Numerical Optimization

Graduate Course

Constrained smooth optimization

Part III: Nonlinear programming

Wen Huang

School of Mathematical Sciences
Xiamen University

Compiled on April 6, 2022

Nonlinear Programming

Nonlinear Programming

- Penalty and augmented Lagrangian methods
- Sequential quadratic programming (Active set methods)
- Interior point methods

Nonlinear Programming

Nonlinear program:

$$\min_{x \in \mathbb{R}^n} f(x)$$
$$\text{s.t. } c_i(x) = 0 \quad i \in \mathcal{E}$$
$$c_i(x) \geq 0 \quad i \in \mathcal{I}$$

where f and $c_i, i \in \mathcal{E} \cup \mathcal{I}$ are sufficiently smooth as needed.

$$\min_{x\in\mathbb{R}^n} f(x) \quad \text{s.t. } c_i(x) = 0 \;\; i \in \mathcal{E}, \quad c_i(x) \geq 0 \;\; i \in \mathcal{I}$$

Quadratic penalty function:

$$Q(x;\mu) = f(x) + \frac{\mu}{2}\sum_{i\in\mathcal{E}} c_i^2(x) + \frac{\mu}{2}\sum_{i\in\mathcal{I}}([c_i(x)]^-)^2,$$

where $[y]^- = \max(-y, 0)$.

- $([y]^-)^2$ is not C^2 at $y = 0$
- We only consider quadratic penalty method with equation constraints

$$\min_{x\in\mathbb{R}^n} f(x) \quad \text{s.t.}\ \ c_i(x)=0\ \ i\in\mathcal{E}, \quad c_i(x)\ge 0\ \ i\in\mathcal{I}$$

Quadratic penalty function:

$$Q(x;\mu) = f(x) + \frac{\mu}{2}\sum_{i\in\mathcal{E}} c_i^2(x) + \frac{\mu}{2}\sum_{i\in\mathcal{I}}([c_i(x)]^-)^2,$$

where $[y]^- = \max(-y,0)$.

- $([y]^-)^2$ is not C^2 at $y=0$
- We only consider quadratic penalty method with equation constraints

Theorem 1

Suppose that each x_k is the exact global minimizer of $Q(x; \mu_k)$ and that $\mu_k \uparrow \infty$. Then every limit point x^* of the sequence $\{x_k\}$ is a global solution of the problem.

See detailed proofs in [NW06, Theorem 17.1].

- Global minimizer for each subproblem
- Approximate solutions?
- May converge to infeasible points!

Theorem 2

Suppose the subproblem are approximated solved in the sense that $\|\nabla_x Q(x, \mu_k)\| \le \tau_k$, where $\tau_k \to 0$ and $\mu_k \uparrow \infty$. Then if a limit point x^* of $\{x_k\}$ is infeasible, it is a stationary point of $\|c(x)\|^2$. If a limit point x^* is feasible and the constraint gradients $\nabla c_i(x^*)$ are linearly independent, then x^* is a KKT point. In addition, we have for any infinite subsequence $\mathcal{K}$ such that $\lim_{k \in \mathcal{K}} x_k = x^*$ that

$$\lim_{k \in \mathcal{K}} -\mu_k c_i(x_k) = \lambda_i^*, \quad \textit{for all } i \in \mathcal{E},$$

where λ^* is the multiplier vector that satisfies the KKT conditions.

See detailed proofs in [NW06, Theorem 17.2].

- Newton method for $Q(x;\mu_k)$ needs $\nabla^2_{xx}Q(x;\mu_k)$;
- Ill conditioning of $\nabla^2_{xx}Q(x;\mu_k)$:

$$\nabla^2_{xx}Q(x;\mu_k) = \nabla^2 f(x) + \sum_{i\in\mathcal{E}} \mu_k c_i(x)\nabla^2 c_i(x) + \mu_k A(x)^T A(x)$$

  where $A(x)^T = [\nabla c_i(x)]_{i\in\mathcal{E}}$.
- Theorem 2 $\Rightarrow$ $\nabla^2_{xx}Q(x;\mu_k) \approx \nabla^2_{xx}\mathcal{L}(x,\lambda^*) + \mu_k A(x)^T A(x)$.
- $A(x)^T A(x)$ rank deficient $\Rightarrow$ $\nabla^2_{xx}Q(x;\mu_k)$ ill conditioned
- Reformulation can be used to fix the ill-conditioning problem, but it still produces unreliable search direction for the Newton equation $\nabla^2_{xx}Q(x;\mu_k)p = -\nabla_x Q(x,\mu_k)$

$$\min_{x \in \mathbb{R}^n} f(x) \quad \text{s.t.} \;\; c_i(x) = 0$$

ℓ_1 penalty function:

$$\phi_1(x;\mu) = f(x) + \mu \sum_{i \in \mathcal{E}} |c_i(x)|$$

where $[y]^- = \max(-y, 0)$.

- $[y]^-$ and $|y|$ is not C^1 at $y = 0$;
- ℓ_1 norm can be replaced by other nonsmooth vector norms
- More difficult to minimize

Theorem 3

If x^* is a strict local minimizer of the nonlinear program and the KKT conditions are satisfied with Lagrange multipliers $\lambda_i^*, i \in \mathcal{E} \cup \mathcal{I}$. Then x^* is a local minimizer of $\phi_1(x, \mu)$ for all $\mu > \|\lambda^*\|_\infty = \max_{i \in \mathcal{E} \cup \mathcal{I}} |\lambda_i^*|$.

See detailed proofs in [NW06, Theorem 17.1].

Equality constraints:

$$\min_{x \in \mathbb{R}^n} f(x) \quad \text{s.t. } c_i(x) = 0 \;\; i \in \mathcal{E}$$

The augmented Lagrangian function

$$\mathcal{L}_A(x, \lambda; \mu) = f(x) - \sum_{i \in \mathcal{E}} \lambda_i c_i(x) + \frac{u}{2} \sum_{i \in \mathcal{E}} c_i^2(x)$$

- $\nabla_x \mathcal{L}_A(x, \lambda; \mu) = \nabla f(x) - \sum_{i \in \mathcal{E}} (\lambda_i - \mu c_i(x)) \nabla c_i(x)$
- If KKT conditions are satisfied at x^* with Lagrange multiplier $\lambda_i^*, i \in \mathcal{E}$, then $\nabla_x \mathcal{L}_A(x^*, \lambda^*; \mu) = 0$ for all μ
- Estimate λ^* during iterations: $\lambda_i^{k+1} = \lambda_i^k - \mu c_i(x_k), i \in \mathcal{E}$

Theorem 4

Let x^* be a local solution of the nonlinear program at which the LICQ is satisfied, and the second-order sufficient conditions are satisfied for $\lambda = \lambda^*$. Then there is a threshold value $\bar{\mu}$ such that for all $\mu \geq \bar{\mu}$, x^* is a strict local minimizer of $\mathcal{L}_A(x, \lambda^*; \mu)$.

See detailed proofs in [NW06, Theorem 17.5].

Theorem 4

Let x^* be a local solution of the nonlinear program at which the LICQ is satisfied, and the second-order sufficient conditions are satisfied for $\lambda = \lambda^*$. Then there is a threshold value $\bar{\mu}$ such that for all $\mu \geq \bar{\mu}$, x^* is a strict local minimizer of $\mathcal{L}_A(x, \lambda^*; \mu)$.

See detailed proofs in [NW06, Theorem 17.5].

- Suggest: increase μ during iterations
- See theoretical results in [NW06, Theorem 17.6] for the case that $\lambda \approx \lambda^*$

- Theorem 4 needs the knowledge of λ^*
- Theorem below does not and is more realistic

Theorem 5

Let x^* be a local solution of the nonlinear program at which the LICQ is satisfied, and the second-order sufficient conditions are satisfied for $\lambda = \lambda^*$. Let $\bar{\mu}$ be chosen as in Theorem 4. Then there exist positive scalars δ, ϵ, and M such that the following claims hold:

- *For all λ^k and μ_k satisfying $\|\lambda^k - \lambda^*\| \leq \mu_k \delta, \mu_k \geq \bar{\mu}$, the problem $\min_x \mathcal{L}_A(x, \lambda^k; \mu_k)$, s.t.$\|x - x^*\| \leq \epsilon$ has a unique solution x_k. Moreover, we have $\|x_k - x^*\| \leq M \|\lambda^k - \lambda^*\| / \mu_k$;*
- *For all λ^k and μ_k that satisfy $\|\lambda^k - \lambda^*\| \leq \mu_k \delta, \mu_k \geq \bar{\mu}$, we have $\|\lambda^{k+1} - \lambda^*\| \leq M \|\lambda^k - \lambda^*\| / \mu_k$;*
- *For all λ^k and μ_k that satisfy $\|\lambda^k - \lambda^*\| \leq \mu_k \delta, \mu_k \geq \bar{\mu}$, the matrix $\nabla^2_{xx} \mathcal{L}_A(x_k, \lambda^k; \mu_k)$ is positive definite and the constraint gradient $\nabla c_i(x_k), i \in \mathcal{E}$ are linear independent.*

- ALM: simple subproblem
- Subproblem: difficult in general

Example 6

Consider the following types of problems:

$$\min_{x\in\mathbb{R}^{n_1}, z\in\mathbb{R}^{n_2}} \; f_1(x) + f_2(z), \text{ s.t. } Ax + Bz = c.$$

Its augmented Lagrangian function is $\mathcal{L}_A(x, z; s) = f_1(x) + f_2(z) + s^T(Ax + Bz - c) + \frac{\mu}{2}\|Ax + Bz - c\|^2$. The subproblem in ALM is therefore $\min_{x,z}\left[f_1(x) + f_2(z) + s^T(Ax + Bz - c) + \frac{\mu}{2}\|Ax + Bz - c\|^2\right]$.

- The subproblem may be as difficult as the original problem
- One step of an alternating direction method for the subproblem $\Longrightarrow$ ADMM

Penalty and Augmented Lagrangian Methods

Alternating direction method of multipliers (ADMM)

Algorithm 1 ADMM for Example 6

Input: $z^0 \in \mathbb{R}^{n_2}$, $y^0 \in \mathbb{R}^m$, $\mu > 0$;

for $k = 0, 1, \ldots$ **do**

$\quad x^{k+1} = \arg\min_x \left[f_1(x) + (s^k)^T(Ax + Bz^k - c) + \frac{\mu}{2}\|Ax + Bz^k - c\|^2 \right]$;

$\quad z^{k+1} = \arg\min_x \left[f_2(x) + (s^k)^T(Ax^{k+1} + Bz - c) + \frac{\mu}{2}\|Ax^{k+1} + Bz - c\|^2 \right]$;

$\quad s^{k+1} = s^k + \mu(Ax^{k+1} + Bz^{k+1} - c)$;

end for

- Convex f_1 and $f_2 \Longrightarrow O(1/k)$ convergence rate, e.g. [Bec17]
- Strong convex f_1 or $f_2 \Longrightarrow$ linear convergence rate, e.g., [DY16]

$$\min_{x \in \mathbb{R}^n} f(x) \quad \text{s.t. } c_i(x) = 0 \;\; i \in \mathcal{E}$$

- The Lagrangian function:

$$\mathcal{L}(x, \lambda) = f(x) - \sum_{i \in \mathcal{E}} \lambda_i c_i(x).$$

- First order:

$$\nabla_x \mathcal{L}(x, \lambda) = \nabla f(x) - A(x)^T \lambda,$$

where $A(x)^T = [\nabla c_1(x), \nabla c_2(x), \ldots, \nabla c_m(x)],\; m = |\mathcal{E}|.$

Assumption 1

- *The constraint Jacobian $A(x)$ has full low rank.*
- *The matrix $\nabla^2_{xx}\mathcal{L}(x,\lambda)$ is positive definite on the tangent space of the equality constraints, i.e., $d^T\mathcal{L}^2_{xx}(x,\lambda)d > 0$ for all $d \neq 0$ such that $A(x)d = 0$.*

- x^* is the local solution $\Longrightarrow$ KKT conditions with the Lagrange multiplier λ^* hold:

$$\nabla_x\mathcal{L}(x^*,\lambda^*) = 0.$$

- SPD of $\nabla^2_{xx}\mathcal{L}(x,\lambda)$ over the tangent space of the equality constraints.
- Quadratic approximation for the Lagrangian function

Local model at iterate x_k

$$\begin{aligned} \min_{p} \;\; & f(x_k) + \nabla f(x_k)^T p + \frac{1}{2} p^T \nabla^2_{xx} \mathcal{L}(x_k, \lambda_k) p \\ \text{s.t.} \;\; & \nabla c_i(x_k)^T p + c_i(x_k) = 0, \quad i \in \mathcal{E} \end{aligned}$$

- Second order approximation of $\mathcal{L}(x_k, \lambda_k)$ around x_k:

$$\mathcal{L}(x_k, \lambda_k) \approx f(x_k) + \nabla_x \mathcal{L}(x_k, \lambda_k)^T p + \frac{1}{2} p^T \nabla^2_{xx} \mathcal{L}(x_k, \lambda_k) p$$

- $\nabla_x \mathcal{L}(x_k, \lambda_k)^T p = \nabla f(x_k)^T p - \lambda_k^T A(x_k) p = \nabla f(x_k)^T p + \lambda_k^T c(x_k)$
- Quadratic program

Local model at iterate x_k

$$\begin{aligned}\min_{p} \quad & f(x_k) + \nabla f(x_k)^T p + \frac{1}{2} p^T \nabla^2_{xx} \mathcal{L}(x_k, \lambda_k) p \\ \text{s.t.} \quad & \nabla c_i(x_k)^T p + c_i(x_k) = 0, \quad i \in \mathcal{E} \\ & \nabla c_i(x_k)^T p + c_i(x_k) \geq 0, \quad i \in \mathcal{I}\end{aligned} \tag{1}$$

Also a quadratic program

Sequential Quadratic Programming

A SQP algorithm

Algorithm 2 A preliminary SQP algorithm

Input: Initial iterate x_0 and λ_0

for $k = 0, 1, 2, \ldots$ **do**

  Solve (1) for p_k

  $x_{k+1} \leftarrow x_k + p_k$

  λ_{k+1} is the Lagrange multiplier of (1) at p_k

end for

- Analogous to Newton's method, it does not converge globally
- $\nabla^2_{xx}\mathcal{L}$ may not be positive definite
- Modification to $\nabla^2_{xx}\mathcal{L}$
- Line search or trust region
- Merit function

$$\min_{x\in\mathbb{R}^n} f(x) \quad \text{s.t.}\;\; c_i(x) = 0 \;\; i \in \mathcal{E}, \quad c_i(x) \geq 0 \;\; i \in \mathcal{I}$$

$$\Longleftrightarrow$$

$$\min_{(x,s)\in\mathbb{R}^{n+|\mathcal{I}|}} f(x) \quad \text{s.t.}\;\; c_i(x) = 0 \;\; i \in \mathcal{E}, \quad c_i(x) - s = 0 \;\; i \in \mathcal{I} \;\; s \geq 0,$$

where $s \geq 0$ is typically not monitored by the merit function.

Only consider equalition constraints:

$$\min_{x\in\mathbb{R}^n} f(x) \quad \text{s.t.}\;\; c_i(x) = 0 \;\; i \in \mathcal{E}$$

ℓ_1-merit function:

$$\phi_1(x;\mu) = f(x) + \mu\|c(x)\|_1.$$

- Sufficient descent condition:

$$\phi_1(x_k + \alpha_k p_k; \mu_k) \leq \phi_1(x_k, \mu_k) + c_1\alpha_k \mathrm{D}\phi_1(x_k;\mu)[p_k]$$

- The descent of p_k for sufficient large μ is guaranteed by Theorem 7
- Increase μ after each iteration if necessary (See [NW06, P.542] for details)

Theorem 7

We have

$$\begin{aligned}\mathrm{D}\phi_1(x_k;\mu)[p_k] =& \nabla f(x_k)^T p_k - \mu\|c(x_k)\|_1, \qquad \textit{moreover}\\ \mathrm{D}\phi_1(x_k;\mu)[p_k] =& -p_k^T \nabla^2_{xx}\mathcal{L}(x_k,\lambda_k)p_k - (\mu - \|\lambda_{k+1}\|_\infty)\|c(x_k)\|_1.\end{aligned}$$

Sequential Quadratic Programming

A practical line search SQP algorithm

Algorithm 3 A practical line search SQP algorithm

Input: $\eta \in (0, 0.5), \kappa \in (0, 1)$, and (x_0, λ_0);

```
Evaluate f(x_0), ∇f(x_0), c(x_0), A(x_0), and k ← 0;
for k = 0, 1, 2, ... do
    Solve (1) for p_k; Let λ̂ be the corresponding multiplier;
    Set p_λ ← λ̂ − λ_k;
    Choose μ_k appropriately, e.g., μ ≥ (2∇f(x_k)^T p_k + p_k^T ∇²_xx L_k p_k)/‖c(x_k)‖_1;[1]
    Set α_k ← 1;
    while φ_1(x_k + α_k p_k; μ_k) > φ_1(x_k; μ_k) + η α_k D_1 φ_1(x_k; μ_1)[p_k] do
        Reset α_k ← κ α_k;
    end while
    x_{k+1} ← x_k + α_k p_k and λ_{k+1} ← λ_k + α_k p_λ;
    Evaluate f(x_{k+1}), ∇f(x_{k+1}), c(x_{k+1}), A(x_{k+1});
    k ← k + 1;
end for
```

[1] See [NW06, 18.36]

- IQP: Solve a general inequality-constrained quadratic program
  - compute a step and
  - estimate the optimal active set
  - The previous algorithm is an IQP
- EQP: Solve two programs
  - One to estimate the optimal active set
  - The other one to compute a step
  - One example: Sequential linear-quadratic programming (SLQP)

$$\mathrm{LP}:\begin{cases} \min_p \quad f(x_k)+\nabla f(x_k)^T p \\ \nabla c_i(x_k)^T p + c_i(x_k) = 0, \quad i \in \mathcal{E} \\ \nabla c_i(x_k)^T p + c_i(x_k) \geq 0, \quad i \in \mathcal{I} \\ \|p\|_\infty \leq \Delta_k^{\mathrm{LP}} \end{cases} \quad \text{and}$$

$$\mathrm{QP}:\begin{cases} \min_p \quad f(x_k)+\nabla f(x_k)^T p + \frac{1}{2} p^T \nabla_{xx}^2 \mathcal{L}_k p \\ \nabla c_i(x_k)^T p + c_i(x_k) = 0, \quad i \in \mathcal{E} \cap \mathcal{W}_k \\ \nabla c_i(x_k)^T p + c_i(x_k) = 0, \quad i \in \mathcal{I} \cap \mathcal{W}_k \\ \|p\|_2 \leq \Delta_k \end{cases}$$

- Merit function needs to be descent
- Sometimes prevent from rapid convergence

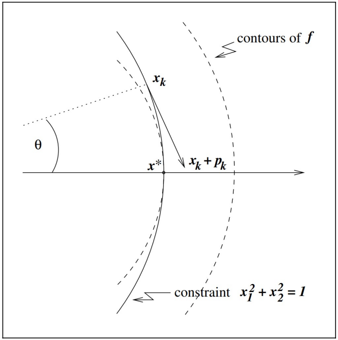

Figure: Maratos Effect

Example:

$$\min f(x_1, x_2) = 2(x_1^2 + x_2^2 - 1) - x1, \text{ subject to } x_1^2 + x_2^2 - 1 = 0.$$

- $x^* = (1, 0)^T$, $\lambda^* = 3/2$ and $\nabla^2_{xx}\mathcal{L} = I$
- Suppose $x_k = (\cos\theta, \sin\theta)^T$ and search direction $p_k = (\sin^2\theta, -\sin\theta\cos\theta)$
- The trial point $x_k + p_k = (\cos\theta + \sin^2\theta, \sin\theta(1 - \cos\theta))$
- Note that $\|x_k + p_k - x^*\|_2 = 2\sin^2(\theta/2)$ and $\|x_k - x^*\| = 2|\sin(\theta/2)|$
- $\frac{\|x_k - x^*\|}{\|x_k + p_k - x^*\|_2^2} = \frac{1}{2} \Rightarrow$ Q-quadratic convergence.
- $f(x_k + p_k) = \sin^2\theta - \cos\theta > -\cos\theta = f(x_k)$
- $c(x_k + p_k) = \sin^2\theta > 0 = c(x_k)$
- $x_k + p_k$ is rejected by any merit function

Remedies: Second-order correction and nonmonotone technieques

- Consider equation constraints for instance
- The linearization of $c(x_k + p_k) \approx c(x_k) + A(x_k)p_k$
- The search direction p_k satisfies: $A(x_k)p_k + c(x_k) = 0$
- $c(x_k + p_k) \approx 0$, not $c(x_k + p_k) = 0$
- Correct p_k by adding $\hat{p}_k$, such that $c(x_k + p_k + \hat{p}_k)$ is closer to zero, i.e.,

$$\hat{p}_k = \arg\min_p \|A(x_k)p + c(x_k + p_k)\|_2^2$$

yields

$$\hat{p}_k = -A(x_k)^T (A(x_k)A(x_k)^T)^{-1} c(x_k + p_k)$$

- Allow merit function increases: nonmonotonic line search

- Lagrangian function:

$$\mathcal{L}(x, y, z) = f(x) - y^T c_{\mathcal{E}}(x) - z^T c_{\mathcal{I}}(x),$$

  where $x \in \mathbb{R}^n$, $y \in \mathbb{R}^{m_1}$, and $z \in \mathbb{R}^{m_2}$;

- KKT conditions:

$$\begin{aligned}
&\nabla f(x) - A_{\mathcal{E}}^T(x)y - A_{\mathcal{I}}^T(x)z = 0, \\
&c_{\mathcal{E}}(x) = 0, \\
&c_{\mathcal{I}}(x) \geq 0, \\
&z \geq 0, \\
&z_i c_i(x) = 0, \quad i \in \mathcal{I},
\end{aligned}$$

  where

$$A_{\mathcal{E}} = \{\nabla c_i(x)^T\}_{i \in \mathcal{E}} \in \mathbb{R}^{m_1 \times n} \text{ and } A_{\mathcal{I}} = \{\nabla c_i(x)^T\}_{i \in \mathcal{I}} \in \mathbb{R}^{m_2 \times n}.$$

KKT conditions:

$$\left.\begin{array}{c} \nabla f(x) - A_{\mathcal{E}}^T(x)y - A_{\mathcal{I}}^T(x)z = 0, \\ c_{\mathcal{E}}(x) = 0, \\ c_{\mathcal{I}}(x) \geq 0, \\ z \geq 0, \\ z_i c_i(x) = 0, \quad i \in \mathcal{I}, \end{array}\right\} \xRightarrow{s = c_{\mathcal{I}}(x)} \begin{array}{c} \nabla f(x) - A_{\mathcal{E}}^T(x)y - A_{\mathcal{I}}^T(x)z = 0, \\ c_{\mathcal{E}}(x) = 0, \\ c_{\mathcal{I}}(x) - s = 0, \\ z_i s_i = 0, i \in \mathcal{I}, \\ (z, s) \geq 0, \end{array}$$

Note that the primal variables are (x, s), the dual variable is (y, z).

Interior Point Methods

KKT to nonlinear system

- Define $F : \mathbb{R}^{n+m_1+2m_2} \to \mathbb{R}^{n+m_1+2m_2}$ by

$$F(x, s, y, z) = \begin{bmatrix} \nabla f(x) - A_{\mathcal{E}}^T(x)y - A_{\mathcal{I}}^T(x)z \\ c_{\mathcal{E}}(x) \\ c_{\mathcal{I}}(x) - s \\ ZS\mathbf{1} \end{bmatrix}$$

- KKT conditions: $F(x, y, s, \lambda) = 0$ and $(z, s) \geq 0$;

KKT conditions $\Longrightarrow$ a variant:

$$\left.\begin{array}{c} F(x,s,y,z) = \begin{bmatrix} \nabla f(x) - A_{\mathcal{E}}^T(x)y - A_{\mathcal{I}}^T(x)z \\ c_{\mathcal{E}}(x) \\ c_{\mathcal{I}}(x) - s \\ ZS\mathbf{1} \end{bmatrix} = 0 \\ (z,s) \geq 0, \end{array}\right\}$$

$$\Longrightarrow \left\{\begin{array}{c} \tilde{F}(x,s,y,z) = \begin{bmatrix} \nabla f(x) - A_{\mathcal{E}}^T(x)y - A_{\mathcal{I}}^T(x)z \\ c_{\mathcal{E}}(x) \\ c_{\mathcal{I}}(x) - s \\ ZS\mathbf{1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \tau\mathbf{1} \end{bmatrix} \\ (z,s) > 0, \end{array}\right. \qquad (2)$$

where $\tau \geq 0$. Under a few assumptions, (2) has a unique solution, denoted by $(x^{(\tau)}, y^{(\tau)}, s^{(\tau)}, \lambda^{(\tau)})$

Assumption 2

Suppose LICQ, the strict complementarity condition, and the second-order sufficient conditions are satisfied at a solution (x^*, s^*, y^*, z^*) *of the nonlinear program. It holds that for sufficient small positive value of* τ, *the system* (2) *has a locally unique solution, which is denoted by* $(x^{(\tau)}, y^{(\tau)}, s^{(\tau)}, \lambda^{(\tau)})$.

- The central path: $\mathcal{C} = \{(x^{(\tau)}, y^{(\tau)}, s^{(\tau)}, \lambda^{(\tau)}) : 0 < \tau < \delta\}$ for a sufficient small δ
- $(x^{(\tau)}, y^{(\tau)}, s^{(\tau)}, \lambda^{(\tau)}) \to$ a solution as $\tau \to 0$
- Approximately solve

$$\begin{bmatrix} \nabla f(x) - A_{\mathcal{E}}^T(x)y - A_{\mathcal{I}}^T(x)z \\ c_{\mathcal{E}}(x) \\ c_{\mathcal{I}}(x) - s \\ ZS\mathbf{1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \tau\mathbf{1} \end{bmatrix}$$
$$(z, s) > 0,$$

e.g., by a step of the Newton's method

- Reduce τ appropriately in every iteration

- Duality measure $\mu = s^T z / m_2$
-

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L} & 0 & -A^T_{\mathcal{E}}(x) & -A^T_{\mathcal{I}}(x) \\ 0 & Z & 0 & S \\ A_{\mathcal{E}}(x) & 0 & 0 & 0 \\ A_{\mathcal{I}}(x) & -I & 0 & 0 \end{bmatrix} \begin{bmatrix} p_x \\ p_s \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} -r^d \\ -SZ\mathbf{1} + \sigma\mu\mathbf{1} \\ -r^y \\ -r^z \end{bmatrix} \tag{3}$$

  where $r^d = \nabla f(x) - A^T_{\mathcal{E}} y - A^T_{\mathcal{I}} z$, $r^y = c_{\mathcal{E}}(x)$, and $r^z = c_{\mathcal{I}}(x) - s$.
- Update:

$$\begin{aligned} x_+ &= x + \alpha^{\max}_s p_x, & s_+ &= s + \alpha^{\max}_s p_s \\ y_+ &= y + \alpha^{\max}_z p_y, & z_+ &= z + \alpha^{\max}_z p_z \end{aligned} \tag{4}$$

  where

$$\begin{aligned} \alpha^{\max}_s &= \max(\alpha \in (0,1] : s + \alpha p_s \geq (1-\kappa)s), \\ \alpha^{\max}_z &= \max(\alpha \in (0,1] : z + \alpha p_z \geq (1-\kappa)z), \end{aligned}$$

  where $\kappa \in (0,1)$

- Error function: $E(x, s, y, z; \tau) = \max(\|r^d\|, \|SZ\mathbf{1} - \tau\mathbf{1}\|, \|r^p\|, \|r^y\|)$

Algorithm 4 A basic interior point algorithm

Input: x_0 and $s_0 > 0$ and compute initial y_0 and $z_0 > 0$; Initial $\tau_0 > 0$, $\sigma, \kappa \in (0, 1)$; set $k \leftarrow 0$;

1: **loop**
2: **while** $E(x_k, s_k, y_k, z_k; \tau_k) \leq \tau_k$ **do**
3: Solve (3) with $(x, s, y, z) = (x_k, s_k, y_k, z_k)$ to obtain the search direction (p_x, p_s, p_y, p_z);
4: Use (4) to obtain $(x_{k+1}, s_{k+1}, y_{k+1}, z_{k+1}) = (x_+, s_+, y_+, z_+)$;
5: $\tau_{k+1} \leftarrow \tau_k$ and $k \leftarrow k + 1$;
6: **end while**
7: Choose $\tau_k \in (0, \sigma\tau_k)$;
8: **end loop**

Linear system:

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L} & 0 & -A^T_{\mathcal{E}}(x) & -A^T_{\mathcal{I}}(x) \\ 0 & Z & 0 & S \\ A_{\mathcal{E}}(x) & 0 & 0 & 0 \\ A_{\mathcal{I}}(x) & -I & 0 & 0 \end{bmatrix} \begin{bmatrix} p_x \\ p_s \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} -r^d \\ -SZ\mathbf{1} + \sigma\mu\mathbf{1} \\ -r^y \\ -r^z \end{bmatrix}$$

- Symmetric formulation:

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L} & 0 & A^T_{\mathcal{E}}(x) & A^T_{\mathcal{I}}(x) \\ 0 & ZS^{-1} & 0 & -I \\ A_{\mathcal{E}}(x) & 0 & 0 & 0 \\ A_{\mathcal{I}}(x) & -I & 0 & 0 \end{bmatrix} \begin{bmatrix} p_x \\ p_s \\ -p_y \\ -p_z \end{bmatrix} = \begin{bmatrix} -r^d \\ -Z\mathbf{1} + \sigma\mu S^{-1}\mathbf{1} \\ -r^y \\ -r^z \end{bmatrix}$$

- Eliminating p_s

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L} & A^T_{\mathcal{E}}(x) & A^T_{\mathcal{I}}(x) \\ A_{\mathcal{E}}(x) & 0 & 0 \\ A_{\mathcal{I}}(x) & 0 & -SZ^{-1} \end{bmatrix} \begin{bmatrix} p_x \\ -p_y \\ -p_z \end{bmatrix} = \begin{bmatrix} -r^d \\ -r^y \\ -r^z - s + \sigma\mu Z^{-1}\mathbf{1} \end{bmatrix}$$

- Eliminating p_z

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L} + A^T_{\mathcal{I}}(x)(S^{-1}Z)A^T_{\mathcal{I}}(x) & A^T_{\mathcal{E}}(x) \\ A_{\mathcal{E}}(x) & 0 \end{bmatrix} \begin{bmatrix} p_x \\ -p_y \end{bmatrix} = \begin{bmatrix} -r^d + A^T_{\mathcal{I}}(x)(-r^z - s + \sigma\mu Z^{-1}\mathbf{1}) \\ -r^y \end{bmatrix}$$

- Symmetric indefinite factorization:

$$P^T K P = L B L^T,$$

where L is lower triangular, B is block diagonal, with block of size 1×1 or 2×2, and P is a permutations.

- Iterative methods: GMRES, QMR, or LSQR.

Adaptive strategies: update the parameter $\tau = \sigma\mu$ every iteration

- $\mu_k = s_k^T z_k / m_2$ and $\tau_k = \sigma_k \mu_k$
- Approach 1 for σ_k: (Used in LOQO package)

$$\sigma_k = 0.1 \min\left(0.05 \frac{1-\xi_k}{\xi_k}, 2\right)^3, \text{ where } \xi_k = \frac{\min_i (s_k)_i (z_k)_i}{(s_k^T z_k / m_2)}$$

- Approach 2 for σ_k: (Similar to LP)
    - Affine scaling direction: $(\Delta x^{\mathrm{aff}}, \Delta s^{\mathrm{aff}}, \Delta y^{\mathrm{aff}}, \Delta z^{\mathrm{aff}})$
    - Compute α_s^{aff}, α_z^{aff}
    - $\mu_{\mathrm{aff}} = (s_k + \alpha_s^{\mathrm{aff}} \Delta s^{\mathrm{aff}})^T (z_k + \alpha_z^{\mathrm{aff}} \Delta z^{\mathrm{aff}}) / m_2$
    - $$\sigma_k = \left(\frac{\mu_{\mathrm{aff}}}{s_k^T z_k / m_2}\right)^3$$

Interior Point Methods

Nonconvexity and singularity

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L} & 0 & A_{\mathcal{E}}^T(x) & A_{\mathcal{I}}^T(x) \\ 0 & ZS^{-1} & 0 & -I \\ A_{\mathcal{E}}(x) & 0 & 0 & 0 \\ A_{\mathcal{I}}(x) & -I & 0 & 0 \end{bmatrix} \Longrightarrow \begin{bmatrix} \nabla^2_{xx}\mathcal{L} + \delta I & 0 & A_{\mathcal{E}}^T(x) & A_{\mathcal{I}}^T(x) \\ 0 & ZS^{-1} & 0 & -I \\ A_{\mathcal{E}}(x) & 0 & -\gamma I & 0 \\ A_{\mathcal{I}}(x) & -I & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L} & A_{\mathcal{E}}^T(x) & A_{\mathcal{I}}^T(x) \\ A_{\mathcal{E}}(x) & 0 & 0 \\ A_{\mathcal{I}}(x) & 0 & -SZ^{-1} \end{bmatrix} \Longrightarrow \begin{bmatrix} \nabla^2_{xx}\mathcal{L} + \delta I & A_{\mathcal{E}}^T(x) & A_{\mathcal{I}}^T(x) \\ A_{\mathcal{E}}(x) & -\gamma I & 0 \\ A_{\mathcal{I}}(x) & 0 & -SZ^{-1} \end{bmatrix}$$

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L} + A_{\mathcal{I}}^T(x)(S^{-1}Z)A_{\mathcal{I}}^T(x) & A_{\mathcal{E}}^T(x) \\ A_{\mathcal{E}}(x) & 0 \end{bmatrix} \Longrightarrow \begin{bmatrix} \nabla^2_{xx}\mathcal{L} + \delta I + A_{\mathcal{I}}^T(x)(S^{-1}Z)A_{\mathcal{I}}^T(x) & A_{\mathcal{E}}^T(x) \\ A_{\mathcal{E}}(x) & -\gamma I \end{bmatrix}$$

Heuristic approach:

- $\delta \geq 0$ such that $\nabla^2_{xx}\mathcal{L} + \delta I$ is SPD
- γI prevents the singularity of $A_{\mathcal{E}}(x)$

$$\phi_\nu(x,s) = f(x) - \tau \sum_{i=1}^{m} \log(s_i) + \nu\|c_\mathcal{E}(x)\| + \nu\|c_\mathcal{I}(x) - s\|,$$

where $\|\cdot\|$ can be the ℓ_1 or the ℓ_2 norm. Consider ℓ_1 here.

- (p_x, p_s) is a descent direction of ϕ_ν for sufficient large ν; (Similar to [NW06, Theorem 18.2])
- Choices of ν
  - $\nu > \max(\|y\|_\infty, \|z\|_\infty)$
  - $\nu > \frac{\nabla f(x)^T p_x}{(1-\rho)\|c_{\mathcal{E}\cup\mathcal{I}}(x)\|_1}$ and $\rho \in (0,1)$
  - $\nu > \frac{\nabla f(x)^T p_x + (\sigma/2) p_x^T \nabla_{xx}^2 \mathcal{L} p_x}{(1-\rho)\|c_{\mathcal{E}\cup\mathcal{I}}(x)\|_1}$ and $\rho \in (0,1)$, where $\sigma = 1$ if $p_x^T \nabla_{xx}^2 \mathcal{L} p_x > 0$; $\sigma = 0$ otherwise.

Review the maximum primal and dual step sizes

$$\alpha_s^{\max} = \max(\alpha \in (0,1] : s + \alpha p_s \geq (1-\kappa)s),$$
$$\alpha_z^{\max} = \max(\alpha \in (0,1] : z + \alpha p_z \geq (1-\kappa)z),$$

- Backtracking algorithm to find $\alpha_s \in (0, \alpha_s^{\max}]$ satisfying

$$\phi_\nu(x + \alpha_s p_x, s + \alpha_s p_s) \leq \phi_\nu(x, s) + c_1 \alpha_s \mathrm{D}\phi_\nu(x, s)[p_x, p_s] \qquad (5)$$

- $\alpha_z = \alpha_s \alpha_z^{\max} / \alpha_s^{\max}$ for instance

Algorithm 5 A line search interior point algorithm

Input: x_0 and $s_0 > 0$ and compute initial y_0 and $z_0 > 0$; Initial $\tau_0 > 0$, $\sigma, \kappa \in (0, 1)$; Tolerance ϵ_{τ_0} and ϵ_{TOL}; Set $k \leftarrow 0$;

while $E(x_k, s_k, y_k, z_k; 0) > \epsilon_{\mathrm{TOL}}$ **do**
  while $E(x_k, s_k, y_k, z_k; \tau_k) > \epsilon_{\tau_k}$ **do**
    Solve (3) with $(x, s, y, z) = (x_k, s_k, y_k, z_k)$ and the modifications for nonconvexity and singularity to obtain the search direction (p_x, p_s, p_y, p_z);
    Find step size α_s and α_z by backtracking to satisfy (5);
    $(x_{k+1}, s_{k+1}, y_{k+1}, z_{k+1}) = (x_+, s_+, y_+, z_+)$ by (4);
    $k \leftarrow k + 1$;
  end while
  Choose $\tau_{k+1} \leftarrow \sigma\tau_k$ and $\epsilon_{\tau_{k+1}} \leftarrow \sigma\epsilon_{\tau_k}$;
end while

- The line search interior point algorithm: not converge globally

$$\begin{aligned} &\min_{x\in\mathbb{R}} x \\ \text{s.t. } &x^2 - s_1 - 1 = 0 \\ &x - s_2 - \frac{1}{2} = 0 \\ &s_1 \geq 0, s_2 \geq 0 \end{aligned}$$

- Safeguard must be used, e.g., penalizations of the constraints

References I

Amir. Beck.
First-Order Methods in Optimization.
Society for Industrial and Applied Mathematics, Philadelphia, PA, 2017.

Wei Deng and Wotao Yin.
On the global and linear convergence of the generalized alternating direction method of multipliers.
Journal of Scientific Computing, 66:889–916, 2016.

J. Nocedal and S. J. Wright.
Numerical Optimization.
Springer, second edition, 2006.