PRO 5961 Métodos de Otimização Não Linear

Celma de Oliveira Ribeiro

Aula 9 - 2023

Quadratic programming

Departamento de Engenharia de Produção
Universidade de São Paulo

PrGen

Consider problem

$$\begin{array}{lll} & \text{minimize } f(x) & \\ \text{s.t} & g_i(x) \leq 0 & \text{i} \in \mathcal{I} = \{1, 2, \ldots m\} \\ & h_i(x) = 0 & \text{i} \in \mathcal{E} = \{1, 2, \ldots l\} \end{array}$$

Definition

x^* is KKT point if there are lagrange multipliers vectors λ^* and μ^*, such that $\begin{bmatrix} x^* & \lambda^* & \mu^* \end{bmatrix}^t$ satisfies:

$$\begin{array}{lll} \nabla_x \mathcal{L}(x^*, \lambda^*, \mu^*) & = & 0 \\ g(x^*) & \leq 0 & \\ h(x^*) & = 0 & \\ \mu^* & \geq 0 & \\ \mu_i g_i(x^*) & = 0 & \forall i \in \mathcal{I} \end{array}$$

Quadratic Programming

Quadratic Programming represents a special class of nonlinear programming in which the objective function is quadratic and the constraints are linear.

The general quadratic problem (QP) can be stated as

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}x^t Q x + c^t x$$

$$\begin{aligned} s.t \quad & Ax \leq a \\ & Bx = b \end{aligned}$$

with $Q \in \mathbb{R}^{n\times n}$ a symetric matrix (not necessarily positive definite), $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m\times n}$,$a \in \mathbb{R}^m$, $B \in \mathbb{R}^{\times n}$,$b \in \mathbb{R}^l$

Quadratic Programming - SQP

Example

Consider the feasible set: $\mathbb{X} = \left\{ x \in \mathbb{R} \;\middle|\; \begin{array}{ll} 2x_1 + 4x_2 \leq 28 & x_1 + x_2 \leq 10 \\ x_1 \leq 10 \quad x_2 \leq 6 & x_1 \geq 0 \quad x_2 \geq 0 \end{array} \right\}$

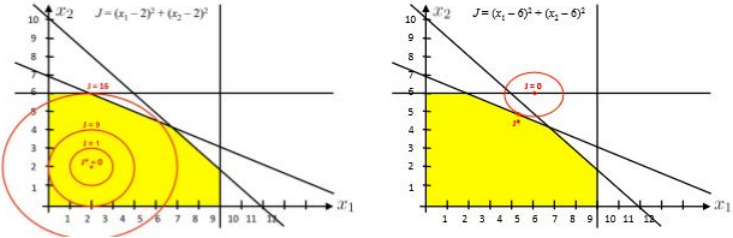

Figure 2: An interior optimum [LEFT] and boundary optimum [RIGHT] for a QP solved graphically.

Figure 1: lecture notes - Professor Scott Moura - University of California, Berkeley

- There is a profound interest in QP problems and their real-time computing.

Quadratic Programming

- There is a profound interest in QP problems and their real-time computing.
    - Practical engineering applications frequently lead to large-scale QP problems.

- There is a profound interest in QP problems and their real-time computing.
    - Practical engineering applications frequently lead to large-scale QP problems.
    - The solution of nonlinear optimization problems can be attained through methods of quadratic optimization and techniques of numerical linear algebra.

Quadratic Programming

- There is a profound interest in QP problems and their real-time computing.
    - Practical engineering applications frequently lead to large-scale QP problems.
    - The solution of nonlinear optimization problems can be attained through methods of quadratic optimization and techniques of numerical linear algebra.
    - Powerful and reliable techniques/codes are available to solve convex QPs

- There is a profound interest in QP problems and their real-time computing.
  - Practical engineering applications frequently lead to large-scale QP problems.
  - The solution of nonlinear optimization problems can be attained through methods of quadratic optimization and techniques of numerical linear algebra.
  - Powerful and reliable techniques/codes are available to solve convex QPs
- Numerical methods depend on
  - the properties of the matrix Q
  - The presence of inequalities
  - The structure of the matrices (sparsity, block structure)
  - if the are only bound constraints $x_{min} \leq x \leq x_{max}$

Quadratic Programming

- A positive definite quadratic programming (QP) with linear constraints can be solved (or shown to be unfeasible) in polynomial time (similar to linear programming)

Quadratic Programming

- A positive definite quadratic programming (QP) with linear constraints can be solved (or shown to be unfeasible) in polynomial time (similar to linear programming)
- If the problem is subject only to equality constraints, A is full rank, and Q is positive definite on $\{h : Ah = 0\}$ then the unique solution is obtained via solution of a linear system of equations.

Quadratic Programming

- A positive definite quadratic programming (QP) with linear constraints can be solved (or shown to be unfeasible) in polynomial time (similar to linear programming)
- If the problem is subject only to equality constraints, A is full rank, and Q is positive definite on $\{h : Ah = 0\}$ then the unique solution is obtained via solution of a linear system of equations.
- QPs are [strictly] convex programs provided that the matrix Q in the objective function is positive semi-definite [positive definite]

Quadratic Programming

- A positive definite quadratic programming (QP) with linear constraints can be solved (or shown to be unfeasible) in polynomial time (similar to linear programming)
- If the problem is subject only to equality constraints, A is full rank, and Q is positive definite on $\{h : Ah = 0\}$ then the unique solution is obtained via solution of a linear system of equations.
- QPs are [strictly] convex programs provided that the matrix Q in the objective function is positive semi-definite [positive definite]
- Nonconvex QPs, in which Q is an indefinite matrix, can be more challenging because they can have several stationary points and local minima

Quadratic Programming

- A positive definite quadratic programming (QP) with linear constraints can be solved (or shown to be unfeasible) in polynomial time (similar to linear programming)
- If the problem is subject only to equality constraints, A is full rank, and Q is positive definite on $\{h : Ah = 0\}$ then the unique solution is obtained via solution of a linear system of equations.
- QPs are [strictly] convex programs provided that the matrix Q in the objective function is positive semi-definite [positive definite]
- Nonconvex QPs, in which Q is an indefinite matrix, can be more challenging because they can have several stationary points and local minima
- These problems are seldon solved analytically

According to matrix Q, different possibilities

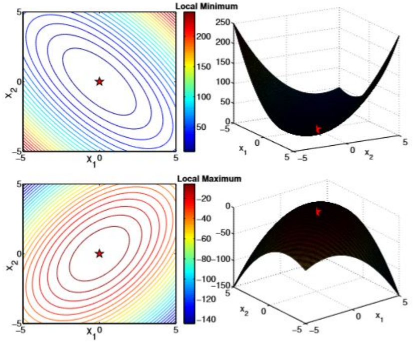

Figure 3: Visualizations of stationary points of different nature. In each case, the objective functions take the form $f(x_1, x_2) = x^T Q x$ and have stationary points at the origin $(x_1^\dagger, x_2^\dagger) = (0, 0)$, denoted by the red star. The Hessian for each case are: Local Minimum $Q = [3\ 2;\ 2\ 3]$, Local Maximum $Q = [-2\ 1;\ 1\ -2]$.

Figure 2: lecture notes from Professor Scott Moura - University of California, Berkeley

According to matrix Q, different possibilities

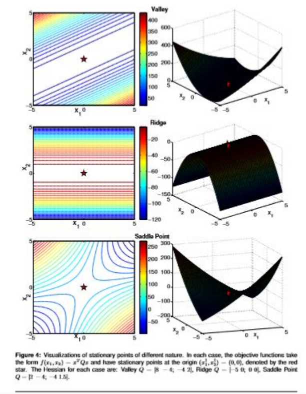

Figure 3: lecture notes from Professor Scott Moura - University of California, Berkeley

Equality-Constrained quadratic programs

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}x^t Qx + c^t x$$

$$s.t \quad Ax = b$$

Exercice Consider

$$\min f(x) = 2x_1^2 - 4x_1 + 2x_2^2 - 6x_2$$

$$x_1 + x_2 = 3$$
$$-2x_1 + x_2 = 2$$

- Write matrices Q, c, A, and b
- Determine the first order necessary conditions for x^* to be a solution of the problem

Quadratic Programming

Example

Consider

$$\min f(x) = (x_1 - 2)^2 + 2(x_2 - 1)^2 - 6$$
$$x_1 + 4x_2 = 3$$

Example

Consider

$$\min f(x) = (x_1 - 2)^2 + 2(x_2 - 1)^2 - 6$$
$$x_1 + 4x_2 = 3$$

$$f(x) = x_1^2 + 2x_2^2 - 4x_1 - 4x_2$$

Example

Consider

$$\min f(x) = (x_1 - 2)^2 + 2(x_2 - 1)^2 - 6$$
$$x_1 + 4x_2 = 3$$

$$f(x) = x_1^2 + 2x_2^2 - 4x_1 - 4x_2$$

In this case

$$Q = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} \quad c = \begin{bmatrix} -4 \\ -4 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 4 \end{bmatrix} \qquad b = [3]$$

Let's write KKT conditions for this problem

Lagrangian

$$\mathcal{L}(x,\lambda) = \frac{1}{2}x^t Qx + c^t x + \lambda^t(Ax - b)$$

$$\mathcal{L}(x,\lambda) = f(x) + \lambda^t(Ax - b)$$

Let's write KKT conditions for this problem

Lagrangian

$$\mathcal{L}\,(x,\lambda) = \frac{1}{2}x^tQx + c^tx + \lambda^t(Ax - b)$$
$$\mathcal{L}(x,\lambda) = f(x) + \lambda^t(Ax - b)$$

$$\mathcal{L}(x,\lambda) = x_1^2 + 2x_2^2 - 4x_1 - 4x_2 + \lambda(x_1 + 4x_2 - 3)$$

Quadratic Programming

Let's write KKT conditions for this problem

Lagrangian

$$\mathcal{L}(x, \lambda) = \frac{1}{2} x^t Q x + c^t x + \lambda^t (Ax - b)$$

$$\mathcal{L}(x, \lambda) = f(x) + \lambda^t (Ax - b)$$

$$\mathcal{L}(x, \lambda) = x_1^2 + 2x_2^2 - 4x_1 - 4x_2 + \lambda(x_1 + 4x_2 - 3)$$

KKT conditions

$$\begin{cases} \nabla_x \mathcal{L}(x^*, \lambda^*) & = & 0 \\ Ax^* & = b & \end{cases}$$

$\boxed{\text{This can be written as a linear system}}$

For the example, the system is given as

$$\left[\begin{array}{cc|c} 2 & 0 & 1 \\ 0 & 4 & 4 \\ \hline 1 & 4 & 0 \end{array}\right] \left[\begin{array}{c} x_1^* \\ x_2^* \\ \lambda^* \end{array}\right] = \left[\begin{array}{c} 4 \\ 4 \\ \hline 3 \end{array}\right]$$

This is a first order necessary condition for a solution.

If Q is positive semi-definite, the problem is convex and the solution of the system gives an optimal solution for the quadratic problem.

Equality-Constrained quadratic programs

There are two cases:

a) Q is symmetric and positive semi-definite $\Rightarrow$ QP is convex $\Rightarrow$ the solution of the KKT-conditions is a solution for the QP.

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

Equality-Constrained quadratic programs

There are two cases:

a) Q is symmetric and positive semi-definite $\Rightarrow$ QP is convex $\Rightarrow$ the solution of the KKT-conditions is a solution for the QP.

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

Lagrange function: $\mathcal{L}(x, \lambda) = \frac{1}{2}x^t Qx + c^t x + \lambda^t(Ax - b)$

KKT conditions

$$\begin{cases} \nabla_x \mathcal{L}(x^*, \lambda^*) & = \quad 0 \\ Ax^* & = b \end{cases} \Rightarrow$$

$$\begin{bmatrix} Q & A^t \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix}$$

Equality-Constrained quadratic programs

There are two cases:

a) Q is symmetric and positive semi-definite $\Rightarrow$ QP is convex $\Rightarrow$ the solution of the KKT-conditions is a solution for the QP.

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

Lagrange function: $\mathcal{L}(x, \lambda) = \frac{1}{2}x^t Qx + c^t x + \lambda^t (Ax - b)$

KKT conditions

$$\begin{cases} \nabla_x \mathcal{L}(x^*, \lambda^*) & = 0 \\ Ax^* & = b \end{cases} \Rightarrow$$

$$\begin{bmatrix} Q & A^t \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix}$$

- The optimization problem is reduced to the solution of a (possibly large-scale) system of linear equations .
- The matrix is symmetric, but it may or may not be positive definite

Example

Solve the following QP problem:

$$minf(x_1; x_2) = x_1^2 + x_2^2$$

subject to

$$x_1 + x_2 = 5$$

- Write the problem in the matrix form
- The problem is convex?
- Write the necessary conditions (KKT)
- Solve the resulting system. The solution is optimal?

Plotting the problem

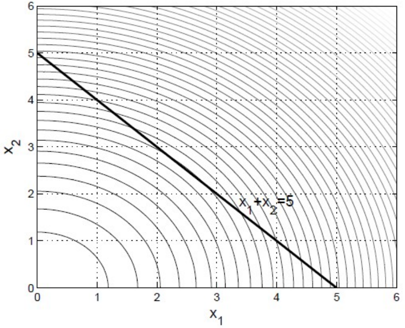

Figure 1.1: The contour plot of $f(x_1, x_2)$ and the constraint

Figure 4: Level curves and constraint

Plotting the problem

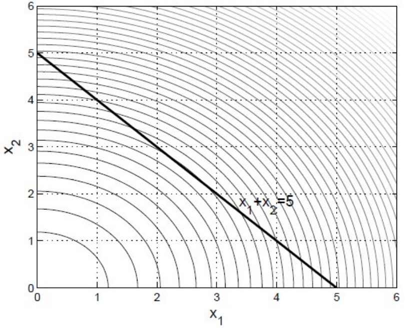

Figure 1.1: The contour plot of $f(x_1, x_2)$ and the constraint

Figure 4: Level curves and constraint

Writing the problem in the matrix form:

$$f(x_1, x_2) = f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \mathbf{x} + [0 \;\; 0]^T \mathbf{x}$$

$$[1 \;\; 1] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 5$$

Quadratic Programming

Writing the problem in the matrix form:

$$f(x_1, x_2) = f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \mathbf{x} + [0 \;\; 0]^T \mathbf{x}$$

$$[1 \;\; 1] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 5$$

Necessary conditions

The necessary condition for optimum is obtained from the linear system given by:

$$\begin{bmatrix} 2 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 5 \end{bmatrix}$$

8

Quadratic Programming

The necessary condition for optimum is obtained from the linear system given by:

$$\begin{bmatrix} 2 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 5 \end{bmatrix}$$

8

The solution of the system (KKT conditions) is: $x_1 = 2,5$; $x_2 = 2:5$; $\lambda = -5$

Verify that $f(x_1, x_2)$ is convex and conclude about optimality

Entrega aula

Dê um exemplo ($x \in \mathbb{R}^3$) de programação quadrática com restrições de igualdade e repita o exemplo anterior.

Quadratic programs with inequalities

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}x^t Q x + c^t x$$

$$A_i x \leq a_i, \quad i \in \{1, 2, \ldots m\}$$
$$A_j x = b_j \quad j \in \{m+1, \ldots m+p\}$$

Q is a symmetric $n \times n$ positive semi-definite matrix

How to solve QP problems?

- The majority of methods for solving quadratic programs can be categorized into either active-set methods or interior methods.
- Briefly, active-set methods are iterative methods that solve a sequence of equality-constrained quadratic subproblems.
- The goal of the method is to predict the active set, the set of constraints that are satisfied with equality, at the solution of the problem.
- The conventional active-set method is divided into two phases; the first focuses on feasibility, while the second focuses on optimality.

Wong, E. (2011). Active-set methods for quadratic programming. UC San Diego

Active Set Methods

The basic idea is as follows:

- Given a point $x_k \in S$, we define a subproblem of minimization with equality constraints determined by the active constraints at x_k. (Keep in mind the definition of active constraints)

Active Set Methods

The basic idea is as follows:

- Given a point $x_k \in S$, we define a subproblem of minimization with equality constraints determined by the active constraints at x_k. (Keep in mind the definition of active constraints)
- If x_k is not optimal for this subproblem, we continue trying to solve the subproblem by choosing a feasible descent direction and performing a linear search Taking this step may involve adding one or more constraints. If this happens, the subproblem changes, and we continue working with a new subproblem.

Active Set Methods

The basic idea is as follows:

- Given a point $x_k \in S$, we define a subproblem of minimization with equality constraints determined by the active constraints at x_k. (Keep in mind the definition of active constraints)
- If x_k is not optimal for this subproblem, we continue trying to solve the subproblem by choosing a feasible descent direction and performing a linear search Taking this step may involve adding one or more constraints. If this happens, the subproblem changes, and we continue working with a new subproblem.
- If x_k is the optimum of the subproblem, we test if x_k is the optimal solution of the problem. If it is not, we choose a new feasible descent direction and perform a linear search to determine x_{k+1}. Also, with this shift, we switch to a new subproblem, and the described process repeats.

Active Set Method

In general an active-set algorithm has the following structure: (from wikipedia!!!)

- Find a feasible starting point
- repeat until "optimal enough"
    - solve the equality problem defined by the active set (approximately)
    - compute the Lagrange multipliers of the active set
    - remove a subset of the constraints with negative Lagrange multipliers
    - search for infeasible constraints

Active Set Method

The approach is similar to unconstrained programming

- 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.

Active Set Method

The approach is similar to unconstrained programming

- 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. This direction is obtained by solving a QP equality constrained subproblem.

Active Set Method

The approach is similar to unconstrained programming

- 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. This direction is obtained by solving a QP equality constrained subproblem.
  - The subproblem solved to obtain d_k will have a subset of constraints imposed as equalities and referred as the working set, $\mathcal{W}_k$, consisting of all m equality constraints and some of the active inequalities.
  - New iterates are calculated and the working set is modified until the optimality conditions are satisfied, or all Lagrange multipliers are positive (the KKT conditions are satisfied).

For a given x^*, the active set is
$\mathcal{A}(x^*) = \{i \in \{1, 2, ...m\} | A_i x^* = a_i\} \cup \{i \in \{m+1, ...m+p\} | A_i x^* = b_i\}$

This will be initial the working set $\mathcal{W}_k$

Idea of the algorithm

- For the current (feasible) point, x_k, check whether x_k minimizes the quadratic objective function in the subspace defined by the working set, i.e. the Lagrange multipliers corresponding to the inequality constraints are non negative. (that is, verify the KKT conditions).
- If the optimality conditions are not satisfied, we compute a direction, d_k, to move to the next point $x_{k+1} = x_k + d_k$

Direction finding problem at iteration k:

$$\min_{x \in \mathbb{R}^n} \frac{1}{2}(x_k + d_k)^t Q(x_k + d_k) + c^t(x_k + d_k)$$

$$A_i(x_k + d_k) = a_i, \quad i \in \mathcal{A}(x_k)$$

Direction finding problem at iteration k:

This problem can be formulated as

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} d_k^t Q d_k + g^t d_k$$

$$A_i d_k = 0, \quad i \in \mathcal{A}(x_k)$$

with $g_k = Qx_k + c$

$$\boxed{\text{An equality constrained QP !!!!}}$$

The solution d_k of this quadratic problem is obtained solving the following system:

$$\begin{bmatrix} Q & \tilde{A}^t \\ \tilde{A} & 0 \end{bmatrix} \begin{bmatrix} d \\ \lambda \end{bmatrix} = \begin{bmatrix} -g_k \\ 0 \end{bmatrix}$$

with $\tilde{A}$ being the matrix having the rows a_i^t , for all indices i in the working set

- If $d_k \neq 0$ we shall move in this direction.
- If $d_k = 0$ we have to check for optimality, i.e. all the Lagrange multipliers for the inequality constraints must be non negative.

Case 1 $d_k = 0$

if the lagrange multipliers $\lambda_j \geq 0$ then Stop and return the current point x_k

if there exists $\lambda_j < 0$, find the most negative λ_j and remove constraint j from the working set $\mathcal{W}_k$. Keep the same point for the next step: $x_{k+1} = x_k$

Case 2 $d_k \neq 0$

In this case we shall decide about the step lenght α_k

The constraints in $\mathcal{W}_k$ are satisfied for any α_k. The step lenght will be determined such that $x_k + \alpha_k d_k$ satisfies the constraints $j \notin \mathcal{W}_k$.

The step length is given as

$$\alpha_k = \min_{i \notin W_k,\ \mathbf{a}_i^T \mathbf{d}_k > 0} \left(1, \frac{b_i - \mathbf{a}_i^T \mathbf{x}_k}{\mathbf{a}_i^T \mathbf{d}_k}\right)$$

(Try to believe...)

- If $\alpha_k < 1$, the movement along d_k was blocked by some constraint that does not belong to $\mathcal{W}_k$. This is a blocking constraint and will be added to the new working set.
- If $\alpha_k = 1$ no new constraints are active for $x_k + \alpha_k d_k$ and there are no blocking constraints at this stage.

Example

Example

$$min f(x_1; x_2) = x_1^2 + x_2^2 - 4x_1 - 4x_2$$

subject to

$$x_1 + x_2 \leq 2$$

$$x_1 - 2x_2 \leq 2$$

$$-x_1 - x_2 \leq 1$$

$$-2x_1 + x_2 \leq 2$$

Algorithm

Algorithm 1 Active set method

Define the quadratic objective function: matrix $\mathbf{Q}$ and vector $\mathbf{c}$
Define the constraints: vectors $\mathbf{a}_i^T$ and scalars b_i, $i = \overline{1,p}$
Select an initial feasible point $\mathbf{x}_0$
Find the initial working set W_0
Compute the gradient of the objective function at the current point: $\mathbf{g}_0 = \mathbf{Q}\mathbf{x}_0 + \mathbf{c}$
Compute the matrix $\mathbf{A}$ having the rows $\mathbf{a}_i$, $i \in W_k$
Solve the linear system

$$\begin{bmatrix} \mathbf{Q} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{d}_0 \\ \lambda \end{bmatrix} = \begin{bmatrix} -\mathbf{g}_0 \\ \mathbf{0} \end{bmatrix} \tag{1.65}$$

Set $k = 0$

Algorithm

Set $k = 0$

while not all $\lambda_i \geq 0,\ i \in W_k \cap \{1, 2, ..., m\}$ or $\mathbf{d}_k \neq 0$ **do**

  if $\mathbf{d}_k = 0$ **then**

    Check optimality:

    if $\lambda_i \geq 0, i \in W_k \cap \{1, 2, ..., m\}$ **then**

      Stop and return the current point $\mathbf{x}_k$

    else

      Find the most negative λ_j

      Remove constraint j from the working set W_k

      Keep the same point for the next step: $\mathbf{x}_{k+1} = \mathbf{x}_k$

    end if

  else

    Compute the step length α_k from:

$$\alpha_k = \min_{i \notin W_k,\ \mathbf{a}_i^T \mathbf{d}_k > 0} \left(1, \frac{b_i - \mathbf{a}_i^T \mathbf{x}_k}{\mathbf{a}_i^T \mathbf{d}_k}\right) \tag{1.66}$$

    Compute a new point: $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$

    If $\alpha_k < 1$, find the blocking constraint i and add it to the working set W_k

  end if

  Set $k \leftarrow k + 1$

Quadratic Programming

Algorithm

Set $k \leftarrow k+1$

Compute the gradient of the objective function at the current point:

$$\mathbf{g}_k = \mathbf{Q}\mathbf{x}_k + \mathbf{c} \tag{1.67}$$

Compute the matrix $\mathbf{A}$ having the rows $\mathbf{a}_i$, $i \in W_k$

Solve the linear system

$$\begin{bmatrix} \mathbf{Q} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{d}_k \\ \lambda \end{bmatrix} = \begin{bmatrix} -\mathbf{g}_k \\ \mathbf{0} \end{bmatrix} \tag{1.68}$$

end while

Important issues:

- How to determine a starting point x_0 for the active-set algorithm.
- Active set methods require an efficient strategy for the determination of active sets at each x_k .

Advantages of active set methods

- Since only active constraints are considered at each iteration x_k , the QP with equality (QPe) usually has only a few constraints and can be solved fast. Thus, Large-scale (QPi) '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 QPE can be used to solve the next QPE known as warm starting.
- All iterates xk are feasible to (QP)I . This is an important property, e.g, in Sequential Quadratic Programming.

Advantages of active set methods

- Since only active constraints are considered at each iteration x_k , the QP with equality (QPe) usually has only a few constraints and can be solved fast. Thus, Large-scale (QPi) '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 QPE can be used to solve the next QPE known as warm starting.
- All iterates xk are feasible to (QP)I . This is an important property, e.g, in Sequential Quadratic Programming.

Disadvantages of active set methods

- Since the active-set $\mathcal{A}_k$ may vary from step to step, the structure and properties, e.g. sparsity, of constraint matrices may change.
- ASM may become slower near to the optimal solution;.
- For some problems ASM can be computationally expensive.