

**Project 4: Sequential quadratic programming**
**Objective:**

The aim of this project is to implement the Sequential Quadratic Programming (SQP) algorithm, test it on different problems and compare the locally-convergent version to the globally-convergent one.

**Requirements:**

The student will implement and apply algorithm 21.1 to the following problems (Hock & Schittkowski, 1981):

a)

$$\begin{aligned}
 &\underset{x \in \mathbb{R}^2}{\text{minimize}} && \ln(1 + x_1^2) - x_2 \\
 &\text{subject to} && (1 + x_1^2)^2 + x_2^2 = 4 \\
 &&& -4 \leq x_1 \leq 4 \\
 &&& -4 \leq x_2 \leq 4
 \end{aligned}$$

Optimal solution  $f(x^*) = -\sqrt{3}$ ,  $x^* = [0, \sqrt{3}]$ ,  $\lambda_1^* = 0.2887$ .

Suggested starting point:  $x_0 = [2, 2]$ .

b)

$$\begin{aligned}
 &\underset{x \in \mathbb{R}^{10}}{\text{minimize}} && \sum_{i=1}^{10} e^{x_i} (c_i + x_i - \ln(\sum_{k=1}^{10} e^{x_k})) \\
 &&& e^{x_1} + 2e^{x_2} + 2e^{x_3} + e^{x_6} + e^{x_{10}} = 2 \\
 &&& e^{x_4} + 2e^{x_5} + e^{x_6} + e^{x_7} = 1 \\
 &&& e^{x_3} + e^{x_7} + e^{x_8} + 2e^{x_9} + e^{x_{10}} = 1 \\
 &&& -100 \leq x_i \leq 100 \quad \forall i = 1, \dots, 10.
 \end{aligned}$$

with

$$\begin{aligned}
 c_1 &= -6.089 & c_2 &= -17.164 & c_3 &= -34.054 & c_4 &= -5.914 & c_5 &= -24.721 \\
 c_6 &= -14.986 & c_7 &= -24.1 & c_8 &= -10.708 & c_9 &= -26.662 & c_{10} &= -22.179
 \end{aligned}$$



Enseignant: M.Bierlaire

Optimization and  
simulation  
Spring 2015  
Assistant: M.Y. Maknoon

---

### Project 4: Sequential quadratic programming

---

Optimal solution:  $f(x^*) = -47.76109026$

$x_1^* = -3.201212$   $x_2^* = -1.912060$   $x_3^* = 0.2444413$   $x_4^* = -6.537489$

$x_5^* = -0.7231524$   $x_6^* = -7.267738$   $x_7^* = -3.596711$   $x_8^* = -4.017769$

$x_9^* = -3.287462$   $x_{10}^* = -2 : 335582$

$\lambda_1^* = 9.78$   $\lambda_2^* = 13$   $\lambda_3^* = 15.2$

Suggested starting point:  $x_0 = [-2.3; \dots; -2.3]$ .

#### The student will proceed in the following way:

1. Solve the problems with algorithm 21.2 (global version). Let  $x^*$  be a local optimum.
2. Randomly generate several starting point in a  $\epsilon$ -ray ball centered in  $x^*$ , with, for example,  $\epsilon = 1, 10, 100, 1000$ .
3. For every value of  $\epsilon$ , make statistics on the number of times that algorithm 21.1 (local version) converges.
4. For every starting point such that the local algorithm has converged, compare the number of iterations of the two algorithms. What's the impact of the globalization on the efficiency of the method?

We encourage the students to compare their results with the Matlab built-in optimisation algorithms and justify the differences (if there exist any)

**Remark** Problems' formulation must comply with the required formulation

$$\begin{aligned} & \underset{x \in \mathbb{R}^n}{\text{minimize}} && f(x) \\ & \text{subject to} && h(x) = 0 \end{aligned}$$

#### Implementation

- Algorithms 11.4, 21.1, 21.2
- Useful Matlab functions for this project: **chol**.