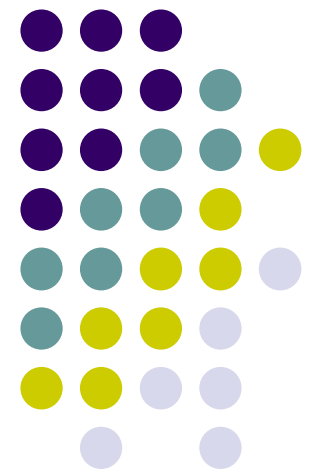


# CS5321

# Numerical Optimization

---

## 18 Sequential Quadratic Programming (Active Set methods)





# Local SQP model

- The problem  $\min_x f(x)$  subject to  $c(x)=0$  can be modeled as a quadratic programming at  $x=x_k$

$$\begin{aligned} \min_p \quad & m_k(p) = f_k + \nabla f_k^T p + \frac{1}{2} p^T \nabla_{xx}^2 L_k p \\ \text{s.t.} \quad & A_k p + c_k = 0 \end{aligned}$$

- Assumptions:
  - $A(x)$ , the constraint Jacobian, has **full row rank**.
  - $\nabla_{xx}^2 L(x, \lambda)$  is positive definite on the tangent space of constraints, that is,  $d^T \nabla_{xx}^2 L(x, \lambda) d > 0$  for all  $d \neq 0, Ad=0$ .



# Inequality constraints

- For  $\min_x f(x)$   
s.t.  $c_i(x) = 0, i \in \mathbf{E}$   
 $c_i(x) \geq 0, i \in \mathbf{I}$

- The local quadratic model is

$$\begin{aligned} \min_p \quad & m_k(p) = f_k + \nabla f_k^T p + \frac{1}{2} p^T \nabla_{xx}^2 L_k p \\ \text{s.t.} \quad & \nabla c_i(x_k)^T p + c_i(x_k) = 0, i \in \mathbf{E} \\ & \nabla c_i(x_k)^T p + c_i(x_k) \geq 0, i \in \mathbf{I} \end{aligned}$$

# Theorem of active set methods



- Theorem 18.1 (Robinson 1974)
  - If  $x^*$  is a local solution of the original problem with some  $\lambda^*$ , and the pair  $(x^*, \lambda^*)$  satisfies the KKT condition, the LICO condition, and the second order sufficient conditions, then for  $(x_k, \lambda_k)$  sufficiently close to  $(x^*, \lambda^*)$ , then there is a local quadratic model whose active set is the same as that of the original problem.



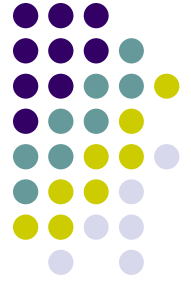
# Sequential QP method

1. Choose an initial guess  $x_0, \lambda_0$
  2. For  $k = 1, 2, 3, \dots$ 
    - a) Evaluate  $f_k, \nabla f_k, \nabla_{xx}^2 L_k, c_k$ , and  $\nabla c_k (=A_k)$
    - b) Solve the local quadratic programming
    - c) Set  $x_{k+1} = x_k + p_k$
- How to choose the active set?
  - How to solve 2(b)?
    - Haven't we solved that in chap 16? (Yes and No)



# Algorithms

- Two types of algorithms to choose active set
  - Inequality constrained QP (IQP): Solve QPs with inequality constraints and take the local active set as the optimal one.
  - Equality constrained QP (EQP): Select constraints as the active set and solve equality constrained QPs.
- Basic algorithms to solve 2(b)
  1. Line search methods
  2. Trust region methods
  3. Nonlinear gradient projection



# Solving SQP

- All techniques in chap 16 can be applied.
- But there are additional problems need be solved
  - Linearized constraints may not be consistent
  - Convergence guarantee (Hessian is not positive def)
- And some useful properties can be used
  - Hessian can be updated by the quasi-Newton method
  - Solutions can be used as an initial guess (warm start)
  - The exact solution is not required.



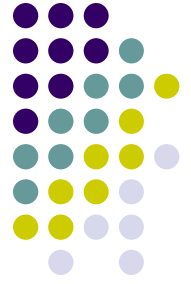
# Inconsistent linearizations

- Linearizing  $1 - x \geq 0$  at  $x_k=1$  gives  $-p \geq 0$   
 $x^2 - 4 \geq 0$   $2p - 3 \geq 0$
- The constraints cannot be enforced since they may not be exact or consistent. Use penalty function

$$\min_{x, v, w, t} f(x) + \mu \sum_{i \in \mathbf{E}} (v_i + w_i) + \mu \sum_{i \in \mathbf{I}} t_i$$

$$\begin{aligned} \text{subject to } c_i(x) &= v_i + w_i & i \in \mathbf{E} \\ c_i(x) &\geq -t_i & i \in \mathbf{I} \\ v, w, t &\geq 0 \end{aligned}$$





# Quasi-Newton approximations

$$s_k = x_{k+1} - x_k$$

- Recall for  $y_k = \nabla_x L(x_{k+1}, \lambda_{k+1}) - \nabla_x L(x_k, \lambda_k)$  the update of Hessian is (BFGS, chap 6)

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}$$

- If the updated Hessian  $B_{k+1}$  is not positive definite
  - Condition  $s_k^T B_{k+1} s_k = s_k^T y_k > 0$  fails.
  - Define  $r_k = \theta_k y_k + (1 - \theta_k) B_k s_k$  for  $\theta \in (0, 1)$

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{r_k r_k^T}{r_k^T s_k}$$



# BFGS for reduced-Hessian

- Let  $A^T = \begin{pmatrix} Q_1 & Q_2 \end{pmatrix} \begin{pmatrix} R \\ 0 \end{pmatrix}$ ,  $p = Q_1 p_1 + Q_2 p_2$   
 $R p_1 = -h$

$$\begin{aligned} \text{Need to solve } (Q_2^T G Q_2) p_2 &= -Q_2^T G Q_1 p_1 - Q_2^T g \\ R^T \lambda^* &= Q_1^T (g + G p) \end{aligned}$$

1. Solve  $\lambda^*$  first to obtain an active set.
2. Ignore  $Q_2^T G Q_1 p_1$  term. Solve  $(Q_2^T G Q_2) p_2 = -Q_2^T g$

- The reduced secant formula is

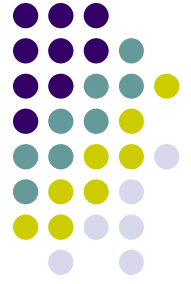
$$(Q_2^T G Q_2)_{k+1} (\alpha_k p_k) = Q_2^T [\nabla_x L(x_{k+1}, \lambda_{k+1}) - \nabla_x L(x_k, \lambda_{k+1})]$$

- Use BFGS on this equation.



# 1. Line search SQP method

- Set step length  $\alpha_k$  such that the *merit function*  $\phi_1(x, \mu) = f(x) + \mu \|c(x)\|_1$  is sufficiently decreased?  
$$\phi_1(x_k + \alpha_k p_k, \mu_k) \leq \phi_1(x_k, \mu_k) + \alpha_k D(\phi_1(x_k, \mu), p_k)$$
  - One of the Wolfe condition (chap 3)
  - $D(\phi_1, p_k)$  is the directional derivative of  $\phi_1$  in  $p_k$ .  
$$D(\phi_1(x_k, \mu), p_k) = \nabla f_k^T p_k - \mu \|c_k\|_1$$
 (theorem 18.2)
  - Let  $\alpha_k = 1$  and decrease it until the condition is satisfied.



## 2.Trust region SQP method

- Problem modification

$$\begin{aligned} \min_p \quad & m_k(p) = f_k + \nabla f_k^T p + \frac{1}{2} p^T \nabla_{xx}^2 L_k p \\ \text{s.t.} \quad & A_k p + c_k = r_k, \quad \|p\|_2 \leq \Delta_k \end{aligned}$$

1. Relax the constraints ( $r_k=0$  in the original algorithm)
2. Add trust region radius as a constraint

- There are smart ways to choose  $r_k$ . For example,

$$\begin{aligned} \min_v \quad & r_k(v) = \|A_k v + c_k\|_2 \\ \text{s.t.} \quad & \|v\|_2 \leq 0.8\Delta_k \end{aligned}$$



## 3. Nonlinear gradient projection

- Let  $B_k$  be the s.p.d. approximation to  $\nabla^2 f(x_k)$ .

$$\begin{aligned} \min_x \quad & q_k(x) = f_k + \nabla f_k^T (x - x_k) + \frac{1}{2}(x - x_k)^T B_k (x - x_k) \\ \text{s.t.} \quad & l \leq x \leq u \end{aligned}$$

- Step direction is  $p_k = x - x_k$
- Combine with the line-search dir  $x_{k+1} = x_k + \alpha_k p_k$ .
  - Choose  $\alpha_k$  s.t.  $f(x_{k+1}) \leq f(x_k) + \eta \alpha_k \nabla f_k^T p_k$ .
- Combine with the trust region bounds  $\|p_k\| \leq \Delta_k$ .

$$\begin{aligned} \min_x \quad & q_k(x) = f_k + \nabla f_k^T (x - x_k) + \frac{1}{2}(x - x_k)^T B_k (x - x_k) \\ \text{s.t.} \quad & \max(l, x_k - \Delta_k e) \leq x \leq \min(u, x_k + \Delta_k e) \end{aligned}$$