Robotics

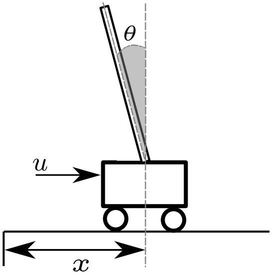
Control Theory

optimal control, HJB equation, infinite horizon case, Linear-Quadratic optimal control, Riccati equations (differential, algebraic, discrete-time), controllability, stability, eigenvalue analysis, Lyapunov function

University of Stuttgart
Winter 2019/20

Lecturer: Duy Nguyen-Tuong
Bosch Center for AI

Cart pole example



state $\boldsymbol{x} = (x, \dot{x}, \theta, \dot{\theta})$

$$\ddot{\theta} = \frac{g\sin(\theta) + \cos(\theta)\left[-c_1 u - c_2\dot{\theta}^2\sin(\theta)\right]}{\frac{4}{3}l - c_2\cos^2(\theta)}$$

$$\ddot{x} = c_1 u + c_2\left[\dot{\theta}^2\sin(\theta) - \ddot{\theta}\cos(\theta)\right]$$

Control Theory

- Concerns controlled systems of the form

$$\dot{x} = f(x, u) + \text{noise}(x, u)$$

and a controller of the form

$$\pi : (x, t) \mapsto u$$

- We'll neglect stochasticity here

- When analyzing a given controller π, one analyzes **closed-loop system** as described by the differential equation

$$\dot{x} = f(x, \pi(x, t))$$

(E.g., analysis for convergence & stability)

Core topics in Control Theory

- **Stability***
  Analyze the stability of a closed-loop system
  $\rightarrow$ Eigenvalue analysis or Lyapunov function method
- **Controllability***
  Analyze which dimensions (DoFs) of the system can actually in principle be controlled
- **Transfer function**
  Analyze the closed-loop transfer function, i.e., "how frequencies are transmitted through the system". ($\rightarrow$ Laplace transformation)
- **Controller design**
  Find a controller with desired stability and/or transfer function properties
- **Optimal control***
  Define a cost function on the system behavior. Optimize a controller to minimize costs

Control Theory references

- Robert F. Stengel: *Optimal control and estimation*
  Online lectures:
  `http://www.princeton.edu/~stengel/MAE546Lectures.html` (esp. lectures 3,4 and 7-9)

- From robotics lectures:
  Stefan Schaal's lecture Introduction to Robotics: `http://www-clmc.usc.edu/Teaching/TeachingIntroductionToRoboticsSyllabus`
  Drew Bagnell's lecture on Adaptive Control and Reinforcement Learning `http://robotwhisperer.org/acrls11/`
  Jonas Buchli's lecture on Optimal & Learning Control for Autonomous Robots `http://www.adrl.ethz.ch/doku.php/adrl:education:lecture:fs2015`

Outline

- We'll first consider *optimal control*
  Goal: understand Algebraic Riccati equation
  significance for local neighborhood control

- Then controllability & stability

Optimal control

Optimal control (discrete time)

Given a controlled dynamic system

$$x_{t+1} = f(x_t, u_t)$$

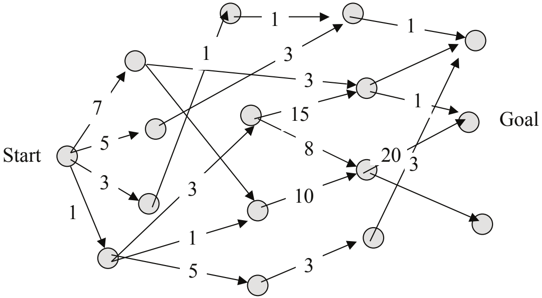
We define a cost function

$$J^\pi = \sum_{t=0}^{T} c(x_t, u_t) + \phi(x_T)$$

Find a policy $\pi = [\pi_0, \cdots, \pi_T]$ that minimizes the defined cost function. With x_0 and the state-feedback policy $u_t = \pi_t(x_t)$, the sequences $x_{1:T}$ and $u_{0:T}$ are determined.

Dynamic Programming & Bellman principle

An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.



$$\text{“}V(\text{state}) = \min_{\text{edge}}[c(\text{edge}) + V(\text{next-state})]\text{”}$$

Bellman equation (discrete time)

Define the **value function** or optimal **cost-to-go function**

$$V_t(x) = \min_{\pi} \Big[\sum_{s=t}^{T} c(x_s, u_s) + \phi(x_T) \Big]_{x_t = x}$$

- Minimization is over policy
- $V_t(x)$ is the cost-to-go, when using the optimal policy with $x_t = x$

Bellman equation (discrete time)

$V_t(x)$ is also called the Bellman value function

$$V_t(x) = \min_u \left[c(x,u) + V_{t+1}(f(x,u))\right]$$

Derivation:

$$\begin{aligned} V_t(x) &= \min_\pi \Big[\sum_{s=t}^{T} c(x_s,u_s) + \phi(x_T)\Big] \\ &= \min_{u_t} \Big[c(x,u_t) + \min_\pi [\sum_{s=t+1}^{T} c(x_s,u_s) + \phi(x_T)]\Big] \\ &= \min_{u_t} \Big[c(x,u_t) + V_{t+1}(f(x,u_t))\Big] \end{aligned}$$

- When x_t is known, we can minimize over action u_t and remaining policies $\pi = [\pi_{t+1}, \cdots, \pi_T]$

Bellman equation (discrete time)

$V_t(x)$ is also called the Bellman value function

$$V_t(x) = \min_u \left[c(x,u) + V_{t+1}(f(x,u))\right]$$

Derivation:

$$\begin{aligned} V_t(x) &= \min_\pi \Big[\sum_{s=t}^{T} c(x_s,u_s) + \phi(x_T)\Big] \\ &= \min_{u_t} \Big[c(x,u_t) + \min_\pi [\sum_{s=t+1}^{T} c(x_s,u_s) + \phi(x_T)]\Big] \\ &= \min_{u_t} \Big[c(x,u_t) + V_{t+1}(f(x,u_t))\Big] \end{aligned}$$

- When x_t is known, we can minimize over action u_t and remaining policies $\pi = [\pi_{t+1}, \cdots, \pi_T]$
- The argmin gives the optimal control signal:

$$u_t^* = \pi_t^*(x) = \operatorname*{argmin}_{u_t} \Big[c(x,u_t) + V_{t+1}(f(x,u_t))\Big]$$

$c(x,u_t)$ is the current cost and $V_{t+1}(f(x,u_t))$ is the future cost you have to pay when following the optimal policy

Solving the Bellman equation

- Solving the Bellman equation is not straightforward, as we are interested in finding a function $V(x)$ that satisfies that equation
- Many solution methods available*, e.g. iteration over the value function, guess and verify, approximate solution, etc.
- The need of discretization of the state space (i.e. the curse of dimensionality)
- The dynamics of the system has to be known

- (*) Literature:
  D.P. Bertsekas. Dynamic Programming and Optimal Control

  R. Sutton. Reinforcement Learning: An Introduction

Optimal Control (continuous time)

Given a controlled dynamic system

$$\dot{x} = f(x, u)$$

we define a cost function with horizon T

$$J^\pi = \int_0^T c(x(t), u(t))\, dt + \phi(x(T))$$

where the start state $x(0)$ and the controller $\pi : (x, t) \mapsto u$ are given, which determine the closed-loop system trajectory $x(t), u(t)$ via $\dot{x} = f(x, \pi(x, t))$ and $u(t) = \pi(x(t), t)$

Hamilton-Jacobi-Bellman equation (continuous time)

- Define the **value function** or optimal **cost-to-go function**

$$V(x,t) = \min_\pi \Big[\int_t^T c(x(s),u(s))\ ds + \phi(x(T)) \Big]_{x(t)=x}$$

- Hamilton-Jacobi-Bellman equation

$$-\tfrac{\partial}{\partial t} V(x,t) = \min_u \Big[c(x,u) + \tfrac{\partial V}{\partial x} f(x,u) \Big]$$

The argmin gives the optimal control signal: $\pi^*(x) = \mathrm{argmin}_u \Big[\cdots \Big]$

Derivation: Apply the discrete-time Bellman equation for V_t and V_{t+dt}:

$$\begin{aligned}
V(x,t) &= \min_u \Big[\int_t^{t+dt} c(x,u)\ dt + V(x(t+dt), t+dt) \Big] \\
&= \min_u \Big[\int_t^{t+dt} c(x,u)\ dt + V(x,t) + \int_t^{t+dt} \frac{dV(x,t)}{dt}\ dt \Big] \\
0 &= \min_u \Big[\int_t^{t+dt} c(x,u) + \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} \dot x\ dt \Big] \\
0 &= \min_u \Big[c(x,u) + \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(x,u) \Big]
\end{aligned}$$

Infinite horizon case

$$J^\pi = \int_0^\infty c(x(t), u(t))\; dt$$

- This cost function is stationary (time-invariant)!
  $\to$ the optimal value function is stationary ($V(x,t) = V(x)$)
  $\to$ the optimal control signal depends on x but not on t
  $\to$ the optimal controller π^* is stationary

- The HJB and Bellman equations remain "the same" but with the same (stationary) value function independent of t:

$$0 = \min_u \Big[c(x,u) + \frac{\partial V}{\partial x} f(x,u) \Big] \qquad \text{(cont. time)}$$

$$V(x) = \min_u \Big[c(x,u) + V(f(x,u)) \Big] \qquad \text{(discrete time)}$$

  The argmin gives the optimal control signal: $\pi^*(x) = \operatorname{argmin}_u \Big[\cdots \Big]$

Infinite horizon examples

- Cart-pole balancing:
  – You always want the pole to be upright ($\theta \approx 0$)
  – You always want the car to be close to zero ($x \approx 0$)
  – You want to spare energy (apply low torques) ($u \approx 0$)
  You might define a cost

$$J^\pi = \int_0^\infty \|\theta\|^2 + \epsilon\|x\|^2 + \rho\|u\|^2$$

- Reference following:
  – You always want to stay close to a reference trajectory $r(t)$
  Define $\tilde{x}(t) = x(t) - r(t)$ with dynamics $\dot{\tilde{x}}(t) = f(\tilde{x}(t) + r(t), u) - \dot{r}(t)$
  Define a cost

$$J^\pi = \int_0^\infty \|\tilde{x}\|^2 + \rho\|u\|^2$$

- Many many problems in control can be framed this way

Comments

- The Bellman equation is fundamental in optimal control theory, but also Reinforcement Learning
- The HJB eq. is a differential equation for $V(x,t)$ which is in general hard to solve
- The (time-discretized) Bellman equation can be solved by Dynamic Programming starting backward:

$$V_T(x) = \phi(x) \;, \quad V_{T\text{-}1}(x) = \min_u \Big[c(x,u) + V_T(f(x,u)) \Big] \qquad \text{etc.}$$

  But it might still be hard or infeasible to represent the functions $V_t(x)$ over continuous x!

- Both become significantly simpler under linear dynamics and quadratic costs:

  $\to$ Riccati equation

Linear-Quadratic Optimal Control

linear dynamics

$$\dot{x} = f(x, u) = Ax + Bu$$

quadratic costs

$$c(x, u) = x^\top Q x + u^\top R u \;, \quad \phi(x_T) = x_T^\top F x_T$$

- Note: Dynamics neglects constant term; costs neglect linear and constant terms. This is because
  - – constant costs are irrelevant
  - – linear cost terms can be made away by redefining x or u
  - – constant dynamic term only introduces a constant drift

Linear-Quadratic Control as Local Approximation

- LQ control is important also to control non-LQ systems in the **neighborhood** of a desired state!

  Let x^* be such a desired state (e.g., cart-pole: $x^* = (0, 0, 0, 0)$)
  – linearize the dynamics around x^*
  – use 2nd order approximation of the costs around x^*
  – control the system *locally* as if it was LQ
  – pray that the system will never leave this neighborhood!

Riccati differential equation = HJB eq. in LQ case

- In the Linear-Quadratic (LQ) case, the value function always is a quadratic function of x!
- Let $V(x,t) = x^\top P(t) x$, then the HJB equation becomes:

$$
\begin{aligned}
-\frac{\partial}{\partial t} V(x,t) &= \min_u \left[c(x,u) + \frac{\partial V}{\partial x} f(x,u) \right] \\
-x^\top \dot{P}(t) x &= \min_u \left[x^\top Q x + u^\top R u + 2x^\top P(t)(Ax + Bu) \right] \\
&= \min_u \left[x^\top Q x + u^\top R u + 2x^\top P(t) A x + 2x^\top P(t) B u \right] \\
&= \min_u \left[x^\top Q x + u^\top R u + x^\top P(t) A x + x^\top A^\top P(t) x + 2x^\top P(t) B u \right] \\
0 &= \frac{\partial}{\partial u} \left[x^\top Q x + u^\top R u + x^\top P(t) A x + x^\top A^\top P(t) x + 2x^\top P(t) B u \right] \\
&= 2u^\top R + 2x^\top P(t) B \\
u^* &= -R^{-1} B^\top P x
\end{aligned}
$$

$\Rightarrow$ **Riccati differential equation**

$$
-\dot{P} = A^\top P + PA - PBR^{-1}B^\top P + Q
$$

Riccati differential equation

$$-\dot{P} = A^\top P + PA - PBR^{-1}B^\top P + Q$$

- This is a differential equation for the matrix $P(t)$ describing the quadratic value function. If we solve it with the finite horizon constraint $P(T) = F$ we solved the optimal control problem

- The optimal control $u^* = -R^{-1}B^\top P x$ is called **Linear Quadratic Regulator**

  Note: If the state is dynamic (e.g., $x = (q, \dot{q})$) this control is linear in the positions and linear in the velocities and is an instance of **PD control**
  The matrix $K = R^{-1}B^\top P$ is therefore also called **gain matrix**
  For instance, if $x(t) = (q(t) - r(t), \dot{q}(t) - \dot{r}(t))$ for a reference $r(t)$ and $K = \begin{bmatrix} K_p & K_d \end{bmatrix}$ then

$$u^* = K_p(r(t) - q(t)) + K_d(\dot{r}(t) - \dot{q}(t))$$

Riccati equations

- Finite horizon continuous time
  Riccati differential equation

$$-\dot{P} = A^\top P + PA - PBR^{-1}B^\top P + Q \;, \quad P(T) = F$$

- Infinite horizon continuous time
  Algebraic Riccati equation (ARE)

$$0 = A^\top P + PA - PBR^{-1}B^\top P + Q$$

- Finite horizon discrete time $(J^\pi = \sum_{t=0}^{T} \|x_t\|_Q^2 + \|u_t\|_R^2 + \|x_T\|_F^2)$

$$P_{t-1} = Q + A^\top [P_t - P_t B(R + B^\top P_t B)^{-1} B^\top P_t] A \;, \quad P_T = F$$

- Infinite horizon discrete time $(J^\pi = \sum_{t=0}^{\infty} \|x_t\|_Q^2 + \|u_t\|_R^2)$

$$P = Q + A^\top [P - PB(R + B^\top PB)^{-1} B^\top P] A$$

Example: 1D point mass

- Dynamics:

$$\ddot{q}(t) = u(t)/m$$

$$x = \begin{pmatrix} q \\ \dot{q} \end{pmatrix}, \quad \dot{x} = \begin{pmatrix} \dot{q} \\ \ddot{q} \end{pmatrix} = \begin{pmatrix} \dot{q} \\ u(t)/m \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 1/m \end{pmatrix} u$$

$$= Ax + Bu, \quad A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 1/m \end{pmatrix}$$

- Costs:

$$c(x, u) = \epsilon \|x\|^2 + \varrho \|u\|^2, \quad Q = \epsilon \mathbf{I}, \quad R = \varrho \mathbf{I}$$

- Algebraic Riccati equation:

$$P = \begin{pmatrix} a & c \\ c & b \end{pmatrix}, \quad u^* = -R^{\text{-}1} B^\top P x = \frac{-1}{\varrho m} [cq + b\dot{q}]$$

$$0 = A^\top P + PA - PBR^{\text{-}1} B^\top P + Q$$

$$= \begin{pmatrix} 0 & 0 \\ a & c \end{pmatrix} + \begin{pmatrix} 0 & a \\ 0 & c \end{pmatrix} - \frac{1}{\varrho m^2} \begin{pmatrix} c^2 & bc \\ bc & b^2 \end{pmatrix} + \epsilon \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Example: 1D point mass (cont.)

- Algebraic Riccati equation:

$$P = \begin{pmatrix} a & c \\ c & b \end{pmatrix}, \quad u^* = -R^{-1} B^\top P x = \frac{-1}{\varrho m}[cq + b\dot{q}]$$

$$0 = \begin{pmatrix} 0 & 0 \\ a & c \end{pmatrix} + \begin{pmatrix} 0 & a \\ 0 & c \end{pmatrix} - \frac{1}{\varrho m^2}\begin{pmatrix} c^2 & bc \\ bc & b^2 \end{pmatrix} + \epsilon \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

First solve for c, then for $b = m\sqrt{\varrho}\sqrt{2c+\epsilon}$ and $a = \frac{bc}{\varrho m^2}$. Whether the damping ration $\xi = \frac{b}{\sqrt{4mc}}$ depends on the choices of ϱ and ϵ.

- The Algebraic Riccati equation is usually solved numerically. (E.g. `are`, `care` or `dare` in Octave)

Optimal control comments

- HJB or Bellman equation are very powerful concepts

- Even if we can solve the HJB eq. and have the optimal control, we still don't know how the system really behaves?
  – Will it actually reach a "desired state"?
  – Will it be stable?
  – It is actually "controllable" at all?

- Last note on optimal control:
  Formulate as a constrainted optimization problem with objective function J^π and constraint $\dot{x} = f(x, u)$. $\lambda(t)$ are the Langrange multipliers. It turns out that $\frac{\partial}{\partial x} V(x, t) = \lambda(t)$. (See Stengel.)

Relation to other topics

- Optimal Control:

$$\min_{\pi} \; J^\pi = \int_0^T c(x(t), u(t)) \; dt + \phi(x(T))$$

- Inverse Kinematics:

$$\min_{q} \; f(q) = \|q - q_0\|^2_W + \|\phi(q) - y^*\|^2_C$$

- Operational space control:

$$\min_{u} \; f(u) = \|u\|^2_H + \|\ddot\phi(q) - \ddot y^*\|^2_C$$

- Trajectory Optimization: (control hard constraints could be included)

$$\min_{q_{0:T}} \; f(q_{0:T}) = \sum_{t=0}^T \|\Psi_t(q_{t\text{-}k}, .., q_t)\|^2 + \sum_{t=0}^T \|\Phi_t(q_t)\|^2$$

- Reinforcement Learning:
  – Markov Decision Processes $\leftrightarrow$ discrete time stochastic controlled system $P(x_{t+1} \,|\, u_t, x_t)$
  – Bellman equation $\to$ Basic RL methods (Q-learning, etc)

Controllability

Controllability

- As a starting point, consider the claim:
  "Intelligence means to gain maximal controllability over all degrees of freedom over the environment."

Controllability

- As a starting point, consider the claim:
  "Intelligence means to gain maximal controllability over all degrees of freedom over the environment."

  Note:
  – controllability (ability to control) $\neq$ control
  – What does controllability mean exactly?

- I think the general idea of *controllability* is really interesting
  – Linear control theory provides one specific definition of controllability, which we introduce next..

Controllability

- Consider a linear controlled system

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

  How can we tell from the matrices A and B *whether we can control* x *to eventually reach any desired state?*

- Example: x is 2-dim, u is 1-dim:

$$\begin{pmatrix}\dot{x}_1 \\ \dot{x}_2\end{pmatrix} = \begin{pmatrix}0 & 0 \\ 0 & 0\end{pmatrix}\begin{pmatrix}x_1 \\ x_2\end{pmatrix} + \begin{pmatrix}1 \\ 0\end{pmatrix}u$$

  Is x "controllable"?

Controllability

- Consider a linear controlled system

$$\dot x = Ax + Bu$$

How can we tell from the matrices A and B *whether we can control* x *to eventually reach any desired state?*

- Example: x is 2-dim, u is 1-dim:

$$\begin{pmatrix}\dot x_1 \\ \dot x_2\end{pmatrix} = \begin{pmatrix}0 & 0 \\ 0 & 0\end{pmatrix}\begin{pmatrix}x_1 \\ x_2\end{pmatrix} + \begin{pmatrix}1 \\ 0\end{pmatrix}u$$

Is x “controllable”?

$$\begin{pmatrix}\dot x_1 \\ \dot x_2\end{pmatrix} = \begin{pmatrix}0 & 1 \\ 0 & 0\end{pmatrix}\begin{pmatrix}x_1 \\ x_2\end{pmatrix} + \begin{pmatrix}0 \\ 1\end{pmatrix}u$$

Is x “controllable”?

Controllability

We consider a linear stationary (=time-invariant) controlled system

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

- **Complete controllability:** All elements of the state can be brought from arbitrary initial conditions to zero in finite time

Controllability

We consider a linear stationary (=time-invariant) controlled system

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

- **Complete controllability:** All elements of the state can be brought from arbitrary initial conditions to zero in finite time
- A system is completely controllable iff the **controllability matrix**

$$C := \begin{bmatrix} B & AB & A^2B & \cdots & A^{n\text{-}1}B \end{bmatrix}$$

has full rank $\dim(x)$ (that is, all rows are linearly independent)

Controllability

We consider a linear stationary (=time-invariant) controlled system

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

- **Complete controllability:** All elements of the state can be brought from arbitrary initial conditions to zero in finite time
- A system is completely controllable iff the **controllability matrix**

$$C := \begin{bmatrix} B & AB & A^2B & \cdots & A^{n\text{-}1}B \end{bmatrix}$$

has full rank $\dim(x)$ (that is, all rows are linearly independent)

- *Meaning of C:*

The ith row describes how the ith element x_i can be influenced by u

"B": $\dot{x}_i$ is directly influenced via B

"AB": $\ddot{x}_i$ is "indirectly" influenced via AB (u directly influences some $\dot{x}_j$ via B; the dynamics A then influence $\ddot{x}_i$ depending on $\dot{x}_j$)

"A^2B": $\dddot{x}_i$ is "double-indirectly" influenced

etc...

Note: $\ddot{x} = A\dot{x} + B\dot{u} = AAx + ABu + B\dot{u}$

$\dddot{x} = A^3x + A^2Bu + AB\dot{u} + B\ddot{u}$

Controllability

- When all rows of the controllability matrix are linearly independent $\Rightarrow$ $(u, \dot{u}, \ddot{u}, ...)$ can influence all elements of x independently
- If a row is zero $\rightarrow$ this element of x cannot be controlled at all
- If 2 rows are linearly dependent $\rightarrow$ these two elements of x will remain coupled forever

Controllability examples

$$\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\end{pmatrix} = \begin{pmatrix}0 & 0\\ 0 & 0\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix} + \begin{pmatrix}1\\ 1\end{pmatrix}u \qquad C = \begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix}$$ rows linearly dependent

$$\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\end{pmatrix} = \begin{pmatrix}0 & 0\\ 0 & 0\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix} + \begin{pmatrix}1\\ 0\end{pmatrix}u \qquad C = \begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}$$ 2nd row zero

$$\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\end{pmatrix} = \begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix} + \begin{pmatrix}0\\ 1\end{pmatrix}u \qquad C = \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}$$ good!

Controllability

Why is it important/interesting to analyze controllability?

- The Algebraic Riccati Equation will always return an “optimal” controller – but controllability tells us whether such a controller even has a chance to control x

Controllability

Why is it important/interesting to analyze controllability?

- The Algebraic Riccati Equation will always return an "optimal" controller – but controllability tells us whether such a controller even has a chance to control x

- *"Intelligence means to gain maximal controllability over all degrees of freedom over the environment."*
  – real environments are non-linear
  – "to have the ability to *gain* controllability over the environment's DoFs"

Stability

Stability

- One of the most central topics in control theory
- Instead of designing a controller by first designing a cost function and then applying Riccati,
  design a controller such that the desired state is provably a stable equilibrium point of the closed loop system

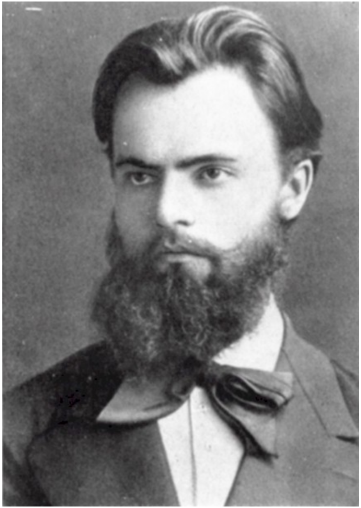
Stability

- Stability is an analysis of the *closed loop* system. That is: for this analysis we don't need to distinguish between system and controller explicitly. Both together define the dynamics

$$\dot{x} = f(x, \pi(x,t)) =: f(x)$$

- The following will therefore discuss stability analysis of general differential equations $\dot{x} = f(x)$

- What follows:
  – 3 basic definitions of stability
  – 2 basic methods for analysis by Lyapunov



Aleksandr Lyapunov (1857–1918)

Stability – 3 definitions

$\dot{x} = f(x)$ with equilibrium point $x_0 = 0$

- x_0 is an **equilibrium point** $\iff f(x_0) = 0$

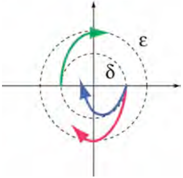
Stability – 3 definitions

$\dot{x} = f(x)$ with equilibrium point $x_0 = 0$

- x_0 is an **equilibrium point** $\iff f(x_0) = 0$

- **Lyapunov stable** or **uniformly stable** $\iff$

$$\forall \epsilon : \exists \delta \text{ s.t. } \|x(0)\| \leq \delta \Rightarrow \forall t : \|x(t)\| \leq \epsilon$$



(when it starts off δ-near to x_0, it will remain ϵ-near forever)

Stability – 3 definitions

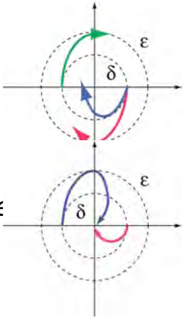
$\dot{x} = f(x)$ with equilibrium point $x_0 = 0$

- x_0 is an **equilibrium point** $\iff$ $f(x_0) = 0$

- **Lyapunov stable** or **uniformly stable** $\iff$

$$\forall \epsilon : \ \exists \delta \text{ s.t. } \|x(0)\| \le \delta \ \Rightarrow \ \forall t : \ \|x(t)\| \le \epsilon$$

(when it starts off δ-near to x_0, it will remain ϵ-near foreve



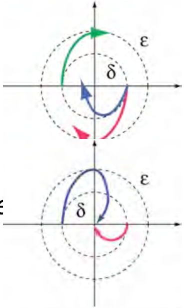
- **asymtotically stable** $\iff$

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

Stability – 3 definitions

$\dot{x} = f(x)$ with equilibrium point $x_0 = 0$

- x_0 is an **equilibrium point** $\iff$ $f(x_0) = 0$



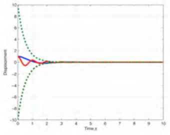
- **Lyapunov stable** or **uniformly stable** $\iff$

$$\forall \epsilon : \ \exists \delta \text{ s.t. } \|x(0)\| \le \delta \ \Rightarrow \ \forall t : \ \|x(t)\| \le \epsilon$$

(when it starts off δ-near to x_0, it will remain ϵ-near foreve

- **asymtotically stable** $\iff$

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



- **exponentially stable** $\iff$

asymtotically stable and $\exists \alpha, a$ s.t. $\|x(t)\| \le a e^{-\alpha t} \|x(0)\|$

($\rightarrow$ the "error" time integral $\int_0^\infty \|x(t)\| dt \le \frac{a}{\alpha} \|x(0)\|$ is bounded!)

Linear Stability Analysis

("Linear" $\leftrightarrow$ "local" for a system linearized at the equilibrium point.)

- Given a linear system

$$\dot{x} = Ax$$

Let λ_i be the **eigenvalues** of A (i.e., Frobenius companion matrix[1])

– The system is *asymptotically stable* $\iff \forall_i : \ \mathrm{real}(\lambda_i) < 0$

– The system is *unstable* $\iff \exists_i : \ \mathrm{real}(\lambda_i) > 0$

– The system is *marginally stable* $\iff \forall_i : \ \mathrm{real}(\lambda_i) \leq 0$

[1] J. Lunze. Regelungstechnik 1. Edition 3

Linear Stability Analysis

("Linear" $\leftrightarrow$ "local" for a system linearized at the equilibrium point.)

- Given a linear system

$$\dot{x} = Ax$$

Let λ_i be the **eigenvalues** of A (i.e., Frobenius companion matrix[1])
– The system is *asymptotically stable* $\iff \forall_i : \ \mathrm{real}(\lambda_i) < 0$
– The system is *unstable* $\iff \exists_i : \ \mathrm{real}(\lambda_i) > 0$
– The system is *marginally stable* $\iff \forall_i : \ \mathrm{real}(\lambda_i) \leq 0$

- Meaning: An eigenvalue describes how the system behaves along one state dimension (along the eigenvector):

$$\dot{x}_i = \lambda_i x_i$$

As for the 1D point mass the solution is $x_i(t) = a e^{\lambda_i t}$ and
– imaginary $\lambda_i \rightarrow$ oscillation
– negative $\mathrm{real}(\lambda_i) \rightarrow$ exponential decay $\propto e^{-|\lambda_i| t}$
– positive $\mathrm{real}(\lambda_i) \rightarrow$ exponential explosion $\propto e^{|\lambda_i| t}$

[1] J. Lunze. Regelungstechnik 1. Edition 3

Linear Stability Analysis: Example

- Let's take the 1D point mass $\ddot{q} = u/m$ in *closed loop* with a PD $u = -K_p q - K_d \dot{q}$

Linear Stability Analysis: Example

- Let's take the 1D point mass $\ddot{q} = u/m$ in *closed loop* with a PD $u = -K_p q - K_d \dot{q}$
- Dynamics:

$$\dot{x} = \begin{pmatrix} \dot{q} \\ \ddot{q} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x + 1/m \begin{pmatrix} 0 & 0 \\ -K_p & -K_d \end{pmatrix} x$$

$$A = \begin{pmatrix} 0 & 1 \\ -K_p/m & -K_d/m \end{pmatrix}$$

Linear Stability Analysis: Example

- Let's take the 1D point mass $\ddot{q} = u/m$ in *closed loop* with a PD $u = -K_p q - K_d \dot{q}$
- Dynamics:

$$\dot{x} = \begin{pmatrix} \dot{q} \\ \ddot{q} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x + 1/m \begin{pmatrix} 0 & 0 \\ -K_p & -K_d \end{pmatrix} x$$

$$A = \begin{pmatrix} 0 & 1 \\ -K_p/m & -K_d/m \end{pmatrix}$$

- Eigenvalues:

The equation $\lambda \begin{pmatrix} q \\ \dot{q} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -K_p/m & -K_d/m \end{pmatrix} \begin{pmatrix} q \\ \dot{q} \end{pmatrix}$ leads to the equation $\lambda \dot{q} = \lambda^2 q = -K_p/mq - K_d/m\lambda q$ or $m\lambda^2 + K_d\lambda + K_p = 0$ with solution (compare slide 03:12)

$$\lambda = \frac{-K_d \pm \sqrt{K_d^2 - 4mK_p}}{2m}$$

For $K_d^2 - 4mK_p$ negative, the $\mathrm{real}(\lambda) = -K_d/2m$

$\Rightarrow$ Positive derivative gain K_d makes the system stable.

Side note: Stability for discrete time systems

- Given a discrete time linear system

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

  Let λ_i be the **eigenvalues** of A
  – The system is *asymptotically stable* $\iff \forall_i : |\lambda_i| < 1$
  – The system is *unstable stable* $\iff \exists_i : |\lambda_i| > 1$
  – The system is *marginally stable* $\iff \forall_i : |\lambda_i| \leq 1$

Linear Stability Analysis comments

- The same type of analysis can be done locally for non-linear systems, as we will do for the cart-pole in the exercises

- We can design a controller that minimizes the (negative) eigenvalues of A:
  $\leftrightarrow$ controller with fastest asymtopic convergence

  This is a real alternative to optimal control!

Lyapunov function method

- A method to analyze/prove stability for general **non-linear** systems is the famous “Lyapunov's second method”

  Let D be a region around the equilibrium point x_0
- A **Lyapunov function** $V(x)$ for a system dynamics $\dot{x} = f(x)$ is
  – positive, $V(x) > 0$, everywhere in D except...
  at the equilibrium point where $V(x_0) = 0$
  – always decreases, $\dot{V}(x) = \frac{\partial V(x)}{\partial x}\dot{x} < 0$, in D except...
  at the equilibrium point where $f(x) = 0$ and therefore $\dot{V}(x) = 0$

- If there exists a D and a Lyapunov function $\Rightarrow$ the system is *asymtotically stable*

  If D is the whole state space, the system is *globally stable*

Lyapunov function method

- The Lyapunov function method is very general. $V(x)$ could be "anything" (energy, cost-to-go, whatever). Whenever one finds some $V(x)$ that decreases, this proves stability

- The problem though is to think of some $V(x)$ given a dynamics! (In that sense, the Lyapunov function method is rather a method of proof than a concrete method for stability analysis.)

Lyapunov function method

- The Lyapunov function method is very general. $V(x)$ could be "anything" (energy, cost-to-go, whatever). Whenever one finds some $V(x)$ that decreases, this proves stability

- The problem though is to think of some $V(x)$ given a dynamics! (In that sense, the Lyapunov function method is rather a method of proof than a concrete method for stability analysis.)

- In standard cases, a good guess for the Lyapunov function is either the energy or the value function

Lyapunov function method – Energy Example

- Let's take the 1D point mass $\ddot{q} = u/m$ in *closed loop* with a PD $u = -K_p q - K_d \dot{q}$, which has the solution (slide 03:15):

$$q(t) = be^{-\xi/\lambda\ t}\ e^{i\omega_0\sqrt{1-\xi^2}\ t}$$

- Energy of the 1D point mass:
  (only the real part of $q(t)$ is used to define $V(t)$)

$$\begin{aligned} V(t) &:= \frac{1}{2}m\dot{q}^2 = \frac{1}{2}mb^2(-\xi/\lambda)^2 e^{-2\xi/\lambda\ t} \\ \dot{V}(t) &= m\dot{q}\ddot{q} = mb^2(-\xi/\lambda)^3 e^{-2\xi/\lambda\ t} \\ &= -(2\xi/\lambda)e^{-2\xi/\lambda\ t}V(0) \end{aligned}$$

  (using that the energy of an undamped oscillator is conserved)

- $\dot{V}(t) < 0 \iff \xi > 0 \iff K_d > 0$
  Same result as for the eigenvalue analysis

Lyapunov function method – value function Example

- Consider infinite horizon linear-quadratic optimal control. The solution of the Algebraic Riccati equation gives the optimal controller.
- The value function satisfies

$$
\begin{aligned}
V(x) &= x^\top P x \\
\dot V(x) &= [Ax + Bu^*]^\top P x + x^\top P [Ax + Bu^*] \\
u^* &= -R^{-1} B^\top P x = Kx \\
\dot V(x) &= x^\top [(A + BK)^\top P + P(A + BK)] x \\
&= x^\top [A^\top P + PA + (BK)^\top P + P(BK)] x \\
0 &= A^\top P + PA - PBR^{-1} B^\top P + Q \\
\dot V(x) &= x^\top [PBR^{-1} B^\top P - Q + (PBK)^\top + PBK] x \\
&= -x^\top [Q + K^\top R K] x
\end{aligned}
$$

(We could have derived this easier! $x^\top Q x$ are the immediate state costs, and $x^\top K^\top R K x = u^\top R u$ are the immediate control costs—and $\dot V(x) = -c(x, u^*)$! See slide 11 bottom.)

- That is: V is a Lyapunov function if $Q + K^\top R K$ is positive definite!

Observability & Adaptive Control

- When some state dimensions are not directly observable: analyzing higher order derivatives to *infer* them.
  Very closely related to controllability: Just like the controllability matrix tells whether state dimensions can (indirectly) be controlled; an observation matrix tells whether state dimensions can (indirectly) be inferred.

- Adaptive Control: When system dynamics $\dot{x} = f(x, u, \beta)$ has unknown parameters β.
  – One approach is to estimate β from the data so far and use optimal control.
  – Another is to design a controller that has an additional internal update equation for an estimate $\hat{\beta}$ and is provably stable. (See Schaal's lecture, for instance.)