Software package to solve LQR problems (regular/singular) for state-space and DAE systems.

Imrul Qais, Chayan Bhawal, and Debasattam Pal
Funding Agency: SERB SRG/2021/000721

1 Problem statement

Problem 1.1 *Consider a DAE-based LTI system having a minimal input-state-output (i/s/o) representation*

$$E\dot{x}(t) = Jx(t) + Lu(t), \textit{where } J, E \in \mathbb{R}^{\mathtt{n}\times\mathtt{n}}, L \in \mathbb{R}^{\mathtt{n}\times\mathtt{m}} \textit{ and } \mathtt{det}(E) = 0. \tag{1}$$

For any initial condition $x_0 \in \mathbb{R}^{\mathtt{n}}$, find input $u(t)$ such that the following performance-index

$$J(x_0, u) = \int_0^\infty \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} dt, \textit{ where } \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \geqslant 0 \tag{2}$$

is minimized with $\lim_{t\to\infty} x(t) = 0$.

The algorithm to find the PD controller is explained in Section 1.1. The basic flow of data to solve the problem is shown in the diagram below:

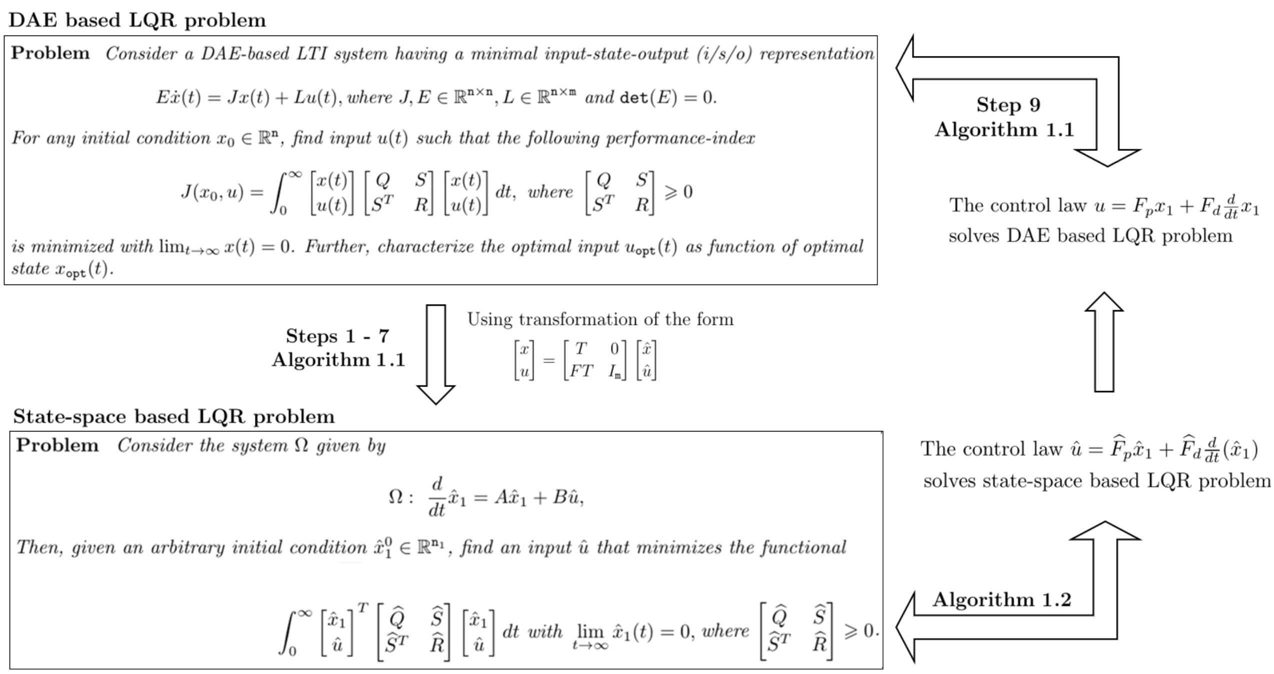

Figure 1: Flow of the algorithm to design PD feedback controllers to solve DAE and state-space based LQR Problem

Lemma 1.2 and the discussion thereafter provides the necessary matrices to convert the problem to state-space based LQR problem (Problem 1.3).

Lemma 1.2 *Assume that the system Σ defined in Problem 1.1 is impulse controllable. Define* $\mathtt{n}_1 := \mathtt{rank}\, E$ *and* $\mathtt{n}_2 := \mathtt{n} - \mathtt{n}_1$. *Let* $\widehat{P}, \widehat{T} \in \mathbb{R}^{\mathtt{n}\times\mathtt{n}}$ *such that* $\widehat{P}E\widehat{T} = \begin{bmatrix} I_{\mathtt{n}_1} & 0 \\ 0 & 0 \end{bmatrix}$. *Define*

$$\widehat{P}J\widehat{T} =: \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix} \text{ and } \widehat{P}L =: \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \tag{3}$$

where $J_{11} \in \mathbb{R}^{\mathtt{n}_1\times\mathtt{n}_1}, J_{22} \in \mathbb{R}^{\mathtt{n}_2\times\mathtt{n}_2}, B_2 \in \mathbb{R}^{\mathtt{n}_2\times\mathtt{m}}$. *Then the following statements are true:*

1. *There exists* $F_2 \in \mathbb{R}^{\mathtt{m}\times\mathtt{n}_2}$ *such that* $(J_{22} + B_2F_2)$ *is non-singular.*
2. *There exist* $A \in \mathbb{R}^{\mathtt{n}_1\times\mathtt{n}_1}, B \in \mathbb{R}^{\mathtt{n}_1\times\mathtt{m}}$ *and* $B_2 \in \mathbb{R}^{\mathtt{n}_2\times\mathtt{m}}$ *such that the system*

$$\begin{bmatrix} I_{\mathtt{n}_1} & 0 \\ 0 & 0 \end{bmatrix} \frac{d}{dt}\hat{x} = \begin{bmatrix} A & 0 \\ 0 & I_{\mathtt{n}_2} \end{bmatrix} \hat{x} + \begin{bmatrix} B \\ B_2 \end{bmatrix} \hat{u} \tag{4}$$

is isomorphic to Σ *with* $\begin{bmatrix} x \\ u \end{bmatrix} = \begin{bmatrix} T & 0 \\ FT & I_{\mathtt{m}} \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{u} \end{bmatrix}$ *for some* $F \in \mathbb{R}^{\mathtt{m}\times\mathtt{n}}$ *and a non-singular matrix* $T \in \mathbb{R}^{\mathtt{n}\times\mathtt{n}}$. *In particular, the following matrices can be used*

$$\left.\begin{aligned} F &:= \begin{bmatrix} 0_{\mathtt{m},\mathtt{n}_1} & F_2 \end{bmatrix} \widehat{T}^{-1} \\ P &:= \begin{bmatrix} I_{\mathtt{n}_1} & -(J_{12} + B_1F_2)(J_{22} + B_2F_2)^{-1} \\ 0 & I_{\mathtt{n}_2} \end{bmatrix} \widehat{P} \\ T &:= \widehat{T} \begin{bmatrix} I_{\mathtt{n}_1} & 0 \\ -(J_{22} + B_2F_2)^{-1}J_{21} & (J_{22} + B_2F_2)^{-1} \end{bmatrix} \end{aligned}\right\} \tag{5}$$

Note that from the above proof it is easy to infer that $PET = \begin{bmatrix} I_{\mathtt{n}_1} & 0 \\ 0 & 0 \end{bmatrix}$ and $P(J + LF)T = \begin{bmatrix} A & 0 \\ 0 & I_{\mathtt{n}_2} \end{bmatrix}$, where

$$\left.\begin{aligned} A &:= J_{11} - (J_{12} + B_1F_2)(J_{22} + B_2F_2)^{-1}J_{21}, \\ B &:= B_1 - (J_{12} + B_1F_2)(J_{22} + B_2F_2)^{-1}B_2. \end{aligned}\right\} \tag{6}$$

Next, define

$$\begin{bmatrix} \widehat{Q} & \widehat{S} \\ \widehat{S}^T & \widehat{R} \end{bmatrix} := \begin{bmatrix} I_{\mathtt{n}_1} & 0 \\ 0 & -B_2 \\ 0 & I_{\mathtt{m}} \end{bmatrix}^T \begin{bmatrix} T & 0 \\ FT & I_{\mathtt{m}} \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} T & 0 \\ FT & I_{\mathtt{m}} \end{bmatrix} \begin{bmatrix} I_{\mathtt{n}_1} & 0 \\ 0 & -B_2 \\ 0 & I_{\mathtt{m}} \end{bmatrix}. \tag{7}$$

Since $\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \geqslant 0$, it follows that $\begin{bmatrix} \widehat{Q} & \widehat{S} \\ \widehat{S}^T & \widehat{R} \end{bmatrix} \geqslant 0$. Using this new cost matrix, we define the following LQR problem (possibly singular) for a state space system:

Problem 1.3 *Consider the system* Ω *given by*

$$\Omega: \ \frac{d}{dt}\hat{x}_1 = A\hat{x}_1 + B\hat{u}, \tag{8}$$

where A *and* B *are as defined in eq.* (6). *Then, given an arbitrary initial condition* $\hat{x}_1^0 \in \mathbb{R}^{\mathtt{n}_1}$, *find an input* $\hat{u}$ *that minimizes the functional*

$$\int_0^\infty \begin{bmatrix} \hat{x}_1 \\ \hat{u} \end{bmatrix}^T \begin{bmatrix} \widehat{Q} & \widehat{S} \\ \widehat{S}^T & \widehat{R} \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{u} \end{bmatrix} dt \text{ with } \lim_{t\to\infty} \hat{x}_1(t) = 0, \tag{9}$$

where $\begin{bmatrix} \widehat{Q} & \widehat{S} \\ \widehat{S}^T & \widehat{R} \end{bmatrix} \geqslant 0$ *is as defined in eq.* (7).

1.1 Algorithm to find the feedback controller for Problem 1.1

In this section we present the complete algorithm to design a PD feedback controller to solve Problem 1.1.

Algorithm 1.1 *LQR-DAE* [Algorithm to find PD controller for DAE based based LQR problem]

Input: E, J, L, Q, S, R (from Problem 1.1).
Output: F_p and F_d
Comment: The control law $u = F_p x_1 + F_d \frac{d}{dt} x_1$ solves the LQR Problem 1.1.
Assumptions: The system is impulse controllable and stabilizable.

1: Find matrices $\widehat{P}$ and $\widehat{T}$ such that $\widehat{P}E\widehat{T} = \begin{bmatrix} I_{\mathtt{n}_1} & 0 \\ 0 & 0 \end{bmatrix}$

2: Partition $\widehat{P}J\widehat{T} =: \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix}$ and $\widehat{P}L =: \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$, where $J_{11} \in \mathbb{R}^{\mathtt{n}_1 \times \mathtt{n}_1}$, $J_{22} \in \mathbb{R}^{\mathtt{n}_2 \times \mathtt{n}_2}$, $B_2 \in \mathbb{R}^{\mathtt{n}_2 \times \mathtt{m}}$.

3: Construct the following matrices:

$$\begin{aligned} A &:= J_{11} - (J_{12} + B_1 F_2)(J_{22} + B_2 F_2)^{-1} J_{21} \\ B &:= B_1 - (J_{12} + B_1 F_2)(J_{22} + B_2 F_2)^{-1} B_2 \end{aligned}$$

4: Find F_2 such that $J_{22} + B_2 F_2$ is non-singular.

5: Construct $F := \begin{bmatrix} 0_{\mathtt{m},\mathtt{n}_1} & F_2 \end{bmatrix} \widehat{T}^{-1}$.

6: Construct the following matrix

$$T := \widehat{T} \begin{bmatrix} I_{\mathtt{n}_1} & 0 \\ -(J_{22} + B_2 F_2)^{-1} J_{21} & (J_{22} + B_2 F_2)^{-1} \end{bmatrix}$$

7: Construct the matrices $\widehat{Q} \in \mathbb{R}^{\mathtt{n}_1 \times \mathtt{n}_1}$, $\widehat{S} \in \mathbb{R}^{\mathtt{n}_1 \times \mathtt{m}}$, $\widehat{R} \in \mathbb{R}^{\mathtt{m} \times \mathtt{m}}$ using the equation below:

$$\begin{bmatrix} \widehat{Q} & \widehat{S} \\ \widehat{S}^T & \widehat{R} \end{bmatrix} := \begin{bmatrix} I_{\mathtt{n}_1} & 0 \\ 0 & -B_2 \\ 0 & I_{\mathtt{m}} \end{bmatrix}^T \begin{bmatrix} T & 0 \\ FT & I_{\mathtt{m}} \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} T & 0 \\ FT & I_{\mathtt{m}} \end{bmatrix} \begin{bmatrix} I_{\mathtt{n}_1} & 0 \\ 0 & -B_2 \\ 0 & I_{\mathtt{m}} \end{bmatrix}.$$

8: Call Algorithm 1.2:

$$[\widehat{F}_p, \widehat{F}_d] = \textit{LQR-Statespace}\left(A, B, \widehat{Q}, \widehat{S}, \widehat{R}\right).$$

9: Construct the feedback gain matrices:

$$F_p := F + \begin{bmatrix} \widehat{F}_p & 0_{\mathtt{m},\mathtt{n}_2} \end{bmatrix} T^{-1} \text{ and } F_d := \begin{bmatrix} \widehat{F}_d & 0_{\mathtt{m},\mathtt{n}_2} \end{bmatrix} T^{-1}.$$

Algorithm 1.2 *LQR-Statespace* [Algorithm to find PD controller for state-space based LQR problem]

Input: $A, B, \widehat{Q}, \widehat{S}, \widehat{R}$ (from Problem 1.3).
Output: $\widehat{F}_p$ and $\widehat{F}_d$
Comment: The control law $\hat{u} = \widehat{F}_p \hat{x}_1 + \widehat{F}_d \frac{d}{dt}(\hat{x}_1)$ solves the LQR Problem 1.3.
Assumptions: The system is stabilizable.

Find an orthogonal matrix $U \in \mathbb{R}^{\mathtt{m}\times\mathtt{m}}$ such that $U^T \widehat{R} U = \begin{bmatrix} 0 & 0 \\ 0 & R_e \end{bmatrix}$, where $R_e \in \mathbb{R}^{\mathtt{r}\times\mathtt{r}}$ and $\mathtt{r} := \mathtt{rank}\,\widehat{R}$.

Compute $\begin{bmatrix} B_r & B_e \end{bmatrix} := BU$ and $\begin{bmatrix} 0 & S_e \end{bmatrix} := \widehat{S}U$, where $B_e, S_e \in \mathbb{R}^{\mathtt{n}_1\times\mathtt{r}}$.

Compute the matrices $A_r := A - B_e R_e^{-1} S_e^T$, $Q_r := Q - S_e R_e^{-1} S_e^T$, and $M := B_e R_e^{-1} B_e^T$.

Construct the matrices $E_r = \begin{bmatrix} I_{\mathtt{n}_1} & 0 & 0 \\ 0 & I_{\mathtt{n}_1} & 0 \\ 0 & 0 & 0_{\mathtt{d},\mathtt{d}} \end{bmatrix}$ and $H_r = \begin{bmatrix} A_r & -M & B_r \\ -Q_r & -A_r^T & 0 \\ 0 & B_r^T & 0_{\mathtt{d},\mathtt{d}} \end{bmatrix}$, where $\mathtt{d} := \mathtt{m} - \mathtt{r}$.

Define $\mathtt{n_s} := \dfrac{\mathtt{degdet}\,(sE_r - H_r)}{2}$ and $\mathtt{n_f} := \mathtt{n}_1 - \mathtt{n_s}$.

Find a full column-rank matrix $V_\Lambda =: \begin{bmatrix} V_{1\Lambda} \\ V_{2\Lambda} \\ V_{3\Lambda} \end{bmatrix}$ with $V_{1\Lambda}, V_{2\Lambda} \in \mathbb{R}^{\mathtt{n}_1\times\mathtt{n_s}}$ and $V_{3\Lambda} \in \mathbb{R}^{\mathtt{d}\times\mathtt{n_s}}$ such that $H_r V_\Lambda = E_r V_\Lambda \Gamma$, where $\Gamma \in \mathbb{R}^{\mathtt{n_s}\times\mathtt{n_s}}$ and $\sigma(\Gamma) \subseteq \mathbb{C}_-$.

Define $\mathtt{p} := \mathtt{rank} \begin{bmatrix} \widehat{Q} & 0 & S_e \\ 0 & 0_{\mathtt{d},\mathtt{d}} & 0 \\ S_e^T & 0 & R_e \end{bmatrix}$. Then, obtain the matrices $C \in \mathbb{R}^{\mathtt{p}\times\mathtt{n}_1}$ and $D_e \in \mathbb{R}^{\mathtt{p}\times\mathtt{r}}$ such that $C^T C = \widehat{Q}$, $C^T D_e = S_e$ and $D_e^T D_e = R_e$.

Construct the matrix (we call it Markov matrix)

$$\mathcal{M} := \begin{bmatrix} 0 & \dots & 0 & C_r B_r \\ 0 & \dots & C_r B_r & C_r A_r B_r \\ \vdots & \ddots & \vdots & \vdots \\ C_r B_r & \dots & C_r A_r^{\mathtt{n_f}-\mathtt{d}-2} B_r & C_r A_r^{\mathtt{n_f}-\mathtt{d}-1} B_r \end{bmatrix}, \text{ for } \mathtt{n_f} > \mathtt{d}; \qquad 0_{\mathtt{p},\mathtt{d}} \text{ for } \mathtt{n_f} = \mathtt{d}$$

where $C_r := C - D_e R_e^{-1} S_e^T$.

Obtain a matrix $\widetilde{N}$ of appropriate dimension such that $\mathcal{M}\widetilde{N} = 0$, where $\mathtt{rank}\,\widetilde{N} = \mathtt{n_f} - \mathtt{d}$ and $\widetilde{N}$ is full column-rank.

Obtain $\widetilde{W} := \begin{bmatrix} B_r & A_r B_r & \dots & A_r^{\mathtt{n_f}-\mathtt{d}-1} B_r \end{bmatrix} \widetilde{N}$ for $\mathtt{n_f} > \mathtt{d}$; $\quad B_r$ for $\mathtt{n_f} = \mathtt{d}$

Obtain a matrix $W_e \in \mathbb{R}^{\mathtt{n}_1\times\mathtt{d}}$ such that $\mathtt{im} \begin{bmatrix} \widetilde{W} & W_e \end{bmatrix} = \mathtt{im} \begin{bmatrix} B_r & A_r \widetilde{W} \end{bmatrix}$ for $\mathtt{n_f} > \mathtt{d}$; $W_e = [\,]$ for $\mathtt{n_f} = \mathtt{d}$.

Define $X_1 := \begin{bmatrix} V_{1\Lambda} & \widetilde{W} & W_e \end{bmatrix}$ $X_2 := \begin{bmatrix} V_{1\Lambda} & B_r & A_r \widetilde{W} \end{bmatrix}$.

Compute $K_{\mathtt{max}} := \begin{bmatrix} V_{2\Lambda} & 0 \end{bmatrix} X_1^{-1} = \begin{bmatrix} V_{2\Lambda} & 0 \end{bmatrix} X_2^{-1}$.

Obtain $G_0 \in \mathbb{R}^{\mathtt{d}\times(\mathtt{n_f}-\mathtt{d})}$ and $G_1 \in \mathbb{R}^{\mathtt{d}\times\mathtt{d}}$ such that $(A_r - MK_{\mathtt{max}} + B_r \begin{bmatrix} V_{3\Lambda} & G_0 & G_1 \end{bmatrix} X_1^{-1})$ is non-singular.

Compute $\widehat{F}_p := U \begin{bmatrix} \begin{bmatrix} V_{3\Lambda} & G_0 & G_1 \end{bmatrix} X_1^{-1} \\ -R_e^{-1}(B_e^T + S_e^T K_{\mathtt{max}}) \end{bmatrix}$ and $\widehat{F}_d := U \begin{bmatrix} \begin{bmatrix} 0 & I_{\mathtt{d},\mathtt{d}} & -G_0 \end{bmatrix} X_2^{-1} \\ 0 \end{bmatrix}$.

Then, the feedback law $\hat{u} = \widehat{F}_p \hat{x}_1 + \widehat{F}_d \frac{d}{dt}(\hat{x}_1)$ solves the LQR Problem 1.3.

2 Procedure to use LQR_Problem SCILAB Package

The steps to be followed to run the package:

1. The code is written for SCILAB. SCILAB is a free and open-source computational package that can be downloaded from `https://www.scilab.org/`

2. Download the entire package LQR_Problem from `https://github.com/chayanbhawal/chayanbhawal.github.io/tree/master/files/LQR_Problem` to a suitable location on your desktop.

3. Open SCILAB and change the working directory to the directory where you have saved the files in the previous step.

4. Open the `input_data_matrix.sci` file in SCILAB. This Scilab function initializes the system matrices and cost matrices involved in the problem. Four different example cases are already written in the file. In order to use it for your problem, change the system matrices E, J, L and the cost matrix Q, S, R. (see Problem 1.1 to know the meaning of the matrices)

5. Open and Run `Main_file.sce` file. This Scilab program computes PD state-feedback closed-loop gain matrices to solve the LQR problem for both state-space and DAE systems.

   - Input to the program
     - LQR cost matrix represented as $\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix}$.
     - System matrices E, J, L, where the DAE system is given by $E\frac{d}{dt}x = Jx + Lu$.
     - For the state-space system, E is the identity matrix.
   - Output of the program:
     - Feedback gain matrices F_p, F_d where $u = F_p x + F_d(dx/dt)$ that solves the LQR problem.
     - Dimension of the good slow subspace (`dim_good_slow`) and fast subspace (`dim_fast`) of the (underlying) LQR problem.
     - Expected closed loop poles (`expect_cl_poles`).
   - Assumptions:
     - For DAE problem: The system must be impulse controllable and stabilizable.
     - For standard state-space problem: The system must be stabilizable.

References

[1] L. Dai. Singular Control Systems. Springer-Verlag Berlin, Heidelberg, 1989.

[2] G.R. Duan. Analysis and Design of Descriptor Linear Systems. Springer New York, NY, 2010.

[3] J. Heiland and E. Zuazua. Classical system theory revisited for turnpike in standard state space systems and impulse controllable descriptor systems. SIAM Journal on Control and Optimization, 59(5):3600–3624, 2021.

[4] I. Qais, C. Bhawal, and D. Pal. Optimal singular LQR problem: a PD feedback solution. SIAM Journal on Control and Optimization, 61(4):2655–2681, 2023.

[5] I. Qais, C. Bhawal, and D. Pal. Software package to solve LQR problems (regular/singular) for state-space and DAE systems. `https://github.com/chayanbhawal/chayanbhawal.github.io/tree/master/files/LQR_Problem`, 2023.