Dynamic Programming and Model Predictive Control

Anders Rantzer

Automatic Control LTH
Lund University
Sweden

Course Outline

Tuesday 08:45

- Optimal control
- Dynamic programming
- Bellman's equation
- Value iteration
- Relaxed DynP

Wednesday 08:45

- Tuesday summary
- MPC with terminal cost
- MPC without terminal cost
- Adaptive horizon
- Receding horizon estimation

Tuesday 09:45

- Model Predicitve Control
- Stability
- Terminal constraints
- Introduction to exercise

Wednesday 09:45

- Dual Decomposition
- Distributed MPC
- Gradient methods
- Large-scale systems

Wednesday afternoon: Computer exercise

Optimal Control

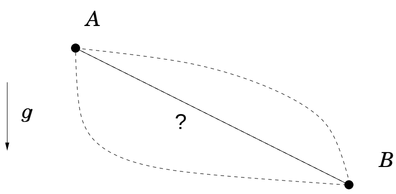
Idea:
Formulate a control synthesis problem in terms of optimization

+ Gives systematic design procedure
+ Can be used on nonlinear models
+ Can capture limitations as constraints
– Hard to find suitable criterium?!
– Can be hard to find the optimal controller

The beginning

John Bernoulli: The bracistochrone problem 1696:

Let a particle slide along a frictionless curve. Find the curve that takes the particle from A to B in shortest time

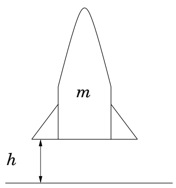


Solved by John and James Bernoulli, Newton, l'Hospital

Example: Goddard's Rocket Problem (1910)

How should one send a rocket as high up in the air as possible?

$$\frac{d}{dt}\begin{pmatrix} v \\ h \\ m \end{pmatrix} = \begin{pmatrix} \frac{u-D}{m} - g \\ v \\ -\gamma u \end{pmatrix}$$



where u = motor force, $D(v,h)$ = air resistance, m = mass.

Maximize $h(t_f)$ when $0 \leq u \leq u_{max}$ and $m(t_f) \geq m_1$

Low v when air resistance is high. Burn fuel at higher level.

Took about 50 years before a complete solution was found.

Read more about Goddard at `http://www.nasa.gov/centers/goddard/`

Optimal Control

- The space race (Sputnik 1957)
- Putting satellites in orbit
- Trajectory planning for interplanetary travel
- Reentry into atmosphere
- Minimum time problems
- Pontryagin's maximum principle, 1956
- Dynamic programming, Bellman 1957
- Vitalization of a classical field

Lecture 1

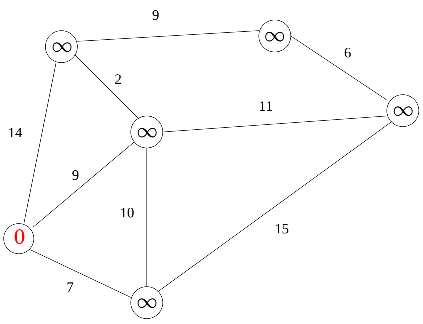
- Optimal control
- **Dynamic Programming and Bellman's Equation**
- Value iteration
- Relaxed Dynamic Programming
- Examples

Dynamic Programming, Richard E. Bellman 1957

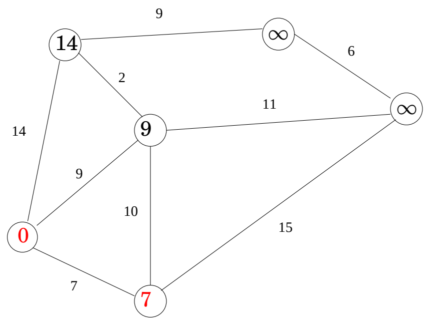
An optimal trajectory on the time interval $[T_1, T]$ must be optimal also on each of the subintervals $[T_1, T_1+\epsilon]$ and $[T_1+\epsilon, T]$.

Example: Dijkstra's Algorithm

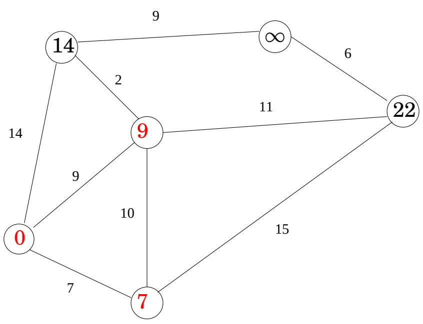
For each node, find the shortest path to the goal!



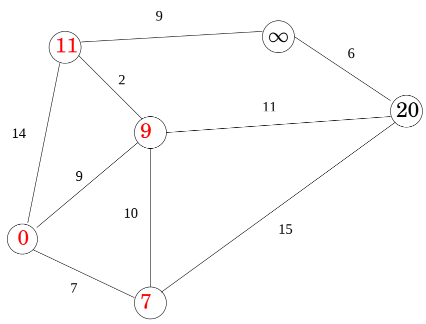
Example: Dijkstra's Algorithm



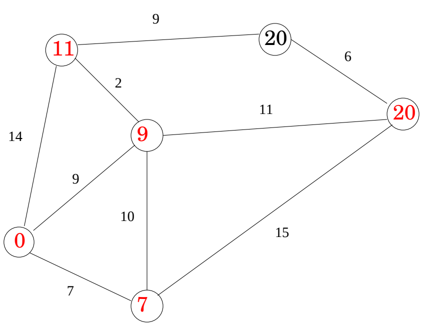
Example: Dijkstra's Algorithm



Example: Dijkstra's Algorithm



Example: Dijkstra's Algorithm



Dynamic Programming in Discrete Time

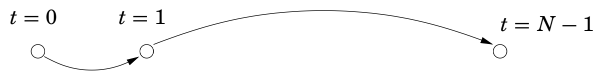
Minimize $\sum_{t=0}^{N-1} l(x(t), u(t))$

subject to $x(t+1) = f(x(t), u(t)) \qquad t = 0, 1, 2, \ldots, N-1$

$x(0) = x_0$

Let $V_N(x_0)$ denote the minimal value. The value function V_N satisfies the *Bellman equation*

$$V_N(x) = \min_u \left[l(x, u) + V_{N-1}(f(x, u)) \right]$$



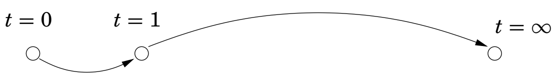
Infinite horizon Bellman equation

Minimize $\sum_{t=0}^{\infty} l(x(t), u(t))$

subject to $x(t+1) = f(x(t), u(t)) \qquad x(0) = x_0$

Let $V_\infty(x_0)$ denote the minimal value. The value function V_∞ satisfies the *Bellman equation*

$$V_\infty(x) = \min_u \left[l(x, u) + V_\infty(f(x, u)) \right]$$



Lecture 1

- Optimal control
- Dynamic Programming and Bellman's Equation
- **Value iteration**
- Relaxed Dynamic Programming
- Examples

Value iteration

Minimize $\sum_{t=0}^{N-1} l(x(t), u(t))$

subject to $x(t+1) = f(x(t), u(t)) \qquad x(0) = x_0 \in X$

Recall that the value function V_N satisfies the Bellman equation

$$V_N(x) = \min_u [l(x,u) + V_{N-1}(f(x,u))]$$

Starting from $V_{-1}(x) \equiv 0$ the iteration gives $V_N(x)$ with

$$0 \le V_1(x) \le V_2(x) \le V_3(x) \cdots$$
$$\lim_{N\to\infty} V_N(x) \to V_\infty(x), \quad N \to \infty$$

Often extremely complex when $X = \mathbf{R}^n$. Useful for finite X.

Theorem: Value Iteration Convergence

Suppose the condition $0 \le V_\infty(f(x,u)) \le \gamma l(x,u)$ holds uniformly for some $\gamma < \infty$ and that $0 \le \eta V_\infty \le V_0^* \le \delta V_\infty$.

Then the sequence defined iteratively by

$$V_{j+1}^*(x) = \min_u \left[V_j^*(f(x,u)) + l(x,u)\right] \qquad j \ge 0$$

approaches V_∞ according to the inequalities

$$\left[1 + \frac{\eta - 1}{(1+\gamma^{-1})^j}\right] V_\infty(x) \le V_j^*(x) \le \left[1 + \frac{\delta - 1}{(1+\gamma^{-1})^j}\right] V_\infty(x)$$

Proof idea

Use assumptions $\eta V_\infty \le V_0^*$ and $V_\infty(f(x,u)) \le \gamma l(x,u)$ to get

$$\begin{aligned} V_1^*(x) &= \min_u [V_0(f(x,u)) + l(x,u)] \\ &\ge \min_u [\eta V_\infty(f(x,u)) + l(x,u)] \\ &\ge \min_u [\eta_1 V_\infty(f(x,u)) + \eta_1 l(x,u)] \\ &= \eta_1 V_\infty(x) \end{aligned}$$

for some $\eta_1 > \eta$. The smaller γ is, the bigger η_1 becomes.

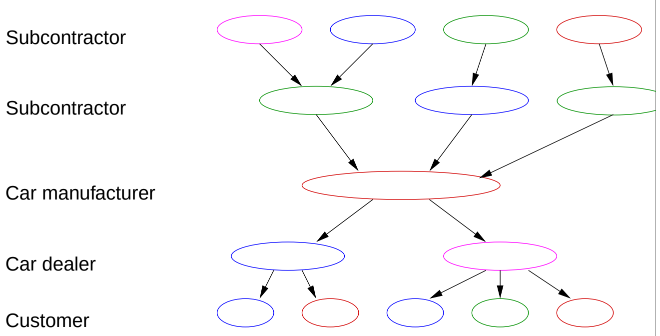
Repeat the argument to get $V_k^*(x) \ge \eta_k V_\infty(x)$, where $\eta < \eta_1 < \eta_2 < \eta_3 < \ldots$

Convergence from above is obtained the same way.

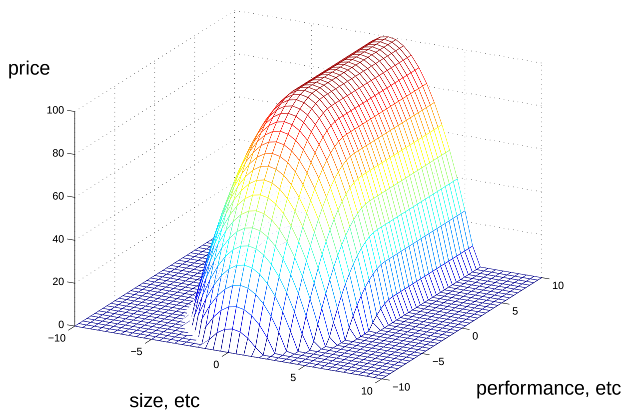
Lecture 1

- Optimal control
- Dynamic Programming and Bellman's Equation
- Value iteration
- **Relaxed Dynamic Programming**
- Examples

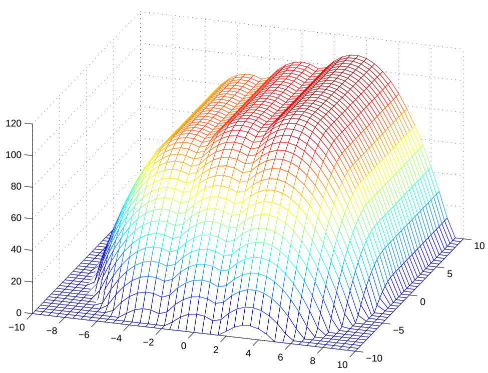
Who decides the price of a Volvo?



Valuation by the customer



Valuation by the car dealer



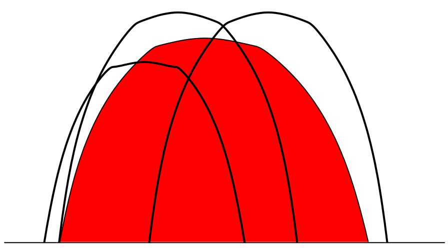
Customers: Andersson, Pettersson and Lundström

The key: Simplified valuation

Exact value-iteration gives absurd complexity.

Every subcontractor of Volvo would have to modify his prices when Andersson expands his garage.

Of course, pricing is not done like that.
Approximations are done in every step.



Bounds on the Optimal Cost

If

$$V(x) \le \min_u [V(f(x,u)) + l(x,u)]$$

then V is a lower bound on the optimal cost.

Conversely, if

$$\min_u [V(f(x,u)) + l(x,u)] \le V(x)$$

then V is an upper bound on the optimal cost.

Relaxed Value Iteration

Replace the Bellman equation by an inequality:

$$\min_u [V(f(x,u)) + l(x,u)/\alpha] \le V(x) \le \min_u [V(f(x,u)) + \alpha l(x,u)]$$

where $\alpha > 1$.

From the inequalities, it follows that

$$V^*(x)/\alpha \le V(x) \le \alpha V^*(x)$$

The recursive conditions become

$$\min_u [V_j(f(x,u)) + l(x,u)/\alpha] \quad \le V_{j+1}(x) \le \min_u [V_j(f(x,u)) + \alpha l(x,u)]$$

The interval for $V_{j+1}(x)$ makes it possible to work with a simplified parameterization of V_j.

Theorem: Relaxed Value Iteration Convergence

Suppose the condition $0 \le V_\infty(f(x,u)) \le \gamma l(x,u)$ holds uniformly for some $\gamma < \infty$ and that $0 \le \eta V_\infty \le V_0^* \le \delta V_\infty$.

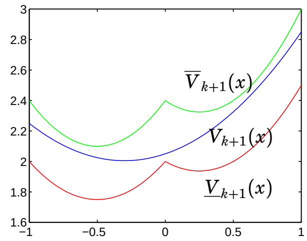
Then a sequence satsifying

$$\min_u [V_j(f(x,u)) + l(x,u)/\alpha] \quad \le V_{j+1}(x) \le \min_u [V_j(f(x,u)) + \alpha l(x,u)]$$

approaches the interval $[\alpha^{-1} V_\infty, \alpha V_\infty]$ according to the inequalities

$$\left[1 + \frac{\eta - 1}{(1+\gamma^{-1})^j}\right] \alpha^{-1} V_\infty \le V_j^*(x) \le \left[1 + \frac{\delta - 1}{(1+\gamma^{-1})^j}\right] \alpha V_\infty(x)$$

Relaxed Dynamic Programming



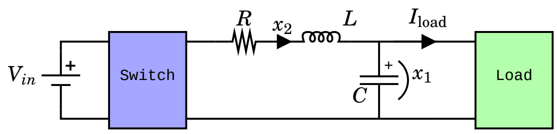
$$\underbrace{\min_u \left\{V_k(f(x,u)) + l(x,u)/\alpha\right\}}_{\underline{V}_{k+1}(x)} \le V_{k+1}(x) \le \underbrace{\min_u \left\{V_k(f(x,u)) + \alpha l(x,u)\right\}}_{\overline{V}_{k+1}(x)}$$

Lecture 1

- Optimal control
- Dynamic Programming and Bellman's Equation
- Value iteration
- Relaxed Dynamic Programming
- **Examples**

Example: Switched voltage converter

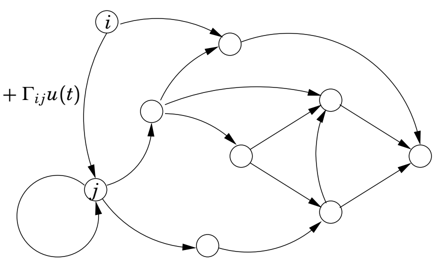
A step-down DC/DC converter.



- A linear system except for the switching actuator
- Objective: Keep output voltage constant.

Optimize switches for continuous dynamics

$$x(t+1) = \Phi_{ij} x(t) + \Gamma_{ij} u(t)$$

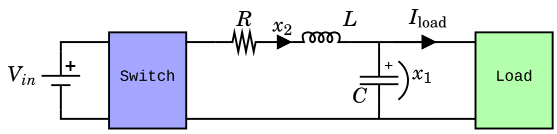


$$x(t+1) = \Phi_{jj} x(t) + \Gamma_{jj} u(t)$$

Minimize $\sum_t x(t)^T Q_{i(t)j(t)} x(t) + u(t)^T R_{i(t)j(t)} u(t)$

Two types of inputs, both affect the penalty

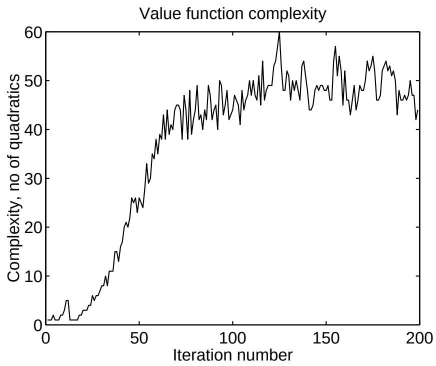
Example: Switched voltage converter



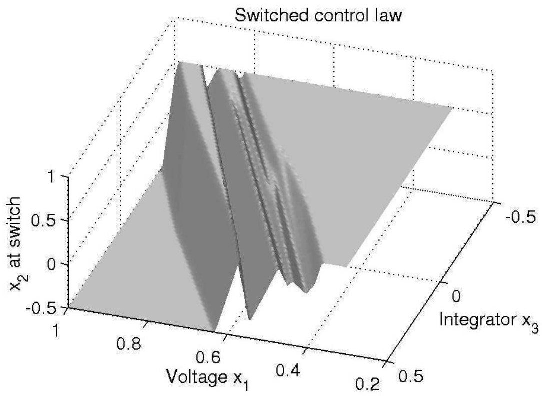
$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} \frac{1}{C}\left(x_2 - I_{\text{load}}\right) \\ -\frac{1}{L}x_1 - \frac{R}{L}x_2 + \frac{1}{L}s(t)V_{\text{in}} \\ V_{\text{ref}} - x_1 \end{bmatrix}$$

$$l(x) = q_P(x_1 - V_{\text{ref}})^2 + q_I x_3^2 + q_D(x_2 - I_{\text{load}})^2$$

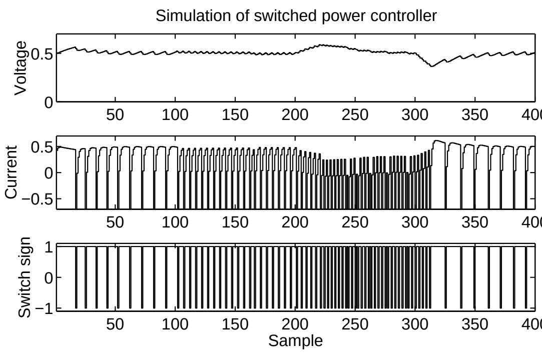
Example: Switched voltage converter



Example: Switched voltage converter



Example: Switched voltage converter

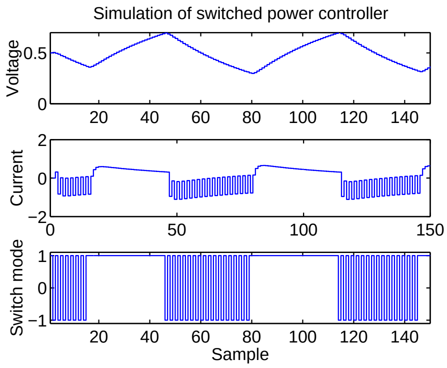


$I_{\text{load}} =$

$\{0.3A \quad 0.1A \quad -0.2A \quad 0.3A\}$

Example: Switched voltage converter

Frequency weights in the cost function can be used to suppress undesired harmonics. This increases state dimension, but has no significant effect on computational complexity.



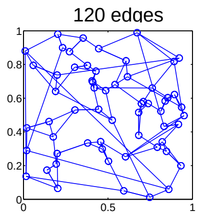
More on Control of DC-DC Converters

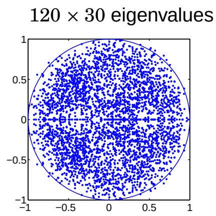

1126 IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, VOL. 18, NO. 5, SEPTEMBER 2010

Comparison of Hybrid Control Techniques for Buck and Boost DC-DC Converters

Sébastien Mariéthoz, *Member, IEEE*, Stefan Almér, Mihai Bâja, Andrea Giovanni Beccuti, Diego Patino, Andreas Wernrud, Jean Buisson, Hervé Cormerais, Tobias Geyer, *Member, IEEE*, Hisaya Fujioka, *Member, IEEE*, Ulf T. Jönsson, *Member, IEEE*, Chung-Yao Kao, *Member, IEEE*, Manfred Morari, *Fellow, IEEE*, Georgios Papafotiou, *Member, IEEE*, Anders Rantzer, *Fellow, IEEE*, and Pierre Riedinger

Optimal control: 60 discrete states, 30 continuous





Minimize

$\sum_t z(t)^T Q_{i(t)u(t)} z(t)$

Continuous dynamics: $z(t+1) = A_{i(t)u(t)} z(t) \quad z(0) = z_0 \in \mathbf{R}^{30}$

Discrete jumps: $i(t+1) = u(t) \quad i(0) = i_0$

Four steps of approximate value iteration

After four iterations we have one 30×30 matrix P^i for each node such that the following switch law is within a factor 3.81 from optimality:

$$\begin{cases} \text{Jump to node } n & \text{if } z^T[A_{in}^T P^n A_{in} + Q_{in}]z < z^T[A_{im}^T P^m A_{im} + Q_{im}]z \\ \text{Jump to node } m & \text{else} \end{cases}$$

Two versions of relaxed value iteration

$$\min_u [V_j(f(x,u)) + l(x,u)/\alpha] \leq V_{j+1}(x) \leq \min_u [V_j(f(x,u)) + l(x,u)]$$

Decentralized computations!

$$\min_u [V_j(f(x,u)) + l(x,u)/\alpha] \leq V_{j+1}(x) \leq \min_u [V_{j+1}(f(x,u)) + l(x,u)]$$

Global convergence!

If simple approximation exists, we will find one!

Assume V^{S} is "simple" and satisfies

$$\min_u \left[V^*(f(x,u)) + l(x,u)/\alpha\right] \quad \leq V^{\mathsf{S}}(x) \leq \min_u \left[V^{\mathsf{S}}(f(x,u)) + l(x,u)\right]$$

Then $V^*/\alpha < V^{\mathsf{S}} < V^*$ and the following relaxed value iteration with $V_0 = 0$ is feasible in every step:

$$\min_u \left[V_k(f(x,u)) + l(x,u)/\alpha\right] \quad \leq V_{k+1}(x) \leq \min_u \left[V_{k+1}(f(x,u)) + l(x,u)\right]$$

Moreover

$$V^*(x)/\alpha < \limsup_{k\to\infty} V_k(x) < V^*(x)$$

Conclusions of Lecture 1

- Dynamic programming — Off-line controller optimization
- Main limitation: Complexity of optimal value function
- Relaxed Dynamic Programming (with error bounds)
- Next: On-line optimization — Model Predictive Control

[Lincoln and Rantzer, *Relaxing Dynamic Programming*, TAC 51:8, 2006]

[Rantzer, *Relaxing Dynamic Programming in Switching Systems*, IEE Proceeding on Control Theory and Applications, 153:5, 2006]

Course Outline

Tuesday 08:45
- Optimal control
- Dynamic programming
- Bellman's equation
- Value iteration
- Relaxed DynP

Wednesday 08:45
- Tuesday summary
- MPC with terminal cost
- MPC without terminal cost
- Adaptive horizon
- Receding horizon estimation

Tuesday 09:45
- Model Predicitve Control
- Stability
- Terminal constraints
- Introduction to exercise

Wednesday 09:45
- Dual Decomposition
- Distributed MPC
- Gradient methods
- Large-scale systems

Wednesday afternoon: Computer exercise