27.3 Sequential Quadratic Programming

(1) Review: Quadratic Programming

- Quadratic Program QP

DEF: $\min \; q(x) = \frac{1}{2} x^T G x$ s.t. $\begin{cases} a_i^T x = b_i & i \in \mathcal{E} \\ a_i^T x \geq b_i & i \in \mathcal{I} \end{cases}$

with: G symmetric $n \times n$ Hessian matrix → Hint: p. 20.2:9

c, x, a_i s.t. $i \in \mathcal{E} \cup \mathcal{I}$ are vectors in $\mathbb{R}^n$

DEF: A QP is → convex if the Hessian matrix G is positive semidefinite

↳ strict convex if G is definite positive

↳ non convex if G is an indefinite matrix

↳ A QP can be solved (or shown that they diverge) in a finite number of steps!

Hint: depends on the characteristics of the objective function and on the number of constraints!

- QP with equality constraints

Hint: techniques for this case are valid also for pb. with inequality constraints (with an iterative approach)

27.3.2

DEF: $\min_x q(x) = \frac{1}{2} x^T G x + x^T c$ s.t. $Ax = b$

with $\begin{cases} A \text{ is } m \times n \text{ Jacobian matrix of constraints} \\ A \text{ has rows } a_i^T,\ i \in \mathcal{E} \\ b \in \mathbb{R}^m \text{ is a vector with } b_i,\ i \in \mathcal{E} \end{cases}$

Hint: Jacobian: $J = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \cdots & \frac{\partial f}{\partial x_n} \end{bmatrix}$;

$$= \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

Hint: if A has full rank, constraints are consistent

- <u>First-order necessary condition</u> for x^* to be a solution of a minimization function is that there is λ^* s.t.

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

Hint: by writing optimal solution x^* as a suboptimal solution x plus a variance ε ; $x^* = x + \varepsilon$

$$\begin{bmatrix} G & -A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} -\varepsilon \\ \lambda^* \end{bmatrix} = \begin{bmatrix} g \\ h \end{bmatrix}$$

s.t. $\begin{cases} h = Ax - b \\ g = c + Gx \end{cases}$

K = Karush-Kuhn-Tucker

<u>KKT-matrix</u>

<u>KKT-system</u>

27.3:3

- Solution caracterization from KKT-matrix

LEMMA: suppose A has full row rank, $Z \in \mathbb{R}^{n\times(n-m)}$ has columns basis of null space of A → Hint: Z has full rank and satisfies $AZ = 0$

There is a pair $(x^*, \lambda^*)^T$ satisfying the min. pb. if $Z^T G Z$ is positive definite (if KKT matrix is non singular) → Hint: if the $\det(K) \neq 0$

TH: suppose: A has full rank, reduced Hessian matrix: $Z^T G Z$ is positive definite, if x^* is a solution of the first order necessary condition, then it is the unique global solution of the minimization pb.

COROLLARY: if $Z^T G Z$ is positive semi definite x^* is local minimizer
↳ is negative x^* is a stationary point

- QP with equality and inequality constraints → Hint: Methods to solve convex QP

Review
- → Gradient projection methods.
- → Interior point methods (large optim. pb.)
- → Active set methods (small optim. pb.)

27.3:4

Hint:

↳ Lagrangian L for QP is:

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

$$\mathcal{A}(x^*) = \{ i \in E \cup I \;\; s.t. \;\; a_i^T x^* = b_i \}$$ ← Hint: Active set p. 20.2:3

TH: suppose G positive semidefinite, x^* is:

a global solution of QP,

∃ a Lagrange multiplier vector λ^* with components λ_i^*, $i \in \mathcal{A}(x^*)$ s.t.

stationarity — i. $Gx^* + c - \sum_{i \in \mathcal{A}(x^*)} \lambda_i^* a_i = 0$

primal feasibility — ii. $a_i^T x^* = b_i \quad \forall i \in \mathcal{A}(x^*)$

iii. $a_i^T x^* \geq b_i \quad \forall i \in I \setminus \mathcal{A}(x^*)$

dual feasibility — iv. $\lambda_i^* \geq 0 \quad \forall i \in I \cap \mathcal{A}(x^*)$

• Active set methods

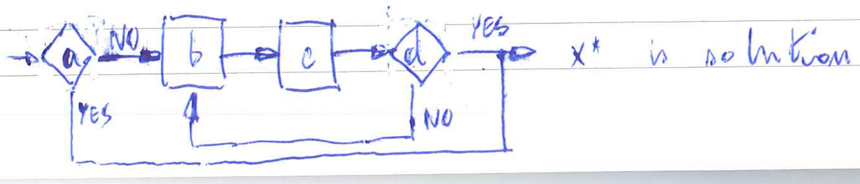
↳ simplex method (linear programming)

a. guess of x^* (optimal?)

b. drop the index from estimate of $\mathcal{A}(x^*)$
(with data from gradient and Lagrange multiplier)

c. add new index

d. check if new x^* is optimal?



↳ Primal active set method (QP)

Hint: In each iterate solve a QP sub-problem in which some inequality $\mathcal{I}$ and all equality $\mathcal{E}$ constraints are imposed as equalities (Working set W_k)

---

② Sequential Quadratic Programming SQP

Hint: widely used active set methods, very useful with many non linear constraints

↳ SQP Methods

- Inequality QP (IQP) Methods
  - ↳ At each iteration:
    - ↳ generate optimal set $\mathcal{A}(x^*)$
    - ↳ compute a step

    } together
- Equality QP (EQP) Methods
  - ↳ At each iteration
    - ↳ generate optimal set $\mathcal{A}(x^*)$
    - ↳ solve an EQP to find step

    } separate

• Local SQP Method

Let's consider equality-constrained problem

$\min f(x)$ s.t. $c(x) = 0$ with $c \in \mathbb{R}^m$ a vector of constraints

↳ then { 1. Model the opt. problem in a QP sub-problem x_k
2. To define a new iterate x_{k+1}

Hint: "hard" task!

27.3 : 6

Hint: says how to define new iterate / new future step

↳ <u>Standard Newton iteration method for KKT opt.</u>

$A(x)^T = [\nabla c_1(x), \nabla c_2(x), \dots, \nabla c_m(x)]$ Hint: Jacobian matrix of constraints

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

$$F(x, \lambda) = \begin{bmatrix} \nabla f(x) - A(x)^T \lambda \\ c(x) \end{bmatrix} = 0$$

Hint: the first order KKT condition for the optimization problem

Any solution (x^*, λ^*) for which $A(x^*)$ has full rank satisfies $F(x^*, \lambda^*) = 0$;

$$F'(x, \lambda) = \begin{bmatrix} \nabla^2_{xx} \mathcal{L}(x, \lambda) & -A(x)^T \\ A(x) & 0 \end{bmatrix}$$

Hint: Jacobian of $F(x, \lambda)$ with respect to x, λ

- We can write <u>the future step</u> as an improvement of the previous step:

$$\begin{bmatrix} x_{k+1} \\ \lambda_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ \lambda_k \end{bmatrix} + \begin{bmatrix} \varepsilon_x \\ \varepsilon_\lambda \end{bmatrix}$$

Hint: see page 27.3:2

⇓ <u>KKT system</u>

<u>KKT matrix</u>

$$F'_k \, \varepsilon = -F_k \implies \begin{bmatrix} \nabla^2_{xx} \mathcal{L}_k & -A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} \varepsilon_x \\ \varepsilon_\lambda \end{bmatrix} = \begin{bmatrix} -\nabla f_k + A_k^T \lambda_k \\ -c_k \end{bmatrix}$$

The iteration is well defined when KKT matrix is non singular

Hint: when 1. $A(x)$ has full rank (or $\det(A(x)) \neq 0$)

2. $d^T \mathcal{L}(x, \lambda) d$ positive definite $\forall d \neq 0$ s.t. $A(x) d = 0$

27.3:7

→ SQP iteration method for KKT opt.

At each iterate (x_k, λ_k) we can model opt. pb. with an optimization sub-problem where we minimize the ε

$\nabla f_k^T \varepsilon = \nabla_x L_k^T \varepsilon \Rightarrow$ is a quadratic approximation of Lagrangian

$$\min_{\varepsilon} \; f_k + \nabla f_k^T \varepsilon + \frac{1}{2} \varepsilon^T \nabla^2_{xx} L_k \varepsilon \quad \text{s.t.} \quad A_k \cdot \varepsilon + c_k = 0$$

We can write the future step as an improvement of the previous (in a different way):

$$\begin{bmatrix} \varepsilon_x \\ \varepsilon_\lambda \end{bmatrix} = \begin{bmatrix} x_{k+1} - x_k \\ \lambda_{k+1} - \lambda_k \end{bmatrix} \quad \text{KKT system} \Rightarrow \quad F'_k \cdot \varepsilon = -F_k$$

$$\begin{bmatrix} \nabla^2_{xx} L_k & -A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} x_{k+1} - x_k \\ \lambda_{k+1} - \lambda_k \end{bmatrix} + \begin{bmatrix} -A_k^T \lambda_k \\ 0 \end{bmatrix} = \begin{bmatrix} -\nabla f_k + \cancel{A_k^T \lambda_k} \\ -c_k \end{bmatrix} + \begin{bmatrix} \cancel{-A_k^T \lambda_k} \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} \nabla^2_{xx} L_k (x_{k+1} - x_k) - A_k^T (\lambda_{k+1} - \lambda_k) \\ A_k (x_{k+1} - x_k) \end{bmatrix} + \begin{bmatrix} -A_k^T \lambda_k \\ 0 \end{bmatrix} = \begin{bmatrix} -\nabla f_k \\ -c_k \end{bmatrix}$$

$$\begin{bmatrix} \nabla^2_{xx} L_k (x_{k+1} - x_k) - A_k^T \lambda_{k+1} \\ A_k (x_{k+1} - x_k) \end{bmatrix} = \begin{bmatrix} -\nabla f_k \\ -c_k \end{bmatrix}$$

$$\begin{bmatrix} \nabla^2_{xx} L_k & -A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} x_{k+1} - x_k \\ \lambda_{k+1} \end{bmatrix} = \begin{bmatrix} -\nabla f_k \\ -c_k \end{bmatrix}$$

3) Local SQP Algorithm

a. Replace non linear pb. by pb. of minimizing Lagrangian

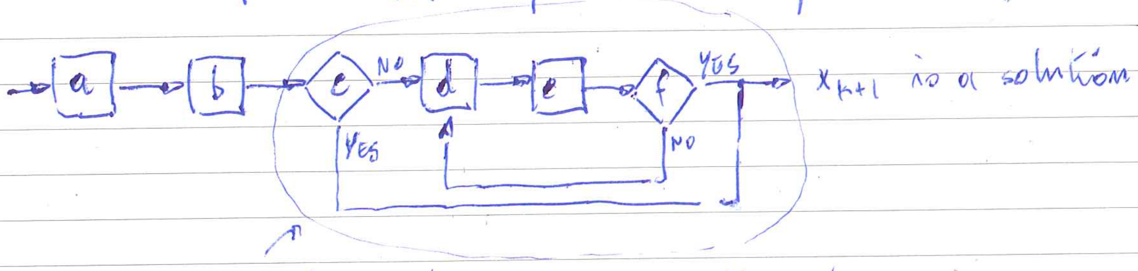
b. Make { 1. Quadratic approx of Lagrangian
2. Linear approx to the constraints

c. Guess initial pair (x_k, λ_k) s.t. $k=0$ (optimal?)

d. Get $f_k, \nabla f_k, \nabla^2_{xx} L_k, c_k, A_k$; to solve (b)

e. Set $x_{k+1} = x_k + \varepsilon_k$, λ_{k+1}; where $\varepsilon_k, \lambda_{k+1}$ comes from (d)

f. check if new x_{k+1} is optimal (convergence?)?



Hint: this is the same as Active set methods at p. 27.3:4 (simplex method)!

In fact, SQP solves a QP problem at each iterate!

---

④ Line Search SQP Algorithm

- Opt. pb. with equality inequality constraints to solve:

$$\min_{\varepsilon} \; f_k + \nabla f_k^T \varepsilon + \frac{1}{2} \varepsilon^T \nabla^2_{xx} L_k \varepsilon \quad \text{s.t.} \begin{cases} \nabla c_i(x_k)^T \varepsilon + c_i(x_k) = 0 & i \in \mathcal{E} \\ \nabla c_i(x_k)^T \varepsilon + c_i(x_k) \geq 0 & i \in \mathcal{I} \end{cases}$$

27.3.19

a. Define parameters $\omega \in (0, 0.5)$, $\tau \in (0, 1)$;

b. Make
- 1. Quadratic approx of Lagrangian
- 2. Linear approx of the constraints

c.
- Guess initial pair (x_k, λ_k) s.t. $k = 0$ (optimal ?)
- Define parameters $\omega \in (0, 0.5)$, $\tau \in (0, 1)$

d. Get f_k, ∇f_k, $\nabla^2_{xx} L_k$, c_k, A_k ; to solve (b)

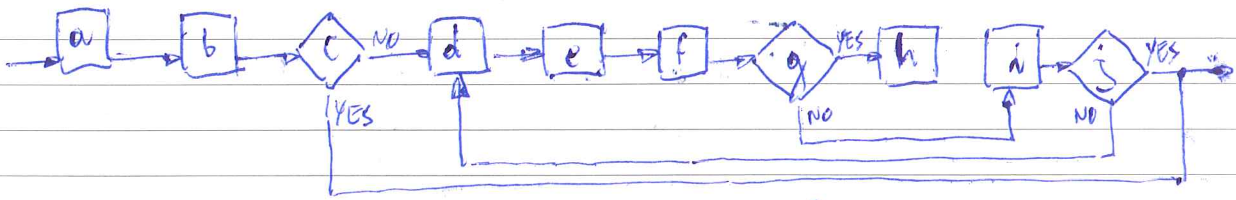
e. Find corresponding multiplier $\hat{\lambda}$ and set $\varepsilon_k \leftarrow \hat{\lambda} - \lambda_k$

f.
- Find $\mu_k = (\geq) \dfrac{\nabla f_k^T \varepsilon_k + \frac{1}{2} \varepsilon_k^T \nabla^2_{xx} L_k \varepsilon_k}{(1-\rho) \|c_k\|}$ s.t. $\rho \in (0, 1)$
- Set $\alpha_k = 1$

h. Set $\alpha_k \leftarrow \tau_\alpha \alpha_k$ s.t. $\tau_\alpha \in (0, \tau]$

i. Set $x_{k+1} = x_k + \alpha_k \varepsilon$, $\lambda_{k+1} \leftarrow \lambda_k + \alpha_k \varepsilon_\lambda$

j. Check if new x_{k+1} is optimal (convergence ?) ?



int: directional derivative of ϕ: $D(\phi(x_k, \mu_k), \varepsilon_k) = \nabla f_k^T \varepsilon_k - \mu \|c_k\|$

g. $\phi(x_k + \alpha_k \varepsilon_k, \mu_k) > \phi(x_k, \mu_k) + \omega \alpha_k D(\phi(x_k, \mu_k), \varepsilon_k)$ (?)

Hint: non smooth merit function

$$\phi(x_k + \alpha_k \varepsilon_k, \mu_k) = f(x_k + \alpha_k \varepsilon_k) + \mu_k \|c(x_k + \alpha_k \varepsilon_k)\|$$