Optimal Control WS19/20 - **Homework exercise 1**
Solution of finite horizon open loop optimal control problems via nonlinear programming

Lucas Rath (3449668)

Lets consider the following optimal control problem

$$\min \quad J = \int_0^{t_f} x(t)^\top Q x(t) + u(t)^\top R u(t) dt \tag{1}$$

$$\text{s.t.} \quad \dot{x}(t) = \underbrace{\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix}}_{A} x(t) + \underbrace{\begin{bmatrix} 1 \\ 2 \end{bmatrix}}_{B} u(t) \tag{2}$$

A) One can discretize equation (1), by approximating the integral with a rectangle method, which considers $\int_{t_k=k*h}^{t_{k+1}=(k+1)*h} f(t)dt \approx h\, f(t_k)$.

In this way, we can rewrite (1) as:

$$J \approx h \Big[\sum_{k=0}^{N-1} x(k.h)^T Q x(k.h) + u(k.h)^T R u(k.h) \Big] \tag{3}$$

$$= h \Big[\sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k \Big] \tag{4}$$

with $x_k = x(t_k)$ and $u_k = u(t_k)$.

B) The exact solution of linear systems of type $\dot{x}(t) = Ax(t) + Bu(t)$ is:

$$x(t) = e^{A(t-t_0)} x(0) + \int_{t_0}^{t} e^{A(t-\tau)} B u(\tau) d\tau \tag{5}$$

By considering $t_0 = k.h = t_k$ and $t = (k+1).h = t_{k+1}$ in (5), and removing $u(\tau)$ from the integral, since it is piecewise constant in the interval $[t_k, t_{k+1})$ we have:

$$x_{k+1} = e^{A.h} x_k + \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)} B d\tau \; u(t_k) \tag{6}$$

Replacing the variable τ by $s = t_{k+1} - \tau$, we have that $ds = -d\tau$ and the integral is now evaluate from $s = h$ to $s = 0$. We now rewrite (6) as:

$$x_{k+1} = e^{A.h} x_k + \int_h^0 -e^{A.s} B\, ds\, u(t_k) \tag{7}$$

$$= \underbrace{e^{A.h}}_{A_D} x_k + \underbrace{\int_0^h e^{A.s} B\, ds}_{B_D} \; u_k \tag{8}$$

C) By using the Euler integration

$$\dot{x}(t_k) \approx \frac{x(t_{k+1}) - x(t_k)}{h} \tag{9}$$

in (2) we get:

$$x_{k+1} = (\underbrace{I + h.A}_{_EA_D})x_k + (\underbrace{B.h}_{_EB_D})u_k \tag{10}$$

where we denote $_EA_D$ and $_EB_D$ the Euler approximation for the discrete state-space matrices.

Equation (10) can be viewed as an approximation of the exact solution (exact iff $u(t)$ is piecewise constant in $[t_k, t_{k+1})$, as mentioned before) by considering the approximation of the matrix exponential

$$e^{A.h} = I + h.A + \frac{h^2}{2}.A^2 + \dots \tag{11}$$

In this sense, we have that:

$$A_k = e^{A.h} = I + h.A + \mathcal{O}(h^2) \tag{12}$$

$$\Rightarrow \boxed{A_k =_E A_D + \mathcal{O}(h^2)} \tag{13}$$

and

$$B_k = \int_0^h e^{A.s} B \, ds \tag{14}$$

$$= \int_0^h \Big(I + A.s + \mathcal{O}(s^2)\Big) B \, ds \tag{15}$$

$$= h.B + A.\frac{h^2}{2}.B + \mathcal{O}(h^3) \tag{16}$$

$$= h.B + \mathcal{O}(h^2) \tag{17}$$

$$\Rightarrow \boxed{B_k = \ _EB_D + \mathcal{O}(h^2)} \tag{18}$$

To this end, we conclude that the Euler discretization is an approximation of the exact discretization with residual $\mathcal{O}(h^2)$.

Lets consider again the discretization of the optimal control problem as defined in (4)

$$min \quad J = \sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k \tag{19}$$

$$s.t. \quad x_{k+1} = A_D x_k + B_D \tag{20}$$

where we basically just dropped h, since this does not change the argmin of the function J.

D) Lets start with the dynamics constraints. Repeated use of $x_{k+1} = A_D x_k + B_D u_k$ with initial condition $x(0) = x_0$ gives us:

$$\underbrace{\begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ \vdots \\ x(N-1) \end{bmatrix}}_{\boldsymbol{x}} = \underbrace{\begin{bmatrix} I \\ A_D \\ A_D^2 \\ \vdots \\ A_D^{N-1} \end{bmatrix}}_{\Omega} x_0 + \underbrace{\begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ B_D & 0 & \cdots & 0 & 0 \\ A_D B_D & B_D & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ A_D^{N-2} B_D & A_D^{N-3} B_D & \cdots & B_D & 0 \end{bmatrix}}_{\Gamma} \underbrace{\begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(N-2) \\ u(N-1) \end{bmatrix}}_{\boldsymbol{u}} \tag{21}$$

which we can write in a more compact notation:

$$\boldsymbol{x} = \Omega x_0 + \Gamma \boldsymbol{u} \tag{22}$$

Defining the a new variable vector as $\boldsymbol{y} = [\boldsymbol{x}, \boldsymbol{u}]^T = [x(0), ..., x(N-1), u(0), ..., u(N-1)]^T$, we get the redefined constraint equations as:

$$\underbrace{[I \quad -\Gamma]}_{A_{eq}} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{u} \end{bmatrix} = \underbrace{\Omega\, x_0}_{b_{eq}} \tag{23}$$

$$A_{eq}\, \boldsymbol{y} = b_{eq} \tag{24}$$

Now lets redefine the cost function in terms of the new variable $\boldsymbol{y}$

$$\begin{aligned} J &= h \sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k && (25) \\ &= h(\boldsymbol{x}^T \bar{Q} \boldsymbol{x} + \boldsymbol{u}^T \bar{R} \boldsymbol{u}) && (26) \\ &= h \boldsymbol{y}^T \begin{bmatrix} \bar{Q} & 0 \\ 0 & \bar{R} \end{bmatrix} \boldsymbol{y} && (27) \\ &= \frac{1}{2} \boldsymbol{y}^T \underbrace{2h \begin{bmatrix} \bar{Q} & 0 \\ 0 & \bar{R} \end{bmatrix}}_{H} \boldsymbol{y} && (28) \end{aligned}$$

with $\bar{Q} = diag(Q, Q, ..., Q)$ and $\bar{R} = diag(R, R, ..., R)$ block diagonal matrices with N blocks.

The new discretized optimal control problem can be then rewritten as:

$$\begin{aligned} \min_{\boldsymbol{y}} \quad & J(\boldsymbol{y}) = \frac{1}{2} \boldsymbol{y}^\top H \boldsymbol{y} + f^\top \boldsymbol{y} + d \\ \text{s.t.} \quad & A_{\text{eq}}\, \boldsymbol{y} = b_{\text{eq}} \\ & A_{\text{ineq}}\, \boldsymbol{y} \leq b_{\text{ineq}} \end{aligned} \tag{29}$$

where the not previously defined matrices f, d, A_{ineq}, b_{ineq} are equal zero with the corresponding correct dimension.

E) For the case with only equality constraints, the KKT necessary conditions are given by:

$$\begin{aligned} \nabla_{\boldsymbol{y}} \mathcal{L}(\boldsymbol{y}, \lambda) &= 0 \\ h(\boldsymbol{y}) &= 0 \end{aligned} \tag{30}$$

with Lagrangian $\mathcal{L}(\boldsymbol{y}, \lambda) = J(\boldsymbol{y}) + \lambda^T h(\boldsymbol{y})$ and equality constraint $h(\boldsymbol{y}) = A_{eq}\boldsymbol{y} - b_{eq} = 0$.

Equation (30) can be solved using Newton's method: $f(x) = 0$, which first uses Taylor expansion to approximate $f(x + \Delta x) \approx f(x) + \nabla f(x)^T \Delta x$ around x and solves for Δx until $x \leftarrow x + \Delta x$ converges. Therefore, we first have:

$$\begin{bmatrix} \nabla_{\boldsymbol{y}}\mathcal{L}(\boldsymbol{y}+\Delta y, \lambda + \Delta\lambda) \\ h(\boldsymbol{y}+\Delta y) \end{bmatrix} \approx \begin{bmatrix} \nabla_{\boldsymbol{y}}\mathcal{L}(\boldsymbol{y}, \lambda) \\ h(\boldsymbol{y}) \end{bmatrix} + \begin{bmatrix} \nabla^2_{\boldsymbol{y}}\mathcal{L}(\boldsymbol{y}, \lambda) & \nabla_{\boldsymbol{y},\lambda}\mathcal{L}(\boldsymbol{y}, \lambda) \\ \nabla_{\boldsymbol{y}} h(\boldsymbol{y})^T & 0 \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

which simplifies to:

$$\begin{bmatrix} \nabla^2_{\boldsymbol{y}}\mathcal{L}(\boldsymbol{y}, \lambda) & \nabla_{\boldsymbol{y}} h(\boldsymbol{y}) \\ \nabla_{\boldsymbol{y}} h(\boldsymbol{y})^T & 0 \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta\lambda \end{bmatrix} = -\begin{bmatrix} \nabla_{\boldsymbol{y}}\mathcal{L}(\boldsymbol{y}, \lambda) \\ h(\boldsymbol{y}) \end{bmatrix} \tag{31}$$

to continue, lets use the definitions of (29) and calculate:

$$\begin{aligned} \nabla^2_{\boldsymbol{y}}\mathcal{L}(\boldsymbol{y}, \lambda) &= H \\ \nabla_{\boldsymbol{y}} h(\boldsymbol{y}) &= A_{eq}^T \\ \nabla_{\boldsymbol{y}}\mathcal{L}(\boldsymbol{y}, \lambda) &= H\boldsymbol{y} + f + A_{eq}^T\lambda \end{aligned} \tag{32}$$

in order to substitute these results in (33) to get:

$$\begin{bmatrix} H & A_{eq}^T \\ A_{eq} & 0 \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta\lambda \end{bmatrix} = -\begin{bmatrix} H\boldsymbol{y} + f + A_{eq}^T\lambda \\ A_{eq}\boldsymbol{y} - b_{eq} \end{bmatrix} \tag{33}$$

Finally, after using the fact that $\Delta y = y^+ - \boldsymbol{y}$ and $\Delta\lambda = \lambda^+ - \lambda$, we can simplify (33) to:

$$\begin{bmatrix} H & A_{eq}^T \\ A_{eq} & 0 \end{bmatrix} \begin{bmatrix} y^+ \\ \lambda^+ \end{bmatrix} = \begin{bmatrix} -f \\ b_{eq} \end{bmatrix} \tag{34}$$

By performing block Gaussian elimination, equation (34) can be solved directly first for λ^+ and then for y^+ (remember that A_eq might not be invertible), which results in:

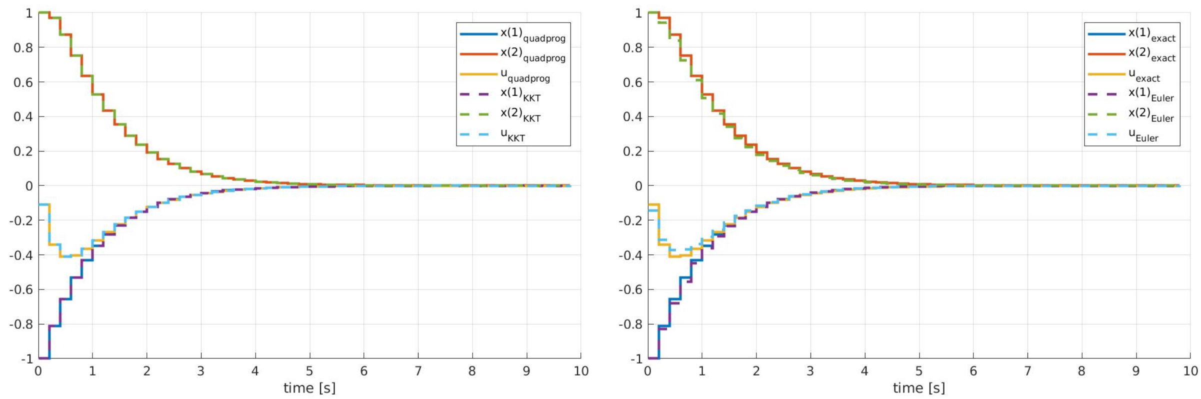
$$\lambda^+ = -(A_{eq}H^{-1}A_{eq}^T)^{-1}(A_{eq}H^{-1}f + b_eq) \tag{35}$$

$$y^+ = -H^{-1}(A_{eq}^T\lambda^+ + f) \tag{36}$$

Since y^+ and λ^+ can be solved analytically (independent of any previous iterate), we conclude that the Newton's method converges in one single step. This result is expected, since the cost function is quadratic in $\boldsymbol{y}$ and the constraints are linear, which implies that (30) is linear and therefore the Taylor approximation used is actually exact everywhere.

The conditions derived by the Newton's method is actually only necessary. However, since the optimization problem (29) is convex for $\boldsymbol{y}$ and λ, necessary optimality conditions are also sufficient.

F) Figure 1 show the comparison of solution obtained when using different solvers (Matlab `quadprog` and the derived KKT closed-formula) and different discretization methods (exact or Euler).



(a) Comparison of **quadprog** *and KKT solution (using exact discretization)*

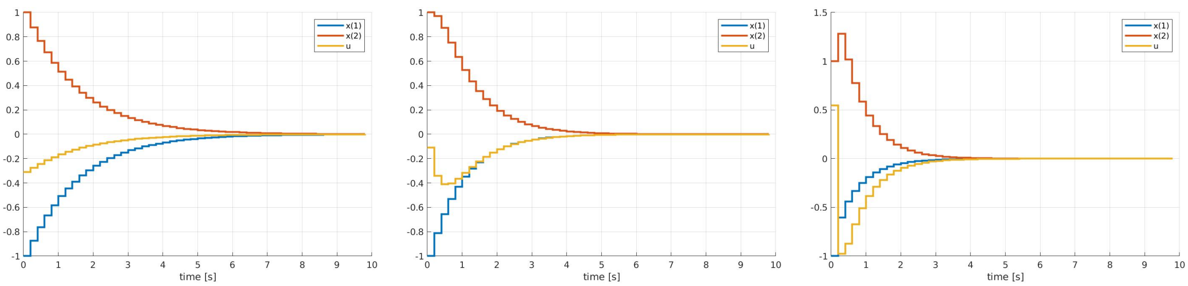
(b) Comparison exact and Euler discretization methods (using **quadprog** *solver)*

Figure 1: *Comparison of different solvers and discretization methods for the optimal control problem*

Clearly, `quadprog` and the derived KKT closed-formula give very similar results (error norm $= 3.3796e - 14$), which is in theoretically expected. Since the problem is convex, the iterative `quadprog` should converge to the derived exact KKT solution.

On the other hand, one can see that the results obtained with Euler discretization are not exactly the same as the results obtained with exact discretization (error norm $= 0.0758$). Theoretically, the error should increase quadratically with h ($\mathcal{O}(h^2)$). We therefore expect an reasonable error while using $h = 0.2s$.

G) Increasing Q in relation to R, makes state deviations from the origin very costly in comparison to high control input values. The direct effect is that the system will stabilize faster but will present higher control input values. On the other hand, decreasing Q, penalizes more higher inputs than state deviations making the system slower but with lower control input values. Both conclusions can be confirmed to hold when analysing Figure 2.



(a) $Q = 0.2I_2$ *(b)* $Q = I_2$ *(c)* $Q = 40I_2$

Figure 2: *Comparison of the optimal control solututions for different values of the state cost* Q

H) We simulate the continuous time dynamical system of equation (2), with the piece-wise constant optimal control input sequence obtained from (29) when using exact discretization. We then compare the continuous time simulation results with the discrete time and show the error between them in Figure 3.

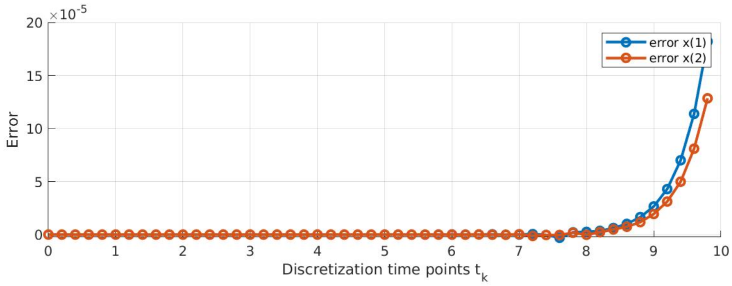


Figure 3: *Error between continuous time and discrete time simulations using the piece-wise constant optimal control input sequence.*

The two simulation results are practically the same. Note the scale of the y-axis on the Figure 3, which is 10^{-5}. This result is expected, since the discretization performed is exact for piece-wise constant control inputs and therefore the result should be exactly the same.