Optimal Control WS20/21: Homework 1

Daniel Bergmann

January 6, 2021

1 Theory Part

First, all answers to theoretical questions are given. The regarding plots can be found in section 2.

a) Discretize cost functional:
Discretizing with time step size h

$$x_k \overset{\text{def}}{=} x(kh) \overset{\text{def}}{=} x(t_k)$$

and evaluate the running cost at each t_k for all $k = 0, \dots, N$. This yields

$$J \approx x_N^T Q x_N + \sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k. \tag{1}$$

b) Matrix representation of discretized linear dynamics:

We know

$$x(t) = e^{At} x(0) + \int_0^t e^{A(t-\tau)} B u(\tau) d\tau \tag{2}$$

Discretizing with time step size h

$$x_k \overset{\text{def}}{=} x(kh) \overset{\text{def}}{=} x(t_k)$$

and inserting it into (2) yields therefore for the state x_{k+1} the expression

$$\begin{aligned} x_{k+1} &= e^{Ah(k+1)} x_0 + \int_0^{(k+1)h} e^{A((k+1)h-\tau)} B u(\tau) d\tau & (3) \\ &= e^{Ah} \left[e^{Akh} x_0 + \int_0^{kh} e^{A(kh-\tau)} B u(\tau) d\tau \right] + \int_{kh}^{(k+1)h} e^{A(kh+h-\tau)} B u(\tau) d\tau. & (4) \end{aligned}$$

We assume that the steering signal u is constant between the discretization time samples t_k. We simplify this expression by substituting with $v(\tau) = kh + h - \tau$ and obtain

$$\begin{aligned} x_{k+1} &= e^{Ah} x_k - \left(\int_{v(kh)}^{v((k+1)h)} e^{Av} dv \right) B u_k & (5) \\ &= e^{Ah} x_k - \left(\int_h^0 e^{Av} dv B \right) u_k. & (6) \\ &= \underbrace{e^{Ah}}_{=:A_d} x_k + \underbrace{\left(\int_0^h e^{Av} dv B \right)}_{=:B_d} u_k. & (7) \end{aligned}$$

This verifies the given Matrix representation.

c) Euler Approximation:

$$\dot{x}(t_k) \approx \frac{x(t_{k+1}) - x(t_k)}{h}. \tag{8}$$

Rearranging (8) yields

$$\begin{aligned} x(t_{k+1}) - x(t_k) &\approx h\dot{x}(t_k) && (9)\\ &= hAx(t_k) + hBu(t_k) && (10)\\ \iff \quad x(t_{k+1}) &\approx (I + Ah)x(t_k) + hBu(t_k) && (11)\\ x_{k+1} &\approx \underbrace{(I + Ah)}_{=:A_{\text{eul}}} x_k + hBu_k. && (12) \end{aligned}$$

Relation to (7):
By the definition of the matrix exponential, we have

$$A_d = e^{Ah} = \sum_{k=0}^{\infty} \frac{A^k h^k}{k!}. \tag{13}$$

Neglecting all quadratic and higher terms yields with A_d from **b)**

$$A_d \approx I + Ah = A_{\text{eul}}$$

✓

which is exactly the matrix obtained by the euler approximation.

d) Bring the discrete OC problem

$$\begin{aligned} \min_{x,u}.\quad & \sum_{k=0}^{N-1} g(x,u) = x_N^T Q x_N + \sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k \\ \text{subject to}\quad & x_{k+1} = A_d x_k + B_d u_k \\ & x_0 = \bar{x} \end{aligned} \tag{14}$$

to the form

$$\begin{aligned} \min_{y}.\quad & \frac{1}{2} y^T H y + f^T y + d \\ \text{subject to}\quad & A_{\text{eq}} y = b_{\text{eq}} \\ & A_{\text{in}} y \leq b_{\text{in}}. \end{aligned} \tag{15}$$

We begin by defining the optimization variables vector y. It consists of all inputs and all states that occur from the initial time to the finite time horizon. So we define

$$u := \begin{pmatrix} u_0 \\ \vdots \\ u_{N-1} \end{pmatrix}; \quad x := \begin{pmatrix} x_0 \\ \vdots \\ x_N \end{pmatrix}; \quad y := \begin{pmatrix} x \\ u \end{pmatrix}. \tag{16}$$

For the objective function, we obtain $g(x,u) = \frac{1}{2} y^T H y + 0^T y + 0$ with the block diagonal matrix

$$H = \operatorname{diag}(\underbrace{Q, \dots, Q}_{N \text{blocks}}, \underbrace{R, \dots, R}_{N-1 \text{blocks}}).$$

✓

Further, we implement the given system dynamics as linear equality constraints. Rearranging the difference equation yields

$$A_d x_k - x_{k+1} + B u_k = 0 \qquad \text{for all } k = 1, \dots, N-1.$$

By stacking these equations for all $k \in \{1, \dots, N-1\}$, we can formulate them as $A_{\text{eq}} y = b_{\text{eq}}$ with $A_{\text{eq}} = \begin{pmatrix} A_{\text{eq}}^x & A_{\text{eq}}^u \end{pmatrix}$, and more detailed,

$$A^x_{\text{eq}} = \begin{pmatrix} I_{n\times n} & 0 & \cdots & \cdots & 0 \\ A_d & -I_{n\times n} & \cdots & \cdots & \vdots \\ 0 & \ddots & \ddots & & 0 \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & \cdots & & A_d & -I_{n\times n} \end{pmatrix},$$

$$A^u_{\text{eq}} = \begin{pmatrix} 0 & \cdots & 0 \\ B_d & \ddots & \vdots \\ \vdots & \ddots & 0 \\ 0 & \cdots & B_d \end{pmatrix}.$$

$$b_{\text{eq}} = \begin{pmatrix} \bar{x} \\ \vdots \\ 0 \end{pmatrix}.$$

The first row of blocks implements the initial condition $x_0 = \bar{x}$.

We have no inequality conditionts to implement, hence we have no inequality constraints.

e) Formulating conditions at an optimum.
For stating criteria on a minimum of our problem, we use the KKT conditions. We know that for convex problems they are neccessary and sufficient. Since $Q, R \succ 0$ and hence $H \succ 0$, the quadratic objective is convex. The equality constraints are affine and we have no inequality constraints. Thus the given problem is convex. For the KKT conditions to hold in a point y, a CQ has to be satisfied. We use the Linear Independence Constraint Qualification (LICQ). It holds automatically, since we have no inequality constraints. Further, we compute the condition on the Lagrangian of (15) at an optimal point y^*

$$\nabla_y L(y^*, \lambda, \nu) = Hy^* + \nu^T A_{\text{eq}} \stackrel{!}{=} 0. \tag{17}$$

Additionally, the point y^* has to be admissible and hence

$$A_{\text{eq}} y^* - b_{\text{eq}} = 0 \tag{18}$$

must hold. By rearranging (17), we obtain

$$y^* = -H^{-1} A^T_{\text{eq}} \nu.$$

Note, that H is invertible since it is positive definite. Plugging this into (18) yields for A_{eq} with full rank

$$A_{\text{eq}}(-H^{-1}A^T_{\text{eq}}\nu) - b_{\text{eq}} = 0 \Longleftrightarrow \nu = -(A_{\text{eq}} H^{-1} A^T_{\text{eq}})^{-1} b_{\text{eq}}.$$

Hence,

$$y^* = H^{-1} A^T_{\text{eq}} (A_{\text{eq}} H^{-1} A^T_{\text{eq}})^{-1} b_{\text{eq}}$$

is an optimal solution.

f) No theoretical question asked, see section 2.

g) High values for α let high state values be punished strongly. Therefore the optimal trajectory contains small values for x_1, x_2. Input values can increase, because their impact on the objective value is not as strong as the state's impact. Small values for α have the opposite effect.

h) Since the applied discretization in the quadratic programming approach is exact, the state values for the simulated and the optimized trajectorie should coincide at every t_k. Nonetheless, small numerical inaccuracies amplify themselves over time. This leads to large differences in late time samples.

2 Plots

2.1 Plots for f)

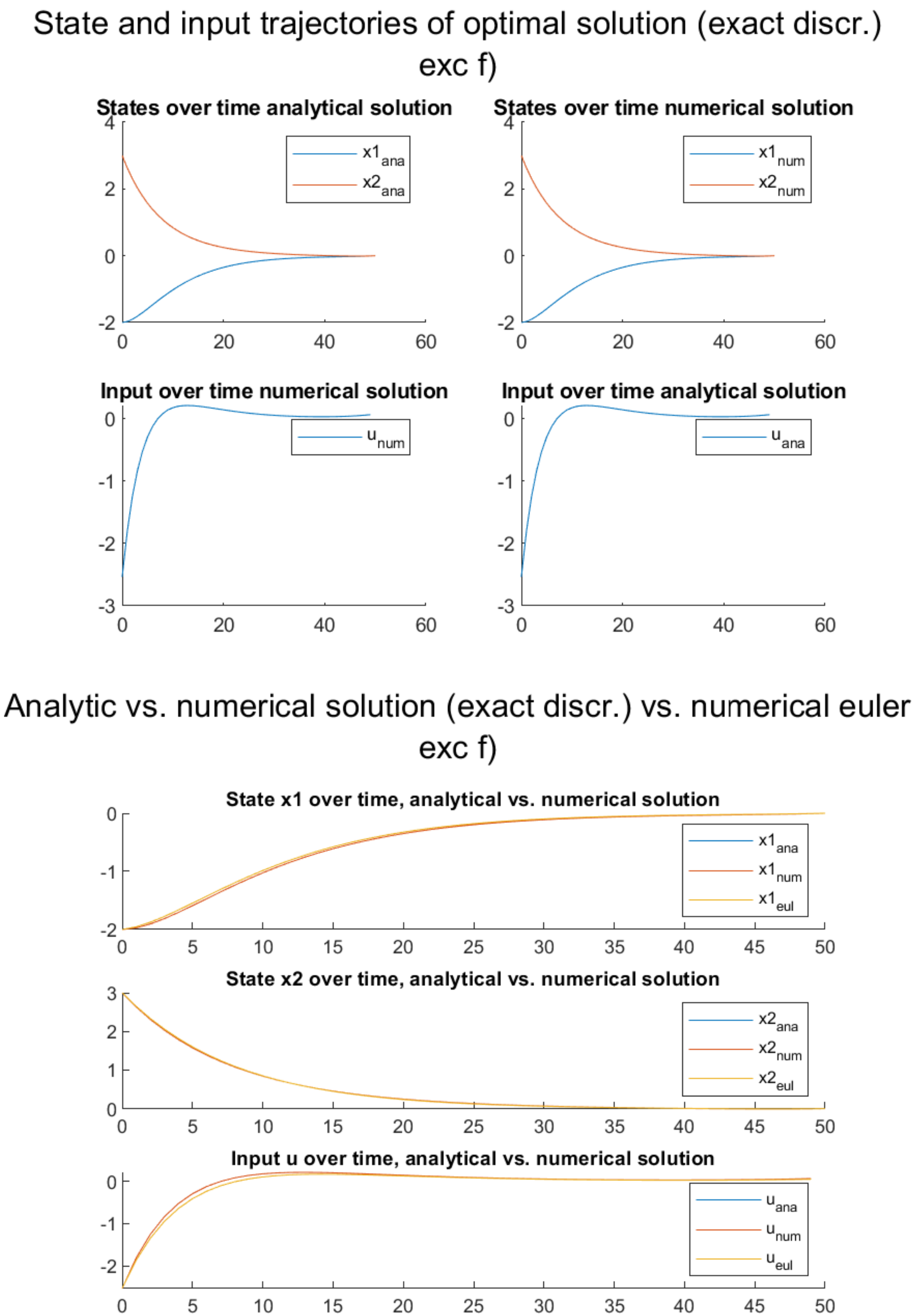

Figure 1: Plot for Exc f)

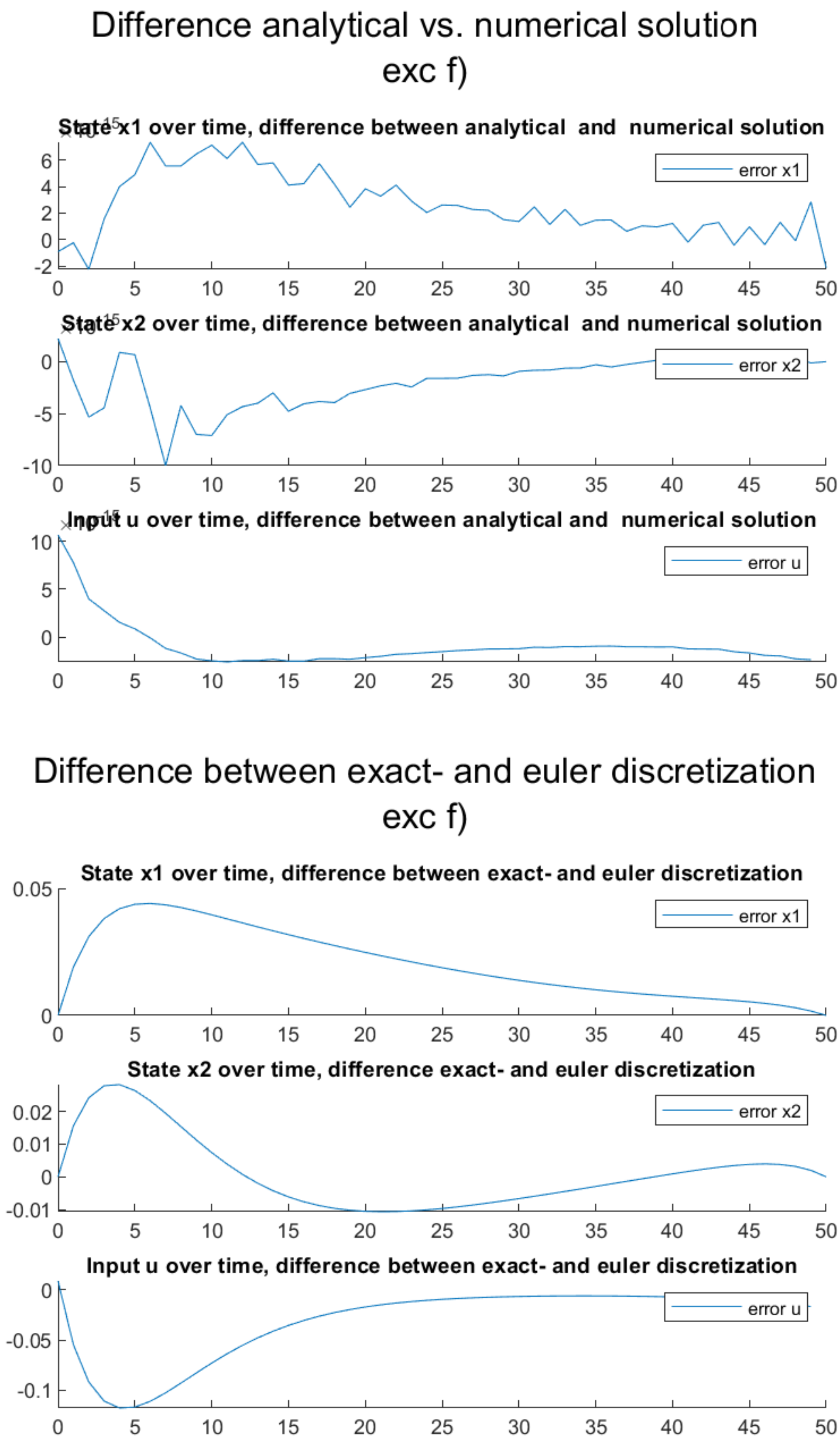

Figure 2: Plot for Exc f)

2.2 Plots for g)

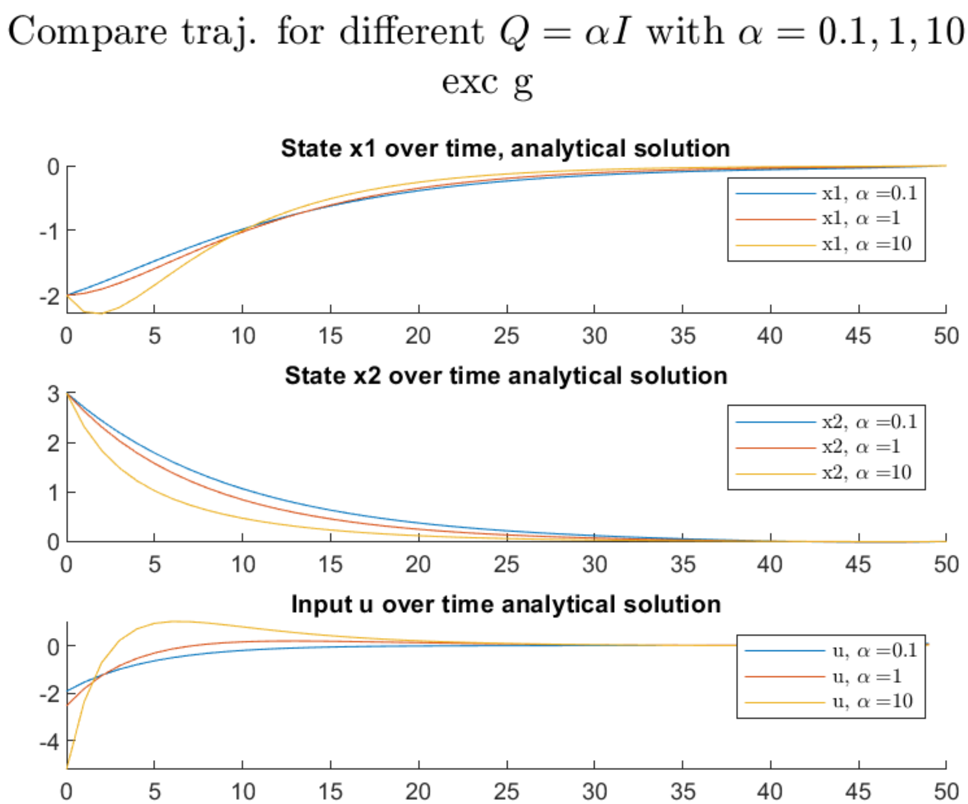

Figure 3: Plot for Exc g)

2.3 Plots for h)

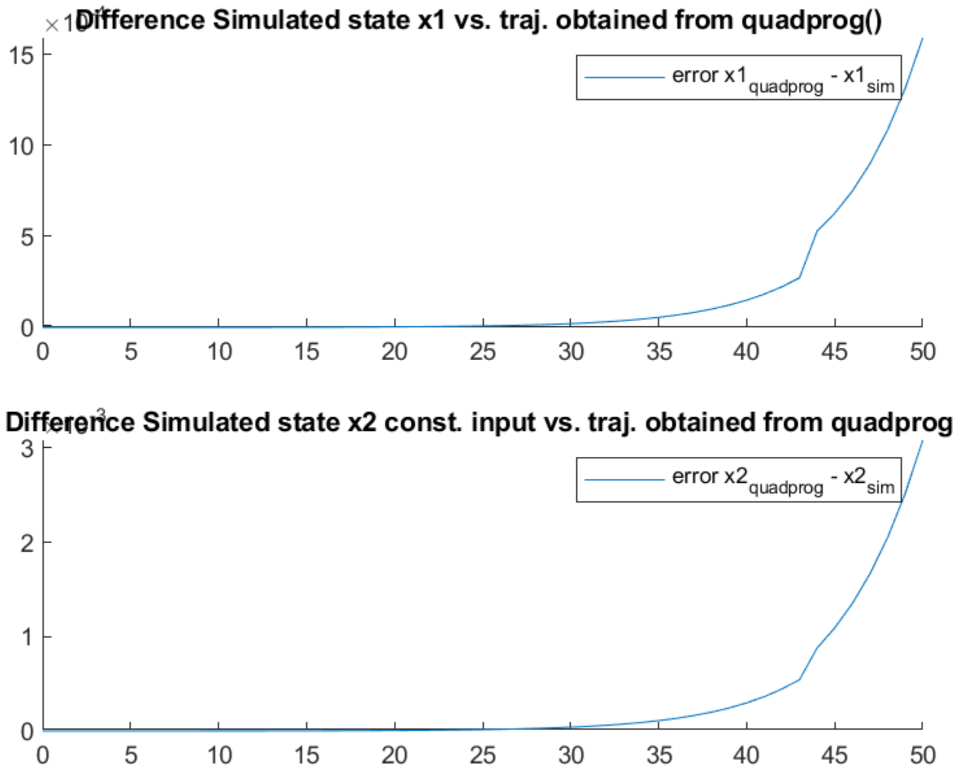

Figure 4: Plot for Exc h)