4F3: An Optimised Based Approach to Control

Kazal Oshodi

January 2023

Contents

1 Intro

1.1 Notation

- $\mathbb{R}^n \rightarrow$ n-dimensional Euclidean space/vector space
- $\forall$ = for all
- Positive definite matrix: $x^T Ax > 0$ for all x $\neq 0$
- Positive semi-definite matrix: $x^T Ax \geq 0$
- $f(\cdot) : A \rightarrow B$ is a function mapping each element in $x \in A$ to an element $f(x) \in B$
- $f(\cdot,\cdot) : A \times B \rightarrow C$ is a function mapping an element in $a \in A$ and an element $b \in B$ to produce an element $f(a, b) \in C$

1.2 Mathematical Optimisation Problem

$$\min_x f_0(x)$$
$$s.t\ f_i(x) \leq b_i, i = 1, ..., m$$
$$h_i(x) = 0, i = 1, ..., p$$

Where:

- $x = (x_1, .., x_n)$ is the optimisation variable
- $f_0 : \mathbb{R}^n \rightarrow R$ is the objective function
- $f_i : \mathbb{R}^n \rightarrow R$ is the inequality constraint function
- $h_i : \mathbb{R}^n \rightarrow R$ is the equality constraint function

The optimal solution x^* has the smallest value of f_0 amongst all vectors that satisfy the constraints

1.3 Least Squares

$$\min_x ||Ax - b||_2^2$$

Analytical solution: $x^* = (A^T A)^{-1} A^T b = A^+ b$
Computation time proportional to $n^2 k$ ($A \in \mathbb{R}^{k \times n}$)

1.4 Linear Programming

$$\min_x c^T x$$
$$s.t a_i^T x \leq b_i, i = 1, ..., m$$

No analytical solution

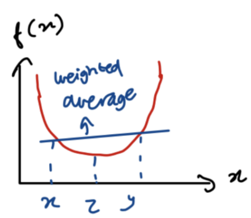
Computation time proportional to $n^2 m$ if m $\geq$ n

1.5 Convex Optimisation

$$\min_x f_0(x)$$
$$\text{s.t } f_i(x) \leq b_i, i = 1, ..., m$$
$$h_i(x) = 0, i = 1, ..., p$$

In this case, both the objective function and inequality constraint functions are convex:

$$f_i(ax + by) \leq af_i(x) + bf_i(y),\ a + b = 1\ a \geq 0, b \geq 0$$



f_i $Z = \alpha x + \beta y$

Special cases are the least-squares + linear programming
There is no analytical solution, and the computational time is roughly proportional to $\max\{n^3, n^2m, F\}$ where F is the cost of evaluating f_i and their first + second derivatives

1.6 Quadratic Programming

Special case of convex optimisation: Assume $P = P^T \geq 0$

$$\begin{aligned}\min_x 0.5x^T Px + q^T x + r \\ s.t\ Gx \leq h \\ Ax = b\end{aligned}$$

1.7 Optimisation in Control

System: $x_{k+1} = f(x_k, u_k),\ y = h(x_k, u_k)$, input u, state x, output y
Example optimal solutions:

- Drive the system optimally:
  - Find the optimal sequence u_k^* - could minimise energy/time
  - Find the optimal trajectory x_k^* - shortest path
- Optimal feedback control
  - Find state-feedback controllers: $u = \sum x$ (complete information) or output feedback controllers $u = \sum y$ (incomplete information) which guarantee optimal closed loop performance/robustness
  - Optimisation assisted control
    * Predictive control (handling constraints)
    * Controllers that learn
    * Computer-assisted control design

2 Optimal Control and Dynamic Programming

2.1 Discrete-time Optimal Control

States and Inputs

- State $x \in X$ e.g $X = \mathbb{R}^n$
- Input $u \in U$ e.g $U = \mathbb{R}^m$

Dynamics

Discrete-time state space system: $x_{k+1} = f(x_k, u_k)$, where $f(\cdot,\cdot) : X \times U \to X$
We assume that the initial condition x_0 is given.

Trajectory

Given x_0 each input sequence $u_0, ..., u_{h-1}$ generates a state sequence $x_0, ..., x_h$ such that $x_{k+1} = f(x_k, u_k)$ for $k = 0, \ldots, h-1$

Finite Horizon Cost Function

$$J(x_0, u_0, ..., u_h - 1) = \sum_{k=0}^{h-1} c(x_k, u_k) + J_h(x_h)$$

Where $\sum_{k=0}^{h-1} c(x_k, u_k)$ is the stage cost and $J_h(x_h)$ is the terminal cost
We then want to find the best input sequence $u_0^*, ..., u_{h-1}^*$ such that:

$$J^*(x_0) = \min_{u_0,...,u_{h-1}} J(x_0, u_0, ..., u_{h-1})$$

Note that: J^* might not be well-defined, and $u_0^*, ..u_{h-1}^*$ might not exist or be non-unique

Bellman's Principle of Optimality

We can truncate the original problem to form:

$$\min_{u_k,...,u_{h-1}} \left(\sum_{i=k}^{h-1} c(x_i, u_i) + J_h(x_h) \right)$$

To solve this, the solution can be defined as $V(\cdot, \cdot) : X \times \{0, ..., h\} \to \mathbb{R}$. Then:

$$V(x, k) \triangleq \min_{u_k,...,u_{h-1}} \left(\sum_{i=k}^{h-1} c(x_i, u_i) + J_h(x_h) \right)$$

In this instance V(x,k) is the value function/cost to go. This is the optimal additional cost from the kth step on.
If we know V(x,k+1) for all x, then we can rewrite V(x,k) as:

$$\begin{aligned}
V(x, k) &= \min_{u_k,...,u_{h-1}} \left(\sum_{i=k}^{h-1} c(x_i, u_i) + J_h(x_h) \right) \\
&= \min_{u_k,...,u_{h-1}} \left(c(x_k, u_k) + \sum_{i=k+1}^{h-1} c(x_i, u_i) + J_h(x_h) \right) \\
&= \min_{u_k} \left(\min_{u_k,...,u_{h-1}} \left(c(x, u_k) + \sum_{i=k+1}^{h-1} c(x_i, u_i) + J_h(x_h) \right) \right) \\
&= \min_{u_k} \left(c(x, u_k) + \min_{u_k,...,u_{h-1}} \left(\sum_{i=k+1}^{h-1} c(x_i, u_i) + J_h(x_h) \right) \right) \\
&= \min_{u_k} \left(c(x, u_k) + V(x_{k+1}, k+1) \right)
\end{aligned}$$

Thus we have recursion to express V(x,k) in terms of V(x,k+1)

Hence we can find the optimal cost and optimal control by solving the dynamic programming equation:

$$V(x, k) = \min_u (c(x, u) + V(f(x, u), k + 1)), \ k = h - 1, h - 2, ...1, 0 \tag{1}$$

With the final condition: $V(x, h) = J_h(x)$

The optimal cost can then be found by:

$$J^*(x_0) = \min_{u_0,...,u_{h-1}} J(x_0, u_0, ..., u_{h-1}) = V(x_0, 0)$$

The optimal input u_k at each step minimises Equation
dp1 for the current value of state x_k. We can also define

$$g(x,k) = \underset{u}{\operatorname{argmin}}(c(x,u) + V(f(x,u), k+1))$$

The optimal control would then be:

$$u_k^* = g(x_k, k) \; k = 0, 1, ..., h-1$$

Note that:

- arg min is the value that achieves the minimum
- This converts the minimisation over a sequence of h inputs to a sequence of h minimisations over 1 input, but all states
- Optimal controls are given by time varying state feedback
- Solution has been computed all for x_0
- If the state and input can only take a finite number of values, then the optimisation can be performed by enumeration

Example

Example

Let $U = \{1, 2, 3\}$ and $X = \{1, 2, 3\}$ and

$$V(x,k) = \min_u (c(x,u) + V(f(x,u), k+1)),$$

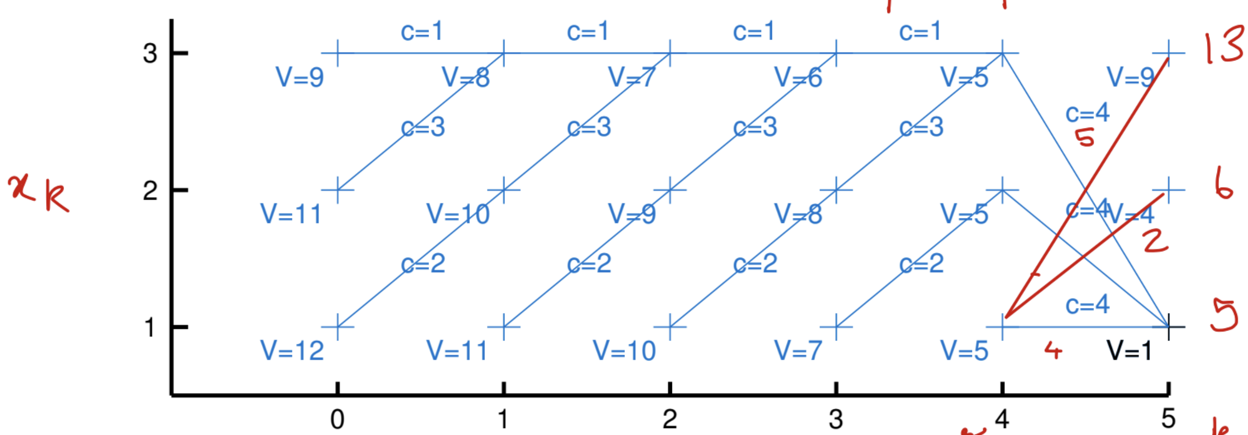
$$x_{k+1} = u_k$$
$$c(x,u) = C_{xu}$$
$$J_h(x) = x^2$$ → terminal cost
$$h = 5$$

$C_{xu} =$

$x \backslash u$	1	2	3
1	4	2	5
2	4	5	3
3	4	2	1

blue is lowest cost path

optimal path



initial conditions

if we were going from k=4 to h=5

Looking at the possible options from k=4 to k=5, where the current state is $x_k = 1$:

- With an input of $u_k = 1$, then $x_{k+1} = 1$, the cost is $C_{11} = 4$, and the total cost is $4 + J_h(1) = 5$
- With an input of $u_k = 2$, then $x_{k+1} = 2$, and the total cost is $C_{12} + J_h(2) = 6$
- With an input of $u_k = 3$, then $x_{k+1} = 3$, the total cost is $C_{13} + J_h(3) = 13$

Thus the optimal path from $x_4 = 1$ to x_5 is with input $u_k = 1$. This is repeated for the other possible paths.

2.2 Discrete-time Linear Quadratic Regulator (LQR)

States and Inputs

State $x \in X = \mathbb{R}^N$, Input $u \in U = \mathbb{R}^m$

Dynamics

$x_{k+1} = Ax_k + Bu_k$, where we assume that the initial condition x_0 is given

Cost Function

We can then define the cost function as:

$$J(x_0, u_0, u_1, ..., u_{h-1}) = \sum_{k=0}^{h-1} \left(x_k^T Q x_k + u_k^T R u_k \right) + x_h^T X_h x_h$$

Where Q,R, X_h are symmetric matrices with Q ≥ 0, R > 0, $X_h \geq 0$. For a matrix to be greater than 0, then $x^T A x > 0$ for all x $\neq 0$, and hence an inverse exists for that matrix.

Lemma 2.1 *(Minimisation for Quadratic Forms)*

For symmetric matrices Q, R, for R > 0, then:

$$\min_u \begin{bmatrix} x^T & u^T \end{bmatrix} \begin{bmatrix} Q & S^T \\ S & R \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} = x^T(Q - S^T R^{-1} S)x$$

This minimum is achieved when:

$$u = -R^{-1} S x$$

We can prove this by completing the square:

$$\begin{aligned}
\begin{bmatrix} x^T & u^T \end{bmatrix} \begin{bmatrix} Q & S^T \\ S & R \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} &= \begin{bmatrix} (x^T Q + u^T S) & (x^T S^T + u^T R) \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \\
&= x^T Q x + u^T S x + x^T S^T u + u^T R u \\
&= (u^T + x^T S^T R^{-1}) R (u + R^{-1} S x) + x^T Q x - x^T S^T R^{-1} S x \\
&\begin{cases} = x^T(Q - S^T R^{-1} S)x & \text{if } u = -R^{-1} S x \\ \geq x^T(Q - S^T R^{-1} S)x & \text{for all u, as the brackets term is positive since R is positive definite} \end{cases}
\end{aligned}$$

This can also be proved by matrix calculus

$$\begin{aligned}
\begin{bmatrix} x^T & u^T \end{bmatrix} \begin{bmatrix} Q & S^T \\ S & R \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} &= \begin{bmatrix} (x^T Q + u^T S) & (x^T S^T + u^T R) \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \\
&= x^T Q x + u^T S x + x^T S^T u + u^T R u
\end{aligned}$$

- There is a stationary point here when $\nabla_u = (x^TQx + u^TSx + x^TS^Tu + u^TRu) = 0$
- At the stationary point $2(Sx + Ru) = 0 \implies u = R^{-1}Sx$
- We can check this is a minimum as: $\nabla_u^2(x^TQx + u^TSx + x^TS^Tu + u^TRu) = 2R > 0$

Thus, the dynamic programming equation (1) can be simplified to:

$$V(x,k) = \min_u \left(\underbrace{x^TQx + u^TRu}_{c(x,u)} + V(\overbrace{Ax + Bu, k+1}^{f(x,u)}) \right)$$

To use this equation, we can look at the penultimate time step h-1, since we know the value function at time step h $\rightarrow$ the terminal cost:

$$\begin{aligned} V(x,h-1) &= \min_u \left(x^TQx + u^TRu + \overbrace{\underbrace{(Ax+Bu)^T}_{X_h^T} X_h \underbrace{(Ax+Bu)}_{X_h}}^{J_x(x_h)} \right) \\ &= \min_u \begin{bmatrix} x^T & u^T \end{bmatrix} \begin{bmatrix} Q + A^TX_hA & A^TX_hB \\ B^TX_hA & R + B^TX_hB \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \end{aligned}$$

This form is similar to (2.1), and thus we can solve this:

$$\begin{aligned} \tilde{Q} &= Q + A^TX_hA, \;\; \tilde{S} = B^TX_hA, \\ \tilde{R} &= R + B^TX_hB, \;\; u = -\tilde{R}^{-1}\tilde{S}x \\ V(x,h-1) &= x^T \underbrace{(Q + A^TX_hA - A^TX_hB(R + B^TX_hB)^{-1}B^TX_hA)}_{X_{h-1}} x \end{aligned}$$

Thus, we can solve the problem recursively:

$$\begin{aligned} V(x,h) &= x^TX_hx \\ V(x,h-1) &= x^TX_{h-1}x \\ &\vdots \\ V(x,0) &= x^TX_0x \end{aligned}$$

Lemma 2.2 Matrix $X_k \geq 0, k = h, h-1, ..., 0$ if $X_h \geq 0, Q \geq 0, R > 0$

Proof:

$$X_{h-1} = Q + A^T(X_h - X_hB(R + B^TX_hB)^{-1}B^TX_h)A$$

For this, it is sufficient to have $Q \geq 0$ and $X_h - X_hB(R + B^TX_hB)^{-1}B^TX_h \geq 0$

Let $M = (B^TX_hB + R)$

$$\begin{bmatrix} I & B^T \end{bmatrix} X_h \begin{bmatrix} I & B \end{bmatrix} \geq 0, \; \begin{bmatrix} 0 & 0 \\ 0 & R \end{bmatrix} \geq 0 \implies \begin{bmatrix} X_h & X_hB \\ B^TX_h & (B^TX_hB + R) \end{bmatrix} \geq 0$$

$$\begin{bmatrix} I & -X_hBM^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} X_h & X_hB \\ B^TX_h & M \end{bmatrix} \begin{bmatrix} I & 0 \\ -M^{-1}B^TX_h & I \end{bmatrix} \geq 0$$

$$\begin{bmatrix} X_h - X_hBM^{-1}B^TX_h & 0 \\ 0 & M \end{bmatrix} \geq 0$$

As this is a diagonal element, then the individual elements must be greater than 0, so the lemma is proved

Overall, to solve the LQR, we need to solve the backwards difference equation:

$$X_{k-1} = Q + A^T X_k A - A^T X_k B(R + B^T X_k B)^{-1} B^T X_k A$$

This has an optimal cost of:

$$x_0^T X_0 x_0$$

when there is a state-feedback control of:

$$u_k = -(R + B^T X_{k+1} B)^{-1} B^T X_{k+1} A x_k$$

2.3 Continuous-Time Dynamic Programming

States and Inputs

$x \in \mathbb{R}^n$, $u \in U \subseteq \mathbb{R}^m$, e.g $U = [-1, 1]$

Dynamics

In this case, it is a continuous-time state space system:

$$\dot{x} = f(x, u),\ f(\cdot, \cdot) : \mathbb{R}^n \times U \to \mathbb{R}^n$$

For example, $\dot{x} = sin(x) + u$

Trajectory

With a initial state $x_0 \in X$, and a horizon $T \geq 0$, then each input function $u(\cdot) : [0, T] \to U$ will produce a state trajectory $x(\cdot) : [0, T] \to \mathbb{R}^n$, where $x(0) = x_0,\ \dot{x}(t) = f(x(t), u(t))$

Cost Function

$$J(x_0, u(\cdot)) = \int_0^T c(x(t), u(t))dt + J_T(x(T))$$

Objective

We want to find the best input function $u^*(\cdot) : [0, T] \to U$ which minimises the cost function:

$$J^*(x_0) = J(x_0, u^*(\cdot)) = \min_{u(\cdot)} J(x_0, u(\cdot))$$

Note that we assume that a unique trajectory exists for each state, and that we can find the minimum cost and input for our problem

Bellman's Principle of Optimality

Again, Bellman's principle can apply here. If we assume that the optimal control $u^*(\cdot) : [0, T] \to U$ gets us from $x(0) = x_0$ to $x(t)$ at time t < T. Then again, the problem can be truncated to show that $u^*(\cdot) : [t, T] \to U$ is a solution to:

$$\min_{u(\cdot)} \int_t^T c(x(\tau), u(\tau))d\tau + J_T(x(T))$$

Solution

We can define a value function/cost to go again $V : X \times [0, T] \to \mathbb{R}$:

$$V(x(t), t) \triangleq \min_{u(\cdot)} \int_t^T c(x(\tau), u(\tau))d\tau + J_T(x(T))$$

In this case, we know that $V(x(T), T) = J_T(x(T))$ is the final cost, and $V(x(0), 0) = J^*(x(0))$ is the optimal cost from the initial condition $x(0)$.
To find the optimal solution, consider an infinitesimal time later h. Then Bellman's Equation becomes:

$$\begin{aligned} V(x(t), t) &= \min_{u(\cdot)} \int_t^{t+h} c(x(\tau), u(\tau))d\tau + \int_{t+h}^T c(x(\tau), u(\tau))d\tau + J_T(x(T)) \\ &= \min_{u(\cdot)} \int_t^{t+h} c(x(\tau), u(\tau))d\tau + V(x(t+h), t+h) \end{aligned}$$

From here, we can consider sampling at every h time step. Thus:

$$x(t+h) = x(t) + \underbrace{f(x(t), u(t))}_{\dot{x}} h + \mathcal{O}(h^2)$$

The infinitesimal cost between t and t+h is then:

$$\int_t^{t+h} c(x(\tau), u(\tau))d\tau = c(x(t), u(t))h + \mathcal{O}(h^2)$$

Hence, the dynamic programming equation can be approximated as:

$$V(x, t) = \min_{u \in U}(c(x, u)h + V(x + f(x, u)h, t + h)) + \mathcal{O}(h^2)$$

With a terminal cost of $V(x, T) = J_T(x)$.
To solve this, we can consider the Taylor series expansion:

$$V(x + \delta x, t + \delta t) = V(x, t) + \frac{\partial V}{\partial x}\delta x + \frac{\partial V}{\partial t}\delta t + \mathcal{O}(\delta x^2, \delta t^2)$$

Subbing this back into the earlier equation produces:

$$\begin{aligned} V(x, t) &= \min_{u \in U}(c(x, u)h + V(x + f(x, u)h, t + h)) + \mathcal{O}(h^2) \\ &= \min_{u \in U}\left(c(x, u)h + V(x, t) + \frac{\partial V(x, t)}{\partial x} f(x, u)h + \frac{\partial V(x, t)}{\partial t} h\right) + \mathcal{O}(h^2) \\ V(x, t) - V(x, t) - \frac{\partial V(x, t)}{\partial x} h &= \min_{u \in U}\left(c(x, u)h + \frac{\partial V(x, t)}{\partial x} f(x, u)h\right) + \mathcal{O}(h^2) \\ -\frac{\partial V(x, t)}{\partial x} &= \min_{u \in U}\left(c(x, u) + \frac{\partial V(x, t)}{\partial x} f(x, u)\right) + \frac{\mathcal{O}(h^2)}{h} \end{aligned}$$

Take the limit as $h \to 0$:

$$-\frac{\partial V(x, t)}{\partial x} = \min_{u \in U}\left(c(x, u) + \frac{\partial V(x, t)}{\partial x} f(x, u)\right) \tag{2}$$

This PDE can be used to find the value function, with boundary condition $V(x,T) = J_T(x)$. The optimal cost is $V(x_0, 0)$ and the optimal input is:

$$u^*(t) = g(x(t), t)$$
$$g(x,t) = \underset{u \in U}{\operatorname{argmin}} \left(c(x,u) + \frac{\partial V(x,t)}{\partial x} f(x,u) \right)$$

This equation (2) is known as the Hamilton-Jacobi-Bellman PDE. It is the infinitesimal version of (1). It is much simpler to solve, as it turns the optimisation over $u(\cdot)$ which is a continuous function with infinite points as continuous time, into a pointwise optimisation over $u \in U$.
Note that in some cases, a solution might not exist, and even if it does exist it might not be able to be computed or make sense physically.

2.4 Continuous-Time Linear Quadratic Regulator

States and Inputs

States $x \in \mathbb{R}^n$, Inputs $u \in \mathbb{R}^m$

Plant

$\dot{x} = Ax + Bu,\ x(0) = x_0$

Cost Function

$$\begin{aligned}
J(x_0, u(\cdot)) &= \int_0^{t_1} c(x,u)dt + J_{t_1}(x(T)) \\
c(x,u) &= x^T Q x + u^T R u \\
R &= R^T > 0 \\
X_{t_1} &= X_{t_1}^T \geq 0 \\
J_{t_1}(x) &= x^T X_{t_1} x \\
Q &= Q^T \geq 0
\end{aligned}$$

To solve this, we use the HJB Equation (2):

$$-\frac{\partial V(x,t)}{\partial x} = \min_{u \in U} \left(c(x,u) + \frac{\partial V(x,t)}{\partial x} f(x,u) \right)$$

Subbing in for the $c(x,u), f(x,u)$ shows that:

$$\begin{aligned}
c(x,u) = x^T Q x + u^T R u, f(x,u) &= Ax + Bu \\
\text{Guess a solution: } V(x,t) &= x^T X(t) x \\
-x^T \dot{X}(t) x &= \min_{u \in \mathbb{R}^m} \left(x^T Q x + u^T R u + 2x^T X(t)(Ax + Bu) \right) \\
&= \min_{u \in \mathbb{R}^m} \left(x^T (Q + XA + A^T X) + u^T R u + x^T X B u + u^T B^T X x \right) \\
&= x^T (Q + XA + A^T X - XBR^{-1}B^T X) x \\
u^*(t) &= -R^{-1} B^T X(t) x(t)
\end{aligned}$$

Essentially, this is solving the ODE (Riccati Equation):

$$-\dot{X} = Q + XA + A^T X - XBR^{-1}B^T X$$
$$X(t_1) = X_{t_1}$$

The optimal cost is given by $x_0^T X(0)x_0$, and the optimal input is $u(t) = -R^{-1}B^T X(t)x(t)$.
To do the backwards numerical integration, we can use that:

$$\dot{X}(t) \simeq \frac{X(t) - X(t-\Delta t)}{\Delta t}$$
$$X(t-\Delta t) \simeq X(t) - \dot{X}(t)\Delta t$$
$$\simeq X(t) + (Q + X(t)A + A^T X(t) - X(t)BR^{-1}B^T X(t))\Delta t$$

Example: Minimum energy input to reach a state

Consider a cost:

$$J(\tilde{x}(0), u(\cdot)) = \int_0^{t_1} \tilde{u}(t)^T \tilde{u}(t)dt + \frac{1}{\epsilon}\tilde{x}(t_1)^T \tilde{x}(t_1)$$

Where $\tilde{x} = x_1 - x$ and $\tilde{u} = u_1 - u$. Essentially x_1, u_1 is a fixed point, which could be equilibrium i.e $Ax_1 + Bu_1 = 0$.
In this case $c(\tilde{x}, \tilde{u}) = \tilde{u}^T \tilde{u}$, Q = 0, R = I.
To do this, we solve the Riccati Equation:

$$\dot{X} = Q + XA + A^T X - XBR^{-1}B^T X = XA + A^T X - XBB^T X$$

Where the terminal condition X(T) = $\frac{1}{\epsilon}$ I. The optimal input would again be:

$$\tilde{u}^*(t) = -B^T X(t)\tilde{x}(t)$$

As $\epsilon \to 0$ and $\tilde{x}(0) = x_1, u = \tilde{u}^*(\cdot) + u_1$. u is the minimum energy input to drive the state from $x(0) = 0$ to $x(t_1) = x_1$.
Consider again the Riccati Equation:

$$-\dot{X} = XA + A^T X - XBB^T X$$

To solve this, we can use the substitution $Y = -X^{-1}$.
Note that: $\frac{d}{dt}Y^{-1} = -Y^{-1}\dot{Y}Y^{-1}$. Then we get that:

$$-Y^{-1}\dot{Y}Y^{-1} = -Y^{-1}A - A^T Y^{-1} - Y^{-1}BB^T Y^{-1}$$

The optimal control in this case would satisfy:

$$u^*(t) = B^T Y^{-1}(t)\tilde{x}(t) \quad (\tilde{x}(0) = x_1)$$

where:

$$\dot{Y} = AY + YA^T + BB^T$$

and $Y(t_1) = \epsilon I$ (this is the Lyapunov equation - easier to solve + integrate4)

2.5 Infinite Horizon Linear Quadratic Regulator - continuous case

Plant

$\dot{x} = Ax + Bu$, $x(0) = x_0$, $z = \begin{bmatrix} Cx \\ u \end{bmatrix}$

Cost Function

$$J(x_0, u(\cdot)) = \int_0^\infty z(t)^T z(t) dt$$
$$= \int_0^\infty \left(\underbrace{x(t)^T C^T C x(t)}_{\text{output energy}} + \underbrace{u(t)^T R u(t)}_{\text{input energy, R = I}} \right) dt$$

Assumptions

(A,B) controllable, (A,C) observable

Solution

We can assume that an infinite horizon is a finite but very long horizon

$$\lim_{t_1 \to \infty} \int_0^{t_1} x(t)^T C^T C x(t) + u(t) R u(t) dt$$

Therefore, we can find the solution from the Riccati equation:

$$-\dot{X}(t) = C^T C + X(t)A + A^T X(t) - X(t)BRB^T X(t)$$

where there can be any final condition $X(t_1) = X^T(t_1) > 0$

Solution

We can guess that the solution will be of the form $u(t) = -B^T X x(t)$, where $X = X^T$ solves the Control Algebraic Riccati Equation (CARE):

$$0 = C^T C + XA + ATX - XBB^T X \tag{3}$$

The closed-loop dynamics would be controlled by:

$$\dot{x} = Ax + Bu = (A - BB^T X)x$$

and we would want $(A - BB^T X)$ is stable (all eigenvalues in left-half plane)

Fact

Based on the earlier assumptions, the CARE (3) has a unique, symmetric, positive,definite solution $X = X^T > 0$, which is stabilising ($(A - BB^T X)$ is stable). We can also obtain this solution from $\lim_{t \to -\infty} X(t)$, where $X(t)$ solves:

$$-\dot{X}(t) = C^T C + X(t)A + A^T X(t) - X(t)BB^T X(t)$$

This is essentially solving backwards in time. This case applies for any final condition $X(T) = X^T(T) > 0$

Summary

Let $X = X^T$ be the stabilising solution to CARE (3). Then the optimal control is given by $u(t) = -B^T X x(t)$, and the optimal cost is $x(0)^T X x(0)$

2.6 Infinite Horizon Linear Quadratic Regulator - Discrete Case

Plant

$x_{k+1} = Ax_k + Bu_k,\ x(0) = x_0,\ z = \begin{bmatrix} Cx \\ u \end{bmatrix}$

Cost Function

$$J(x_0, u_0, u_1, ...) = \sum_{k=0}^{\infty} x_k^T Q x_k + u_k^T R u_k$$

Assumptions

(A,B) controllable, (A,Q) observable

Solution

Again, think of an infinite horizon as a very long finite horizon. Using previous results, the optimal state-feedback control is:

$$u_k = -(R + B^T X B)^{-1} B^T X A x_k$$

Where $X = X^T$ solves the Discrete Algebraic Riccati Equation

$$X = Q + A^T X A - A^T X B(R + B^T X B)^{-1} B^T X A \tag{4}$$

for $X = X^T > 0$. The optimal cost is:

$$x_0^T X x_0$$

3 $\mathcal{H}_2$ norm

3.1 The $\mathcal{H}_2$ norm

Consider the stable linear system:

$$\dot{x} = Ax + Bu,\ y = Cx$$

Where A has all its eigenvalues in the left half plane. The system would then have a transfer function:

$$G(s) = C(sI - A)^{-1} B$$

Then the $\mathcal{H}_2$ norm is defined as:

$$||G||_2^2 = \int_{-\infty}^{\infty} \text{trace}\{\bar{G(j\omega)}^T G(j\omega)\} d\omega$$

$$||G||_2^2 = \sum_i ||G_i||_2^2$$

Where G_i is the transfer function from the ith input to the output i.e $G_i : T_{ui} \to y$.
Therefore, you can show that:

$$||y||_\infty \leq \frac{1}{\sqrt{2\pi}} ||G||_2 ||u||_2$$

$$||y||_\infty = \sup_t \sqrt{y^T(t) y(t)}$$

$$||u||_2 = \sqrt{\int_{-\infty}^{\infty} u^T(t) u(t) dt}$$

Where sup is the smallest upper bound of a set. The aim of $\mathcal{H}_2$ optimal control is to minimise the $\mathcal{H}_2$ norm of a closed-loop transfer function.
Let the impulse response of G(s) be g(t). As $G(s) = C(sI - A)^{-1}B$, and $g(t) = \mathcal{L}^{-1}G(s) = Ce^{At}B$.
From Parseval's Theorem, we know that:

$$\frac{1}{\sqrt{2\pi}}||G(s)||_2 = ||g(t)||_2,\ ||g(t)||_2^2 = \sum_i ||g_i(t)||_2^2$$

$g_i(t)$ is the response to an impulse on the ith input with $x(0^-) = 0$.
Hence, $g_i(t) = 0$ for t < 0, but for $t \geq 0$ it is equal to the response of the system where:

$$x(0^+) = B_i$$

the ith column of B. This works if we compare the initial response: $Ce^{At}x_0$, to the general response: $Ce^{At}B$, under input u=0.
Then the problem is now computing the response of the system under u=0 starting at the appropriate initial conditions $x(0) = x_0$.
Consider the function $V(t) = x(t)^T Lx(t), L = L^T$. If u $= 0$, then:

$$\begin{aligned}
\dot{V}(t) + y(t)^T y(t) &= \frac{d}{dt}(x(t)^T Lx(t)) + y(t)^T y(t) \\
&= (Ax(t))^T Lx(t) + x(t)^T LAx(t) + x(t)^T C^T Cx(t) \\
&= x(t)^T (A^T L + LA + C^T C)x(t)
\end{aligned}$$

If we choose $L = L^T > 0$ to be the solution of the Lyapunov Equation:

$$A^T L + LA + C^T C = 0 \tag{5}$$

i.e L is the observability gramian, then the equation simplifies to:

$$\dot{V}(t) + y(t)^T y(t) = 0$$

If we integrate from 0 to ∞ then:

$$[V(t)]_0^\infty + ||y||_2^2 = 0$$

Since A is stable, and u$=0$ then:

$$\lim_{t\to\infty} x(t) = 0$$

Therefore:

$$\lim_{t\to\infty} V(t) = \lim_{t\to\infty} x^T(t)Lx(t) = 0$$

We also know that $V(0) = x_0^T Lx_0$, so:

$$||y||_2^2 = x_0^T Lx_0$$

So the response is also bounded. To find the impulse response for each input (i.e set $x_0 = B_i$:

$$\begin{aligned}
||\tilde{y}_i||_2^2 &= B_i^T LB_i \\
\sum_i ||\tilde{y}_i(t)||_2^2 &= \sum_i (B_i^T LB_i) = \text{trace}(B^T LB)
\end{aligned}$$

Summary

$$\frac{1}{\sqrt{2\pi}}||G(s)||_2 = \sqrt{\text{trace}(B^T L B)},\ L = L^T$$

This solves the Lyapunov equation (5). We also know that:

$$\frac{1}{2\pi}||T_{u\to y}||_2^2 = \sum_i ||y(t)|_{u(t)=e_i\delta(t)}||_2^2$$

and it can be shown that:

$$||y||_\infty = \sup_t \sqrt{y(t)^T y(t)} \leq \frac{1}{\sqrt{2\pi}}||T_{u\to y}||_2||u||_2$$