fact, for equality constraints only, there is no "interior" and the methods we discuss here will reduce to SQP methods.)

Consider the constrained optimization problem:

$$\begin{aligned} &\min_x \; f(x) \\ &\text{s.t. } c_{\mathcal{E}}(x) = 0 \\ &\qquad c_{\mathcal{I}}(x) + s = 0, \;\; s \geq 0 \end{aligned}$$

or

$$\begin{aligned} &\min_x \; f(x) \\ &\text{s.t. } c_{\mathcal{I}}(x) = 0, \;\; x \geq 0 \end{aligned}$$

- Any problem can be put into these forms (and similar ones).
- Reformulation WILL affect the practical/theoretical behavior of an algorithm.
- We will use the former reformulation...

Two interpretations

- As for SQP, we will develop the central idea in two ways.
- Each interpretation leads to the same subproblem.
- However, the chosen interpretation WILL affect the algorithm...

- The challenge is ALL with the inequalities/bounds. (If there were equalities only, then use Newton!)
- Thus, create a subproblem that "maintains" the bounds in an easier way:

$$\begin{aligned} \min_x \ & f(x) \\ \text{s.t. } & c_{\mathcal{E}}(x) = 0 \\ & c_{\mathcal{I}}(x) + s = 0, \ \ s \geq 0 \end{aligned} \quad \Longrightarrow \quad \begin{aligned} \min_x \ & f(x) - \mu \sum_{i \in \mathcal{I}} \ln s_i \\ \text{s.t. } & c_{\mathcal{E}}(x) = 0 \\ & c_{\mathcal{I}}(x) + s = 0, \ \ s \geq 0 \end{aligned}$$

- The log-barrier term forces $s > 0$.
- (Note: Do not think that a log term is the only option! It is only the most common since it seems to work the best in general.)
- For a given $\mu > 0$, this is an equality constrained problem: use Newton!
- Solve for a sequence of barrier parameters such that $\mu \to 0$.

- Define the Lagrangian

$$L(x, s, \lambda_{\mathcal{E}}, \lambda_{\mathcal{I}}) = f(x) - \mu \sum_{i \in \mathcal{I}} \ln s_i + \lambda_{\mathcal{E}}^T c_{\mathcal{E}}(x) + \lambda_{\mathcal{I}}^T (c_{\mathcal{I}}(x) + s).$$

- Assuming a constraint qualification, the optimality conditions are:

$$\begin{aligned} \nabla f(x) + \nabla c_{\mathcal{E}}(x)\lambda_{\mathcal{E}} + \nabla c_{\mathcal{I}}(x)\lambda_{\mathcal{I}} &= 0 \\ -\mu S^{-1} e + \lambda_{\mathcal{I}} &= 0 \\ c_{\mathcal{E}}(x) &= 0 \\ c_{\mathcal{I}}(x) + s &= 0 \end{aligned}$$

where $S = \text{diag}(s)$

- A Newton iteration for the above conditions involves the linear system:

$$\begin{bmatrix} \nabla^2_{xx} L^k & 0 & \nabla c_{\mathcal{E}}^k & \nabla c_{\mathcal{I}}^k \\ 0 & \mu {S^k}^{-2} & 0 & I \\ \nabla c_{\mathcal{E}}^k & 0 & 0 & 0 \\ \nabla c_{\mathcal{I}}^k & I & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ d_s \\ \delta_{\mathcal{E}} \\ \delta_{\mathcal{I}} \end{bmatrix} = - \begin{bmatrix} \nabla f^k + \nabla c_{\mathcal{E}}^k \lambda_{\mathcal{E}}^k + \nabla c_{\mathcal{I}}^k \lambda_{\mathcal{I}}^k \\ -\mu {S^k}^{-1} e + \lambda_{\mathcal{I}}^k \\ c_{\mathcal{E}}^k \\ c_{\mathcal{I}}^k + s^k \end{bmatrix}$$

$$\begin{aligned} &\min_x \ f(x) \\ &\text{s.t. } c_{\mathcal{E}}(x) = 0 \\ &\qquad c_{\mathcal{I}}(x) + s = 0, \ \ s \geq 0 \end{aligned}$$

Define the Lagrangian

$$L(x, s, \lambda_{\mathcal{E}}, \lambda_{\mathcal{I}}) = f(x) + \lambda_{\mathcal{E}}^T c_{\mathcal{E}}(x) + \lambda_{\mathcal{I}}^T (c_{\mathcal{I}}(x) + s).$$

Assuming a constraint qualification, the optimality conditions are:

$$\begin{aligned} \nabla f(x) + \nabla c_{\mathcal{E}}(x)\lambda_{\mathcal{E}} + \nabla c_{\mathcal{I}}(x)\lambda_{\mathcal{I}} &= 0 \\ \lambda_{\mathcal{I}} &\geq 0 \\ c_{\mathcal{E}}(x) &= 0 \\ c_{\mathcal{I}}(x) + s &= 0 \\ \lambda_{\mathcal{I}} \cdot s &= 0. \end{aligned}$$

- The challenge now lies in the bounds and complementarity equations.
- If we perturb these and use a perturbation and continuation approach, that may work...

We replace $\lambda_{\mathcal{I}} \geq 0$ and $\lambda_{\mathcal{I}} \cdot s = 0$ with the perturbed equation $\lambda_{\mathcal{I}} \cdot s = \mu e$:

$$\begin{aligned} \nabla f(x) + \nabla c_{\mathcal{E}}(x)\lambda_{\mathcal{E}} + \nabla c_{\mathcal{I}}(x)\lambda_{\mathcal{I}} &= 0 \\ \lambda_{\mathcal{I}} \cdot s &= \mu e \\ c_{\mathcal{E}}(x) &= 0 \\ c_{\mathcal{I}}(x) + s &= 0. \end{aligned}$$

A Newton iteration for the above conditions involves the linear system

$$\begin{bmatrix} \nabla^2_{xx} L^k & 0 & \nabla c^k_{\mathcal{E}} & \nabla c^k_{\mathcal{I}} \\ 0 & \Lambda^k_{\mathcal{I}} & 0 & S^k \\ \nabla c^k_{\mathcal{E}} & 0 & 0 & 0 \\ \nabla c^k_{\mathcal{I}} & I & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ d_s \\ \delta_{\mathcal{E}} \\ \delta_{\mathcal{I}} \end{bmatrix} = - \begin{bmatrix} \nabla f^k + \nabla c^k_{\mathcal{E}} \lambda^k_{\mathcal{E}} + \nabla c^k_{\mathcal{I}} \lambda^k_{\mathcal{I}} \\ -\mu e + \lambda^k_{\mathcal{I}} \cdot s^k \\ c^k_{\mathcal{E}} \\ c^k_{\mathcal{I}} + s^k \end{bmatrix}$$

- Perturbing the optimization problem leads to the optimality conditions:

$$
\begin{aligned}
\nabla f(x) + \nabla c_{\mathcal{E}}(x)\lambda_{\mathcal{E}} + \nabla c_{\mathcal{I}}(x)\lambda_{\mathcal{I}} &= 0 \\
-\mu S^{-1} e + \lambda_{\mathcal{I}} &= 0 \\
c_{\mathcal{E}}(x) &= 0 \\
c_{\mathcal{I}}(x) + s &= 0
\end{aligned}
$$

- Perturbing the optimality conditions directly leads to:

$$
\begin{aligned}
\nabla f(x) + \nabla c_{\mathcal{E}}(x)\lambda_{\mathcal{E}} + \nabla c_{\mathcal{I}}(x)\lambda_{\mathcal{I}} &= 0 \\
\lambda_{\mathcal{I}} \cdot s - \mu e &= 0 \\
c_{\mathcal{E}}(x) &= 0 \\
c_{\mathcal{I}}(x) + s &= 0.
\end{aligned}
$$

- Clearly, the solution sets of these equations are the same:

$$
S(-\mu S^{-1} e + \lambda_{\mathcal{I}}) = \lambda_{\mathcal{I}} \cdot s - \mu e.
$$

- Perturbing the optimization problem leads to the Newton system:

$$\begin{bmatrix} \nabla^2_{xx} L^k & 0 & \nabla c^k_{\mathcal{E}} & \nabla c^k_{\mathcal{I}} \\ 0 & \mu {S^k}^{-2} & 0 & I \\ \nabla c^k_{\mathcal{E}} & 0 & 0 & 0 \\ \nabla c^k_{\mathcal{I}} & I & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ d_s \\ \delta_{\mathcal{E}} \\ \delta_{\mathcal{I}} \end{bmatrix} = - \begin{bmatrix} \nabla f^k + \nabla c^k_{\mathcal{E}} \lambda^k_{\mathcal{E}} + \nabla c^k_{\mathcal{I}} \lambda^k_{\mathcal{I}} \\ -\mu {S^k}^{-1} e + \lambda^k_{\mathcal{I}} \\ c^k_{\mathcal{E}} \\ c^k_{\mathcal{I}} + s^k \end{bmatrix}$$

  In particular, note the equation: $\mu {S^k}^{-2} d_s = -\lambda^k_{\mathcal{I}} + \mu {S^k}^{-1} e$.
- Perturbing optimality conditions leads to the Newton system:

$$\begin{bmatrix} \nabla^2_{xx} L^k & 0 & \nabla c^k_{\mathcal{E}} & \nabla c^k_{\mathcal{I}} \\ 0 & \Lambda^k_{\mathcal{I}} & 0 & S^k \\ \nabla c^k_{\mathcal{E}} & 0 & 0 & 0 \\ \nabla c^k_{\mathcal{I}} & I & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ d_s \\ \delta_{\mathcal{E}} \\ \delta_{\mathcal{I}} \end{bmatrix} = - \begin{bmatrix} \nabla f^k + \nabla c^k_{\mathcal{E}} \lambda^k_{\mathcal{E}} + \nabla c^k_{\mathcal{I}} \lambda^k_{\mathcal{I}} \\ -\mu e + \lambda^k_{\mathcal{I}} \cdot s^k \\ c^k_{\mathcal{E}} \\ c^k_{\mathcal{I}} + s^k \end{bmatrix}$$

  In particular, note the equation: $\Lambda^k_{\mathcal{I}} d_s = -\lambda^k_{\mathcal{I}} \cdot s^k + \mu e$.
- The solution sets of these systems are not the same!

$$\mu {S^k}^{-2} \neq {S^k}^{-1} \Lambda^k_{\mathcal{I}}$$

  though, if converging to a solution, $\lambda^k \to \mu {S^k}^{-1} e$ in the limit.

- Idea: Apply Newton's method to perturbed problems/optimality conditions.
- Benefit: Search directions require only a linear system solve (no QPs!).
- Variations in subproblem formulation, but each probably OK.
- Many algorithmic components to decide upon:
  - maintaining positivity of the slack variables
  - ensuring global convergence/defining a way to judge progress
  - handling nonconvexity
  - handling ill-conditioning/rank-deficiency of the constraints

Outline

We now have in mind an algorithm that attempts to solve the following system:

$$\begin{aligned} \nabla f(x) + \nabla c_{\mathcal{E}}(x)\lambda_{\mathcal{E}} + \nabla c_{\mathcal{I}}(x)\lambda_{\mathcal{I}} &= 0 \\ \lambda_{\mathcal{I}} \cdot s - \mu e &= 0 \\ c_{\mathcal{E}}(x) &= 0 \\ c_{\mathcal{I}}(x) + s &= 0. \end{aligned}$$

- Note that since $\mu > 0$, we must either have

$$(\lambda_i, s_i) > 0 \quad \text{or} \quad (\lambda_i, s_i) < 0, \quad \forall i \in \mathcal{I}.$$

- In fact, we want them both positive! Have to ensure this in the algorithm.

- One typically maintains $s^k > 0$ (it may be wise to maintain $\lambda_{\mathcal{I}}^k > 0$ as well).
- Incorporated into the line search through a fraction-to-the-boundary rule.
- Suppose at (x^k, s^k) a step (d_x^k, d_s^k) is computed.
- Choose $\bar{\alpha}_k \in (0, 1]$ as the largest value such that

$$s^k + \bar{\alpha}_k d_s^k \geq \tau s^k,$$

  where $\tau \in (0, 1)$ is a small positive constant.
- The steplength may be cut further in the line search...

- As for any other type of method, we need a way to judge progress.
- One way is to use a ℓ_1-penalty function:

$$\phi(x, s; \mu, \nu) = f(x) - \mu \sum_{i \in \mathcal{I}} \ln s_i + \nu \left\| \begin{bmatrix} c_{\mathcal{E}}(x) \\ c_{\mathcal{I}}(x) + s \end{bmatrix} \right\|.$$

Solving the Newton system is equivalent to solving the QP:

$$\begin{aligned}\min_{d_x, d_s} \quad & \begin{bmatrix} \nabla f(x^k) \\ -\mu {S^k}^{-1} e \end{bmatrix}^T \begin{bmatrix} d_x \\ d_s \end{bmatrix} + \frac{1}{2} \begin{bmatrix} d_x \\ d_s \end{bmatrix}^T \begin{bmatrix} \nabla^2_{xx} L^k & 0 \\ 0 & {S^k}^{-2} \end{bmatrix} \begin{bmatrix} d_x \\ d_s \end{bmatrix} \\ \text{s.t.} \quad & c_{\mathcal{E}}(x^k) + \nabla c_{\mathcal{E}}(x^k)^T d_x = 0 \\ & c_{\mathcal{I}}(x^k) + \nabla c_{\mathcal{I}}(x^k)^T d_x + s^k + d_s = 0. \end{aligned}$$

Issues arise when this subproblem is nonconvex or has rank-deficient constraints.

- Nonconvexity: Matrix modifications, trust regions, etc.
- Rank-deficient constraints: Matrix modifications, trust regions, etc.

Algorithm Overview

Initialize $\mu_0 > 0$, ν_0 and $(x^0, s^0, \lambda_{\mathcal{E}}^0, \lambda_{\mathcal{I}}^0)$ with $s^0 > 0$
for $k = 0, 1, 2, \ldots$ **do**
    Solve the Newton system

$$\begin{bmatrix} \nabla_{xx}^2 L^k & 0 & \nabla c_{\mathcal{E}}^k & \nabla c_{\mathcal{I}}^k \\ 0 & \Lambda_{\mathcal{I}}^k & 0 & S^k \\ \nabla c_{\mathcal{E}}^k & 0 & 0 & 0 \\ \nabla c_{\mathcal{I}}^k & I & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ d_s \\ \delta_{\mathcal{E}} \\ \delta_{\mathcal{I}} \end{bmatrix} = - \begin{bmatrix} \nabla f^k + \nabla c_{\mathcal{E}}^k \lambda_{\mathcal{E}}^k + \nabla c_{\mathcal{I}}^k \lambda_{\mathcal{I}}^k \\ -\mu e + \lambda_{\mathcal{I}}^k \cdot s^k \\ c_{\mathcal{E}}^k \\ c_{\mathcal{I}}^k + s^k \end{bmatrix}$$

    Compute the largest $\bar{\alpha}_k \in (0, 1]$ satisfying a fraction-to-the-boundary rule.
    Increase ν to ensure decrease in $\phi(x, s; \mu, \nu)$
    Compute a stepsize $\alpha_k \in (0, \bar{\alpha}]$ yielding progress in $\phi(x, s; \mu, \nu)$
    Update $x^{k+1} \leftarrow x^k + \alpha_k d_x^k$ and $s^{k+1} \leftarrow s^k + \alpha_k d_s^k$. Set $\lambda_{\mathcal{E}}^{k+1}$ and $\lambda_{\mathcal{I}}^{k+1}$.
    If the barrier subproblem is solved sufficiently accurately, then decrease μ.
end for

- Generally, interior-point methods are able to attain global convergence and fast local convergence in the neighborhood of (nice) solution points.
- Convergence follows similar to other Newton-type methods, except that care must be taken in updating the barrier parameter μ.
- Interior-point methods for linear/quadratic/convex programming certainly have difficulties for some problems (e.g., degeneracy issues, etc.). However, they generally work and can be proved to work.
- The same is generally true in the general nonlinear case, but there is one very important case to consider: A class of line-search interior-point methods that cannot be proved to converge in general.

Consider any interior-point method in which the search direction computation solves a linearization of the constraints:

$$\begin{aligned} c_{\mathcal{E}}(x^k) + \nabla c_{\mathcal{E}}(x^k)^T d_x &= 0 \\ c_{\mathcal{I}}(x^k) + \nabla c_{\mathcal{I}}(x^k)^T d_x + s^k + d_s &= 0. \end{aligned}$$

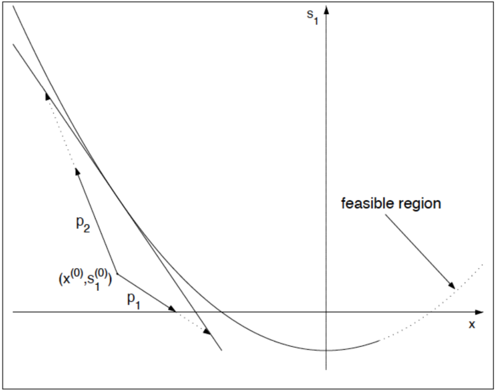
This includes the methods we have discussed and all of those considered in linear/quadratic/convex optimization.

Suppose also that the method requires the fraction-to-the-boundary rule

$$s^k + \alpha d_s \geq \tau s^k$$

Now consider the problem

$$\begin{aligned} \min_{x, s_1, s_2} \quad & x \\ \text{s.t.} \quad & 1 - x^2 + s_1 = 0 \\ & \tfrac{1}{2} - x + s_2 = 0. \end{aligned}$$



Linearized constraints + fraction-to-the-boundary $\Longrightarrow$ fails to converge!