Optimalisering og regulering TTK4135 - Assignment 4

Ola Kristoffer Hoff

16^{th} February, 2023

1 Quadratic Programming

a Under which conditions is a QP-problem convex? Why is convexity an important property?

A QP is convex if: $G \geq 0$. Convexity is an important property because it can be solved much faster than other harder problems. QP's are also widely used as the basis for other more complicated problems, so if those can be represented as convex problems, it will help solving them.

b Go through and understand the proof of Theorem 16.2. If you were to formulate the lemma with the condition $Z^T GZ \geq 0$ instead of $Z^T GZ > 0$, how would you change the wording of the theorem? How would the proof change?

It's hard to paraphrase so I will quote and try to explain the reasoning: "When the reduced Hessian matrix $Z^T GZ$ is positive semidefinite with zero eigenvalues, the vector x^* satisfying (16.4) is a local minimizer but not a strict local minimizer. If the reduced Hessian has negative eigenvalues, then x^* is only a stationary point, not a local minimizer." (N&W, page. 454)

As I understand it it boils down to the last equation of the proof:

$$q(x) = \frac{1}{2}u^T Z^T GZu + q(x^*)$$

The proof shows here that given a strict greater than zero property, the point x must be larger than the point x^*, hence x^* must be the global solution. If we say, it might be equal, then that all falls apart, and we can at most state that it's a local solution, if it has zero eigenvalues.

c Based on Example 16.4 in the textbook, show how Algorithm 16.3 finds the solution if the starting point is $x = \begin{bmatrix} 2 & 0 \end{bmatrix}^T$ and we assume that only the constraint $-x_1 + 2x_2 + 2 \geq 0$ is active. This means $\mathcal{W}_0 = \{3\}$.

$$\begin{aligned}
\min_x \; q(x) &= (x_1 - 1)^2 + (x_2 - 2.5)^2 \\
&s.t. \\
x_1 - 2x_2 + 2 &\geq 0 \\
-x_1 - 2x_2 + 6 &\geq 0 \\
-x_1 - 2x_2 + 2 &\geq 0 \\
x_1 &\geq 0 \\
x_2 &\geq 0
\end{aligned}$$

This gives us the following for the matrix form:

$$G = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$
$$c = \begin{bmatrix} -2 \\ -5 \end{bmatrix}$$
$$A = \begin{bmatrix} 1 & -2 \\ -1 & -2 \\ -1 & 2 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$
$$b = \begin{bmatrix} -2 \\ -6 \\ -2 \\ 0 \\ 0 \end{bmatrix}$$

By following the algorithm (16.3 - Active-Set Method for Convex QP) we get that:

First we need a feasible point for x_0, which we are given. Then we need to set the set $\mathcal{W}_0$ based on x_0, this is also given.

$$p = x - x_k,$$
$$g_k = Gx_k + c$$

$$\min_p \frac{1}{2}p^T Gp + g_k^T p, \ \ s.t. \ \ a_i^T p = 0, \ \ i \in \mathcal{W}_k \tag{1}$$

Then we begin at our iterative loop:

Iteration: $k = 0$

First we need to solve equation 1 to find p_0. I will show this process once.

$$\begin{aligned} &\min_p \frac{1}{2}p^T Gp + g_k^T p \\ &s.t. \\ &a_3^T p = 0 \end{aligned}$$

Given that we have one equality constraint we have one degree of freedom to optimise.

$$a_3^T p = 0$$
$$\begin{bmatrix} -1 & 2 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = 0$$
$$2p_2 - p_1 = 0$$

Using this relation in the objective funciton gives us:

$$
\begin{aligned}
\frac{1}{2}p^TGp + g_k^Tp &= \frac{1}{2}\begin{bmatrix}p_1 & p_2\end{bmatrix}\begin{bmatrix}2 & 0\\0 & 2\end{bmatrix}\begin{bmatrix}p_1\\p_2\end{bmatrix} + (Gx_k + c)^T\begin{bmatrix}p_1\\p_2\end{bmatrix}\\
p_1 &= 2p_2\\
&= \frac{1}{2}\begin{bmatrix}2p_2 & p_2\end{bmatrix}\begin{bmatrix}2 & 0\\0 & 2\end{bmatrix}\begin{bmatrix}2p_2\\p_2\end{bmatrix} + (Gx_k + c)^T\begin{bmatrix}2p_2\\p_2\end{bmatrix}\\
&= \frac{1}{2}\begin{bmatrix}4p_2 & 2p_2\end{bmatrix}\begin{bmatrix}2p_2\\p_2\end{bmatrix} + (\begin{bmatrix}2 & 0\\0 & 2\end{bmatrix}\begin{bmatrix}2\\0\end{bmatrix} + \begin{bmatrix}-2\\-5\end{bmatrix})^T\begin{bmatrix}2p_2\\p_2\end{bmatrix}\\
&= \frac{1}{2}(8p_2^2 + 2p_2^2) + \begin{bmatrix}2\\-5\end{bmatrix}^T\begin{bmatrix}2p_2\\p_2\end{bmatrix}\\
&= 5p_2^2 + 4p_2 - 5p_2\\
&= 5p_2^2 - p_2\\
&= (5p_2 - 1)p_2
\end{aligned}
$$

Since we are dealing with a convex problem we can say that the derivative must be equal to zero.

$$
\begin{aligned}
\nabla q(x) &= 10p_2 - 1 = 0\\
p_2 &= \frac{1}{10} \Rightarrow p_1 = 2p_2 = \frac{1}{5}\\
p &= \frac{1}{10}\begin{bmatrix}2\\1\end{bmatrix}
\end{aligned}
$$

We now have solved for p_0. Since it's not equal to zero we compute α_0 with equation 2:

$$
\alpha_k = min\left(1, \min_{i \notin \mathcal{W}_k, a_i^Tp_k < 0} \frac{b_i - a_i^Tx_k}{a_i^Tp_k}\right) \tag{2}
$$

$$
\begin{aligned}
\alpha_0 &= min\left(1, min\{a_2 : 10\}\right)\\
\alpha_0 &= 1
\end{aligned}
$$

Now that we have α_0 we can update x.

$$
\begin{aligned}
x_1 &= x_0 + \alpha_o p_0\\
x_1 &= \begin{bmatrix}2\\0\end{bmatrix} + 1\frac{1}{10}\begin{bmatrix}2\\1\end{bmatrix}\\
x_1 &= \begin{bmatrix}2.2\\0.1\end{bmatrix}
\end{aligned}
$$

The set $\mathcal{W}$ remains the same, $\mathcal{W}_1 = \mathcal{W}_0$.

Iteration: $k = 1$

Again we solve for the next value, p_1:

$$
\begin{aligned}
&\vdots\\
&= \frac{1}{2}\begin{bmatrix}4p_2 & 2p_2\end{bmatrix}\begin{bmatrix}2p_2\\p_2\end{bmatrix} + (\begin{bmatrix}2 & 0\\0 & 2\end{bmatrix}\begin{bmatrix}2.2\\0.1\end{bmatrix} + \begin{bmatrix}-2\\-5\end{bmatrix})^T\begin{bmatrix}2p_2\\p_2\end{bmatrix}\\
&= \frac{1}{2}(8p_2^2 + 2p_2^2) + \begin{bmatrix}2.4\\-4.8\end{bmatrix}^T\begin{bmatrix}2p_2\\p_2\end{bmatrix}\\
&= 5p_2^2 + 4.8p_2 - 4.8p_2\\
&= 5p_2^2
\end{aligned}
$$

$$\nabla q(x) = 10p_2 = 0$$
$$p_2 = 0 \Rightarrow p_1 = 2p_2 = 0$$
$$p = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Here $p_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, this gives us $\lambda = -2.4$. Since λ is strictly negative we must now remove the element in $\mathcal{W}_1$ that gives the biggest negative multiplier. The set only has one element and will then naturally become empty for the next iteration. The x-value will remain the same as this iteration.

Iteration: $k = 2$

We now have no constraints to limit our search, we have full degree of freedom on our objective function. We solve for p_2:

$$Gp_2 = -g_2$$
$$\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} p_2 = \begin{bmatrix} -2.4 \\ 4.8 \end{bmatrix}$$
$$p_2 = \begin{bmatrix} -1.2 \\ 2.4 \end{bmatrix}$$

p_2 is not equal to zero, so we will try and update x with an α_2, this becomes: $\alpha = \frac{2}{3}$. Here we also get a blocking constraint, constraint 1, and hence it is added to the set $\mathcal{W}$.

$$x_3 = x_2 + \alpha_2 p_2$$
$$x_3 = \begin{bmatrix} 1.4 \\ 1.7 \end{bmatrix}$$

Iteration: $k = 3$

Here solving for p_3 gives us: $p_3 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, and furthermore $\lambda = 0.8$. Since λ is now positive when p is equal to zero, this is a solution. Hence, $x^* = x_3 = \begin{bmatrix} 1.4 \\ 1.7 \end{bmatrix}$.

d Define the dual problem for the QP-problem in Example 16.4. Hint: See Section 12.9 and Example 12.10 and 12.12. Note that in this section, $f(x)$ is the primal objective and $q(x)$ is the dual objective.

The normal form for a QP is:

$$\underset{x}{min}\ \frac{1}{2}x^T Gx + x^T c \ \ s.t. \ \ Ax \geq b$$

Since our G is a symmetric positive definite matrix we can write the dual as(N&W, page. 349, Example 12.12):

$$q(\lambda) = \underset{x}{inf}\ \mathcal{L}(x, \lambda) = \underset{x}{inf}\ \frac{1}{2}x^T Gx + c^T x - \lambda^T (Ax - b)$$
$$\underset{(\lambda,x)}{max}\ -\frac{1}{2}x^T Gx + \lambda^T b, \ \ s.t. \ \ Gx + c - A^T\lambda = 0, \ \ \lambda \geq 0$$

e Explain how the dual optimization problem can be used to give an over-estimate of $q(\bar{x}) - q(x^*)$, when x^* is not known. ($q(x)$ is the objective function in a QP.) Hint: See Theorem 12.11 (and note that in this theorem, $f(x)$ is the primal objective and $q(x)$ is the dual objective).

Given the relation between the dual and the primal we can show that an over-estimate can be found without knowing x^*. *Weak duality* states that: for any feasible $\bar{x}$ and any $\bar{\lambda} \geq 0$, we have that $f(\bar{\lambda}) \leq q(\bar{x})$ (N&W, page. 345, Theorem 12.11). Using x^* as a feasible point gives us:

$$
\begin{aligned}
f(\bar{\lambda}) &\leq q(x^*) \\
f(\bar{\lambda}) + q(\bar{x}) &\leq q(x^*) + q(\bar{x}) \\
q(\bar{x}) - q(x^*) &\leq q(\bar{x}) - f(\bar{\lambda})
\end{aligned}
$$

2 Production Planning and Quadratic Programming

Two reactors, R_I and R_{II}, produce two products A and B. To make $1000kg$ of A, $2hours$ of R_I and $1hour$ of R_{II} are required. To make $1000kg$ of B, $1hour$ of R_I and $3hours$ of R_{II} are required. The order of R_I and R_{II} does not matter. R_I and R_{II} are available for 8 and $15hours$, respectively. We want to maximize the profit from selling the two products. The profit now depends on the production rate:

- the profit from A is $3 - 0.4x_1$ per tonne produced,
- the profit from B is $2 - 0.2x_2$ per tonne produced,

where x_1 is the production of product A and x_2 is the production of product B (both in number of tonnes).

a Formulate this as a quadratic program.

The objective function must be the profit, which is the amount sold times the price.

$$
\begin{aligned}
f(x) &= A_{amount} * A_{price} + B_{amount} * B_{price} \\
f(x) &= x_1 * (3 - 0.4x_1) + x_2 * (2 - 0.2x_2) \\
f(x) &= 3x_1 - 0.4x_1^2 + 2x_2 - 0.2x_2^2 \\
f(x) &= \frac{1}{2}x^T G x + c^T x \\
f(x) &= \frac{1}{2}x^T \begin{bmatrix} 0.8 & 0 \\ 0 & 0.4 \end{bmatrix} x + \begin{bmatrix} 3 & 2 \end{bmatrix} x
\end{aligned}
$$

We also negate the function $f(x)$ so from now on $q(x) = -f(x)$.

Then we have some constraints, the total time used by R_I and R_{II} must be less than or equal to 8 and 15 respectively.

$$
\begin{aligned}
c_1(x) &= R_{IA} * x_1 + R_{IB} * x_2 \leq 8 \\
c_1(x) &= 2x_1 + 1x_2 \leq 8 \\
\\
c_2(x) &= R_{IIA} * x_1 + R_{IIB} * x_2 \leq 15 \\
c_2(x) &= 1x_1 + 3x_2 \leq 15
\end{aligned}
$$

We also have that $x \geq 0$ since we can't produce negative amounts of product.

We now have:

$$\begin{aligned} \min_x \; q(x) &= \frac{1}{2}x^T G x + x^T c \\ \text{s.t.} \quad -2x_1 - x_2 &\geq -8 \\ -x_1 - 3x_2 &\geq -15 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \end{aligned}$$

b Make a contour plot and sketch the constraints.

Based on the diagram from Assignment 3, I created the plot seen in figure 1.

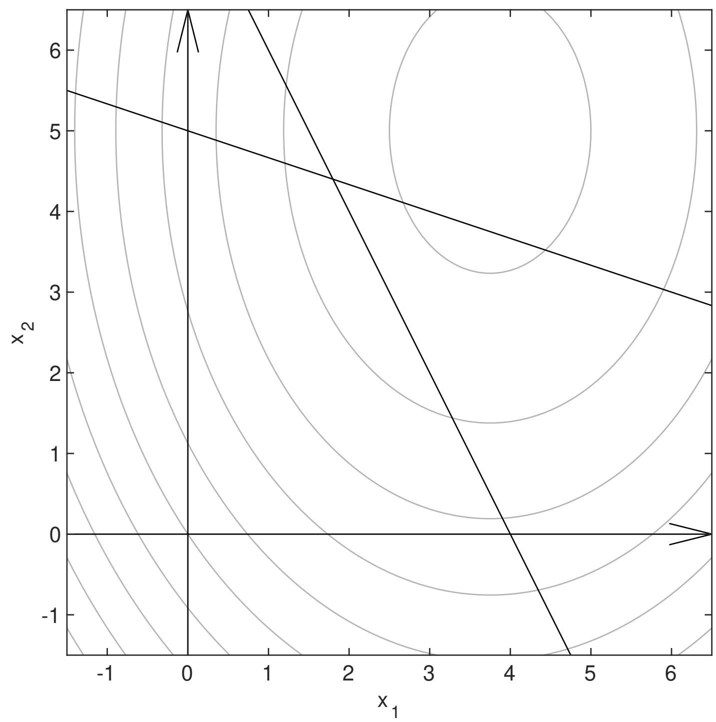

Figure 1: Contour plot with constraints for the QP problem.

c Find the production of A and B that maximizes the total profit. Do this using the MATLAB files posted on Blackboard; modify the file *qp_prodplan.m* so that it solves the problem formulated in a) (define G, c, A, and b and run the file). Is the solution found at a point of intersection between the constraints? Are all constraints active? Mark the iterations on the plot made in b), as well as the iteration number.

The solution is: $x = \begin{bmatrix} 2.25 \\ 3.50 \end{bmatrix}$. The solution is illustrated in figure 2. As you can see the solution is not found at the intersection of two constraints, meaning only one constraint is active.

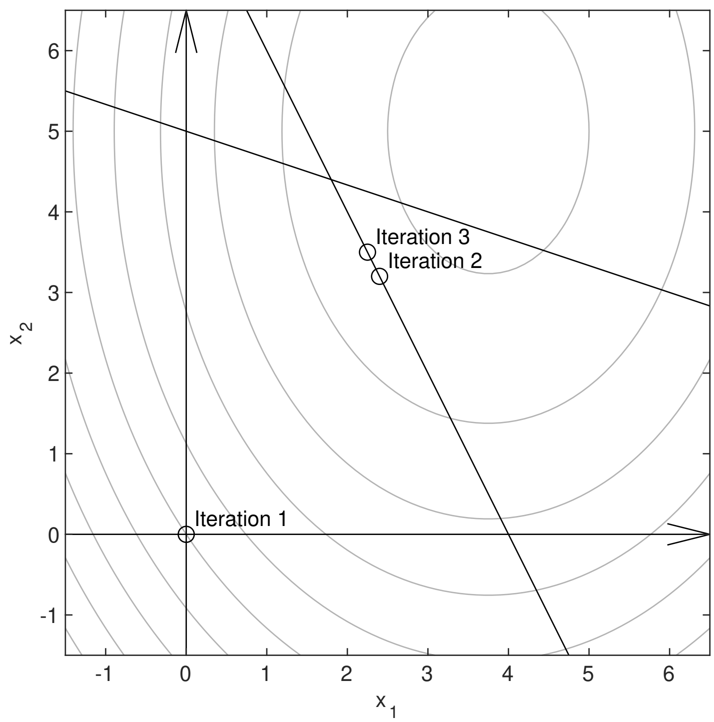

Figure 2: Contour plot with iterations of solving the QP.

d The solution is calculated by an active-set method. Explain how this method works based on the sequence of iterations from c).

First it has the two constraints of $x \geq 0$ active. It then looks for a better solution along these paths and find that it can switch to having one of the other constraints as active with a better estimate. This is where iteration 2 is. Then it again looks along the active constraint for a better value, here it sees that it is "above", then it must find the α-value taking it the proper distance along the constraint and then it lands in iteration 3 which is the solution. What happens here is that the direction of improvement is perpendicular to the linear constraint and it can thus not improve (given that the direction leaves the feasible area, as is in this case).

e Compare the solution in c) with Problem 2 c) in exercise 3 and comment.

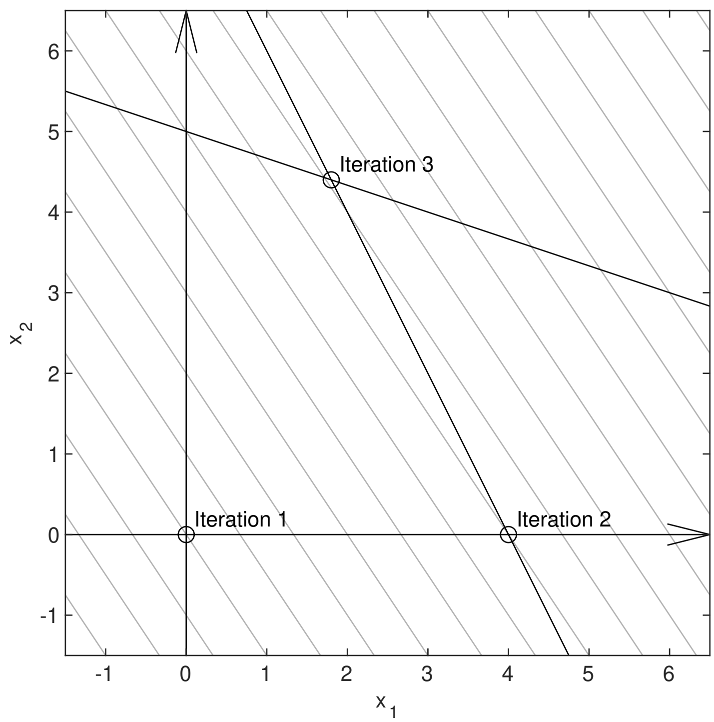

Figure 3: Plot of solution from problem 2c) in exercise 3.

As we can see the solutions are not too different. However, the linear problem will always end up in a corner, intersection of constraints, unless a constraint is parallel to the contour lines, or more precise, perpendicular to the gradient of the objective function. This is also why the solution for the LP "jumps" from corner to corner, while the QP have a more sophisticated method.