Quadratic programming problems - a review on algorithms and applications (Active-set and interior point methods)

Dr. Abebe Geletu

Ilmenau University of Technology
Department of Simulation and Optimal Processes (SOP)

Topics

- Introduction
- Quadratic programming problems with equality constraints
- Quadratic programming problems with inequality constraints
- Primal Interior Point methods for Quadratic programming problems
- Primal-Dual-Interior Point methods Quadratic programming problems
- Linear model-predictive control (LMPC) and current issues
- References and Resources

Introduction - Quadratic optimization (is not programming)

- A general quadratic optimization (programming) problem

$$(QP) \qquad \min_x \left\{ \frac{1}{2} x^\top Q x + q^\top x \right\}$$
$$s.t.$$
$$Ax = a;$$
$$Bx \leq b;$$
$$x \geq 0,$$

where $Q \in \mathbb{R}^{n \times n}$ symmetric (not necessarily positive definite), $A \in \mathbb{R}^{m_1 \times n}$, $B \in \mathbb{R}^{m_2 \times n}$ and $m_1 \leq n$.

- Sometimes

$$\text{instead of } Ax = a \text{ we write} \qquad \mathbf{a}_i^\top x = a_i, i = 1, \ldots, m_1$$
$$\text{instead of } Bx = b \text{ we write} \qquad \mathbf{b}_j^\top x \leq b_j, j = 1, \ldots, m_2;$$

where $\mathbf{a}_i^\top$ is the $i-th$ row of A and $\mathbf{b}_j^\top$ is the j-th row of B.

Introduction...

Some applications of QP's:

- Least Square approximations and estimation
- Portfolio optimization
- Signal and image processing, computer vision, etc.
- Optimal contro, linear model predictive control, etc
- PDE-constrained optimization problems in CFD, CT, topology/shape optimization, etc
- Sequential quadratic programming (SQP) methods for NLP
- etc.

What has been achieved to date for the solution of nonlinear optimization problems has been really attained through methods of quadratic optimization and techniques of numerical linear algebra. As a result nowadays it is not surprising to see a profound interest in QP's and their real-time computing. QP's are now the driving force behind modern control technology.

QP introduction...SQP in brief

Given a constrained optimization problem

$$(NLP) \qquad \min_x f(x) \quad s.t. \quad h_i(x) = 0, i = 1, 2, \ldots, p; \quad g_j(x) \leq 0, j = 1, 2, \ldots, m;$$

with Lagrange function: $\mathcal{L}(x, \lambda, \mu) = f(x) + \lambda^\top h(x) + \mu^\top g(x)$.

A rough scheme for the SQP algorithm:

`Step 0`: start from x^0

`Step k`: $x^{k+1} = x^k + \alpha_k d_k$, where

- the **search-direction** d^k is computed using a quadratic programming problem:

$$(QP)_k \qquad \min_d \left\{ \frac{1}{2} d^\top H_k d + \nabla f(x_k)^\top d \right\}$$
$$s.t. \quad \nabla h_i(x_k)^\top d + h_i(x_k) = 0, i = 1, 2, \ldots, p;$$
$$\nabla g_j(x)^\top d + g_j(x_k) \leq 0, j \in \mathcal{A}(x^k) \subset \{1, 2, \ldots, m\};$$

where H_k is the Hessian of the Lagrangian $\nabla_x^2 \mathcal{L}(x^k, \lambda^k, \mu^k)$ or a Quasi-Newton approximation of it.

- the **step-length** α_k is determine by a 1D minimization of a **merit function** $\mathcal{M}\left(x^k + \alpha d^k\right)$ - a function that guarantees a sufficient decrease in the objective $f(x)$ and satisfaction of constraints along d^k with an appropriate step length α_k.

QP introduction - trajectory tracking for autonomous vehicles

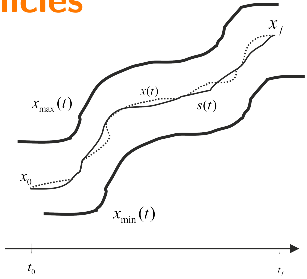

Figure: Trajectory tracking in a safe corridor

$$(OptCtrl) \qquad \min_u \frac{1}{2}\int_{t_0}^{t_f} \left\{ [x(t)-s(t)]^\top M\,[x(t)-u(t)] + u(t)^\top R u(t) \right\}$$

$$\begin{aligned}
&s.t.\\
&\quad \dot{x}(t) = Ax(t) + Bu(t),\, x(t_0) = x_0,\, x(t_f) = x_f;\\
&\quad x_{min}(t) \le x(t) \le x_{max}(t);\\
&\quad u_{min} \le u(t) \le u_{max};\\
&\quad t_0 \le t \le t_f.
\end{aligned}$$

Introduction to QP - Special classes

• Tracking problems for fast systems are better treated using model-predictive control (MPC).

• Practical engineering applications frequently lead to large-scale QP's.
• Seldom are these problems solved analytically.
• Numerical methods strongly depended on
  - the properties of the matrix Q - Q positive definite or not
  - if there are only equality constraints $Ax = b$
  - if the are only <u>bound constraints</u> $x_{min} \leq x \leq x_{max}$
  - if matrices exhibit sparsity properties and/or block structures
  - etc.

Some classification of QP's ....

Unconstraned QP

$$(QP) \quad \min_x \left\{ \frac{1}{2} x^\top Q x + q^\top x \right\}.$$

Box-constrained QP

$$(QP) \qquad \min_x \left\{ \frac{1}{2} x^\top Q x + q^\top x \right\}$$
$$a \leq x \leq b.$$

Equality constrained QP

$$(QP)_E \qquad \min_x \left\{ \frac{1}{2} x^\top Q x + q^\top x \right\}$$
$$Ax = a.$$

Inequality constrained QP

$$(QP)_I \qquad \min_x \left\{ \frac{1}{2} x^\top Q x + q^\top x \right\}$$
$$Ax = a$$
$$Bx \geq b.$$

Introduction to QP - Special classes

(I) Unconstrained QP

$$\min_x \left\{ \frac{1}{2} x^\top Q x + q^\top x \right\}$$

- **Known method:** Conjugate gradient methods (CG).

  Basically CG is used when the matrix Q is symmetric and positive definite.

- **Variants of CG:**
  - Hestens-Steifel 1952;
  - Fletcher-Reeves 1964;
  - Polak-Ribiere 1969

As in all iterative methods, CG methods may require preconditioning techniques to guarantee convergence to the correct solution $\Rightarrow$ leading to preconditioned CG (PCGG). This is necessary for large-scale QP's.

Read: - CG without an agonizing pain by J. R. Shewchuk.
- Matrix Computations by Golub & Van Loan.

Introduction to QP - Special classes

(II) Box-constrained QP's

$$\min_{x} \left\{ \frac{1}{2} x^\top Q x + q^\top x \right\}$$
$$s.t.$$
$$x_{min} \leq x \leq x_{max}.$$

- **This form of QP commonly arises, eg., :**

- in image recovery and deblurring, for instance, with lower and upper bounds on the gray-levels of the image;
- in linear quadratic control or linear model predictive control, with bound constraints on the control, etc. In fact, if

$$\min_{x} \left\{ \frac{1}{2} x^\top Q x + u^\top R u \right\}$$
$$s.t.$$
$$Ax + Bu = b$$
$$u_{min} \leq u \leq u_{max}.$$

Theoretically, we can solve for x in terms of u (quasi-sequential) so that $x = A^{-1}(b - Bu)$ so that we have

$$\min_{x} \left\{ \frac{1}{2} \left[A^{-1}(b - Bu) \right]^\top Q \left[A^{-1}(b - Bu) \right] + u^\top R u \right\}$$
$$s.t. \quad u_{min} \leq u \leq u_{max}.$$

Box constrained QP's

- **Known methods:** Iterative projection algorithms.
- **Variants:**
- The Brazilai-Borwein method (Brazilai & Borowein 1988)
- The Nesetrov gradient method (Nesterov 1983)
- Trust region methods (Celis, Dennis and Tapia 1985)

Important extensions and application-specific modifications of the above: gradient projection methods.

- Friedlander, A.; Martinez, J. M.; Raydan, M. A new method for large-scale box constrained convex quadratic minimization problems. Optim. Methods and Software, 7(1995), 149 – 154.
- Raydan, M. On the Brazilai and Borwein choice of steplength for gradient methods. IMA J. Numer. Anal. 13(1993), 321 – 326.
- Celis, M.; Dennis, J. E.; Tapia, R. A. A trust region strategy for nonlinear equality constrained optimization, in Numerical Optimization 1984, in P. Boggs, R. Byrd and R. Schnabel. eds.. SIAM. Philadelphia, 1985, pp. 71 – 82.
- Conn, A. R.; Gould, N. I. M.; Toint, T. Ph. L. Global convergence of a class of trust region algorithms for optimization with simple bounds. SIAM J. Numer. Anal. 25(1988), 764 – 767.
- Trust Region Methods by A. R. Conn, N. I. M. Gould, and Ph. L. Toint

Some recent engineering applications:

- Beck, A.; Tiboulle, M. Fast Gradient-based algorithms for constrained total variation image denoising and deblurring problems. IEEE Trans. on Image Processing, 18(2009), 2419 – 2434.
- Hauska, B.; Ferreau, H. J.; Diehl, M. An auto-generated real-time iteration algorithm for nonlinear mpc in the microseconds range. Automatica, 47(2011), 2279 – 2285.
- Hu, Y.-Q.; Dai Y.-H. Inexact Barzilai-Borwein method for saddle point problems. Numerical Linear Algebra with Applications. 14(2007), Issue 4, pp. 299–317.
- Zometa, P.; Kügel, M.; Faulwasser, T.; Findeisen, R. Implementation aspects of model predictive control for embedded systems. Procedings of the 2012 American Control Conference, Montreal, Canada, pp. 1205–1210.

Equality constrained QP's

(III) Equality constrained QP's

$$(QP) \qquad \min_{x} \left\{ \frac{1}{2} x^\top Q x + q^\top x \right\} \\ s.t. \\ Ax = a. \tag{1}$$

There are two cases:

(a) Q is symmetric and positive semi-definite $\Rightarrow$ QP is convex.
(b) Q is symmetric but not positive semi-definite $\Rightarrow$ QP is non-convex.

Note: The issue that wether A is of full-rank or rank-deficient (i.e. $rank(A) = m_1$ or $rank(A) < m_1$) is not serious.

Equality constrained QP's

- Lagrange function: $\mathcal{L}(x,\lambda) = \left\{\frac{1}{2}x^\top Qx + q^\top x\right\} + \lambda^\top (Ax - b)$
- **KKT conditions (KKT-equation):**

$$\begin{aligned} Qx + A^\top\lambda &= -q, \\ Ax &= b. \end{aligned} \Rightarrow \underbrace{\begin{bmatrix} Q & A^\top \\ A & \mathbb{O} \end{bmatrix}}_{=:K} \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} -q \\ b \end{bmatrix}$$

$\Rightarrow$ the optimization problem is reduced to the **solution of a (possibly large-scale) system of linear equations**.

- K is a symmetric matrix; but it may or may not be positive definite.

Equality constrained QP's

Some suggestions:

(A) A numerical analyst approach:

▶ If K is positive definite, then use a pre-conditioned conjugate gradient (PCG) method.

▶ If K is not positive definite, then transform $Ky = p$ to $\left(K^\top K\right) y = K^\top p$ and then apply PCG. Notes that the matrix $\left(K^\top K\right)$ is positive definite

• Almost all state-of-the-art linear algebra packages include a PCG solver.

Find more, eg., from: Matrix computations, 2nd ed., by Golub and Van Loan.

(B) An optimization specialist approach:

• Use (gradient) projection along with the conjugate gradient method.

Recommended reading:

• Gould, N.I.M. On practical conditions for the existence and uniqueness of solutions to the general equality quadratic programming problem. Math. Programming, 32(1985), 90 – 99.

• Gould, N.I.M., Hiribar, M.E., Nocedal, J. On the solution of equality constrained quadratic programming problems arising in optimization. SIAM J. Sci. Comput., 23(2001), 1376 – 1395.

Equality constrained QP's

NB:

• If Q is positive definite, the solution of the KKT-equation is a solution for the QP (convex optimization problem);

• otherwise, the solution of the PCG solver needs to be verified.

Special case: when Q is positive definite and A has full rank ($rank(A) = m_1$), the matrix $\begin{bmatrix} Q & A^\top \\ A & \mathbb{O} \end{bmatrix}$ is invertible.

▶ **small-scale quadratic optimization problems** can be solved directly to obtain the solution

$$\Rightarrow \qquad \begin{aligned} x^* &= -Q^{-1}q + Q^{-1}A^\top \left(AQ^{-1}A^\top\right)^{-1} \left(b + AQ^{-1}q\right) \\ \lambda^* &= -\left(AQ^{-1}A^\top\right)^{-1} \left(b + AQ^{-1}q\right). \end{aligned}$$

▶ However, in general, avoid inversion of a matrix in computations.

▶ **Small- to medium-scale** KKT-equations can be efficiently solved by using a direct-linear algebra algorithm, eg., using Choleski or QR factorizations.

Further recommended reading:

• Practical optimization by Gill, Murray and Wright.

QP with inequality constraints

$$(QP)_I \qquad \min_x \left\{ \frac{1}{2} x^\top Q x + q^\top x \right\}$$
$$s.t.$$
$$Ax = a.$$
$$Bx \le b.$$

There are two classes of algorithms:

- the active-set method (ASM)
- the interior point method (IPM).

Quadratic optimization ... Active set Method

Active Set Method

Strategy:

- Start from an arbitrary point x^0
- Find the next iterate by setting $x^{k+1} = x^k + \alpha_k d^k$,

where α_k is a step-length and d^k is search direction.

Question

- How to determine the search direction d^k?
- How to determine the step-length α_k?

(A) **Determination of the search direction:**

- At the current iterate x^k determine the index set of active the inequality constraints

$$\mathcal{A}^k = \{j \mid \mathbf{b}_j^\top x^k - b_j = 0, j = 1, \ldots, m_2\}.$$

Quadratic optimization ... Active set Method

- Solve the direction finding problem

$$\min_d \left\{ \frac{1}{2}(x^k + d)^\top Q(x^k + d) + q^\top (x^k + d) \right\}$$
$$s.t.$$
$$\mathbf{a}_i^\top (x^k + d) = a_i, i = 1, \ldots, m_1;$$
$$\mathbf{b}_j^\top (x^k + d) = b_j, j \in \mathcal{A}^k.$$

Expand

- $\frac{1}{2}(x^k + d)^\top Q(x^k + d) + q^\top (x^k + d) = \frac{1}{2}d^\top Qd + \underline{\frac{1}{2}d^\top Qx^k + \frac{1}{2}(x^k)^\top Qd} + \frac{1}{2}(x^k)^\top Qx^k + q^\top d + q^\top x^k$
- $\mathbf{a}_i \top \left(x^k + d\right) = a_i \Rightarrow \mathbf{a}_i^\top d = a_i - \mathbf{a}_i^\top x^k = 0$. Similarly, $\mathbf{b}_j \top d = b_j - \mathbf{b}_j^\top x^k = 0$.
- Simplify these expressions and drop constants to obtain:

$$\min_d \left\{ \frac{1}{2}d^\top Qd + \left[Qx^k + q\right]^\top d \right\}$$
$$s.t.$$
$$Ad = 0,$$
$$\widetilde{B}d = 0;$$

where

$$\widetilde{B} = \begin{bmatrix} \vdots \\ b_j^\top \\ \vdots \end{bmatrix}, j \in \mathcal{A}^k.$$ Set $g^k = Qx^k + q$.

Quadratic optimization ... Active set Method

- To obtain the search direction d^k, solve the **equality constrained QP:**

$$\min_d \left\{ \frac{1}{2} d^\top Q d + \left[g^k\right]^\top d \right\}$$
$$s.t.$$
$$Ad = 0,$$
$$\widetilde{B} d = 0.$$

The KKT optimality conditions lead to the system:

$$\begin{array}{ll} Qd + g^k + A^\top \lambda + \widetilde{B}^\top \widetilde{\mu} & = 0, \\ Ad & = 0, \\ \widetilde{B} d & = 0. \end{array} \Rightarrow \begin{bmatrix} Q & A^\top & \widetilde{B}^\top \\ A & \mathbb{O}\ldots & \mathbb{O} \\ \widetilde{B} & \mathbb{O}\ldots & \mathbb{O} \end{bmatrix} \begin{bmatrix} d \\ \lambda \\ \widetilde{\mu} \end{bmatrix} = \begin{bmatrix} -g^k \\ 0 \\ 0 \end{bmatrix} (*)$$

where λ and $\widetilde{\mu}$ are Lagrange multipliers corresponding to equality and active inequality constraints, resp.

Quadratic optimization ... Active set Method

- Apply algorithms for equality constrained QP's to obtain d^k.

- If d^k is a solution of QP, then there are λ^k and $\widetilde{\mu}^k$ such that

(KKT conditions for the equality constrained QP)

$$\begin{aligned} Qd^k + g^k + A^\top \lambda^k + \widetilde{B}^\top \widetilde{\mu}^k &= 0, \\ Ad^k &= 0, \\ \widetilde{B} d^k &= 0. \end{aligned}$$

- There are two cases: either $d^k = 0$ or $d^k \neq 0$.

Case 1: $d^k = 0$. Then, the system above reduces to

$$g^k + A^\top \lambda^k + \widetilde{B}^\top \widetilde{\mu}^k = 0, \quad (**)$$

Case 1-a: If $\widetilde{\mu}^k \geq 0$, (i.e. $\widetilde{\mu}_j^k \geq 0, j \in \mathcal{A}^k$), then corresponding to the non-active constraints we can set $\mu_j^k = 0$ for $j \in \{1, 2, \ldots, m_2\} \setminus \mathcal{A}^k$. Hence, we have for the original problem

$$\begin{aligned} g^k + A^\top \lambda^k + B\mu^k &= 0, \\ Ax^k - a &= 0, \\ Bx^k - b &\leq 0, \\ \left(Bx^k - b_k\right)^\top \mu^k &= 0 \\ \mu_k &\geq 0. \end{aligned}$$

Quadratic optimization ... Active set Method

- If $d^k = 0$ and $\widetilde{\mu}_k \geq 0$, $x^{k+1} = x^k$ is a KKT point. Stop!

- **Case 1-b:** If some components of $\widetilde{\mu}^k$ are negative, then x^k is not an optimal solution. Let $\mu_{j_0} = \min\left\{\widetilde{\mu}_j \mid \widetilde{\mu}_j < 0, j \in \mathcal{A}^k\right\}$. Remove the index j_0 from $\mathcal{A}^k$ and solve the quadratic programming problem

$$\min_d \left\{\frac{1}{2} d^\top Q d + \left[g^k\right]^\top d\right\}$$
$$s.t.$$
$$\mathbf{a}_i^\top d = 0,$$
$$\mathbf{b}_j^\top d = 0, j \in \mathcal{A}^k \setminus \{j_0\}.$$

Then the direction obtained is descent direction d^k for $(\mathrm{QP})_I$.

Quadratic optimization ... Active set Method

Case 2: $d^k \neq 0$.

• Determine a step-length α_k that guarantees $x^k + \alpha_k d^k$ is feasible to $(\mathrm{QP})_{In}$.

We need to choose α_k so that $Ax^{k+1} = A(x^k + \alpha_k d^k) = a$ and $Bx^{k+1} = B(x^k + \alpha_k d^k) \leq b$.

• For the equality constraints $Ax = b$, since x^k is feasible to $(\mathrm{QP})_{In}$ and $Ad_k = 0$ (in $(\mathrm{QP})_E$), we have

$$Ax^{k+1} = A(x^k + \alpha_k d^k) = Ax^k + \alpha_k Ad_k = a + 0 = a$$

So the equality constraints are satisfied for any α_k.

• For the active inequality constraints $Bx \leq b$, similarly we have $\mathbf{b}_j^\top x^{k+1} = \mathbf{b}_j^\top \left(x^k + \alpha_k d^k\right) = b_j \leq b_j, j \in \mathcal{A}^k$ for any α_k.

• For the inactive inequality constraints at x^k, i.e. $j \notin \mathcal{A}^k$, we have $\mathbf{b}^\top x^k < b_j$. Thus, we need to determine α_k so that $\mathbf{b}_j^\top (x^k + \alpha_k d^k) \leq b_j$ holds true for $j \notin \mathcal{A}^k$. Hence,

$$\mathbf{b}_j^\top x^k + \alpha_k \mathbf{b}_j^\top d^k) \leq b_j \Rightarrow \alpha_k \mathbf{b}_j^\top d^k \leq b_j - \mathbf{b}_j^\top x^k.$$

(i) If $b_j^\top d^k \leq 0$, then $\alpha_k \mathbf{b}_j^\top d^k \leq b_j - \mathbf{b}_j^\top x^k$ is satisfied for any $\alpha_k > 0$, since $0 \leq b_j - \mathbf{b}_j^\top x^k$ (recall that x^k is feasible for QP_I).

(ii) If $b_j^\top d^k > 0$, then choose α_k

$$\alpha_k = \frac{b_j - b_j^\top d^k}{b_j^\top d^k}.$$

Quadratic optimization ... Active set Method

- A common α_k that guarantees the satisfaction of all constraints is

$$\alpha_k = \min\left\{1, \frac{b_j - b_j^\top d^k}{b_j^\top d^k} \mid j \notin \mathcal{A}^k \text{ and } b_j^\top d^k > 0\right\}.$$

- Updating the active index set:

Observe that if $\alpha_k < 1$, then $\alpha_k = \frac{b_{j_0} - b_{j_0}^\top d^k}{b_{j_0}^\top d^k} =$ for some $j_0 \notin \mathcal{A}^k$. This implies that $\mathbf{b}_{j_0}^\top \left(x^k + \alpha_k d^k\right) = b_{j_0}$.

That is, the inequality constraint corresponding to the index j_0 be comes active.

- If $\alpha_k < 1$, then update the active index set as

$$\mathcal{A}^{k+1} = \mathcal{A}^k \cup \{j_0\}.$$

Algorithm 1: An active-set algorithm for QP

1: **Give a start vector** x^0;
2: **Identify the active index set** $\mathcal{A}^0$;
3: **Set** $k = 0$;
4: **while (no convergence) do**
5:     **Compute** $g^k = Qx^k + q$;
6:     **Obtain** d^k, λ^k **and** $\widetilde{\mu}^k$ **by solving the KKT-equations for**

$$\min_d \left\{ \frac{1}{2} d^\top Qd + \left[g^k\right]^\top d \right\}$$
$$s.t.$$
$$\mathbf{a}_i^\top d = 0,$$
$$\mathbf{b}_j^\top d = 0, j \in \mathcal{A}^k.$$

7:     **if** $(d^k = 0)$ **then**
8:         **if** $\widetilde{\mu}^k \geq 0$ **then**
9:             **STOP!** x^k **is a KKT point.**
10:         **else**
11:             $\widetilde{\mu}_{j_0} = \min\{\widetilde{\mu}_j \mid \widetilde{\mu}_j < 0, j \in \mathcal{A}^k\}$
12:             **Update the index set** $\mathcal{A}^k \leftarrow \mathcal{A}^k \setminus \{j_0\}$ **and GOTO Step 6.**
13:         **end if**
14:     **end if**
15: **end while((continue..))**

16. if ($d^k \neq 0$) 17. Compute the step-length

$$\alpha_k = \min\left\{1, \frac{b_j - b_j^\top d^k}{b_j^\top d^k} \mid j \notin \mathcal{A}^k \text{ and } b_j^\top d^k > 0\right\}.$$

18. Update $x^{k+1} = x^k + \alpha_k d^k$.

19. Update active index-set: if $\alpha_k = 1$, then $\mathcal{A}^{k+1} = \mathcal{A}^k$, else $\mathcal{A}^{k+1} = \mathcal{A}^k \cup \{j_0\}$, where $\alpha_k = \frac{b_{j_0} - b_{j_0}^\top d^k}{b_{j_0}^\top d^k}$ for $b_{j_0}^\top d^k > 0$.

20. Update $k \leftarrow k + 1$.

21 end if

22. end while

Important issues:

- How to determine a starting iterate x^0 for the active-set algorithm.
- ASM requires an efficient strategy for the determination of active sets $\mathcal{A}^k$ at each x^k.

Quadratic optimization ... Active set Method

Advantages of ASM:

• Since only active constraints are consider at each iteration x^k, $(\mathrm{QP})_E$ usually has only a few constraints and can be solved fast.

$\Rightarrow$ Large-scale $(\mathrm{QP})_I$'s are easy to solve.

• In many cases the active set varies slightly from step-to-step, making active set method efficient.

$\Rightarrow$ Data obtained from the current QP_E can be used to solve the next QP_E known as **warm starting**.

• All iterates x^k are feasible to $(\mathrm{QP})_I$. This is an important property, eg, in SQP.

Disadvantages of ASM:

• Since the active-set $\mathcal{A}^k$ may vary from step to step, the structure and properties, eg. sparsity, of constraint matrices may change.

• ASM may become slower near to the optimal solution;.

• For some problems ASM can be computationally expensive.

Quadratic optimization... PD Interior point method

Consider

$$(QP) \qquad \min_x \left\{ \frac{1}{2} x^\top Q x + q^\top x \right\}$$
$$s.t.$$
$$Ax = a.$$
$$x \geq 0,$$

where $A \in \mathbb{R}^{m_1 \times n}$.

Any other constraints can be transformed to the above form.

- If there is $Bx \leq b$, then add slack variables s to obtain $Bx + s = b$, $s \geq 0$ so that the constraints become

$$\begin{bmatrix} A & O \\ B & I_n \end{bmatrix} \begin{bmatrix} x \\ s \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}$$
$$\begin{bmatrix} x \\ s \end{bmatrix} \geq 0.$$

- If there is a box constraint $x_{min} \leq x \leq x_{max}$, then split it in two constraints

$$\begin{aligned} x_{max} - x &\geq 0 \\ x - x_{min} &\geq 0. \end{aligned}$$

Quadratic optimization... PD Interior point metho

$\Rightarrow$ Use the logarithmic barrier function for the non-negative variables x.

$\Rightarrow$ The barrier problem associated to (QP) will be

$$(BQP)_\mu \qquad \min_x \left\{ \frac{1}{2} x^\top Q x + q^\top x - \mu \sum_{j=1}^{n} \log x_i \right\}$$
$$s.t.$$
$$Ax = a.$$

- The Lagrange function of $(\mathrm{BQP})_\mu$

$$\mathcal{L}_\mu(x, \lambda) = \frac{1}{2} x^\top Q x + q^\top x - \lambda^\top (Ax - a) - \mu \sum_{j=1}^{n} \log x_i,$$

where $x > 0$.

Quadratic optimization... PD Interior point method

- The KKT condition for $(\text{BQP})_\mu$ will be

$$\begin{aligned}\frac{\partial \mathcal{L}_\mu}{\partial x} &= 0 \Rightarrow Qx + q - A^\top \lambda - \mu X^{-1}\mathbf{e} = 0, x > 0.\\ \frac{\partial \mathcal{L}_\mu}{\partial \lambda} &= 0 \Rightarrow Ax = b,\end{aligned}$$

where $\mathbf{e} = (1, \ldots, 1)$.

$$\begin{aligned}\Rightarrow Qx - A^\top \lambda - \mu X^{-1}\mathbf{e} &= -q\\ Ax &= b.\end{aligned}$$

- Define $s = \mu X^{-1}\mathbf{e}$. Then $s \in \mathbb{R}^n$, $s > 0$ and $x_i s_i = \mu, i = 1, \ldots, n$.
- The KKT condition for $(\text{BQP})_\mu$ takes the form

$$\begin{aligned}\Rightarrow Qx - A^\top \lambda - s &= -q\\ Ax &= b\\ x_i s_i &= \mu, i = 1, \ldots n.\end{aligned}$$

Quadratic optimization... PD Interior point method

- For each fixed parameter μ, the KKT condition for $(BQP)_\mu$ is a system of non-linear equations (nonlinearity due to the product $x_i s_i = \mu$).
- There are $2n + m_1$ variables (x, λ, s). Set

$$F_\mu(x, \lambda, s) = \begin{bmatrix} Qx - A^\top \lambda - s - q \\ Ax - b \\ XS - \mu \mathbf{e} \end{bmatrix},$$

where $X = diag(x) \in \mathbb{R}^{n \times n}$ and $S = diag(s) \in \mathbb{R}^{n \times n}$.
- For each fixed $\mu > 0$, the equation

$$F_\mu(x, \lambda, s) = 0.$$

should be solved.

Quadratic optimization... PD Interior point methoc

- The system can be solved using a Newton-like algorithm.

A general Newton algorithm:

`Step 0:` Give an initial iterate (x^0, λ^0, s^0) where $(x^0, s^0) > 0$.

`Step k:` Given (x^k, λ^k, s^k):

◦ solve the system of equations

$$J_{F_\mu}\left(x^k, \lambda^k, s^k\right) d = -F_\mu\left(x^k, \lambda^k, s^k\right)$$

to determine the search direction $d^k = (\Delta x_k, \Delta\lambda_k, \Delta s_k)$.

◦ Determine a step-length α_k.

◦ Set

$$\begin{aligned} x^{k+1} &= x^k + \alpha_k \Delta x_k \\ \lambda^{k+1} &= \lambda^k + \alpha_k \Delta\lambda_k \\ s^{k+1} &= s^k + \alpha_k \Delta s_k. \end{aligned}$$

Quadratic optimization... PD Interior point method

- At each iteration the **system of linear equations**

$$J_{F_\mu}\left(x^k, \lambda^k, s^k\right) d = -F\left(x^k, \lambda^k, s^k\right)$$

should be solved.

- The Jacobian matrix $J_{F_\mu}\left(x^k, \lambda^k, s^k\right)$ has the special form

$$J_{F_\mu}\left(x^k, \lambda^k, s^k\right) = \begin{bmatrix} Q & -A^\top & -I \\ A & \mathbb{O} & \mathbb{O} \\ S^k & \mathbb{O} & X^k \end{bmatrix},$$

where $X^k = diag(x^k)$ and $S^k = diag(s^k)$.

In interior-point method, most computation effort is spent for solving the system of linear equations to determine d^k. This should be accomplished by an efficient linear algebra solver.

Quadratic optimization... PD Interior point methoc

• There are various prima-dual interior-point algorithms with modifications on the above general algorithm. ∗ Path-following Algorithm ∗ Affine-scaling Algorithm, ∗ Mehrotra predicator-corrector Algorithm etc.

Algorithm 2: Path-following Algorithm for QP

Choose σ_{min};
Choose initial iterates (x^0, λ^0, s^0) **with** $(x^0, s^0) > 0$.
Set $k = 0$;
while $\left(\left(\left(x^k\right)^\top s^k / n\right) > \varepsilon\right)$ **do**
  Choose $\sigma_k \in [\sigma_{min}, \sigma_{max}]$;
  Solve the system

$$J_{F_\mu}\left(x^k, \lambda^k, s^k\right) d = -\begin{bmatrix} Qx - A^\top \lambda - s - q \\ Ax - b \\ XS - \sigma_k \mu_k \mathbf{e} \end{bmatrix},$$

  to determine $d^k = (\Delta x_k, \Delta\lambda_k, \Delta s_k)$.
  Choose α_{max} **the largest value such that**

$$(x^k, s^k) + \alpha\,(\Delta x_k, \Delta s_k) > 0;$$

  Set $\alpha_k = \min\{1, \eta_k \alpha_{max}\}$ **for some** $\eta_k \in (0, 1)$ **and** $\mu_k = \dfrac{\left(x^k\right)^\top s^k}{n}$;
  Set $(x^{k+1}, \lambda^{k+1}, s^{k+1}) + \alpha_k\left(\Delta x_k, \Delta\lambda^k, \Delta s_k\right)$
  Update $k \leftarrow k + 1$.
end while

Quadratic optimization... PD Interior point metho

- Commonly used termination criteria is

$$\frac{\left(x^k\right)^\top s^k}{n} \leq \varepsilon.$$

for some termination tolerance $\varepsilon > 0$, eg. $\varepsilon = 0.001$, etc.

Some strategies for choice of centering parameter:
(a) $\sigma_k = 0, k = 1, 2, \ldots,$ - affine-scaling approach;
(b) $\sigma_k = 1, k = 1, 2, \ldots,$
(c) $\sigma_k \in [\sigma_{min}, \sigma_{max}] = 1, k = 1, 2, \ldots$ Commonly, $\sigma_{min} = 0.01$ and $\sigma_{max} = 0.75$ (path following method)
(d) $\sigma_k = 1 - \frac{1}{\sqrt{n}}, k = 1, 2, \ldots,$ (with $\alpha_k = 1$ - short-step path-following method)

Quadratic optimization... PD Interior point metho

Advantages of PDIPM:

• Can be used to solve medium- to large-scale QPs $\Rightarrow$ so important in optimal control.

• Fast convergence properties.

• Structure of the Jacobian matrix does not change a lot.

• The Jacobian matrix $J_{F_{\mu_k}}$ takes a special structure, which can be exploited by a linear algebra solver.

Disadvantages of PDIPM:

• Usually requires a strictly feasible initial iterate (x^0, s^0) which is not trivial to obtain.

• All constraints are used in the computation of the search direction d^k, even if some are not active.

$\Rightarrow$ The computation of d^k may be only attained through iterative linear algebra solvers.

Some current Issues

- **Optimization methods for real-time control.**

Applications for fast real-time systems control:
• Control of autonomous (ground, arial or underwater) vehicles and Robots computation of QP's in microseconds.
$\Rightarrow$ Require hardware (embedded systems) implementation of optimization algorithms.

- **Parallel implementation on graphic processors.**

• Computer graphic cards provided an opportunity for parallel linear algebra computation.
$\Rightarrow$ Large QP's can solved on shared memory PC with graphic processors.
Example:

- Graphic cards with processors: Latest NVIDA Graphic cards, AMD Athlon graphic cards.
- Programming Languages: CUDA , OpenCL (Athlon), etc.

- **Parallel implementation on shared and distributed memory (multiprocess) computers.**

For instance in the

- numerical solution of optimization problems with partial differential constrains; Applications: CFD optimization, Computer Tomography, Thermodynamics, etc
- large-scale stochastic simulation and optimization.

Resources

Some use full links:

- A quadratic programming page, N. Gould and P. Toint
  URL: http://www.numerical.rl.ac.uk/qp/qp.html
- Computational Optimization Laboratory, Y. Ye, Stanford University,
  URL: http://www.stanford.edu/ yyye/Col.html
- Convex Optimization, Stephen P. Boyd, Stanford University,
  URL: http://www.stanford.edu/ boyd/software.html (Book, software, etc)
- The HSL Mathematical Software Library (HSL=Harwell Subroutine Library),
  Numerical Analysis Group at the STFC Rutherford Appleton Laboratory
  URL: http://www.hsl.rl.ac.uk/

Software:

- MINQ: General Definite and Bound Constrained Indefinite Quadratic Programming, Arnold Neumaier, University of Vienna
  URL: http://www.mat.univie.ac.at/ neum/software/minq/
- qpOASES - Online Active Set Strategy, H.J. Ferreau, University of Lauven (Belgium),
  URL: http://www.kuleuven.be/optec/software/qpOASES/
- QPC - Quadratic Programming in C Adrian Wills, University of Newcastle,
  URL: http://sigpromu.org/quadprog/
- OOQP is an object-oriented C++ package, M. Gertz and S. Wright
  URL: http://pages.cs.wisc.edu/ swright/ooqp/
- **I**nterior **p**oint **Opt**imizer (IpOpt), A. Wächter and C. Laird URL: https://projects.coin-or.org/Ipopt/
- NLPy is a Python package for numerical optimization, D. Orban, Ecole Polytechnique de Montreal,
  URL: http://nlpy.sourceforge.net/
- Python based optimization solvers
  URL: http://wiki.python.org/moin/NumericAndScientific

References

Some references on active set methods:

• R.A. Bartlett and L.T. Biegler: QPSchur: a dual, active-set- Schur-complement method for large-scale and structured convex quadratic programming. Optim. Eng., 7(2006), 5–32.

• D. Goldfarb, A. Idnani: A numerically stable dual method for solving strictly convex quadratic programming. Math. Prog., 27(1983), 1 – 33.

• C.M. Maes: A Regularized Active-Set Method for Sparse Convex Quadratic Programming. PhD thesis, Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA, August 2010.

• A. Potschka, C. Kirches, H.G: Bock and J.P. Shlöder: Reliable solution of convex quadratic programs with parametric active set methods. Optimization Online 2011.
URL: http://www.optimization-online.org/DB_HTML/2010/11/2828.html.

• M.J.D. Powell: On the quadratic programming algorithm of Goldfarb and Idnani. Math. Programming Stud., 25(1985), 46–61.

• E. Wong: Active-Set Methods for Quadratic Programming. Phd thesis, University of California, San Diego, 2011.

Some references on interior point methods:

• M. D'Apuzzo, V. De Simone, D. di Serafino: On mutual impact of numerical linear algebra and large-scale optimization with focus on interior point methods. Comput. Optim. Appl., 45(2010), 283 – 310. • A. Forsgren, P. E. Gill, M.H. Wright. Interior point methods for nonlinear optimization, SIAM Review, 44(2002), 525 – 597.

• J. Gondzio: Matrix-Free Interior Point Method. Optimization Online, 2009.

• J. Gondzio: Interior point methods 25 years later, European Journal of Operational Research 218 (2012) pp. 587-601. Published online: October 8, 2011. DOI 10.1016/j.ejor.2011.09.017.

• N. Karamarkar: A new polynomial-time algorithm for linear programming, Combinatorica, 4(1984), 373 – 395.

• M. H. Wright: The interior-point revolution in optimization: history, recent developments and lasting consequences. Bull. of the American Math. Society, 62(2004), 39 – 56.

• R.D.C. Monterio and I. Adler: Interior path-following primal dual algorithms, Part II: Convex quadratic programming. Math. Prog. 44(1989), 45 – 66.

• Y. Nesterov and A. Nemirovsky: Interior-point Polynomial Algorithms in Convex Programming. SIAM Philadelphia, PA, 1994.

• S. J. Wright. Prima-dual interior-point methods. SIAM Philadelphia 1997.

• Y. Ye: Interior point algorithms: Theory and analysis. Wiley, NY, 1997.

References

Some references on real-time control:

- R.A. Bartlett, A. Wächter and L.T. Biegler: Active set vs interior point strategies for model predictive control. Proceedings of the American Control Conference, Chicago, Illinois, June 2000.
- A.K. Abbes, F. Bouani and M. Ksouri: A Microcontroller Implementation of Constrained Model Predictive Control. World Academy of Science, Engineering and Technology, 80, 2011.
- H. J. Ferreau, H. G. Bock, and M. Diehl: An online active set strategy to overcome the limitations of explicit MPC. Inter. J. of Robust and Nonlinear Ctrl., 18(2008), 816 – 830.
- K.V. Ling, B.F. Wu and J.M. Maciejowski: Embedded Model Predictive Control (MPC) using a FPGA. Proceedings of the 17th World Congress The International Federation of Automatic Control Seoul, Korea, July 6-11, 2008.
- M.S.K. Lau, S.P. Yue, K.V. Ling and J.M. Maciejowski: A comparison of interior point and active set methods for FPGA implementation of model predictive control. Proc. European Control Conference, Budapest, August 23- 26, 2009.
- C. V. Rao, S. J. Wright and J. B. Rawlings: Application of Interior-Point Methods to Model Predictive Control. J. of Optim. Applic., 99(1998), 723 –757.
- Y. Wang and S. Boyd: Fast model predictive control using online optimization. IEEE Trans. on Ctrl. Sys. Tech. 18(2010), 267 – 278.
- A. Wills and A. Mills and B. Ninness: FPGA implementation of an interior-point solution for linear model predictive control. The 18th IFAC World Congress, Milano (Italy), August 28 - September 2, 2011.

References

Some classical references on conjugate gradient methods:

- R. Fletcher and M.J.D. Powell: Function Minimization by Conjugate Gradients. Comp. J., 7(1964), 149 – 154.
- R. Fletcher and C. Reeves: Function minimization by conjugate gradients. Comput. J., 7(1964), 149 – 154.
- R. Fletcher, Practical Methods of Optimization, John Wiley & Sons, New York, 1987.
- G.H. Golub and C.F. Van Loan: Matrix Computations. Johns Hopkins University Press, Baltimore, 3rd ed., 1996.
- M. R. Hestenes and E. L. Stiefel: Methods of conjugate gradients for solving linear systems, J. Research Nat. Bur. Standards, 49 (1952), 409 – 436.
- E. Polak , Computational Methods in Optimization: A Unified Approach. Academic Press Inc, London, 1971.
- E. Polak and G. Ribiere: Note sur la convergence de directions conjugées, Rev. Francaise Informat Recherche Opertionelle, 3e Année 16(1969), 35 – 43.
- M.J.D. Powell: Some convergence properties of the conjugate gradient method, Math. Prog., 11(1976), 42 – 49.
- M.J.D. Powell: Restart procedures of the conjugate gradient method. Math. Prog., 2(1977), 241 – 254.
- B.T. Polyak: The conjugate gradient method in extremal problems. USSR Comput. Math. Math. Phys., 9(1969), 94 – 112.
- J.R. Shewchuk: An Introduction to the Conjugate Gradient Method Without the Agonizing Pain.
  (URL: http://www.cs.cmu.edu/ quake-papers/painless-conjugate-gradient.pdf)

Some references on box-constrained QP:

- J. Brazilai and J.M. Borwein: Two-point step size gradient methods. IMA J. Numer. Anal. 8(1988), 141 – 148.
- Y. Nesterov: A method of solving convex programming problems with convergence rate $O\left(1/k^2\right)$. Soviet Mathematics Doklady, 27(1983), 372 – 376.