Numerical Optimization

Graduate Course

Constrained smooth optimization

Part III: Quadratic programming

Wen Huang

School of Mathematical Sciences
Xiamen University

Compiled on March 29, 2022

Quadratic Programming

Quadratic Programming

- Equation constraints
- Active set methods
- Interior point methods

Quadratic Programming

The general quadratic program (QP):

$$
\begin{aligned}
\min_x \quad & q(x) = \frac{1}{2}x^T G x + x^T c \\
\text{s.t.} \quad & a_i^T x = b_i, i \in \mathcal{E} \\
& a_i^T x \geq b_i, i \in \mathcal{I}
\end{aligned}
$$

- $c, x, a_i, i \in \mathcal{E} \cup \mathcal{I}$ are vectors in $\mathbb{R}^n$;
- b_i is a scalar and $\{b_i\}_{i \in \mathcal{E} \cup \mathcal{I}}$ is a vector in $\mathbb{R}^m$;
- $|\mathcal{E}| = m_1$, $|\mathcal{I}| = m_2$, and $m_1 + m_2 = m$;
- $G \in \mathbb{R}^{n \times n}$ is a symmetric matrix;

Quadratic Programming

The general quadratic program (QP):

$$\begin{aligned} \min_x \quad & q(x) = \frac{1}{2}x^T Gx + x^T c \\ \text{s.t.} \quad & a_i^T x = b_i, i \in \mathcal{E} \\ & a_i^T x \geq b_i, i \in \mathcal{I} \end{aligned}$$

- G symmetric positive semidefinite: convex quadratic program
- G symmetric positive definite: strictly convex quadratic program
- G symmetric indefinite: nonconvex quadratic program

Quadratic Programming

The general quadratic program (QP):

$$\begin{aligned} \min_x \quad & q(x) = \frac{1}{2} x^T G x + x^T c \\ \text{s.t.} \quad & a_i^T x = b_i, i \in \mathcal{E} \\ & a_i^T x \geq b_i, i \in \mathcal{I} \end{aligned}$$

- G symmetric positive semidefinite: convex quadratic program
- G symmetric positive definite: strictly convex quadratic program
- G symmetric indefinite: nonconvex quadratic program

$$\begin{array}{cc} \min_x & q(x) = \frac{1}{2}x^T G x + x^T c \\ \text{s.t.} & Ax - b = 0, \end{array} \tag{1}$$

where $G \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^m$, and $A \in \mathbb{R}^{m\times n}$ is full row rank.

KKT conditions:

$$\begin{bmatrix} G & -A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix} \Longleftrightarrow \begin{bmatrix} G & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ -\lambda \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix} \tag{2}$$

When is the coefficient matrix nonsingular?

Equation Constraints

Theoretical results

Let $Z \in \mathbb{R}^{n\times(n-m)}$ denote the matrix whose columns are a basis for the null space of A, i.e., $AZ = 0$

Lemma 1

Let A have full row rank, and assume that the reduced-Hessian matrix $Z^T GZ$ is positive definite. Then the KKT matrix

$$K = \begin{bmatrix} G & A^T \\ A & 0 \end{bmatrix}$$

is nonsingular, and hence there is a unique vector pair (x^*, λ^*) satisfying (2).

Theorem 2

Let A have full row rank and assume that the reduced-Hessian matrix $Z^T GZ$ is positive definite. Then the vector x^* satisfying (2) is the unique global solution of the quadratic program with equation constraints (1).

$$\begin{bmatrix} G & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ -\lambda \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix} : \text{ change notation to } \begin{bmatrix} G & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}$$

- Suppose $Z^T G Z$ is symmetric positive definite
- Factorization based methods
- Iterative methods

Equation Constraints

Schur-complement method

Suppose G is symmetric positive definite.

$$\begin{bmatrix} G & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}$$

$$\begin{bmatrix} I & 0 \\ -AG^{-1} & I \end{bmatrix} \begin{bmatrix} G & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} I & 0 \\ -AG^{-1} & I \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$$

$$\begin{bmatrix} G & A^T \\ 0 & -AG^{-1}A^T \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} a \\ b - AG^{-1}a \end{bmatrix}$$

$$\begin{cases} AG^{-1}A^T v = AG^{-1}a - b \\ Gu = a - A^T v \end{cases}$$

$$\begin{cases} AG^{-1}A^T v = AG^{-1}a - b \\ Gu = a - A^T v \end{cases}$$

Factorization method:

- Cholesky decomposition: $G = LL^T$
- Sove for w: $LL^T w = a$
- Compute QR factorization of $L^{-1}A^T$, i.e., $QR = L^{-1}A^T$
- Solve for v: $R^T Rv = Aw - b$
- Solve for u: $LL^T u = a - A^T v$

$$\begin{cases} AG^{-1}A^T v = AG^{-1}a - b \\ Gu = a - A^T v \end{cases}$$

Factorization method:

- Cholesky decomposition: $G = LL^T$
- Sove for w: $LL^T w = a$
- Compute QR factorization of $L^{-1}A^T$, i.e., $QR = L^{-1}A^T$
- Solve for v: $R^T R v = Aw - b$
- Solve for u: $LL^T u = a - A^T v$

Note: (i) Dominated computational cost on QR factorization of $L^{-1}A^T$. Note that Q is not needed. (ii) Cholesky factorization for $AG^{-1}A^T$ is relatively inexpensive, but may have numerical problems

$$\begin{cases} AG^{-1}A^Tv = AG^{-1}a - b \\ Gu = a - A^Tv \end{cases}$$

Iterative method:

- CG method for solving $Gw = a$
- CG method for solving $AG^{-1}A^Tv = Aw - b$
- CG method for solving $Gu = a - A^Tv$

$$\begin{cases} AG^{-1}A^T v = AG^{-1}a - b \\ Gu = a - A^T v \end{cases}$$

Iterative method:

- CG method for solving $Gw = a$
- CG method for solving $AG^{-1}A^T v = Aw - b$
- CG method for solving $Gu = a - A^T v$

Note that the second is more involved:

- Let $B = AG^{-1}A^T$; CG for $Bv = Aw - b$
- $s = Bv = AG^{-1}A^T v$ requires multiple steps:

$$z = A^T v \to t = G^{-1}z \to s = At$$

and $G^{-1}z$ requires another CG

Equation Constraints

Null-space method

$$\begin{bmatrix} G & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}$$

- Let $Y \in \mathbb{R}^{n\times m}$ and $\mathrm{span}(Y) = \mathrm{span}(A^T)$, therefore $[Y, Z] \in \mathbb{R}^{n\times n}$ is nonsingular
- Decomposition: $u = Yu_y + Zu_z$
- Solve $AYu_y = b$ for u_y
- $GYu_y + GZu_z + A^T v = a \Longrightarrow Z^T GZu_z = Z^T a - Z^T GYu_y$
- Solve $(Z^T GZ)u_z = Z^T a - Z^T GYu_y$ for u_z
- Solve $(AY)^T v = Y^T(a - Gu)$ for v

$$\begin{bmatrix} G & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}$$

- Let $Y \in \mathbb{R}^{n\times m}$ and $\mathrm{span}(Y) = \mathrm{span}(A^T)$, therefore $[Y, Z] \in \mathbb{R}^{n\times n}$ is nonsingular
- Decomposition: $u = Yu_y + Zu_z$
- Solve $AYu_y = b$ for u_y
- $GYu_y + GZu_z + A^T v = a \Longrightarrow Z^T GZu_z = Z^T a - Z^T GYu_y$
- Solve $(Z^T GZ)u_z = Z^T a - Z^T GYu_y$ for u_z
- Solve $(AY)^T v = Y^T(a - Gu)$ for v

Note: (i) Y can be the Q factor of QR decomposition of A^T, (ii) solving $(Z^T GZ)u_z = Z^T a - Z^T GYu_y$ dominates the cost

Equation Constraints

Null-space method

$$(Z^T GZ)u_z = Z^T a - Z^T GYu_y$$

- Cholesky factorization for $Z^T GZ$
- CG method
- The above approaches requires knowledge of Z
- Note that $ZZ^T = I - A^T(AA^T)^{-1}A \Longrightarrow$ projected CG, see [NW06, P.461].

Equation Constraints

Schur-complement method versus null-space method

- G is symmetric positive definite and $AG^{-1}A^T$ is inexpensive $\Longrightarrow$ Schur-complement method
- Otherwise, null-space method

$$\begin{aligned}
\min_x \quad & q(x) = \frac{1}{2}x^T G x + x^T c \\
\text{s.t.} \quad & a_i^T x = b_i, i \in \mathcal{E} \\
& a_i^T x \geq b_i, i \in \mathcal{I}
\end{aligned}$$

- Active set at x: $\mathcal{A}(x) = \{i \in \mathcal{I} \cup \mathcal{E} : a_i^T x = b_i\}$
- KKT conditions:

$$\begin{aligned}
& Gx + c - \sum_{i \in \mathcal{A}} a_i \lambda_i = 0, & \\
& a_i^T x = b_i, & i \in \mathcal{A}(x), \\
& a_i^T x \geq b_i, & i \in \mathcal{I}/\mathcal{A}(x), \\
& \lambda_i \geq 0, & i \in \mathcal{I} \cap \mathcal{A}(x).
\end{aligned}$$

- The active set $\mathcal{A}(x^*)$ is known $\Longrightarrow$ using methods discussed before
- Define the working set: the union of $\mathcal{E}$ and a subset of $\mathcal{I}$
- Consider equality constraints problem:

$$\begin{aligned} \min_x \quad & q(x) = \frac{1}{2}x^T Gx + x^T c \\ \text{s.t.} \quad & a_i^T x = b_i, i \in \mathcal{W} \end{aligned}$$

- Active set methods: Update the working set iteratively

Initialization:

- Initial working set $\mathcal{W}_0$ such that $a_i, i \in \mathcal{W}_0$ are linear independent
- Initial iterate x_0 such that $a_i^T x_0 = b_i, i \in \mathcal{W}_0$ and $a_i^T x_0 \geq b_i, i \notin \mathcal{W}_0$
- $k \leftarrow 0$

Add an index to the working set:

- Find search direction p_k at x_0:

$$
\begin{aligned}
& p_k = \arg\min_p \tfrac{1}{2}(x_k+p)^T G(x_k+p) + (x_k+p)^T c \\
& \qquad \text{s.t.} \quad a_i^T(x_k+p) = b_i, i \in \mathcal{W}_k \\
\Longleftrightarrow \quad & p_k = \arg\min_p \tfrac{1}{2} p^T G p + p^T(Gx_k + c) \\
& \qquad \text{s.t.} \quad a_i^T p = 0, i \in \mathcal{W}_k
\end{aligned}
$$

- $x_{k+1} = x_k + \alpha_k p_k$, where α_k is the largest value in $[0,1]$ such that all constraints are satisfied.
- $\alpha_k = \min\left(1, \min_{i \notin \mathcal{W}_k, a_i^T p_k < 0} \frac{b_i - a_i^T x_k}{a_i^T p_k}\right)$
- If $\exists j \notin \mathcal{W}_k$ such that $a_j^T(x_k + \alpha_k p_k) = b_k$, then $\mathcal{W}_{k+1} \leftarrow \mathcal{W}_k \cup \{j\}$; Otherwise, $\mathcal{W}_{k+1} \leftarrow \mathcal{W}_k$

Note that $a_i, i \in \mathcal{W}_{k+1}$ are also linear independent

Remove an index from the working set:

- Compute the Lagrange multiplier:

$$\sum_{i \in \mathcal{W}} a_i \lambda_i = Gx_k + c,$$

- If $\lambda_j < 0, j \in \mathcal{W}_k$, then drop j from $\mathcal{W}_k$ and solve

$$\begin{array}{c} p_k = \arg\min_p \frac{1}{2} p^T G p + p^T (Gx_k + c) \\ \text{s.t.} \quad a_i^T p = 0, \quad i \in \mathcal{W}_k / \{j\} \end{array}$$

- $x_{k+1} = x_k + \alpha_k p_k$, where α_k is the largest value in $[0, 1]$ such that all constraints are satisfied.

Theorem 3

Suppose x is a feasible point, $\mathcal{W}$ is a working set, and $a_i^T x = b_i$ for all $x \in \mathcal{W}$. Suppose that $a_i, i \in \mathcal{W}$ are linear independent and there is an index $j \in \mathcal{W}$ such that $\lambda_j < 0$. Let p^* be the solution of

$$\begin{aligned} p^* = \arg\min_p & \tfrac{1}{2} p^T G p + p^T (Gx + c) \\ s.t. \quad & a_i^T p = 0, \quad i \in \mathcal{W}/\{j\}. \end{aligned}$$

Then p is a feasible direction for constraint j, i.e., $a_j^T p \geq 0$. Moreover, if $Z^T G Z$ is positive definite, then $a_j^T p > 0$, and therefore, p is a descent direction.

See detailed proofs in [NW06, Theorem 16.5].

Active Set Methods

An algorithm

Algorithm 1 An active set method for quadratic program

Input: A set $\mathcal{W}_0 \subset \mathcal{E} \cup \mathcal{I}$ such that $a_i, i \in \mathcal{W}_0$ linear independent; $x_0 \in \mathbb{R}^n$ such that $a_i^T x_0 = b_i, i \in \mathcal{W}_0$;

loop
  Solve $p_k = \arg\min_p \frac{1}{2} p^T G p + p^T (G x_k + c)$ such that $a_i^T p = 0, i \in \mathcal{W}_k$;
  if $\|p\| = 0$ **then**
    Compute Lagrange multiplier $\lambda_i, i \in \mathcal{W}$
    If $\lambda_i \geq 0$, then return $x^* = x_k$; otherwise, $j \leftarrow \arg\min_{j \in \mathcal{W}_k \cap \mathcal{I}} \lambda_j$;
    $x_{k+1} \leftarrow x_k$ and $\mathcal{W}_{k+1} \leftarrow \mathcal{W}_k / \{j\}$;
  else
    Compute step size α_k and set $x_{k+1} \leftarrow x_k + \alpha p_k$;
    If there are blocking constraints, obtain $\mathcal{W}_{k+1}$ by adding an index to $\mathcal{W}_k$; otherwise, $\mathcal{W}_{k+1} \leftarrow \mathcal{W}_k$;
  end if
end loop

Active Set Methods

An example

$$\min_x q(x) = (x_1 - 1)^2 + (x_2 - 2.5)^2$$
$$\text{subject to } x_1 - 2x_2 + 2 \geq 0$$
$$-x_1 - 2x_2 + 6 \geq 0$$
$$-x_1 + 2x_2 + 2 \geq 0$$
$$x_1 \geq 0$$
$$x_2 \geq 0$$

- initial iterate $x^0 = [2, 0]^T$, working set $\mathcal{W}_0 = \{3, 5\}$;
- $p_0 = [0, 0]^T$ since 3rd and 5th active constraints determine a single point;
- $a_3\lambda_3 + a_5\lambda_5 = Gx^0 + c \Rightarrow \lambda_3 = -2$ and $\lambda_5 = -1$;
- Remove 3 from the working set $\mathcal{W}_1 = 5$, $x_1 = x_0 = [2, 0]^T$;

Active Set Methods

An example

$$
\begin{aligned}
\min_x q(x) = (x_1 - 1)^2 &+ (x_2 - 2.5)^2 \\
\text{subject to } x_1 - 2x_2 + 2 &\geq 0 \\
-x_1 - 2x_2 + 6 &\geq 0 \\
-x_1 + 2x_2 + 2 &\geq 0 \\
x_1 &\geq 0 \\
x_2 &\geq 0
\end{aligned}
$$

- The working set $\mathcal{W}_1 = 5$, $x^1 = x^0 = [2, 0]^T$;
- The search direction:

$$
p^1 = \arg\min_p (p_1 - 1)^2 + (p_2 - 2.5)^2 + 4p_1 \quad \text{s.t.} \quad p_2 = 0
$$

  yields $p^1 = [-1, 0]^T$
- $x_2 = [1, 0]^T$, $\mathcal{W}_2 = \{5\}$

$$
\begin{aligned}
\min_x q(x) = (x_1 - 1)^2 &+ (x_2 - 2.5)^2 \\
\text{subject to } x_1 - 2x_2 + 2 &\geq 0 \\
-x_1 - 2x_2 + 6 &\geq 0 \\
-x_1 + 2x_2 + 2 &\geq 0 \\
x_1 &\geq 0 \\
x_2 &\geq 0
\end{aligned}
$$

- $x_2 = [1, 0]^T$, $\mathcal{W}_2 = 5$;
- $p_2 = 0$ since the previous step does not have blocking constraints;
- $a_5 \lambda_5 = Gx^2 + c \Rightarrow \lambda_5 = -5$;
- Remove 5 from the working set $\mathcal{W}_3 = \{\}$, $x_3 = x_2 = [1, 0]^T$;

Active Set Methods

An example

$$
\begin{aligned}
\min_x q(x) = (x_1-1)^2 + (x_2-2.5)^2 \\
\text{subject to } x_1 - 2x_2 + 2 \geq 0 \\
-x_1 - 2x_2 + 6 \geq 0 \\
-x_1 + 2x_2 + 2 \geq 0 \\
x_1 \geq 0 \\
x_2 \geq 0
\end{aligned}
$$

- The working set $\mathcal{W}_3 = \{\}$, $x_3 = x_2 = [1, 0]^T$;
- The search direction:

$$p^3 = \arg\min_p (p_1 - 1)^2 + (p_2 - 2.5)^2 + 2p_1$$

  yields $p^3 = [0, 2.5]^T$;
- $x_4 = x_3 + \alpha_3 p_3 \Rightarrow \alpha_3 = 0.6$ and the constraint 1 is the blocking constraint;
- $\mathcal{W}_4 = \{1\}$ and $x_4 = [1, 1.5]^T$;

$$\min_x q(x) = (x_1 - 1)^2 + (x_2 - 2.5)^2$$
$$\begin{aligned} \text{subject to } x_1 - 2x_2 + 2 &\geq 0 \\ -x_1 - 2x_2 + 6 &\geq 0 \\ -x_1 + 2x_2 + 2 &\geq 0 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \end{aligned}$$

- $\mathcal{W}_4 = 1$ and $x_4 = [1, 1.5]^T$;
- The search direction:

$$p^4 = \arg\min_p (p_1 - 1)^2 + (p_2 - 2.5)^2 + 2p_1 + 3p_2$$
$$\text{s.t.} \quad p_1 - 2p_2 = 0$$

  yields $p^4 = [0.4, 0.2]^T$;
- $x^5 = x^4 + \alpha^4 p^4 \Rightarrow \alpha_4 = 1$, no blocking constraints;
- $\mathcal{W}_5 = \{1\}$ and $x^5 = [1.4, 1.7]^T$;

$$\min_x q(x) = (x_1 - 1)^2 + (x_2 - 2.5)^2$$
$$\text{subject to } x_1 - 2x_2 + 2 \geq 0$$
$$-x_1 - 2x_2 + 6 \geq 0$$
$$-x_1 + 2x_2 + 2 \geq 0$$
$$x_1 \geq 0$$
$$x_2 \geq 0$$

- $\mathcal{W}_5 = \{1\}$ and $x^5 = [1.4, 1.7]^T$;
- $p^5 = 0$ since no blocking constraint in previous step
- $a_1\lambda_1 = Gx^5 + c \Rightarrow [1, -2]\lambda_1 = [0.8, -1.6] \Rightarrow \lambda_1 = 0.8$;
- $\lambda_1 > 0$ implies the solution is found;

- Lagrangian function:

$$\mathcal{L}(x, s, \lambda) = \frac{1}{2}x^T Gx + x^T c - \sum_{i\in\mathcal{E}} s_i(a_i^T x - b_i) - \sum_{i\in\mathcal{I}} \lambda_i(a_i^T x - b_i),$$

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

- KKT conditions:

$$\begin{aligned}
&Gx + c - A_{\mathcal{E}}^T s - A_{\mathcal{I}}^T \lambda = 0, && \\
&a_i^T x = b_i, && i \in \mathcal{E}, \\
&a_i^T x \geq b_i, && i \in \mathcal{I}, \\
&\lambda_i \geq 0, && i \in \mathcal{I}, \\
&\lambda_i(a_i^T x - b_i) = 0, && i \in \mathcal{I},
\end{aligned}$$

where

$$A_{\mathcal{E}} = \{a_i^T\}_{i\in\mathcal{E}} \in \mathbb{R}^{m_1\times n} \text{ and } A_{\mathcal{I}} = \{a_i^T\}_{i\in\mathcal{I}} \in \mathbb{R}^{m_2\times n}.$$

KKT conditions:

$$
\left.\begin{array}{c}
Gx + c - A_{\mathcal{E}}^T s - A_{\mathcal{I}}^T \lambda = 0, \\
a_i^T x = b_i, \qquad i \in \mathcal{E}, \\
a_i^T x \geq b_i, \qquad i \in \mathcal{I}, \\
\lambda_i \geq 0, \qquad i \in \mathcal{I}, \\
\lambda_i (a_i^T x - b_i) = 0, \quad i \in \mathcal{I},
\end{array}\right\} \stackrel{y = \{a_i^T x - b\}_{i \in \mathcal{I}}}{\Longrightarrow}
\begin{array}{c}
Gx + c - A_{\mathcal{E}}^T s - A_{\mathcal{I}}^T \lambda = 0, \\
A_{\mathcal{E}} x - b_{\mathcal{E}} = 0, \\
A_{\mathcal{I}} x - b_{\mathcal{I}} - y = 0, \\
(\lambda, y) \geq 0, \\
\lambda_i y_i = 0, i \in \mathcal{I}, \\
\text{where } b_{\mathcal{E}} = \{b_i\}_{i \in \mathcal{E}} \text{ and } b_{\mathcal{I}} = \{b_i\}_{i \in \mathcal{I}}
\end{array}
$$

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

$$
\left.\begin{array}{c}
\min_x \quad q(x) = \frac{1}{2} x^T G x + x^T c \\
\text{s.t.} \quad a_i^T x = b_i, i \in \mathcal{E} \\
a_i^T x \geq b_i, i \in \mathcal{I}
\end{array}\right\} \Longleftrightarrow \left\{\begin{array}{c}
\min_{x,y} \quad q(x, y) = \frac{1}{2} x^T G x + x^T c \\
\text{s.t.} \quad a_i^T x = b_i, i \in \mathcal{E} \\
a_i^T x - b_i - y = 0, i \in \mathcal{I} \\
y \geq 0
\end{array}\right.
$$

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

$$F(x, y, s, \lambda) = \begin{bmatrix} Gx + c - A_{\mathcal{E}}^T s - A_{\mathcal{I}}^T \lambda \\ A_{\mathcal{E}} x - b_{\mathcal{E}} \\ A_{\mathcal{I}} x - b_{\mathcal{I}} - y \\ \Lambda Y \mathbf{1} \end{bmatrix}$$

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

Follow the same idea in the interior point methods for linear programming

KKT conditions $\Longrightarrow$ a variant:

$$\left.\begin{array}{l} F(x,y,s,\lambda) = \begin{bmatrix} Gx + c - A_{\mathcal{E}}^T s - A_{\mathcal{I}}^T \lambda \\ A_{\mathcal{E}} x - b_{\mathcal{E}} \\ A_{\mathcal{I}} x - b_{\mathcal{I}} - y \\ \Lambda Y \mathbf{1} \end{bmatrix} = 0 \\ \qquad\qquad\qquad (\lambda, y) \geq 0, \end{array}\right\}$$

$$\Longrightarrow \left\{\begin{array}{l} \tilde{F}(x,y,s,\lambda) = \begin{bmatrix} Gx + c - A_{\mathcal{E}}^T s - A_{\mathcal{I}}^T \lambda \\ A_{\mathcal{E}} x - b_{\mathcal{E}} \\ A_{\mathcal{I}} x - b_{\mathcal{I}} - y \\ \Lambda Y \mathbf{1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \tau \mathbf{1} \end{bmatrix} \\ \qquad\qquad\qquad (\lambda, y) > 0, \end{array}\right. \tag{3}$$

where $\tau \geq 0$. Note that (3) has a unique solution, denoted by $(x^{(\tau)}, y^{(\tau)}, s^{(\tau)}, \lambda^{(\tau)})$

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

$$\begin{bmatrix} Gx + c - A_{\mathcal{E}}^T s - A_{\mathcal{I}}^T \lambda \\ A_{\mathcal{E}} x - b_{\mathcal{E}} \\ A_{\mathcal{I}} x - b_{\mathcal{I}} - y \\ \Lambda Y \mathbf{1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \tau \mathbf{1} \end{bmatrix}$$
$$(\lambda, y) > 0,$$

  e.g., by a step of the Newton's method
- Reduce τ appropriately in every iteration

- Duality measure $\mu = y^T \lambda / m_2$
- $$\begin{bmatrix} G & 0 & -A_{\mathcal{E}}^T & -A_{\mathcal{I}}^T \\ A_{\mathcal{E}} & 0 & 0 & 0 \\ A_{\mathcal{I}} & -I & 0 & 0 \\ 0 & \Lambda & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} -r^d \\ -r^p \\ -r^y \\ -\Lambda Y \mathbf{1} + \sigma \mu \mathbf{1} \end{bmatrix}$$

  where $r^d = Gx + c - A_{\mathcal{E}}^T s - A_{\mathcal{I}}^T \lambda$, $r^p = A_{\mathcal{E}} x - b_{\mathcal{E}}$, and $r^y = A_{\mathcal{I}} x - b_{\mathcal{I}} - y$.

Algorithm 2 Initial iterate

Input: Initial point (x, y, s, λ)
Output: Initial iterate $(x_0, y_0, s_0, \lambda_0)$;
1: Solve

$$\begin{bmatrix} G & 0 & -A_{\mathcal{E}}^T & -A_{\mathcal{I}}^T \\ A_{\mathcal{E}} & 0 & 0 & 0 \\ A_{\mathcal{I}} & -I & 0 & 0 \\ 0 & \Lambda & 0 & Y \end{bmatrix} \begin{bmatrix} \Delta x^{\mathrm{aff}} \\ \Delta y^{\mathrm{aff}} \\ \Delta s^{\mathrm{aff}} \\ \Delta \lambda^{\mathrm{aff}} \end{bmatrix} = \begin{bmatrix} -r^d \\ -r^p \\ -r^y \\ -\Lambda Y \mathbf{1} \end{bmatrix}$$

2: $x_0 \leftarrow x$, $y_0 \leftarrow \max(1, |y + \Delta y^{\mathrm{aff}}|)$, $s_0 \leftarrow s$, and $\lambda_0 \leftarrow \max(1, |\lambda + \Delta \lambda^{\mathrm{aff}}|)$

Algorithm 3 Predictor-corrector algorithm Part I

Input: Calculas $(x_0, y_0, s_0, \lambda_0)$ by Algorithm 2; $\{\eta_k \in [0.9, 1)\}$ $\forall k$ and $\eta_k \to 1$;

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

2: Solve

$$\begin{bmatrix} G & 0 & -A_{\mathcal{E}}^T & -A_{\mathcal{I}}^T \\ A_{\mathcal{E}} & 0 & 0 & 0 \\ A_{\mathcal{I}} & -I & 0 & 0 \\ 0 & \Lambda_k & 0 & Y_k \end{bmatrix} \begin{bmatrix} \Delta x_k^{\mathrm{aff}} \\ \Delta y_k^{\mathrm{aff}} \\ \Delta s_k^{\mathrm{aff}} \\ \Delta \lambda_k^{\mathrm{aff}} \end{bmatrix} = \begin{bmatrix} -r_k^d \\ -r_k^p \\ -r_k^y \\ -\Lambda_k Y_k \mathbf{1} \end{bmatrix}$$

3: Compute

$$\alpha_{\mathrm{aff}} \leftarrow \max(\alpha \in (0, 1] : (y_k, \lambda_k) + \alpha(\Delta y_k^{\mathrm{aff}}, \Delta \lambda_k^{\mathrm{aff}}) \geq 0)$$

$$\mu_{\mathrm{aff}} \leftarrow \left(y_k + \alpha_{\mathrm{aff}} \Delta y_k^{\mathrm{aff}}\right)^T \left(\lambda_k + \alpha_{\mathrm{aff}} \Delta \lambda_k^{\mathrm{aff}}\right) / n$$

$$\mu_k \leftarrow y_k^T \lambda_k / m_2 \text{ and } \sigma_k = (\mu_{\mathrm{aff}} / \mu_k)^3$$

4: Continue on the next page

Algorithm 4 Predictor-corrector algorithm Part II

1: Solve

$$\begin{bmatrix} G & 0 & -A_{\mathcal{E}}^T & -A_{\mathcal{I}}^T \\ A_{\mathcal{E}} & 0 & 0 & 0 \\ A_{\mathcal{I}} & -I & 0 & 0 \\ 0 & \Lambda_k & 0 & Y_k \end{bmatrix} \begin{bmatrix} \Delta x_k \\ \Delta y_k \\ \Delta s_k \\ \Delta \lambda_k \end{bmatrix} = \begin{bmatrix} -r_k^d \\ -r_k^p \\ -r_k^y \\ -\Lambda_k Y_k \mathbf{1} - \Delta \Lambda_k^{\mathrm{aff}} \Delta Y_k^{\mathrm{aff}} \mathbf{1} + \sigma_k \mu_k \mathbf{1} \end{bmatrix}$$

2: Compute

$$\begin{aligned} \alpha_{\eta_k}^{\mathrm{pri}} &= \max(\alpha \in (0,1] : y_k + \alpha \Delta y_k \geq (1-\eta_k) y_k) \\ \alpha_{\eta_k}^{\mathrm{dual}} &= \max(\alpha \in (0,1] : \lambda_k + \alpha \Delta \lambda_k \geq (1-\eta_k) \lambda_k) \\ \alpha_k &= \min(\alpha_{\eta_k}^{\mathrm{pri}}, \alpha_{\eta_k}^{\mathrm{dual}}) \end{aligned}$$

3: Set

$$(x_{k+1}, y_{k+1}, s_{k+1}, \lambda_{k+1}) = (x_k, y_k, s_k, \lambda_k) + \alpha_k (\Delta x_k, \Delta y_k, \Delta s_k, \Delta \lambda_k)$$

4: **end for**

- Dominated computational cost
- Solve for v:

$$\begin{bmatrix} G & 0 & -A_{\mathcal{E}}^T & -A_{\mathcal{I}}^T \\ A_{\mathcal{E}} & 0 & 0 & 0 \\ A_{\mathcal{I}} & -I & 0 & 0 \\ 0 & \Lambda_k & 0 & Y_k \end{bmatrix} \begin{bmatrix} \Delta x_k^{\text{aff}} \\ \Delta y_k^{\text{aff}} \\ \Delta s_k^{\text{aff}} \\ \Delta \lambda_k^{\text{aff}} \end{bmatrix} = \begin{bmatrix} -r_k^d \\ -r_k^p \\ -r_k^y \\ -\Lambda_k Y_k \mathbf{1} \end{bmatrix}$$

References I

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