
Lecture notes on Control Systems and Reinforcement Learning

Written by
Manuel Hinz

mh@mssh.dev or s6mlhinz@uni-bonn.de

Lecturer

Prof. Dr. Jochen Garcke
[garcke\[at\]math.uni-bonn.de](mailto:garcke[at]math.uni-bonn.de)



University of Bonn
Summer semester 2025
Last update: April 22, 2025

Contents

Chapter 0 Manuel’s notes 2

 0.1 Organization 2

Chapter 1 Control Problems 3

 1.1 State Space Models 4

 1.1.1 Linear State Space Model 5

 1.1.2 State Space Models in continuous Time 5

 1.1.3 Value iteration 13

 1.1.4 Policy iteration 14

 1.1.5 Exploration 15

Journal 16

Bibliography 16

Chapter 0:

Manuel's notes

Warning

These are unofficial lecture notes written by a student. They are messy, will almost surely contain errors, typos and misunderstandings and may not be kept up to date! I do however try my best and use these notes to prepare for my exams. Feel free to email me any corrections to mh@mssh.dev or s6mlhinz@uni-bonn.de.
Happy learning!

Many thanks to Vincent for his feedback and some corrections!

General Information

- Basis: [Basis](#)
- Website: <https://ins.uni-bonn.de/teachings/ss-2025-467-v5e1-advanced-topics/>
- Time slot(s): **Tuesday: 14-16** SR 2.035 and **Thursdays: 16-18** SR 2.035
- Exams: ?
- Deadlines: No exercise sheets / tutorials

0.1 Organization

- Focused on ingredients, won't get to the current state of the art
- Some algorithmic / numerical background (Euler method is fine)
- Control Problems (Steering the bike / car)

Start of lecture 01
(10.4.2025)

The main source for this course is [\[2\]](#). We will follow this somewhat closely, especially in the first part of the course!

Chapter 1: Control Problems

1. u is the control (input / action)
2. y observations (outputs)
3. $\phi : Y \rightarrow U$ policy
4. ff feed forward control (plan we had)

Interactions with the outside world might be hidden in the observations. Typically ff is in regard to some reference state. There might be some disturbances (holes in the road, ...).

The overall aim is to find a policy ϕ that sticks close to $r(k), k \geq 0$.

t is continous, k is step
by step / iterative

$$u(k) = u_{ff}(k) + U_{fb}(k)$$

where u_{ff} is the planing to reach the overall goal and u_{fb} actual steering, updated "all the time".
Some examples from the book:

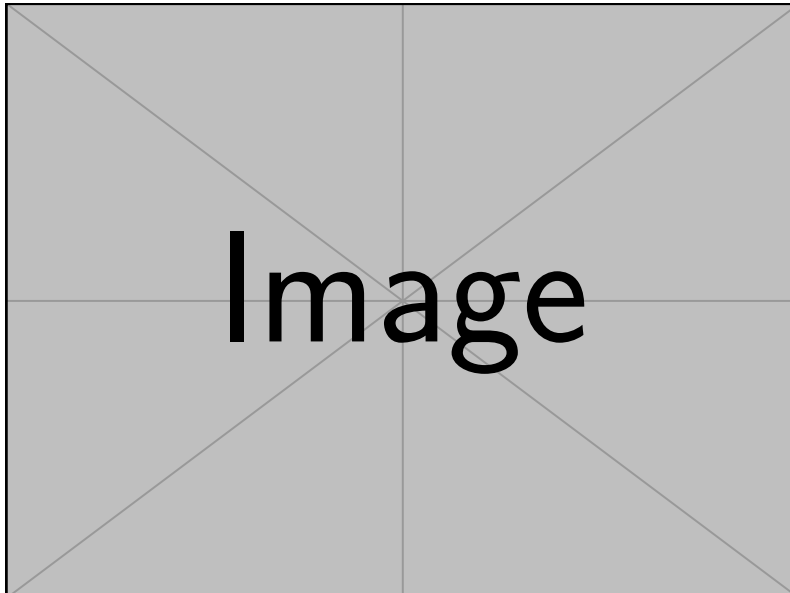


Figure 1.1: Sketch 1.01

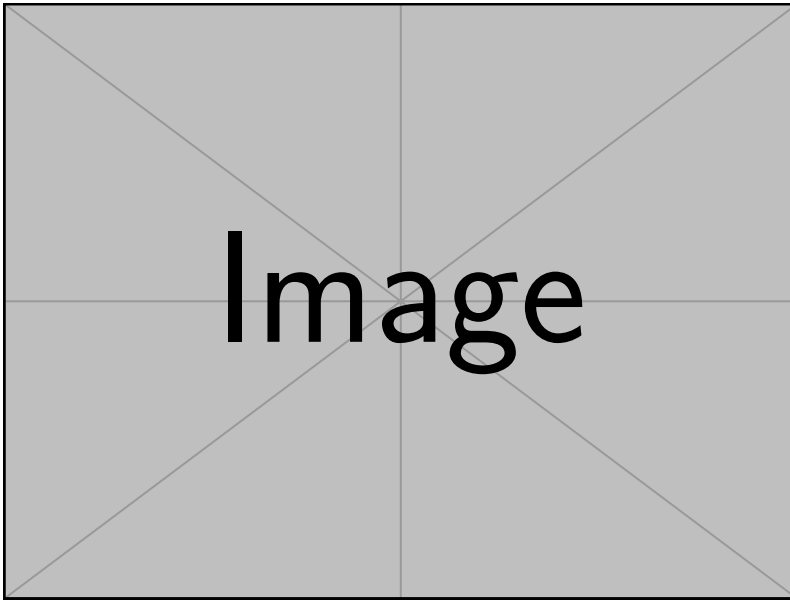


Figure 1.2: Sketch 1.02: Mountain car

Difference: In Reinforcement learning, we don't start with a model / ode.
Some part of reinforcement learning works model-free (i.e. assumes the model only implicitly)

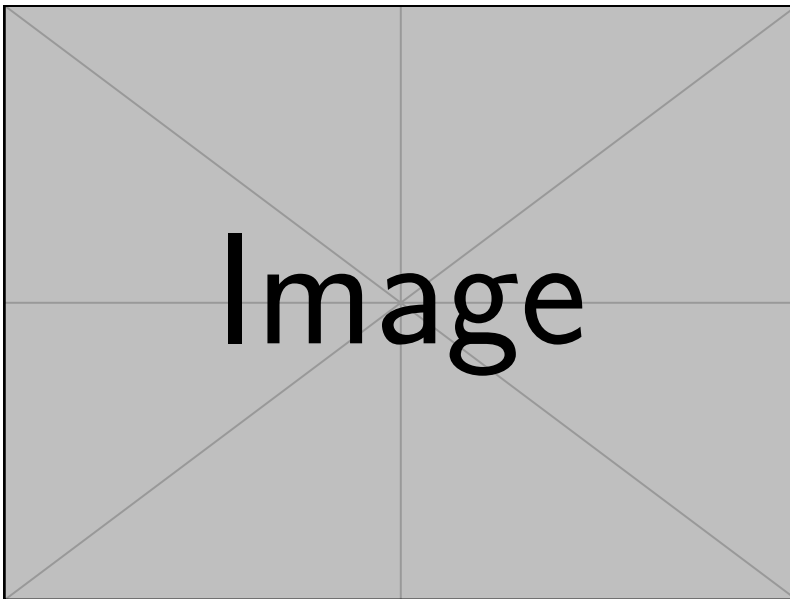


Figure 1.3: Sketch 1.03: cart pole / inverted pendulum

Next example: Acrobot (more than one equilibrium)

1.1 State Space Models

We have some

- state space $X, x \in X$
- action space $U, u \in U$
- action at step $k : u(k) \in U(k)$, i.e. we might have some constraints

- observation space $Y, y \in Y$

Definition 1.1. Given state, action and observation spaces X, U, Y , a state space model is defined by

$$x(k+1) = \mathcal{F}(x(k), u(k)) \quad (1)$$

$$y(k) = \mathcal{C}(x(k), u(k)) \quad (2)$$

$x(k)$ might include the past, might be useful for the stock trading problem

Remark. Overcomplicating problems by loading lots of information into the state space, might make the problem harder!

1.1.1 Linear State Space Model

$$x(k+1) = Fx(k) + Gu(k) \quad (3)$$

$$y(k) = Cx(k) + Du(k) \quad (4)$$

Remark. The representations (in terms of the matrices) might not be unique!

Common scenario for (3) is to keep $x(k)$ near the origin. You have to think about robustness of the system. Disturbances should be handled by the system.

$$u(k) = -Kx(k).$$

Consider a disturbance under the same control:

$$u(k) = -Kx(k) + v(k)$$

inserting this into (3) yields

$$x(k+1) = (F - GK)x(k) - Gv(k)$$

$$y(k) = (C - DK)x(k) + Dv(k)$$

Closed vs open loop: In closed loops we don't change our course based on observations, while in open loop systems we do.

1.1.2 State Space Models in continuous Time

$$\frac{d}{dt}x = f(x, u)$$

for $x \in \mathbb{R}^n, u \in \mathbb{R}^m$. We often write u_t, x_t for u, x at time t . If f is linear we get

$$\frac{d}{dt}x = Ax + Bu$$

$$y = Cx + Du$$



Figure 1.4: Sketch 1.04

To discretize we use the forward Euler method. Given time interval Δ

$$x(k+1) = x(k) + \Delta f(x(k), u(k))$$

so in (1) $\mathcal{F}(x, u) = x + \Delta f(x, u)$. Using Taylor

$$x_{t+\Delta} = x_t \Delta f(x, u) + O(\Delta^2)$$

For the linear model we get $F = I + \Delta A$

$$x(k+1) = x(k) + \Delta A x(k) + \underbrace{\Delta B}_{=:G} u(k)$$

For now fix some policy ϕ , so $u(k) = \phi(x(k))$:

$$x(k+1) = \mathcal{F}(x(k))$$

Assumption 1.2. The state space X is equal to \mathbb{R}^n or a closed subset of \mathbb{R}^n .

Definition 1.3. An equilibrium x^e is a state at which is system is frozen:

$$x^e = \mathcal{F}(x^e).$$

Definition 1.4. Given a cost function $C : X \rightarrow \mathbb{R}_+$ and a policy ϕ we define

$$J_\phi(x) = J(x) = \sum_{k=0}^{\infty} C(x(k)), \quad x(0) = x$$

This is called total cost or value function of the policy ϕ .

Given x^e , we usually assume $C(x^e) = 0$. Generally, we consider a discount factor γ^k in front of $C(x(k))$.

Definition 1.5. Denote by $\mathcal{X}(k; x_0)$ the state step k with initial condition x_0 and following fixed policy ϕ . The equilibrium x^e is stable in the sense of Lyapunov if for all $\epsilon > 0 \exists \delta > 0$ s.t. $\|x_0 - x^e\| < \delta$, then

$$\|\mathcal{X}(k; x_0) - \mathcal{X}(k; x^e)\| < \epsilon \forall k \geq 0$$

The same concept with a different sign comes up in RL under the term reward

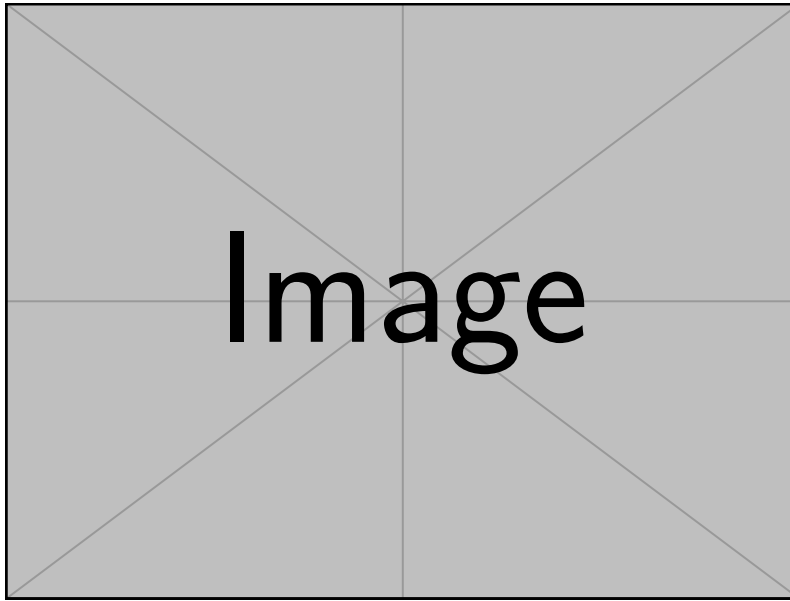


Figure 1.5: Sketch about Lyapunov stability

Definition 1.6. An equilibrium is said to be *asymptotically stable* if x^e is stable in the sense of Lyapunov and for some $\delta_0 > 0$, whenever $\|x_0 - x^e\| < \delta_0$, it follows

$$\lim_{k \rightarrow \infty} \mathcal{X}(k, x_0) = x^e.$$

The set of x_0 for which this holds is the *region of attraction* for x^e . An equilibrium is *globally asymptotically stable* if the region of attraction is X .

Definition 1.7 (Lyapunov function). A function $V : X \rightarrow \mathbb{R}_+$ is called *Lyapunov function*. We frequently assume V is *inf-compact*, i.e.: it holds

$$\forall x^0 \in X : \{x \in X \mid V(x) \leq V(x^0)\} \text{ is a bounded set.}$$

Remark. There is some variability in the definition of Lyapunov functions! We often assume $V(x)$ is large if x is large.

Sublevel sets:

$$S_V(r) = \{x \in X \mid V(x) \leq r\}.$$

One can see with V being inf-compact $S_V(r)$ is either

- empty
- the whole domain X
- a bounded subset of X .

We usually want to avoid this

Usually, $S_V(r) = X$ is impossible, a common assumption is *coersiveness*:

$$\lim_{\|x\| \rightarrow \infty} V(x) = \infty.$$

Example. • $V(x) = x^2$, coercive

- $V(x) = \frac{x^2}{(1+x)^2}$, not coercive, but inf-compact $r > 1 : S_V(r) = \mathbb{R}$, $r < 1 : S_V(r) = [-a, a]$ with $a = \sqrt{\frac{r}{1+r}}$
- $V(x) = e^x$ is neither

Lemma 1.8. Suppose that the cost function C and the value function J from definition 1.5 are non-negative and finite valued.

1. $J(x(k))$ is non-increasing in k and $\lim_{k \rightarrow \infty} J(x(k)) = 0$ for each initial condition.
2. In addition let J be continuous, inf-compact and vanishing only at x^e . Then for each initial condition

$$\lim_{k \rightarrow \infty} x(k) = x^e$$

Proof. Consider $J(x) = \sum_{k=0}^{\infty} c(x(k))$, then

$$\begin{aligned} J(x) &= c(x) + \sum_{k=1}^{\infty} c(x(k)) \\ &= c(x) + \sum_{k=0}^{\infty} c(x^+(k)); \quad x^+(0) = \mathcal{F}(x) \\ &= c(x) + J(\mathcal{F}(x)) \end{aligned}$$

This is the **dynamic programming principle** for a **fixed policy**. It is also called **Bellmann equation**. For 1. from this it follows

$$J(x(k+1)) + c(x) - J(x(k)) = 0$$

summing up from $k = 0$ up to $N - 1$

$$\begin{aligned} J(x) &= J(x(N)) + \sum_{k=0}^{N-1} c(x(k)) \\ &\implies \text{non-increasing} \end{aligned}$$

Taking the limit

$$= \lim_{N \rightarrow \infty} \left[J(x(N)) + \sum_{k=0}^{N-1} c(x(k)) \right] = \left[\lim_{N \rightarrow \infty} J(x(N)) \right] + J(x)$$

using $J(x)$ is finite gives (i).

For 2. with $r = J(x)$, we get $x(k) \in S_J(r) \forall k$. Now suppose $\{x(k_i)\}$ is a convergent subsequence of the trajectory with limit x^∞ . Then $J(x^\infty) = \lim_{i \rightarrow \infty} J(x(k_i)) = 0$ by the continuity of J . We assumed $J(x) = 0 \iff x^e = x \implies x^\infty = x^e$. Finally, the assumption follows, since each convergent subsequence reach the same value x^e .

□

Definition 1.9 (Poisson's inequality). Let $V, c : X \rightarrow \mathbb{R}_+$ and $\eta \geq 0$. Then **Poisson's inequality** states that

$$V(\mathcal{F}(x)) \leq V(x) - c(x) + \eta.$$

We often assume $\eta = 0$

Proposition 1.10. Suppose the Poisson inequality holds with $\eta = 0$. Additionally V shall be continuous, inf-compact and it shall have a unique minima at x^e . Then x^e is stable in the sense of Lyapunov (**sitsoL**).

Proof.

$$\bigcap \{S_V(r) \mid r > V(x^e)\} = \{S_V(r)|_{r=V(x^e)}\}^{\text{unique minimizer}} = \{x^e\}.$$

Using compactness we get: For each $\epsilon > 0$, we can find some $r > V(x^e)$ and some $\delta < \epsilon$ s.t.

$$\{x \in X \mid \|x - x^e\| < \delta\} \subset S_V(r) \subset \{x \in X \mid \|x - x^e\| < \epsilon\}$$

If $\|x_0 - x^e\| < \delta$, then $x_0 \in S_V(r)$ and hence $x(k) \in S_V(r)$ since $V(x(k))$ is non-increasing. With the second inclusion we see

$$\|x(k) - x^e\| < \epsilon \forall k$$

This gives sitsoL.

□

this is a assumption on the value function

We are separating one step!

This is the same Bellman from the curse of dimensionality!

Proposition 1.11 (Comparison theorem). *Poisson's inequality implies*

1. For each $N \geq 1$ and $x = x(0)$

$$V(x(N)) + \sum_{k=0}^{N-1} c(x(k)) \leq V(x) + N\eta$$

2. If $\eta = 0$, then $J(x) \leq V(x) \forall x$

3. Assume $\eta = 0$ and V, c are continuous. Suppose that c is inf-compact and vanishes only at the equilibrium x^e . Then x^e is globally asymptotically stable.

We don't write that explicitly, but we don't start in x^e !

Proof. 1.

$$V(x(k+1)) - V(x(k)) + c(x(k)) \leq \eta$$

summing up from 0 to $N-1$:

$$V(x(N)) - V(x(0)) + \sum_{k=0}^{N-1} c(x(k)) \leq N\eta$$

2. for $\eta = 0$ the above is ≤ 0 , so $\sum_{k=0}^{N-1} c(x(k)) \leq V(x(0)) - V(x(N)) \leq V(x(0))$ where the LHS converges to $J(x(0))$ for $N \rightarrow \infty$

3. Show sitsoL, with $\eta = 0$ it follows from definition 1.9 that $V(x) \geq c(x)$, which gives V is also inf-compact. c is vanishing only at x^e , so $V(x(k))$ is strictly decreasing. When $x(k) \neq x^e$, implies $V(x(k)) \downarrow V(x^e)$ for each $x(0)$. Further

This is important!

$$V(x^e) < V(x(0)) \quad \forall x(0) \in X \setminus \{x^e\}.$$

So it is a unique minimum. V has therefore the properties of proposition 1.10, which gives sitsoL. For global: with 1. we get

$$\lim_{k \rightarrow \infty} c(x(k)) = 0$$

and assumptions give us by lemma 1.8 that $x(k) \rightarrow x^e$ as $k \rightarrow \infty$. So, we converge from any initial condition, which gives global asymptotical stability. □

Proposition 1.12. *Suppose that $V(\mathcal{F}(x)) = V(x) - c(x)$. Further, we assume that*

1. J is continuous, inf-compact, vanishing only at x^e

2. V is continuous

Then $J(x) = V(x) - V(x^e)$.

Proof. As before we sum up:

$$V(x(N)) + \underbrace{\sum_{k=0}^{N-1} c(x(k))}_{J(x(N-1)) \xrightarrow{N \rightarrow \infty} J(x)} = V(x).$$

Lemma 1.8 together with the continuity of V implies that

$$V(x(N)) \rightarrow V(x^e) \quad \text{as } N \rightarrow \infty.$$

This gives

$$V(x^e) + J(x) = V(x) \quad \square$$

Start of lecture 03
(17.04.2025)

Example (Linear state space model). Setting $x(k+1) = \mathcal{F}(x(k))$, now with linear dynamics:

$$x(k+1) = Fx(k) = F^{k+1}x(0) = F^{k+1}x.$$

Assume quadratic cost $c(x) = x^\top Sx$, where S is symmetric and positive definite. Observe

$$c(x(k)) = (F^k x)^\top S F^k x$$

Summing up yields

$$J(x) = x^\top \underbrace{\left[\sum_{k=0}^{\infty} (F^k)^\top S F^k \right]}_{=:M} x$$

This satisfies a linear fixed point equation:

$$M = S + F^\top M F \quad (5)$$

This is also called
discrete time
Lyapunov equation

One can show for the linear state space model, that the following are equivalent:

1. the origin is asymptotically stable
2. the origin is globally asymptotically stable
3. Each eigenvalue λ of F satisfies $|\lambda| < 1$
4. (5) admits a solution M positive semi-definite for any S positive semidefinite.

Reference: [1]

Consider 1.1 without y

$$y(k+1) = \mathcal{F}(x(k), u(k))$$

with

$$c : X \times U \rightarrow \mathbb{R}_+.$$

The total cost J_ϕ for a given ϕ given $u(k) = \phi(x(k))$ is

$$J_\phi(x) = \sum_{k=0}^{\infty} c(x(k), u(k)).$$

The optimal value function is the minimum over all controls

$$J^*(x) = \min_{\underline{U}=[u(0), u(1), \dots]} \sum_{k=0}^{\infty} c(x(k), u(k)), \quad x(0) = x \in X \quad (6)$$

This describes the
optimal control policy
(OCP)

Remark. The minimizer might not be unique! In harder settings this might need to be an inf!

Goal: Find a control sequence that achieves the minimum.

Computationally we can't expect to calculate J_ϕ exactly, but we will approximate it.

and the corresponding
policy

Remark. We are in the infinite horizon setting (infinite time steps) to talk about the stability. For this it is important that the equilibrium has cost 0. Without an equilibrium we can also think about discounted value functions

$$J_\phi = \sum_{k=0}^{\infty} \gamma^k c(x(k), u(k))$$

We will see later that it holds for the sequence x^* achieving the minimum

$$J^*(x^*(k)) = c(x^*(k), u^*(k)) + J^*(x^*(k+1))$$

which is definition 1.9 with $\eta = 0$ and equality.

Proposition 1.11 implies, under some conditions, that x^e is globally asymptotically stable.

Under the following assumptions J^* is finite:

1. there is a (target) state x^e that is an equilibria for some control $F(x^e, u^e) = x^e$
2. $c \geq 0, c(x^e, u^e) = 0$
3. for any initial condition $x(0) = x$ there is a control sequence \underline{u} and a time T , such that $x(T) = x^e$ for $x(0) = x$ using control \underline{u} .

This is sometimes called controllability

Example (Linear Quadratic Regulator). Consider linear dynamics 3 from the first lecture with quadratic cost $c(x, u) = x^\top Sx + u^\top Ru$ with S positive semi-definite and R positive definite. Reminder: $u = -Kx$.

If there is a policy for which J^* is finite, then

$$J^*(x) = x^\top M^* x$$

with M^* positive semi-definite and

$$\phi^*(x) = -K^*(x)$$

with K^* depends on M^*, R, F, G .

and implicitly on c

Bellmann equation

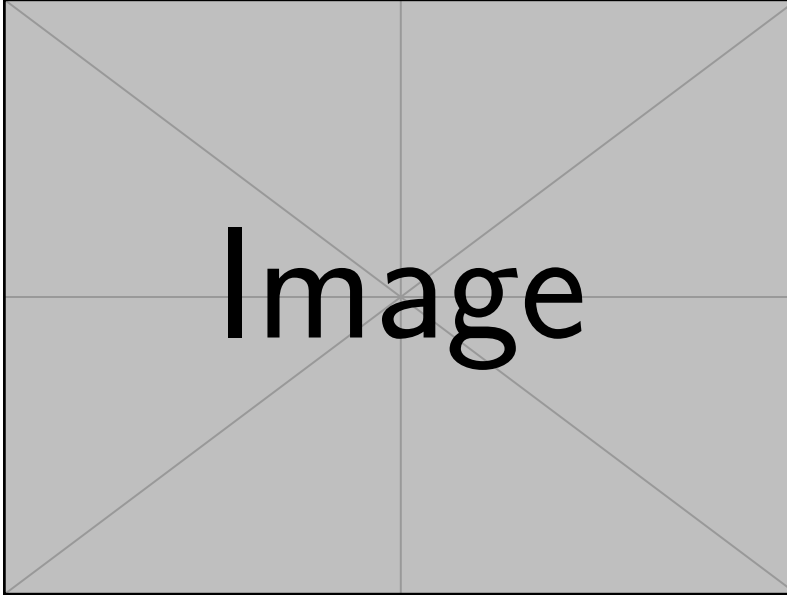


Figure 1.6: Sketch 1.06; Principle of optimality

Observation:

$$\begin{aligned}
 J^*(x) &= \min_{\underline{u}} \left[\sum_{k=0}^{k_m-1} c(x(k), u(k)) + \sum_{k_m}^{\infty} c(x(k), u(k)) \right] \\
 &= \min_{u[0, \dots, k_m-1]} \left[\sum_{k=0}^{k_m-1} c(x(k), u(k)) + \underbrace{\min_{u[k_m, \dots]} \sum_{k_m}^{\infty} c(x(k), u(k))}_{=J^*(x(k_m))} \right]
 \end{aligned}$$

This gives

$$J^*(x) = \min_{u[0, \dots, k_m-1]} \left[\sum_{k=0}^{k_m-1} c(x(k), u(k)) \right] + J^*(x(k_m)).$$

which can be seen as a kind of fix point equation

With $k_m = 1$ we have shown the following theorem

Theorem 1.13 (Bellmann equation, Dynamic Programming equation). Assume that J^* is finite and optimal control u^* solving (6) exists. Then the value function satisfies

$$J^*(x) = \min_u \{c(x, u) + J^*(\mathcal{F}(x, u))\} \quad (7)$$

Suppose the minimum is unique for each x and let $\phi^*(x)$ denote the minimum feedback law at x . Then the optimal control is expressed as

$$u^*(k) = \phi^*(x^*(k)).$$

Definition 1.14 (Q-function). The function of two variables within the minimum in (7) is called Q-function.

$$Q^*(x, u) = c(x, u) + J^*(\mathcal{F}(x, u))$$

In the optimal case we write Q^* . Thus

$$J^*(x) = \min_u Q^*(x, u).$$

The optimal feedback law is then

$$\phi^*(x) \in \operatorname{argmin}_u Q^*(x, u).$$

The Q-function solves the fixed point equation

$$Q^*(x, u) = c(x, u) + \min_u Q^*(\mathcal{F}(x, u), u).$$

This already gives a hint for an algorithm coming later next lecture.

Remark. In RL the difference is that we don't know the model, we only observe state action pairs. This motivates the Q-function.

Definition, which is not so useful for the analysis, but for the practical application!

Some concepts from Reinforcement Learning

Actors and critic:

Given is a parameterized family of policies $\{\phi^\theta \mid \theta \in \mathbb{R}^d\}$. the actors. For each θ , observe the trajectories by their states x and actions u determined by their policy.

The critic approximates the associated value function \tilde{J}_θ . Aim for the minimum

$$\theta^* = \operatorname{argmin}_\theta \langle v, \tilde{J}_\theta \rangle,$$

where the weight vector $v \geq 0$ reflects the weighting of the states. $v(x)$ is large for *important* states.

scalar product in \mathbb{R}^n (all states?)

Temporal differences:

$$J_\theta(x(k)) = c(x(k), u(k \mid \theta)) + J_\theta(x(k+1))$$

Look for an approximation \hat{J} for which the error is small (w.r.t. the equality above).

Temporal differences are

$$D_{k+1}(\hat{J}) := -\hat{J}(x(k)) + \hat{J}(x(k+1)) + c(x(k), u(k)).$$

After N samples

$$\Gamma(\hat{J}) := \frac{1}{N} \sum_{k=0}^{N-1} D_{k+1}(\hat{J})^2.$$

We can optimize / minimize this.

There is a whole class of TD algorithms and those fit into the actors critic approach!

What changes, or what is the information gain

Start of lecture 04
(22.04.2025)

1.1.3 Value iteration

We approximate J^* by a sequence of V^k given an initial value function V^0 .

$$V^{k+1}(x) = \min_u \{c(x, u) + V^k(\mathcal{F}(x, u))\}, x \in X, k \geq 0$$

This is called **value iteration** often shortened to VI.

Proposition 1.15. *Let V^0 be chosen with non-negative entries and $V^0(x^e) = 0$. Further, we assume*

1. X, U are finite sets
2. c is non-negative and vanishes only at (x^e, u^e) , and J^* is finite valued.

Then there is $n_0 \geq 1$ such that

$$V^k(x) = J^*(x), x \in X, k \geq n_0.$$

Proof. Let $\phi^*(x)$ be an optimal policy, and let $n_0 \geq 1$ denote the value such that

$$(x^*(k), u^*(k)) = (x^e, u^e)$$

for $k \geq n_0$. This exists since J^* is finite.

Using the principle of optimality (6) we can show

$$V^n(x) = \min_{u[0, \dots, n-1]} \left\{ \sum_{k=0}^{n-1} c(x(k), u(k)) + V^0(x(n)) \right\}, x(0) \in X \quad (8)$$

This gives

$$V^n(x) \leq \sum_{k=0}^{n-1} c(x(k), u(k)) + V^0(x(n)) \text{ for all } u \text{ including } u(k) = \phi^*(k)$$

$$\stackrel{n \geq n_0}{=} J^*(x) + V^0(x(0)) = J^*(x)$$

For such n , the inequality must be an equality, due to (8) and the use of the optimal policy. \square

VI provides a sequence of policies ϕ^n

$$\phi^n(x) \in \operatorname{argmin}_u \{c(x, u) + V^n(\mathcal{F}(x, u))\}.$$

If we assume that V^0 is non-negative and satisfies for some $\eta \geq 0$ poisson's inequality(1.9)

$$V^0(\mathcal{F}(x, u)) \leq V^0(x) - c(x, \phi^0(x)) + \eta, x \in X$$

then we get the following statement

Proposition 1.16. *Suppose that V^0 is non-negative and it holds*

$$\min_u (c(x, u) + V^0(\mathcal{F}(x, u))) = \{c(x, u) + V^0(\mathcal{F}(x, u))\} |_{u=\phi^0(x)}$$

$$\leq V^0(x) + \eta, x \in X$$

Then a corresponding bound holds for each n

$$\{c(x, u) + V^n(\mathcal{F}(x, u))\} |_{u=\phi^n(x)} \leq V^n(x) + \eta_n, x \in X,$$

where η_i is non-increasing:

$$\eta \geq \eta_0 \geq \eta_1 \dots$$

For infinite state spaces we will have to fix this algorithm for memory related reasons

We really exploit the finiteness!

Proof. Write $B^n(x) = V^{n+1}(x) - V^n(x)$

This is (connected to?)
the Bellman error

$$\eta_n := \sup_x B^n(x).$$

Value iteration gives

$$\begin{aligned} \{c(x, u) + V^n(\mathcal{F}(x, u))\}_{|u=\phi^n(x)} &= \min_u \{c(x, u) + V^n(\mathcal{F}(x, u))\} \\ &= V^{n+1}(x) = V^n(x) + B^n(x) \\ &\leq V^n(x) + \eta_n \end{aligned}$$

For non-increasing, we consider

$$V^1(x) = \{c(x, u) + V^0(\mathcal{F}(x, u))\}_{|u=\phi^0(x)} \stackrel{\text{Assumption}}{\leq} V^0(x) + \eta$$

which gives $B^0(x) \leq \eta \forall x \implies \eta_0 \leq \eta$.

For $n \geq 1$ The trick is using the old control in the second line:

$$\begin{aligned} V^n(x) &= \{c(x, u) + V^{n-1}(\mathcal{F}(x, u))\}_{|u=\phi^{n-1}(x)} \\ V^{n+1}(x) &\leq \{c(x, u) + V^n(\mathcal{F}(x, u))\}_{|u=\phi^{n-1}(x)} \end{aligned}$$

So,

$$V^{n+1}(x) - V^n(x) \leq \{V^n(\mathcal{F}(x, u)) - V^{n-1}(\mathcal{F}(x, u))\}_{|u=\phi^{n-1}(x)} \leq \eta_{n-1}.$$

Hence, $\eta_n = \sup_x B^n(x) \leq \eta_{n-1}$. □

Now consider $\eta = 0$, so for each n

$$\{c(x, u) + V^n(\mathcal{F}(x, u))\}_{|u=\phi^n(x)} \leq V^n(x)$$

with proposition 1.11 it follows

$$J^* \leq V^n(x), \quad x \in X,$$

where J^* is the total cost using policy ϕ^n .

One view of policy iteration is the focus on updating the policy function!

1.1.4 Policy iteration

Start with an initial policy $\phi^0, n = 0$

- Compute the total cost for the policy ϕ^n , this is called policy evaluation

$$J^n(x) = \sum_{k=0}^{\infty} c(x(k), u(k)), \quad u(k) = \phi^n(x(k)) \forall x \in X$$

- perform policy improvement to obtain the next policy

$$\phi^{n-1}(x) \in \operatorname{argmin}_u \{c(x, u) + J^n(\mathcal{F}(x, u))\}, \quad x \in X$$

- while *not good enough*

This is sometimes also called Howard's algorithm.

Remark. The first step is some linearization and the second is the update. Like a generalization of Newton's method

Algorithm 1 This will be fixed soon

Input: $A \in \mathbb{R}^{m \times n}$, $m \geq n$

Output: R von der QR -Zerlegung (A wird zerstört “in place”)

```

for  $j = 1, \dots, n$  do
  for  $i = m, m-1, \dots, j+1$  do
    Berechne  $c, s$ 
     $A[i-1 : i, j : n] = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}^t A[i-1 : i, j : n]$ 
  end for
end for

```

Proposition 1.17. Suppose that J^0 for ϕ^0 is finite valued. Then for each $n \geq 0$

$$\{c(x, u) + J^n(\mathcal{F}(x, u))\}_{|_{u=\phi^{n+1}(x)}} \leq J^n(x), \quad x \in X$$

and consequently, the value functions are non-increasing

$$J^0(x) \geq J^1(x) \geq \dots$$

Proof. Similar to the proof of proposition 1.16, where the non-increasing sequence again follows from proposition 1.11. \square

Here we always assumed that we can compute everything, especially \mathcal{F} and the infinite sum.

1.1.5 Exploration

In RL we learn from observations, each state-action pair, new state and observed cost gives us information. We need *good* and *useful* information.

Consider a policy that is not optimal, but has $x(k) \rightarrow x^e$ reasonably rapidly, where we assume $c(x^e, \cdot) = 0$. Typically we have continuity

$$\begin{aligned} \lim_{k \rightarrow \infty} D_{k+1}(\hat{J}) &= \lim_{k \rightarrow \infty} \left[-\hat{J}(x(k)) + \hat{J}(x(k+1)) + c(x(k), u(k)) \right] \\ &= -\hat{J}(x^e) + \hat{J}(x^e) + 0 = 0. \end{aligned}$$

This is not much information, one cannot further improve the policy!

$$\Gamma^\epsilon(\hat{J}, x^i) = \frac{1}{N_\epsilon} \sum_{k=0}^{N_\epsilon-1} [D_{k+1}(\hat{J})]^2, \quad x(0) = x^i$$

To avoid getting *small* information from long trajectories, one can take a couple of shorter ones.

$$\hat{\Gamma}(\hat{J}) = \frac{1}{M} \sum_{i=1}^M \Gamma^\epsilon(\hat{J}; x^i)$$

How to choose x^i is current research. Much of the theoretical research assume that “every state is assumed regularly”, which is nice for results, but not so nice realistic in most applications.

Another way to get more diverse information is to use [exploration](#). Namely one modifies the trajectories, not strictly follows ϕ^n .

$u(k) = \hat{\phi}(x(k), \zeta(k))$, where $\zeta(k)$ is some form of noise. Typically

1. $\hat{\phi}(x(k), \zeta(k)) = \phi^\theta(k)$ for *most* k
2. Choose action to explore the state-action space (e.g. randomly) the other times

Generally, the trajectory to gather information stems from a different policy than the current estimate ϕ^θ .

This dilemma is called the [exploration-exploitation](#) dilemma.

this is also sometimes called off-policy and on-policy

Journal

- **Lecture 01:** Covering: Introduction, (linear, continuous) State space models, equilibrium, (Lyapunov, asymptotically) stable, region of attraction, globally asymptotically stable . Starting in ‘[Organization](#)’ on page 2 and ending in ‘[State Space Models in continuous Time](#)’ on page 7. Spanning 5 pages
- **Lecture 02:** Covering: Lyapunov function, inf-compactness and coerciveness, sublevel sets, Poisson’s inequality, comparison theorem, a few propositions connecting the value function, equilibria and Lyapunov functions . Starting in ‘[State Space Models in continuous Time](#)’ on page 7 and ending in ‘[State Space Models in continuous Time](#)’ on page 9. Spanning 2 pages
- **Lecture 03:** Covering: discrete time Lyapunov equation, optimal control policy, controllability, linear quadratic regulator, Bellmann equation, principle of optimality, Q-function and some concepts from Reinforcement Learning . Starting in ‘[State Space Models in continuous Time](#)’ on page 9 and ending in ‘[Some concepts from Reinforcement Learning](#)’ on page 12. Spanning 3 pages
- **Lecture 04:** Covering: Value iteration, policy iteration, exploration-exploitation . Starting in ‘[Some concepts from Reinforcement Learning](#)’ on page 12 and ending in ‘[Exploration](#)’ on page 15. Spanning 3 pages

Bibliography

- [1] Tamer Basar, Sean Meyn, and William R. Perkins. *Lecture Notes on Control System Theory and Design*. 2024. arXiv: [2007.01367](https://arxiv.org/abs/2007.01367) [math.OC]. URL: <https://arxiv.org/abs/2007.01367>.
- [2] Sean Meyn. *Control Systems and Reinforcement Learning*. Cambridge University Press, 2022.