Optimal Control: Final Project

Voleti, Rajeev Shobhit

April 2020

Introduction

I started by solving the adaptive optimal control problem discussed in the brief paper by Vrabie et al.[1]. Then summarised the paper by Freeman et al. in the context of what was discussed in class. I set up the policy iteration on Simulink for better visualisation. I have provided a trivial piece of simulink code for the inverse optimality example to compare the responses obtained using the pointwise min-norm and solving the HJI equation explicitly just for completeness.

1 Adaptive Optimal Control

We begin by considering a linear system of the form

$$\dot{x} = Ax + Bu \tag{1}$$

Let the dynamics be given as

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -3 & -2 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \tag{2}$$

The controllability of this system can be observed by using the rank check for the controllability matrix $\mathbf{C}$

$$\mathbf{C} = \begin{bmatrix} B & AB \end{bmatrix} \tag{3}$$

$$= \begin{bmatrix} 0 & 1 \\ 1 & -2 \end{bmatrix} \tag{4}$$

Since $\mathbf{C}$ has full rank, the system is controllable and hence, stabilisable.

The policy iteration algorithm requires the knowledge of an initially stabilising gain K. However, we have chosen an open loop stable system with poles $(-1 + 1.4142i, -1 - 1.4142i)$ So we begin with $P_i = 0_{2x2}$

The cost function to be minimised is of the form

$$J = \min \int_0^{\infty} (x^T Q x + R u^2) dt \tag{5}$$

Now, the solution to the optimal control problem is determined using Bellman's Optimality and is given by

$$\begin{aligned} u &= -Kx \\ &= R^{-1}B^TP \end{aligned}$$

Here P is the solution to the Algebraic Riccati Equation(ARE) given as

$$A^TP + PA - PBR^{-1}B^TP + Q = 0$$

To compare, we find that the optimal P is given as:

$$P = \begin{bmatrix} 1.297 & 0.1623 \\ 0.1623 & 0.3075 \end{bmatrix} \tag{6}$$

Code for the above is given in Simulation folder

If the dynamics are assumed to be unknown equations described by

$$x_t^TP_ix_t = \int_t^{(t+T)} x_\tau^T(Q + K_i^TRK_ix_\tau)d\tau + x_{t+T}^TP_ix_{t+T} \tag{7}$$

$$K_{i+1} = R^{-1}B^TP_i \tag{8}$$

Can be used as a policy iteration algorithm.

Assuming that we start with an initially stabilising gain K. If

$$A_i = (A - BK_i)$$

is stable, solving for P_i in eq (8) is equivalent to finding the solution of the Lyapunov equation

$$A_i^TP_i + P_iA_i = -x_t^T(K_i^TRK_i + Q)x_t$$

Since A_i is stable, and $(K_i^TRK_i + Q) > 0$, there exists a unique solution to the Lyapunov equation, $P_i > 0$. Since $V(x) = \frac{1}{2}x_t^tP_ix_t$ is a Lyapuov function of the system $\dot{x} = A_ix$ and

$$\frac{d(x_t^TP_ix_t)}{dt} = x_t^T(A_i^TP_i + P_iA_i)x_t = -x_t^T(K_i^TRK_i + Q)x_t$$

Integrating the equation on both sides we get

$$\begin{aligned} \int_t^{(t+T)} (x_t^T(K_i^TRK_i + Q)x_t)dt &= -\int_t^{(t+T)} \frac{d(x_t^TP_ix_t)}{dt}dt \\ &= x_t^TP_ix_t - x_{t+T}^TP_ix_{t+T} \end{aligned}$$

for $\forall T > 0$. This implies that the policy iteration algorithm proposed solves the Lyapunov equation uniquely provided A_i is asymptotically stable. Now we must show that our update of K_i leads to a stabilising solution.

We begin by assuming the initial value of K is stabilising. Consider the Lyapunov function for the system.

$$V_i(x_t) = x_t^TP_ix_t$$

The derivative of V using the policy iteration algorithm is given as

$$\begin{aligned} V_i\dot{(x_t)} &= x_t^T[P_i(A - BK_{i+1}) + (A - BK_{i+1})^T P_i]x_t \\ &= x_t^T[P_i(A - BK_i) + (A - BK_i)^T P_i]x_t + x_t^T[P_i B(K_i - K_{i+1}) + (K_i - K_{i+1})^T B^T P_i]x_t \end{aligned}$$

The term after completing squares may be written as

$$x_t^T[P_i B(K_i - K_{i+1}) + (K_i - K_{i+1})^T B^T P_i]x_t = x_t^T[-(K_i - K_{i+1})^T R(K_i - K_{i+1}) - K_{i+1}^T R K_{i+1} + K_i^T R K_i]x_t$$

Utilising Lyapunov equation the equation for $\dot{V}$ may be written as

$$\dot{V}(x_t) = -x_t^T[(K_i - K_{i+1})^T R(K_i - K_{i+1})]x_t - x_t^T[Q + K_{i+1}^T R K_{i+1}]x_t \tag{9}$$

Now, since $Q \geq 0$, $R > 0$, $V_i(x_t)$ is a Lyapunov positive definite function, we notice from 9 that if the initial K is stabilising, the updated K_{i+1} is stabilising as the new Lyapunov candidate function is indeed the Lyapunov function for the updated system.

Hence the policy iteration proposed is stabilising for an intial choice of a stabilising gain

1.1 Online implementation

The policy iteration algorithm is summarised by the flow chart

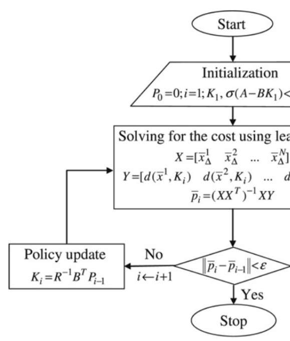

Figure 1: Policy iteration Algorithm

The policy iteration works online by sampling the state space at N points between t and t + T. The cost function of the closed loop system is minimised. Let

$$\dot{V} = x^T(t)Qx(t) + u^T(t)Ru(t) \tag{10}$$

This is assumed to be a new state augmenting the dynamics and propagated. The value of the closed loop cost function

$$d(\bar{x}, K_i) = \int_t^{t+T} x^T(\tau)(Q + K_i^T R K_i)x(\tau)d\tau \tag{11}$$

is measured by taking two measurements of V which implies

$$d(\bar{x}, K_i) = V(t + T) - V(t) \tag{12}$$

This is where the implementation looked a little tricky. In the paper, it is suggested to use $\bar{p_i}$ given by

$$\bar{p_i} = (XX^T)^{-1}XY \tag{13}$$

where $\bar{p_i}$ is a column vector with diagonal elements as the first two elements and the off diagonal element is given by the third element, X, Y are square matrices of the change in state variables between evaluations and target function measurement respectively.

Equation 13 is obtained by minimising the error in least square sense between the measurement obtained from 12 and $\bar{p_i}^T(x(t) - x(t+T))$. In [1] it is stated that the above mechanism is suitable to solve the equation with **regular presence of excitation**. But in another paper by the same author[2], it is mentioned

> In practice, the matrix inversion is not performed, the solution of the equation being obtained using algorithms that involve techniques such as Gaussian elimination, back-substitution, and Householder reflections.

However, I have tried to implement Recursive least Squares(RLS) to try to estimate $\bar{p_i}$ based on the methodology described above and found reasonable success.

1.2 Simulation

The online simulation has been set up on Simulink as shown:

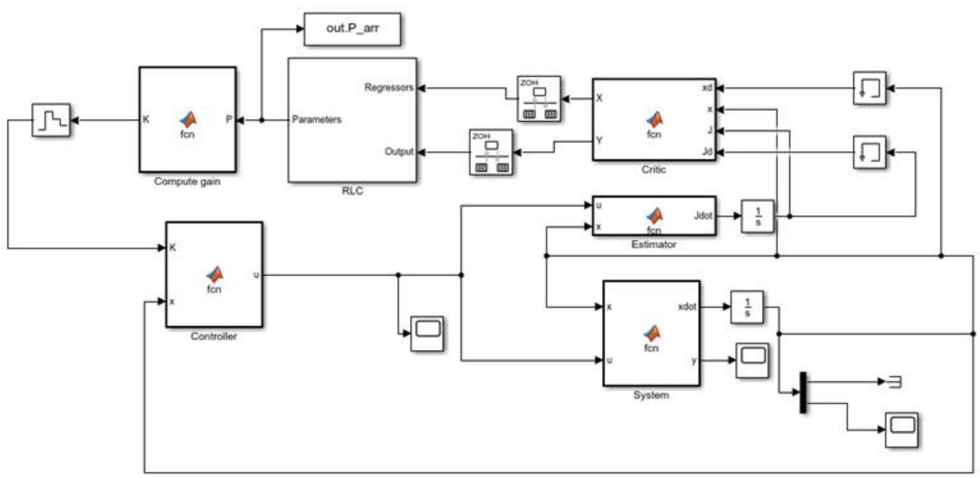

Figure 2: Simulink Implementation

Due to lack of excitation, I think this might not do very well with open loop unstable systems.

On implementing this online without excitation, I have utillised the Recursive Least Squares Estimator(RLS) to estimate my final value of P

The Kronecker product quadratic polynomial vector X defined as

$$X = \begin{bmatrix} \bar{x}_1^2 \\ \bar{x}_1\bar{x}_2 \\ \bar{x}_2\bar{x}_1 \\ \bar{x}_2^2 \end{bmatrix} \tag{14}$$

Here $\bar{x} = x(t) - x(t+T)$ and the vector Y is defined as

$$Y = \begin{bmatrix} V_{11} \\ V_{12} \\ V_{12} \\ V_{22} \end{bmatrix} \tag{15}$$

Here V is the new state introduced in 12

These vectors have been used to estimate $\bar{P}$ using the RLC estimator

The obtained $\bar{P}$ can be used to form the P matrix as

$$P = \begin{bmatrix} \bar{P}_1 & \bar{P}_2/2 \\ \bar{P}_3/2 & \bar{P}_4 \end{bmatrix} \tag{16}$$

This estimate is multiplied by B^T to obtain gain $K = B^T P$ which is fed to the controller to obtain control u = -Kx

1.3 Results

The plots of the states and control are given below.

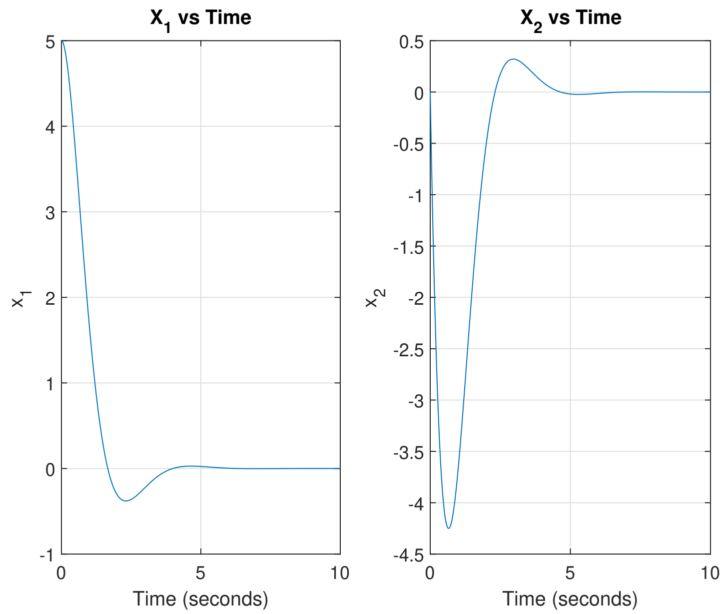

Figure 3: States vs time

Control vs time is given as

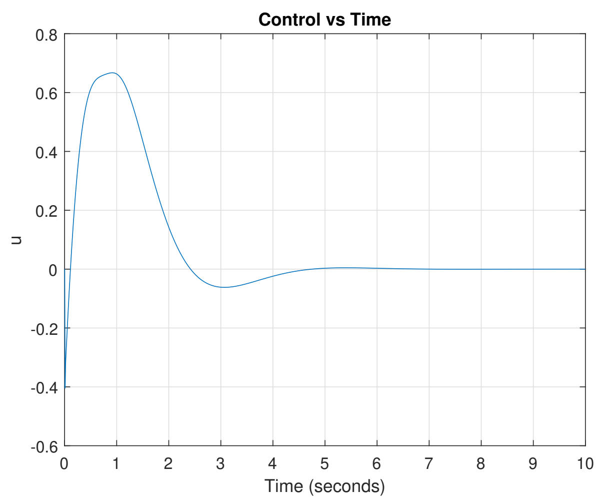

Figure 4: Control vs time

The estimate of the P matrix along with the true value obtained from the ARE is shown below

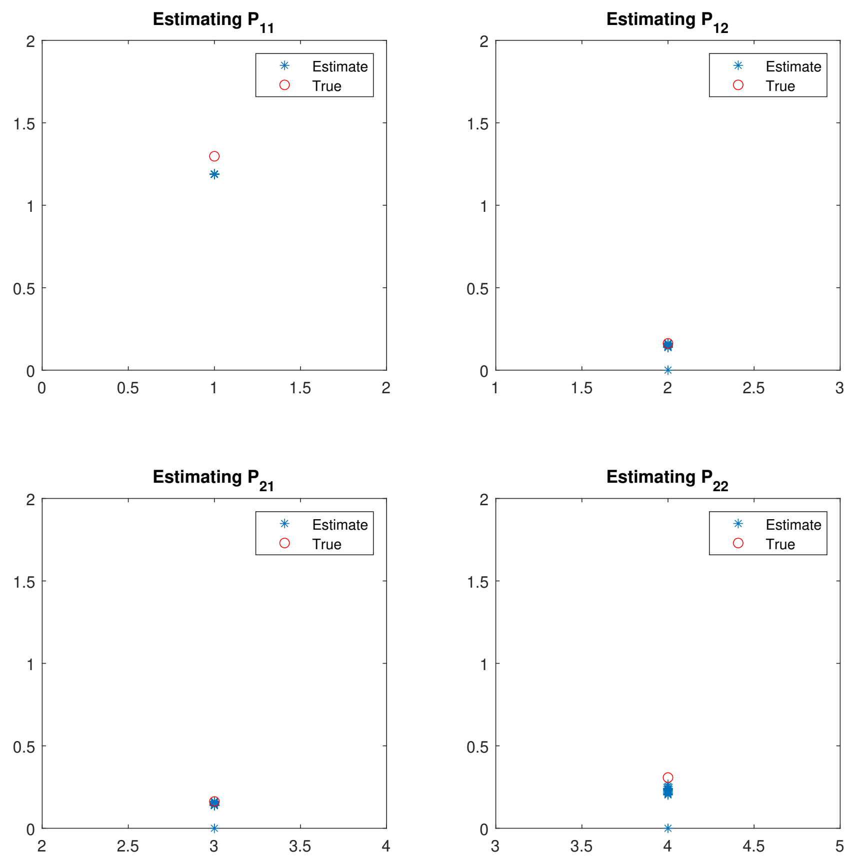

Figure 5: Estimates of P vs real values

We notice a very small error in estimation of P In case this is not clear form above, true value is given as:

$$P = \begin{bmatrix} 1.297 & 0.1623 \\ 0.1623 & 0.3075 \end{bmatrix} \tag{17}$$

And the final P obtained after policy iteration is obtained as

$$P_{est} = \begin{bmatrix} 1.189 & 0.1407 \\ 0.1407 & 0.2408 \end{bmatrix} \tag{18}$$

2 Inverse Optimality in Robust Stabilisation

This section summarises the relevant sections of the paper on inverse optimality by Freeman et al. [3]

An example is used to illustrate the point wise min-norm control law

$$\dot{x} = -x^3 + u + wx \tag{19}$$

The x^3 term here is a beneficial nonlinearity and traditional feedback linearisation would result in wasted control effort. The paper focuses on trying to find a systematic method for choosing the control law that prevents wasteful mistakes.

For system 19, a pointwise min-norm control law is given as

$$u = \begin{cases} x^3 - 2x & x^2 < 2 \\ 0 & x^2 \geq 2 \end{cases} \tag{20}$$

The HJI equation is solved for a cost function

$$J = \int_0^\infty (x^2 + u^2)dt$$

The resulting feedback law is given as

$$u = x^3 - x - x\sqrt{x^4 - 2x^2 + 2} \tag{21}$$

On plotting control laws, Both 20 and 21 utilise the benefit of the nonlinearity and are non positive hence avoiding positive feedback. The pointwise min norm calculation is feasible but the HJI calculation is not.

The proof of optimality of the control law relies on the notion of robust control Lyapunov functions(rclf). A continuous function V is an rclf for the system $\dot{x} = f(x, u, w)$ if there exists a c Such that

$$\inf_{u \in U} \sup_{w \in W} L_f V(x, u, w) < 0 \tag{22}$$

Here, $L_f V = \nabla V.f(x, u, w)$ We notice that the min of the control and the maximum of the noise is considered. The existence of an rclf implies stabilizability.

If V is the rclf of the system given, a fuction m_v may be defined as

$$m_V(x) = \begin{cases} argmin\{||u|| : u \in K_V(x)\} & whenx \in \chi \setminus \Omega_{CV}(V) \\ 0 & whenx \in \Omega_{CV}(V) \end{cases} \tag{23}$$

Here χ is the state space, $\Omega_{CV}(V)$ is a subset of state space such that $V(x) \leq c$, K_V is the intersection of the set of feasible control values and the set L_v which is the set of all values of control 'u' such that the maximum value of noise is less than $-\alpha(x)$

m_V is called the minimal selection for V and is non unique. The existence of m_V and its continuity is proven.

Pointwise min norm have at each point x, their value is the unique element U of the minimum norm that satisfies control constraint and drives the worst case Lyapunov derivative at least as negative as $-\alpha(x)$.

The pointwise min-norm control law at any point is computed by solving the static optimisation problem described in 23 instead of a dynamic programming problem. The solution is simple for **Jointly Affine Systems**.

2.1 Jointly Affine Systems

Assume the system is of the form

$$\dot{x} = f_0(x) + f_1(x)u + f_2(x)w \tag{24}$$

Here all $f_i(x)$ are all considered continuous. let V be the rclf for the system. If we define

$$D_f V(x,u) = \max_{winnoise} L_f X(x,u,w) \tag{25}$$

Based on this expression, for jointly affine systems,

$$D_f V(x,u) = \nabla V(x).f_0(x) + \nabla V(x).f_1(x)u + \|\nabla V(x).f_2(x)\| \tag{26}$$

Choosing a negativity margin $\alpha(x)$ we write

$$D_f V(x,u) + \alpha(x) = \Psi_0(x) + \Psi_1^T(x).u \tag{27}$$

Here $\Psi_0(x) = \nabla V(x).f_0(x) + \|\nabla V(x).f_2(x)\| + \alpha(x)$ and $\Psi_1(x) = [\nabla V(x).f_1(x)]^T$ We set up the set K_v as

$$K_V(x) = \{u : \Psi_0(x) + \Psi_1^T(x).u \leq 0\} \tag{28}$$

Using 23 we notice that the minimum value is given by,

$$m_V(x) = \begin{cases} -\frac{\Psi_0(x)\Psi_1(x)}{\Psi_1^T(x)\Psi_1(x)} & when x \in \chi \setminus \Omega_{CV}(V) and \Psi_0(x) > 0 \\ 0 & Otherwise \end{cases} \tag{29}$$

Applying this to 19 for $V = \frac{1}{2}x^2$ and $\alpha(x) = x^2$, we obtain

$$\Psi_0(x) = -x^4 + 2x^2$$
$$\Psi_1(x) = x$$

Using 29 we get 20

2.2 Inverse optimal robust stabilisation

In this section we will study the optimality of min-norm control law for a meaningful game. Consider the cost function (J)

$$J = \int_0^\infty \{q(x) + r(x,u)\}dt \tag{30}$$

Let the upper value function of the game be defined as

$$\bar{J}(x_0) = \inf_u \sup_w \sup_x J \tag{31}$$

Assume that for the system,

$$\int_0^T (L_f V(x,u,w))dt \geq \int_0^T (D_f V(x,u))dt - \Delta \tag{32}$$

The proof of optimality is through the construction of q and r such that V satisfies the steady state HJI equation given as

$$\min_u \max_w [q(x) + r(x,u) + L_f V(x,u,w)] = 0 \tag{33}$$

For a positive definite function q(x) that satisfies,

$$q(x) + r(x, k^*(x)) + D_f V(x, k^*(x)) = 0 \tag{34}$$

Here k^* is the pointwise min-norm proposed. And r also satisfies the equality

$$\min_u \{r(x,u) + D_f V(x,u)\} = r(x, k^*(x)) + D_f V(x, k^*(x)) \tag{35}$$

Once functions q and r are found, the above equations imply that HJI 33 is satisfied.

2.3 Simulation

A simulation to compare the pointwise min-norm and the HJI solution to system 19 has been implemented. Plots for the control law and states are given as

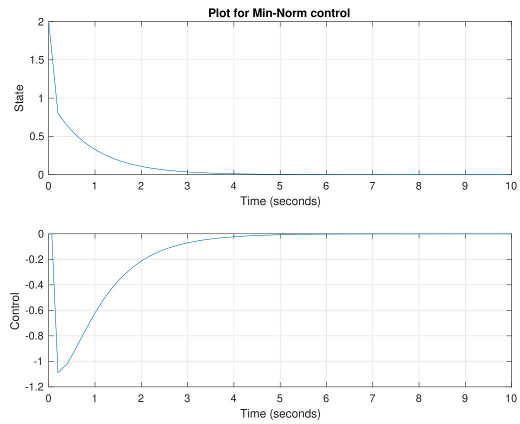

Figure 6: Using Pointwise min-norm

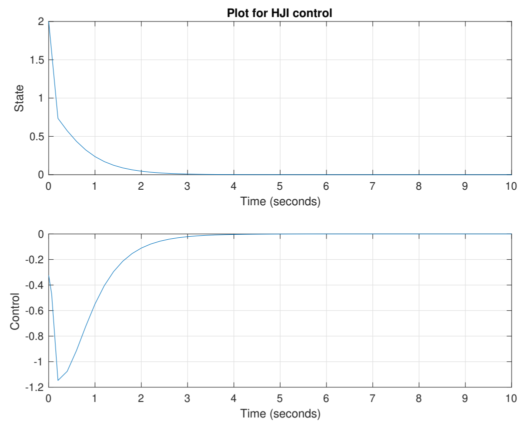

Figure 7: Using HJI control

Both responses are very similar for initial condition x(0) = 2

Appendix

A very short note on RLC:

Given a linear function with measurements and parameter to estimate as

$$y = H^T \theta \tag{36}$$

Given measurement of y and estimate of θ the recursive least squares estimator estimates the value of H.

The process is complicated and the description has been omitted due to lack of time.

References

[1] Vrabie, D., Pastravanu, O., Abu-Khalaf, M. and Lewis, F.L., 2009. Adaptive optimal control for continuous-time linear systems based on policy iteration. Automatica, 45(2), pp.477-484.

[2] Vrabie, D., Pastravanu, O. and Lewis, F.L., 2007, June. Policy iteration for continuous-time systems with unknown internal dynamics. In 2007 Mediterranean Conference on Control & Automation (pp. 1-6). IEEE.

[3] Freeman, R.A. and Kokotovic, P.V., 1996. Inverse optimality in robust stabilization. SIAM journal on control and optimization, 34(4), pp.1365-1391.