

Lecture 11 - Sequential Quadratic Programming (SQP)

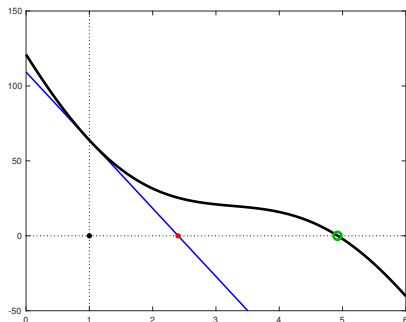
Dinesh Krishnamoorthy

Department of Engineering Cybernetics, NTNU

Spring 2026

Solving a system on nonlinear equations (Root finding)

$$F(w) = 0$$



Newton Root finder

for $k = 0, 1, 2, \dots$

- Compute $F(w_k)$ and $\nabla F(w_k)$
- Compute Newton direction

$$\nabla F(w_k)^T \Delta w = -F(w_k)$$

- Full Newton step

$$w_{k+1} \leftarrow w_k + \Delta w$$

Recap: Newton's method for unconstrained minimization

- Unconstrained problem

$$\min_x f(x)$$

- Necessary optimality condition $\nabla f(x) = 0$
- Solve using Newton method

$$F(w) = \nabla f(x), \quad \nabla F(w) = \nabla^2 f(x)$$

Compute Newton direction

$$s_k = x_{k+1} - x_k = -\nabla^2 f(x_k)^{-1} \nabla f(x_k)$$

- Modified Newton method with line search

$$x_{k+1} = x_k - \alpha \nabla^2 f(x_k)^{-1} \nabla f(x_k)$$

Direct method to solve equality constrained problems

$$\begin{array}{ll}\min_{\mathbf{x}} & f(\mathbf{x}) \\ \text{subject to} & \mathbf{h}(\mathbf{x}) = 0\end{array}$$

with $\mathbf{x} \in \mathbb{R}^n$, and $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^{m_h}$

KKT conditions:

$$F(\mathbf{x}, \lambda) := \begin{bmatrix} \nabla f(\mathbf{x}) + \nabla \mathbf{h}(\mathbf{x})\lambda \\ \mathbf{h}(\mathbf{x}) \end{bmatrix} = 0$$

- ▶ $n + m_h$ equations and $n + m_h$ unknowns $\mathbf{x}$ and λ .
- ▶ Solve the system of nonlinear equations to find $\mathbf{x}^*$ and λ^* that satisfies $F(\mathbf{x}, \lambda) = 0$.

Newton method on the KKT conditions

- Solve $F(x, \lambda) = 0$ using Newton method, where $w = [x, \lambda]^T$

$$F(w) = \begin{bmatrix} \nabla f(x) + \nabla h(x)\lambda \\ h(x) \end{bmatrix}, \quad \nabla F(w) = \begin{bmatrix} \nabla^2 \mathcal{L}(x, \lambda) & \nabla h(x) \\ \nabla h(x)^T & 0 \end{bmatrix}$$

- Compute Newton direction $\Delta w = \begin{bmatrix} s \\ \Delta \lambda \end{bmatrix}$ at x_k and λ_k

$$\begin{bmatrix} s_k \\ \Delta \lambda_k \end{bmatrix} = - \begin{bmatrix} \nabla^2 \mathcal{L}_k & \nabla h_k \\ \nabla h_k^T & 0 \end{bmatrix}^{-1} \begin{bmatrix} \nabla f_k + \nabla h_k \lambda_k \\ h_k \end{bmatrix}$$

- Take full Newton step

$$\begin{bmatrix} x_{k+1} \\ \lambda_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ \lambda_k \end{bmatrix} + \begin{bmatrix} s_k \\ \Delta \lambda_k \end{bmatrix}$$

Newton method on the KKT conditions

Newton method is well defined when the KKT matrix is non-singular.

Question: Under what conditions the KKT matrix is non-singular?

- ▶ $\nabla h(x)$ has full row rank^a
- ▶ Second order sufficient condition (SOSC)
Hessian of the Lagrangian is positive definite $\forall d \neq 0$, such that $h(x)^T d = 0$
$$d^T \nabla^2 \mathcal{L}(x, \lambda) d > 0,$$

^aThis is known as: Linear independence constraint qualification (LICQ)

Newton method on the KKT conditions

The Newton method on the KKT conditions can also be equivalently expressed as

$$\begin{bmatrix} \nabla^2 \mathcal{L}_k & \nabla \mathbf{h}_k \\ \nabla \mathbf{h}_k^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{s}_k \\ \lambda_k + \Delta \lambda \end{bmatrix} = - \begin{bmatrix} \nabla f_k \\ \mathbf{h}_k \end{bmatrix}$$

Exercise: Show that this is equivalent

This insight gives rise to the SQP framework.

Sequential Quadratic Program

The QP subproblem

$$\begin{aligned} \min_s \quad & f_k + \nabla f_k^T s + \frac{1}{2} s^T \nabla^2 \mathcal{L}_k s \\ \text{s.t.} \quad & \nabla h_k^T s + h_k = 0 \end{aligned}$$

has the KKT conditions

$$\begin{aligned} \nabla^2 \mathcal{L}_k s_k + \nabla f_k + \nabla h_k \lambda_{k+1} &= 0 \\ \nabla h_k^T s_k + h_k &= 0 \end{aligned}$$

In matrix form expressed as

$$\begin{bmatrix} \nabla^2 \mathcal{L}_k & \nabla h_k \\ \nabla h_k^T & 0 \end{bmatrix} \begin{bmatrix} s_k \\ \lambda_{k+1} \end{bmatrix} = - \begin{bmatrix} \nabla f_k \\ h_k \end{bmatrix}$$

Sequential quadratic programming

Starting from (x_0, λ_0) , For $k = 0, 1, 2, \dots$

- ▶ Evaluate f_k , ∇f_k , $\nabla^2 \mathcal{L}_k$, h_k , and ∇h_k
- ▶ Solve the following convex QP to obtain s and λ_{k+1}

$$\begin{aligned} s_k = \operatorname{argmin}_s \quad & f_k + \nabla f_k^T s + \frac{1}{2} s^T \nabla^2 \mathcal{L}_k s \\ \text{s.t.} \quad & \nabla h_k^T s + h_k = 0 \end{aligned}$$

- ▶ Set $x_{k+1} = x_k + s_k$.

Solving a sequence of convex QP subproblems is equivalent to solving the Newton iterations.

Lagrange multiplier of the k -th QP subproblem is equal to

$$\lambda_{k+1} = \lambda_k + \Delta \lambda$$

Sequential quadratic programming (SQP)

$$\begin{array}{ll}\min_{x \in \mathbb{R}^n} & f(x) \\ \text{subject to} & g(x) \leq 0 \\ & h(x) = 0\end{array}$$

Two ways of handling inequality constraints:

1. Include linearized inequalities in the QP (referred to as IQP)
 $\Rightarrow$ let QP determine activity of inequality constraints
2. Determine which inequality constraints are treated as equality constraints (referred to as EQP)
 $\Rightarrow$ active set strategy: solve LP problem first at each x_k

Inequality constrained QP (IQP)

Starting from (x_0, λ_0, μ_0) , For $k = 0, 1, 2, \dots$

1. Evaluate f_k , ∇f_k , $\nabla^2 \mathcal{L}_k$, h_k , g_k , ∇h_k , and ∇g_k
2. Solve the following convex QP to obtain s_k , λ_{k+1} , and μ_{k+1}

$$\begin{aligned} s_k = \underset{s}{\operatorname{argmin}} \quad & f_k + \nabla f_k^T s + \frac{1}{2} s^T \nabla^2 \mathcal{L}_k s \\ \text{s.t.} \quad & \nabla h_k^T s + h_k = 0 \\ & \nabla g_k^T s + g_k \leq 0 \end{aligned}$$

3. Set $x_{k+1} = x_k + s_k$.

NB! Solving a general QP can be computationally expensive for large-scale problems.

Example

Consider the Quadratically constrained quadratic programming (QCQP) problem

$$\begin{aligned} \min_x \quad & x^2 \\ \text{s.t.} \quad & x \leq 1 \\ & x^2 \geq 4 \end{aligned}$$

We would like to solve the optimization problem using SQP method by linearizing the constraints at $x_k = 1$, $\mu_k = [0, 0]^T$.

- a) Write the SQP problem for this problem.
- b) Is the problem feasible?

Equality constrained QP (EQP)

1. Select a set of active constraints $g_{\mathbb{A}} \subset g$, known as the working set $\mathcal{W}_k$
2. Solve the following convex equality-constrained QP to obtain s_k , λ_{k+1} , and $\mu_{\mathbb{A},k+1}$

$$\begin{aligned} s_k = \operatorname{argmin}_s \quad & f_k + \nabla f_k^T s + \frac{1}{2} s^T \nabla^2 \mathcal{L}_k s \\ \text{s.t.} \quad & \nabla h_k^T s + h_k = 0 \\ & \nabla g_{\mathbb{A},k}^T s + g_{\mathbb{A},k} = 0 \end{aligned}$$

3. Set $x_{k+1} = x_k + s_k$.

Equality constrained QP (EQP)

The KKT conditions of the equality constrained QP (EQP) is a set of linear equations.

$$\begin{bmatrix} \nabla^2 \mathcal{L}_k & \nabla h_k & \nabla g_{\mathbb{A},k} \\ \nabla h_k^T & 0 & 0 \\ \nabla g_{\mathbb{A},k}^T & 0 & 0 \end{bmatrix} \begin{bmatrix} s_k \\ \lambda_{k+1} \\ \mu_{\mathbb{A},k+1} \end{bmatrix} = - \begin{bmatrix} \nabla f_k \\ h_k \\ g_k \end{bmatrix}$$

So convex EQP requires linear equation solving only!

Active set convex QP method for inequality constraints

- ▶ Active set strategy is a set of rules for changing the working set $\mathcal{W}_k$ during the calculation towards the next iterate $\mathbf{x}_{k+1}$

Outline basic active set strategy¹

1. Set $k \leftarrow 0$; set $\mathbf{x}_0$ (feasible); initialize $\mathcal{W}_0$ (first guess)
2. Solve EQP_k to obtain proposed step $\mathbf{s}_k$ towards point $\mathbf{x}_{k+1}$
3. If $\mathbf{s}_k \neq 0$ determine $\alpha_k \mathbf{s}_k$ that provides feasible $\mathbf{x}_{k+1}$. If $\alpha_k < 1$ **add** blocking constraint to $\mathcal{W}_k$. Set $k \leftarrow k + 1$, return to 2.
4. If $\mathbf{s}_k = 0$ determine Lagrange multipliers. If multipliers for inequality constraints satisfy $\mu \geq 0$, stop with $\mathbf{x}^* = \mathbf{x}_k$, else **delete** inequality constraint with largest negative multiplier value from $\mathcal{W}_k$, set $k \leftarrow k + 1$ and return to 2.

¹Further reading: Sec 16.5, Nocedal and Wright

SQP algorithm

► Basic algorithm

1. Set $k = 0$, and select x_0, λ_0, μ_0
2. Solve QP subproblem k and determine $s_k, \lambda_{k+1}, \mu_{k+1}$
3. Set $x_{k+1} = x_k + s_k$
4. If termination criteria satisfied: stop, else $k = k + 1$ and return to 2

► Enhancement: quasi-Newton

- update $\nabla^2 \mathcal{L}$ using BFGS/DFP
- guaranteed positive definiteness of $\nabla_x^2 L$

► Enhancement: stabilization

- Line search SQP variant: $x_{k+1} = x_k + \alpha s_k$
- Trust region SQP variant: add $\|s\|_\infty \leq \Delta_k$ to QP

Quasi-Newton for SQP

- ▶ SQP needs Hessian of Lagrangian, but this requires second derivatives of objective and constraints, which may be expensive
- ▶ Quasi-Newton (BFGS) very successful for unconstrained optimization - Can we do the same in the constrained case?

$$\hat{B}_{k+1} = \hat{B}_k - \frac{\hat{B}_k s_k s_k^T \hat{B}_k}{s_k^T \hat{B}_k s_k} + \frac{r_k r_k^T}{r_k^T s_k}$$

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

$$y_k = \nabla_x \mathcal{L}(x_{k+1}, \lambda_{k+1}) - \nabla_x \mathcal{L}(x_k, \lambda_{k+1}),$$

$$r_k = \theta_k y_k + (1 - \theta_k) \hat{B}_k s_k \text{ (Damped update)}$$

Recap: line search for unconstrained optimization

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$$

Choose α that satisfies Armijo condition (sufficient decrease)

1. Choose $\bar{\alpha} = 1$, $c_1 \in (0, 1)$, $\rho \in (0, 1)$; Set $\alpha \leftarrow \bar{\alpha}$
2. Evaluate

$$f(\mathbf{x}_k + \alpha \mathbf{s}_k) < f(\mathbf{x}_k) + c_1 \alpha \nabla f(\mathbf{x}_k)^T \mathbf{s}_k, \quad c_1 \in (0, 1)$$

3. If Armijo condition not satisfied, set $\alpha \leftarrow \rho \alpha$, and return to step 2.
4. Terminate with $\alpha_k = \alpha$

Question: How can we evaluate the Armijo condition for a constrained problem?

Line search SQP algorithm

Determine the step length α by checking Armijo's sufficient decreases condition on a merit function $\phi(x, \cdot)$, instead of $f(x)$

► l_1 -merit function

$$\phi(x, r) = f(x) + r \sum_{j=1}^{m_h} |h_j(x)| + r \sum_{j=1}^{m_g} [g_j(x)]^+$$

► Augmented Lagrangian merit function

$$\begin{aligned} \phi(x, \lambda, \mu) = f(x) &+ \sum_{j=1}^{m_h} \lambda_j h_j(x) + \frac{\rho}{2} \sum_{j=1}^{m_h} (h_j(x))^2 \\ &+ \sum_{j=1}^{m_g} \mu_j g_k(x) + \frac{\rho}{2} \sum_{j=1}^{m_g} ([g_j(x)]^+)^2 \end{aligned}$$

Line search SQP

Check Armijo's condition with a merit function

$$\phi(\mathbf{x}_k + \alpha \mathbf{s}_k, \cdot) < \phi(\mathbf{x}_k, \cdot) + c_1 \alpha D(\phi(\mathbf{x}_k, \cdot); \mathbf{s}_k)$$

- ▶ $D(\phi(x); s)$ denotes the directional derivative of $\phi(x)$ in the direction s .
- ▶ l_1 merit functions are not differentiable, but they have a directional derivative.
- ▶ If ϕ is continuously differentiable, then

$$D(\phi(x); s) = \nabla \phi(x)^T s$$

Maratos Effect

Merit functions can reject good steps!

Example

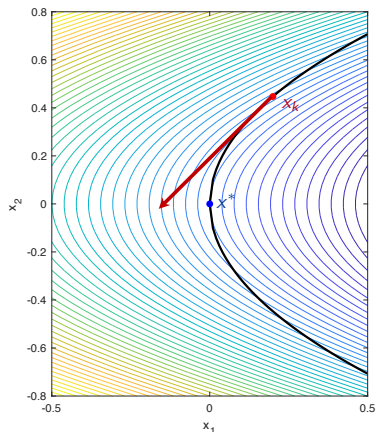
$$\min 3x_2^2 - 2x_1 \quad \text{s.t. } x_1 - x_2^2 = 0$$

Optimum $x^* = [0 \ 0]^\top$

Consider the iterate $x_k = [a^2 \ a]^\top$

Newton step $s_k = -[2a^2 \ a]^\top$

However $f(x_{k+1}) > f(x_k)$ and
 $|g(x_{k+1})| > |g(x_k)|$



No merit function will accept this step $\rightarrow$ creepingly slow convergence!

Trust region SQP algorithm

Use a ball of radius Δ_k where we trust the model.

Unstrained Optimzation (cf. Lecture 4)

$$\begin{aligned} s_k = \operatorname{argmin}_s \quad & f_k + \nabla f_k^T s + \frac{1}{2} s^T \nabla^2 L_k s \\ \text{s.t.} \quad & \|s\|_2 \leq \Delta_k \end{aligned}$$

Constrained Optimzation

$$\begin{aligned} s_k = \operatorname{argmin}_s \quad & f_k + \nabla f_k^T s + \frac{1}{2} s^T \nabla_x^2 L_k s \\ \text{s.t.} \quad & h_k + \nabla h_k^T s = 0 \\ & g_k + \nabla g_k^T s \leq 0 \\ & \|s\|_2 \leq \Delta_k \end{aligned}$$

Question: Is this a QP?

Trust region SQP algorithm

- QP subproblem with trust region

$$\begin{aligned} s_k = \operatorname{argmin}_s \quad & f_k + \nabla f_k s + \frac{1}{2} s^T \nabla_x^2 L_k s \\ \text{s.t.} \quad & h_k + \nabla h_k s = 0 \\ & g_k + \nabla g_k s \leq 0 \\ & \|s\|_\infty \leq \Delta_k \end{aligned}$$

- Infinity norm trust region implies

$$-\Delta_k \leq s_i \leq \Delta_k \quad i = 1, \dots, n$$

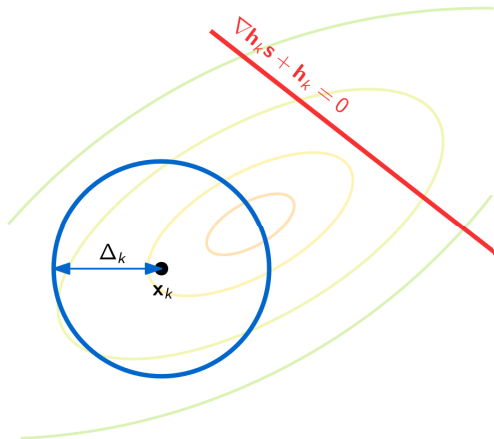
which perfectly matches with the linear constraints

Question: Does this problem always have a solution?²

²Further reading: Section 18.5, Nocedal and Wright

Trust region SQP algorithm

Linearized constraint inconsistent with the trust region constraint



Choosing the trust-region

- Measure quality of approximation:

$$\begin{aligned} r_k &= \frac{\text{actual reduction}}{\text{predicted reduction}} \\ &= \frac{\phi(x_k) - \phi(x_k + s_k)}{\hat{\phi}(0) - \hat{\phi}(s_k)} \end{aligned}$$

with $\hat{\phi}(s)$ predicted and $\phi(x)$ actual merit function

- Update trust-region radius:

$$\Delta_{k+1} = \begin{cases} \text{if } 0.75 < r_k < 1.25 & \rightarrow \text{increase } \Delta \\ \text{if } r_k < 0.25 & \rightarrow \text{decrease } \Delta \\ \text{else} & \rightarrow \text{do not change } \Delta \end{cases}$$

QP subproblem: inexact solution

- ▶ Exact solution of subproblem may be expensive and is often unnecessary

⇒ Find a cheap approximate solution s_k
within trust-region with sufficient merit reduction

- ▶ Condition for sufficient reduction: Cauchy point

Limitations of Active-Set Methods for NLP

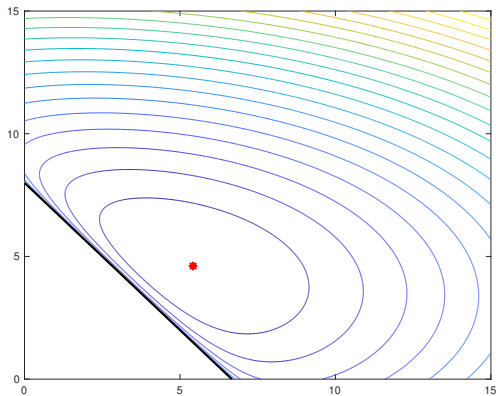
- ▶ Combinatorial nature of active sets
 - ▶ Must identify the correct active constraint set
 - ▶ Potentially exponential number of combinations
- ▶ Sensitive to degeneracy
 - ▶ LICQ failure or weakly active constraints cause instability
 - ▶ Can lead to zig-zagging or cycling
- ▶ Poor scaling for large problems
 - ▶ Frequent changes in working set
 - ▶ Repeated factorization of KKT systems
- ▶ Not well suited for very large, sparse NLPs

Motivation

Interior-point methods avoid explicit active-set identification and instead approach the boundary smoothly via barrier functions.

Interior point methods

$$\min_{x \in \mathbb{R}^n} f(x) - \tau \ln(-g(x))$$



Interior point methods

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^n} \quad & f(\mathbf{x}) - \tau \sum_{j=1}^{m_g} \ln(-g_j(\mathbf{x})) \\ \text{s.t.} \quad & \mathbf{h}(\mathbf{x}) = 0 \end{aligned}$$

KKT conditions

$$\begin{aligned} \nabla f(\mathbf{x}) + \sum_{j=1}^{m_g} \frac{-\tau}{g_j(\mathbf{x})} \nabla g_j(\mathbf{x}) + \nabla \mathbf{h}(\mathbf{x}) \lambda &= 0 \\ \mathbf{h}(\mathbf{x}) &= 0 \end{aligned}$$

- ▶ Notice that $-\mu_j g_j(\mathbf{x}) = \tau$
- ▶ As $\tau \rightarrow 0$, we recover our complementary slackness condition

Dealing with inequality constraints using Barrier method

Advantageous to write $g_j(x) \leq 0$ as equalities by introducing slack variables $\mathbf{s} \geq 0$

$$\begin{array}{ll}\min_{\mathbf{x}, \mathbf{s}} & f(\mathbf{x}) \\ \text{subject to} & g(\mathbf{x}) + \mathbf{s} = 0 \\ & h(\mathbf{x}) = 0 \\ & \mathbf{s} \geq 0\end{array}$$

Barrier problem:

$$\begin{array}{ll}\min_{\mathbf{x}, \mathbf{s}} & f(\mathbf{x}) - \tau \ln(\mathbf{s}) \\ \text{subject to} & h(\mathbf{x}) = 0 \\ & g(\mathbf{x}) + \mathbf{s} = 0\end{array}$$

Dealing with inequality constraints using Barrier method

KKT condition:

$$\begin{bmatrix} \nabla f(\mathbf{x}) - \nabla h(\mathbf{x})\lambda + \nabla g(\mathbf{x})\mu \\ S\mu - \tau \mathbb{1} \\ h(\mathbf{x}) \\ g(\mathbf{x}) + \mathbf{s} \end{bmatrix} = 0$$

where $S = \text{diag}(s_1, s_2, \dots, s_{m_g})$ and $\mathbb{1} = [1, 1, \dots, 1]^T$.

This is a system of nonlinear equations $F(\mathbf{w})$, that can be solved using Newton method with $\mathbf{w} = [\mathbf{x}, \mathbf{s}, \lambda, \mu]^T$.