LQR Problem

Consider a discrete-time system:

$$x_{t+1} = Ax_t + Bu_t, x_0 = x^{init}$$

problem: choose $u_0, u_1, ...$ so that:

- $x_0, x_1, ...$ is 'small', i.e. we get good *regulation* or *control*
- $u_0, u_1, ...$ is 'small', using small*input effort* or *actuator authority*
- there are usually competing objectives, e.g. a large u can drive x to zero fast
- LQR theory addresses this question

general solution
The continuous LQR problem consists of the system:

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

$$y = Cx$$

and the cost function:

$$J = \frac{1}{2}\int_0^\infty (x^T Qx + u^T Ru)\text{dt}$$

we need to find out the state feedback control law:

$$u = -Kx$$

such that the cost function J is minimized.

Plugging in the control law,

$$\dot{x} = Ax + Bu = (A - BK)x$$

we can easily obtain the solution for $x(t)$, $x(t) = e^{(A-BK)t}x(0)$. Then try to evaluate the cost J,

$$J = \frac{1}{2}\int_0^\infty (x^T Qx + u^T Ru)\text{dt}$$

$$= \frac{1}{2}\int_0^\infty (x^T Qx + x^T K^T RKx)\text{dt}$$

$$= \frac{1}{2}\int_0^\infty x^T (Q + K^T RK)x\text{dt}$$

$$= \frac{1}{2}\int_0^\infty x(0)^T e^{(A-BK)^T t}(Q + K^T RK)e^{(A-BK)t}x(0)\text{dt}$$

we've reached a dead end because we could not evaluate the integral. The solution to LQR is not as simple as the pole placement problem. This time, we could not as well use the controllable canonical form to provide a useful solution.

discrete time algebraic Ricatti equation
this section considers only **linear** discrete-time time-invariant systems

$$x(k+1) = Ax(k) + Bu(k)$$

we define quadratic cost function:

$$J(U) = \sum_{\tau=0}^{N-1} (x_\tau^T Q x_\tau + u_\tau^T R u_\tau) + x_N^T Q_f x_N$$

where $U = (u_0, ..., u_{N-1})$ and $Q = Q^T \geq 0, Q_f = Q_f^T \geq 0, R = R^T > 0$ are given state cost, final state cost, and input cost matrices. Later, we'll see why we require $R > 0$.

- N is called time horizon
- first term measures *state deviation*
- second term measures *input size* or *actuator authority*
- last term measures *final state deviation*

we consider only the problem of regulating the state to the origin, **without state or input constraints**.

toy example
Let us consider a simpler scalar example,

$$J = \frac{1}{2} \int_0^\infty (qx^2 + ru^2)\mathrm{dt}$$

- the factor 1/2 is introduced for numerical convenience.
- the weighing factors q and r express the relative importance of keeping x and u near zero.
- if we place more importance on x, then we select q to be large relative to r. In this case, the state x will converge to 0 faster, but the control effort will be bigger, and energy cost higher.
- If we care more about the energy cost rather than the response speed, then we should set higher r.
- Although we are interested in minimizing J, the actual value of J is usually not of interest. This also means that we can set either q or r to unity for convenience because it is their relative weight that is important.

it turns out that feedback-control law that minimizes J is a linear state feedback law,

$$u = -Kx$$

Assume a simple plant, and $r = 1$

$$\dot{x} = x + u$$

$$J = \frac{1}{2}\int_0^\infty (qx^2 + u^2)\mathrm{dt}$$

plug in the control law, we have the closed loop system:

$$\dot{x} = x - Kx = -(K-1)x$$

and for constant K,

$$x = x(0)e^{-(K-1)t}$$

substituting $x(t)$ into J gives,

$$J = \frac{1}{2}(q + K^2)x^2(0)\int_0^\infty e^{-2(K-1)t}\mathrm{dt} = \frac{q + K^2}{4(K-1)}x^2(0)$$

now we can compute $\frac{dJ}{dK} = 0$
we will have $K^2 - 2K - q = 0$. its roots are $K_1 = 1 + \sqrt{1+q}, K_2 = 1 - \sqrt{1+q}$. To ensure the system to be stable we require $K > 1$. K_1 will satisfy this condition.

Lets have more insights by considering other sample problems.

1. Consider a state space plant:

$$\dot{x}_1 = x_1$$

$$\dot{x}_2 = x_2 + u$$

   the performance index:

$$J = \frac{1}{2}\int_0^\infty x_1^2 + u^2\mathrm{dt}$$

   no minimum for J will exist because the state variable x_1 is uncontrollable and of an unstable mode. The response of x_1 will be $x_1(t) = x_1(0)e^t$ regardless of what u and x_2 do, i.e. $x_1(t) \to \infty$ no matter what $u(t)$ does. Hence, we assume that the system, (A, B), is controllable.

2. Consider a state space plant:

$$\dot{x} = x + u$$

$$J = \frac{1}{2}\int_0^\infty (u^2)\mathrm{dt}$$

   to minimize J, $u(t) = 0$. but then $x \to \infty$. The system is unstable. We need the performance index to know all the state variables. Therefore, we assume that the system pair (A,H) is completely observabl, where H is any matrix such that $H^T H = Q$.

3. Consider the cost functions:

$$J = \frac{1}{2}\int_0^\infty x^2 - u^2 \mathrm{dt}$$

$$J = \frac{1}{2}\int_0^\infty -x^2 + u^2 \mathrm{dt}$$

J maybe made as negative as one wishes (when u or $x \to \infty$) therefore we impose the conditions the weighing matrices Q to be semi-positive definite and R to be positive definite respectively.

4. But why do we require $R > 0$
Consider:

$$\dot{x} = -x + u$$
$$J = \frac{1}{2}\int_0^\infty x^2 \mathrm{dt}$$

has Q=1 and R=0 (not positive definite). In this case, $K \to \infty$. Which implies the input is infinity. $u(t)$ must be finite due to physical limitations.

Recursive approach
using the concept from dynamic programming, the recursive solution relies on Bellman's principle of optimality cf. Bellman's equation in RL,

$$q_*(s,a) = \sum_r \sum_{s'} P(s', r|s,a)[r(s,a,s') + \gamma \max_{a'} q_*(s', a')]$$

. It states, "For any solution for steps j to N with $j \geq 0$, taken from the 0 to N solution, must itself be optimal for the $j - to - N$ problem". Therefore, we have for any $j = 0, ..., N$

$$J_j^*(x_j) = \min_{\mathrm{u_j}} J(x_i, u_i) + J_{j+1}^*(x_{j+1})$$

$$\text{subj. to } x_{j+1} = Ax_j + Bu_j$$

Define the "j-step optimal cost-to-go" as the **optimal** cost attainable for the step j problem:

$$J_j^*(x(j)) = \min_{U_j \to N} x_N^T P x_N + \sum_{i=j}^{N-1} (x_i^T Q x_i + u_i^T R u_i)$$

$$\text{subj. to } x_{i+1} = Ax_i + Bu_i, i = j, ..., N-1$$
$$x_j = x(j)$$

where the final state cost $J(x(j)) = x(j)^T P x(j)$ is an assumed Lyapunov function, $P_{k+1} = P_{k+1}^T \geq 0$. This is the minimum cost attainable for the remainder of the horizon after step j.

- Consider the 1-step problem (solved at time N-1)

$$J^*_{N-1}(x_{N-1}) = \min_{U_{N-1}} x^T_{N-1} Q x_{N-1} + u^T_{N-1} R u_{N-1} + x^T_N P_N x_N \quad (1)$$

$$\text{s.t. } x_N = A x_{N-1} + B u_{N-1} \quad (2)$$
$$P_N = P$$

- Substituting (2) into (1)

$$J^*_{N-1}(x_{N-1}) = \min_{u_{N-1}} \{ x^T_{N-1}(A^T P_N A + Q) x_{N-1}$$
$$u^T_{N-1}(B^T P_N B + R) u_{N-1}$$
$$+ 2 x^T_{N-1} A^T P_N B u_{N-1} \}$$

- Solving again by setting the gradient to zero leads to the following optimality condition for u_{N-1}

$$2(B^T P_N B + R) u_{N-1} + 2 B^T P_N A x_{N-1} = 0$$

Optimal 1-step input:

$$u^*_{N-1} = -(B^T P_N B + R)^{-1} B^T P_N A x_{N-1} = F_{N-1} x_{N-1}$$

1-step cost-to-go:

$$J^*_{N-1}(x_{N-1}) = x^T_{N-1} P_{N-1} x_{N-1} \ ,$$

where

$$P_{N-1} = A^T P_N A + Q - A^T P_N B (B^T P_N B + R)^{-1} B^T P_N A.$$

- We can obtain the solution for any given time step i in the horizon

$$u^*_t = -(B^T P_{t+1} B + R)^{-1} B^T P_{t+1} A x_t$$
$$K_t = -(B^T P_{t+1} B + R)^{-1} B^T P_{t+1} A$$

where we can find any P_t by recursive evaluation from $P_N = P$, using

$$P_t = Q + A^T P_{t+1} A - A^T P_{t+1} B (B^T P_{t+1} B + R)^{-1} B^T P_{t+1} A$$

which can be initialized with $P_N = P$, the given terminal weight. This is called the **Discrete Time Riccati equation** or **Riccati Difference equation (RDE)**

- The optimal cost-to-go is

$$J^*_i(x_i) = x^T_i P_i x_i$$

, evaluating down to P_0, we obtain

$$x(0)^T P_0 x(0)$$

Batch approach
The batch solution explicitl represents all future states x_i in terms of initial condition x_0 and inputs $u_0, ..., u_{N-1}$.

$$x_{t+1} = Ax_t + Bu_t$$

$$x_1 = Ax_0 + Bu_0$$
$$x_2 = A(Ax_0 + Bu_0) + Bu_1$$
$$x_3 = A^3 + A^2Bu_0 + ABu_1 + Bu_2$$
$$\vdots$$
$$x_t = A^t x_0 + A^{t-1}Bu_0 + A^{t-2}Bu_1 + \ldots + ABu_{t-2} + Bu_{t-1}$$

In matrix form,

$$\begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} I \\ A \\ \vdots \\ \vdots \\ A^N \end{bmatrix} x(0) + \begin{bmatrix} 0 & \ldots & \ldots & 0 \\ B & 0 & \ldots & 0 \\ AB & B & \ldots & 0 \\ \vdots & \ddots & \ddots & 0 \\ A^{N-1}B & \ldots & AB & B \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ \vdots \\ u_{N-1} \end{bmatrix}$$

the equation above can be written as

$$X = S^X x(0) + S^U U$$

For the cost function, define

$$\bar{Q} = blockdiag(Q, Q, \ldots, P) \quad \text{and} \quad \bar{R} = blockdiag(R, \ldots, R)$$

Then the finite horizon cost function can be written

$$J(x(0), U) = X^T \bar{Q} X + U^T \bar{R} U$$

substituting X,

$$J(x(0), U) = (S^X x(0) + S^U U)^T \bar{Q}(S^X x(0) + S^U U) + U^T \bar{R} U$$

$$= x(0)^T {S^X}^T \bar{Q} S^X x(0) + x(0)^T {S^X}^T \bar{Q} S^U U + U^T {S^U}^T \bar{Q} S^X x(0) + U^T {S^U}^T S^U U + U^R U$$

$$= U^T({S^U}^T \bar{Q} S^U + \bar{R})U + 2x(0)^T {S^X}^T \bar{Q} S^U U + x(0)^T {S^X}^T \bar{Q} S^X x(0)$$

define, $H = {S^U}^T \bar{Q} S^U + \bar{R}$ and $F = {S^X}^T \bar{Q} S^U$ where $H > 0$, since $R > 0$, and ${S^U}^T \bar{Q} S^U \geq 0$.

Since the problem is unconstraned and $J(x(0), U)$ is a positive definite quadratic function U we can solve for the optimal input U^* by setting the gradient with respect to U to zero:

$$\nabla_U J(x(0), U) = 2({S^U}^T \bar{Q} S^U + \bar{R})U + 2{S^U}^T \bar{Q}^T S^X x(0) = 0$$

$$U^*(x(0)) = -({S^U}^T \bar{Q} S^U + \bar{R})^{-1} (S^U)^T \bar{Q}^T S^X x(0)$$
$$= -H^{-1} F^T x(0)$$

which is a linear function of the initial state $x(0)$.
Note: H^{-1} alwas exists, since $H > 0$ and therefore has full rank. by back-substitution, we can recover the optimal cost:

$$J^*(x(0)) = -x(0)^T F H^{-1} F^T x(0) + x(0)^T {S^X}^T \bar{Q} S^X x(0)$$
$$= x(0)^T \left((S^X)^T \bar{Q} S^X - (S^X)^T \bar{Q} S^U \left((S^U)^T \bar{Q} S^U + \bar{R} \right)^{-1} (S^U)^T \bar{Q} S^X \right) x(0)$$

The batch approach expresses the cost function in terms of the initial state $x(0)$ and input sequence U by eliminating the states x_i.
Note: If there are state or input constraints, solvin this problem by matrix inversion is not guaranteed to result in a feasible input sequence.

references:

1. Stephen Boyd, Linear Dynammical Systems EE363 , Lecture Notes.
2. Melanie Zeilinger, Model Predictive Control, Lecture 2: Unconstrained Linear Quadratic Optiaml Control, ETHZ 2020.