Continuous Time Markov Decision Processes: Theory, Applications and Computational Algorithms

Peter Buchholz,
Informatik IV, TU Dortmund, Germany

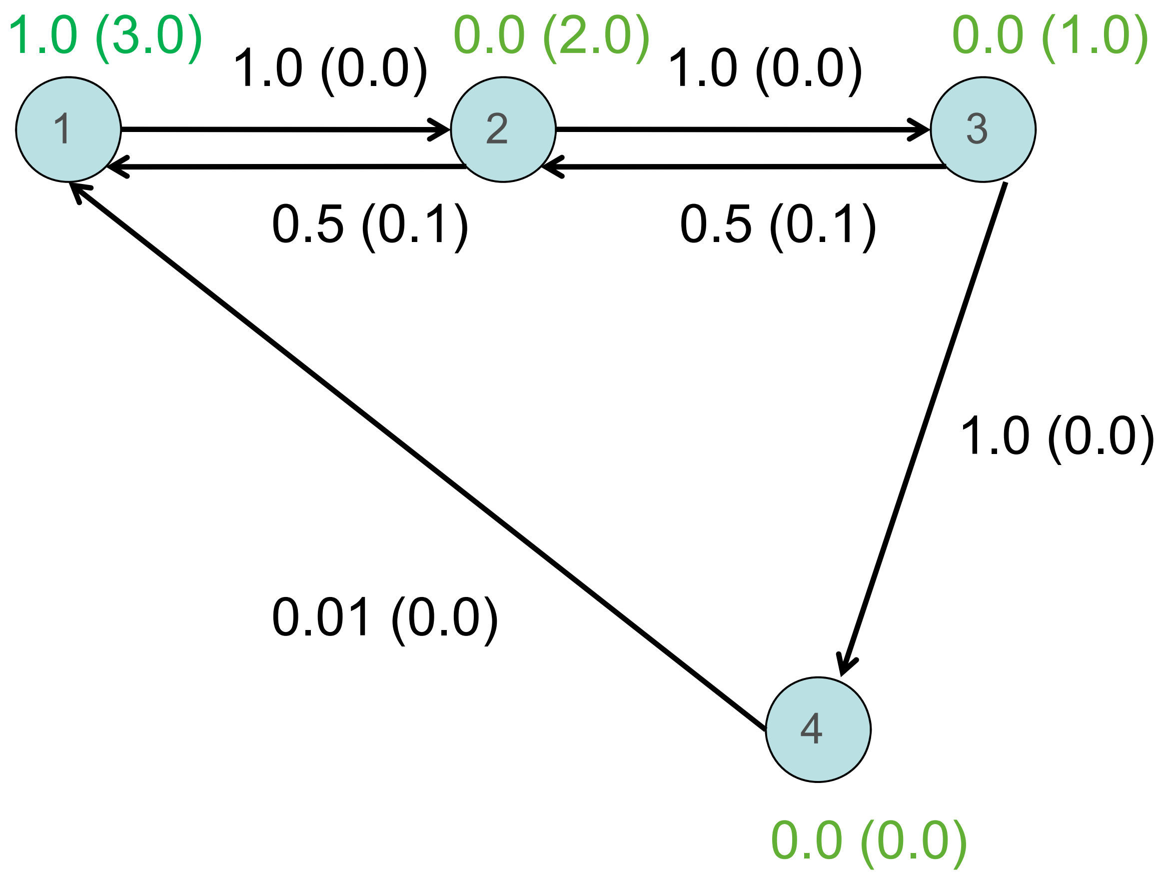
Overview 1

- Continuous Time Markov Decision Processes (CTMDPs)
  - Definition
  - Formalization
  - Applications
- Infinite Horizons
  - Result Measures
  - Optimal Policies
  - Computational Methods

Overview 2

- Finite horizons
  - Result Measures
  - Optimal Policies
  - Computational Methods
- Advanced Topics
  - Model Checking CTMDPs
  - Infinite State Spaces
  - Transition Rate Bounds
  - ….

Continuous Time Markov Chains (CTMCs) with Rewards



Components:

- States
- Initial Probabilities
- Rate Rewards
- Transitions
- Transition Rates
- Impulse Rewards

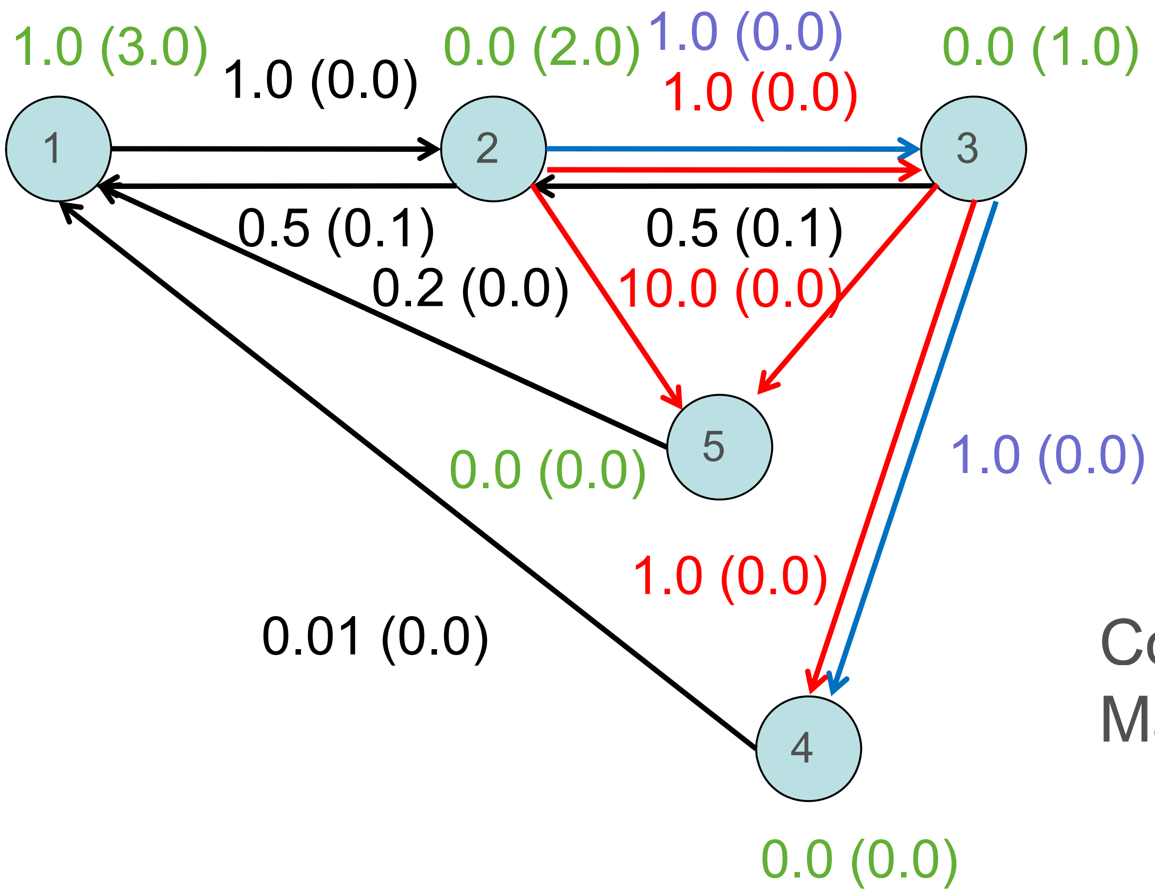
Behavior of the CTMC with rewards:

- Process stays an exponentially distributed time in a state (if it is not absorbing)
- Performs a transition into a successor state (according to the transition rates)
- Accumulates (rate) rewards per time unit in a state and impulse reward per transition

Result measures:

- Average reward accumulated per unit of time in the long run
- Discounted reward
- Accumulated reward during a finite interval
- Accumulated reward before entering a state or subset of states

Extension: Choose between different actions in a state



Decision between

red or blue

in the states 2 and 3!

Continuous Time
Markov Decision Process

Markov Decision Process (with finite state and action spaces)

- State space S = {1,…,n} ($S = \mathbb{Z}_+$ in the countable case)
- Set of decisions $D_i = \{1,\ldots,m_i\}$ for $i \in S$
- Vector of transition rates $\mathbf{q}_i^u \in \mathbb{R}^{1,n}_+$
  where $\mathbf{q}_i^u(j) < \propto$ is the transition rate from i to j ($i \neq j$, $i,j \in S$) under decision $u \in D_i$
  $\Rightarrow$ exponential sojourn time in state i with rate $-\mathbf{q}_i^u(i) = \Sigma_{i \neq j} \mathbf{q}_i^u(j)$
  if decision u is taken, afterwards transition into j with probability $\mathbf{q}_i^u(j) / -\mathbf{q}_i^u(i)$
- $r^u(i) \in \mathbb{R}_+$ the (non-negative, decision dependent) reward in state i, we assume $r^u(i) < \propto$
- $\mathbf{s}^u(i,j)$ the (non-negative, decision dependent) reward of a transition from state i into state j , we assume $\mathbf{s}^u(i,j) < \propto$ and $\mathbf{s}^u(i,j) = 0$ for i=j or $\mathbf{q}_i^u(j) = 0$

Goal: Analysis and control of the system in the interval *[0,T]*

($T = \infty$ is included)

- $\mathbf{d}_t$ is the decision vector at time t where $\mathbf{d}_t(i) \in D_i$
- Decision space $D = X_{i=1..n} D_i$ (size $\Pi_{i=1..n} m_i$)
- $\mathbf{Q}^\mathbf{d} \in \mathbb{R}^{n,n}$ with $\mathbf{Q}^\mathbf{d}(i,j) = \mathbf{q}_i^{\mathbf{d}(i)}(j)$ transition matrix of the CTMC under decision vector **d**
- $\mathbf{r}^\mathbf{d} \in \mathbb{R}^{n,1}$ rate reward vector under decision vector **d**
- $\mathbf{S}^\mathbf{d} \in \mathbb{R}^{n,n}$ impulse reward matrix under decision vector **d**
- $\mathbf{p}_0$ is the initial distribution of the CTMDP at time t=0

Control of CTMDP via policies:

A policy π is a measurable function from *[0,T]* into *D*

(set of all policies $\mathbb{M}$)

$\Rightarrow$ decisions can depend on the time and the state (but not on the history)

π_t defines $\mathbf{d}_t$, the decision vector taken at time t

$\pi_{t,T}$ is the policy π restricted to the interval [t,T]

Policy π is

- **piecewise constant,** iff $0 = t_0 < t_1 < \ldots < t_m = T$ exist such that $\mathbf{d}_t = \mathbf{d}_{t'}$ for $t,t' \in (t_{i-1},t_i]$
- **constant,** iff **d** is independent of *t* (i.e., decisions depend only on the state)

Other forms of policies:

- **randomized policy**: π_t defines a probability distribution over D
- **history dependent**: π_t depends on $(x_{t'}, a_{t'})$ for $0 \le t' < t$

  restricted forms of history dependency

  - **reward dependent**: π_t depends on reward accumulated in $[0,t)$

Policies types can be combined:

E.g., piecewise constant history dependent, constant randomized ….

CTMDP with a fixed policy π: Stochastic Process (X_t, A_t)

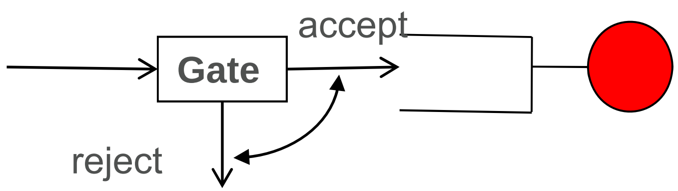
- X_t state process
- A_t action/decision process

Both processes together define

- G_t gain/reward process (i.e., accumulated reward in $[0,t)$)

Behavior of G_t:

- Changes with rate $r^a(i)$ if $X_t = i$ and $A_t = a$
- Makes a jump of height $\boldsymbol{s}^a(i,j)$ if X_t jumps at time t from i to j and $A_t = a$

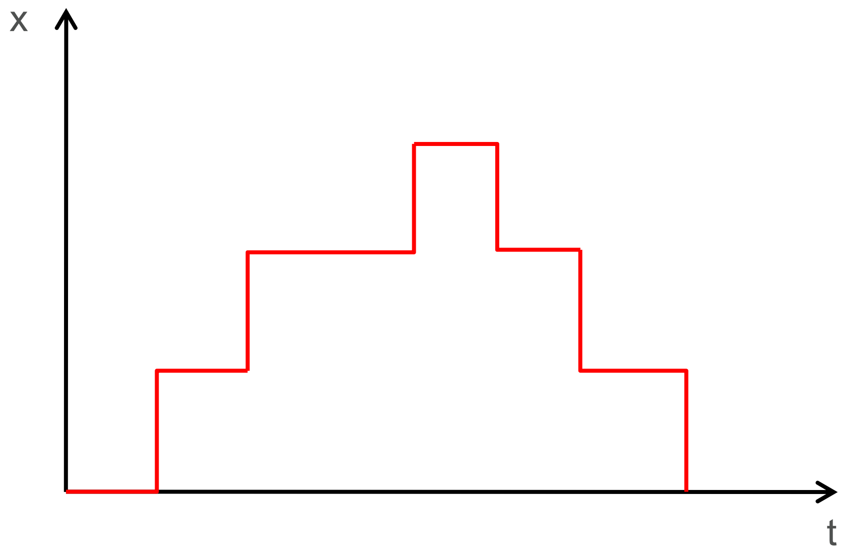


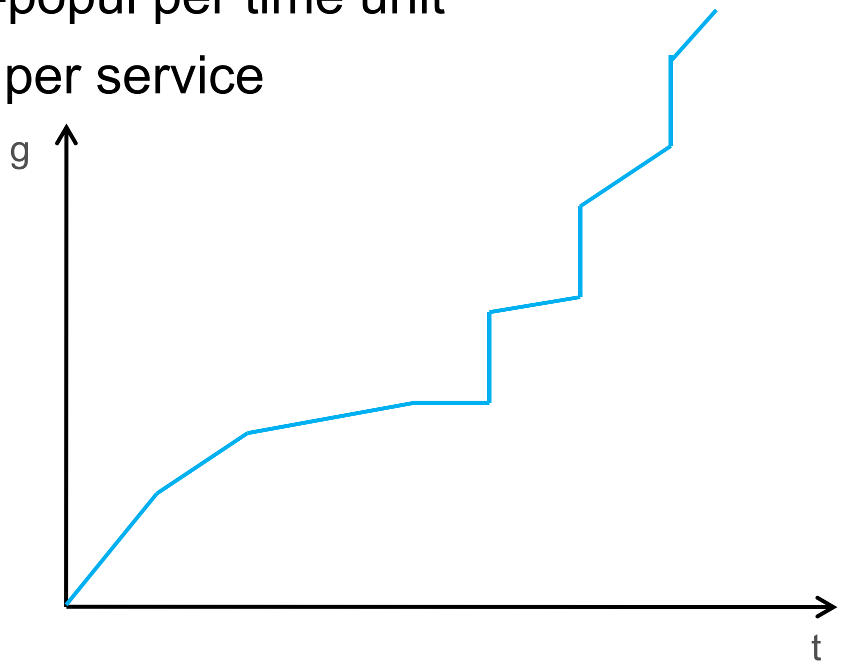
Policy:

- accept if t < 2 and popul < 3
- reject otherwise

Rewards:

- 3-popul per time unit
- 1 per service





Mathematical Basics:

Assume for the moment that policy π is known

Evolution of the state process X_t:

Let $0 \leq t \leq u \leq T$

$$\mathbf{V}^{\pi}_{t,t} = \mathbf{I} \text{ and } \frac{d}{du}\mathbf{V}^{\pi}_{t,u} = \mathbf{V}^{\pi}_{t,u}\mathbf{Q}^{\mathbf{d}_u}$$

$\mathbf{V}^{\pi}_{t,u}(i,j) :=$ Prob. of being in *j* at *u*, if the process is in *i* at *t*

Let $\mathbf{V}^{\pi}_{t} = \mathbf{V}^{\pi}_{0,t}$

We have $\mathbf{p}_u = \mathbf{p}_t\mathbf{V}^{\pi}_{t,u}$ and $\mathbf{p}_t = \mathbf{p}_0\mathbf{V}^{\pi}_{t}$

Evolution of the gain process G_t:

Let $\mathbf{W}^{\mathbf{d}_t}$ be a *n x n* matrix with:

$$\mathbf{W}^{\mathbf{d}_t}(i,j) = \begin{cases} r^{\mathbf{d}_t(i)}(i) & \text{if } i = j \\ s^{\mathbf{d}_t(i)}(i,j) q_i^{\mathbf{d}_t}(j) & \text{otherwise} \end{cases} \qquad \mathbf{w}^{\mathbf{d}_t}(i) = \sum_{j=1}^{n} \mathbf{W}^{\mathbf{d}_t}(i,j)$$

Then $-\frac{d}{dt}\mathbf{g}_t^\pi = \mathbf{w}^\pi + \mathbf{Q}^{\mathbf{d}_t}\mathbf{g}_t^\pi$ (backwards in time!)

$\mathbf{g}_{t,T}^\pi$ is the accumulated gain in the interval [t,T] and

$\mathbf{g}_T^\pi$ the final gain at time T

such that $\mathbf{g}_{t,T}^\pi = \mathbf{V}_{t,T}^\pi \mathbf{g}_T^\pi + \int_t^T \mathbf{V}_{u,T}^\pi \mathbf{w}^\pi du$

For $(T-t) \rightarrow \infty$ usually also $\mathbf{g}^{\pi}_{t,T}(i) \rightarrow \infty$ holds

Results for *[0,∞]* are not meaningful in this case!

Alternative ways of defining the gain vector

- Time averaged gain

$$\mathbf{g}^{\pi}_{t,T} = \frac{1}{T-t}\left(\mathbf{V}^{\pi}_{t,T}\mathbf{g}^{\pi}_{T} + \int_t^T \mathbf{V}^{\pi}_{u,T}\mathbf{w}^{\pi} du\right) \text{ for } T > t$$

- Discounted gain

$$\mathbf{g}^{\pi}_{t,T} = e^{-\beta(T-t)}\mathbf{V}^{\pi}_{t,T}\mathbf{g}^{\pi}_{T} + \int_t^T e^{-\beta(u-t)}\mathbf{V}^{\pi}_{u,T}\mathbf{w}^{\pi} du$$

for discount factor $\beta > 0$

From analysis to optimization problems:

Find a policy π (from a specific class) that maximizes/minimizes the gain

Optimal gain $\mathbf{g}_{0,T}^{+} = \sup_{\pi \in \mathcal{M}} \left(\mathbf{g}_{0,T}^{\pi}\right)$ (or $\mathbf{g}_{0,T}^{-} = \inf_{\pi \in \mathcal{M}} \left(\mathbf{g}_{0,T}^{\pi}\right)$)

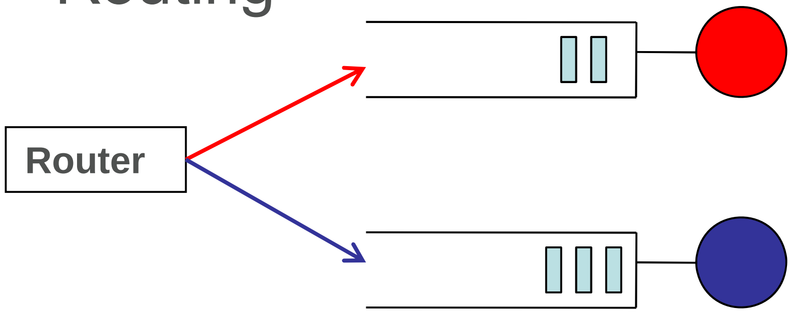
Optimal policy $\pi^{+} = \arg \sup_{\pi \in \mathcal{M}} \left(\mathbf{g}_{0,T}^{\pi}\right)$ (or $\pi^{-} = \arg \inf_{\pi \in \mathcal{M}} \left(\mathbf{g}_{0,T}^{\pi}\right)$)

π^+/π^- need not be unique!

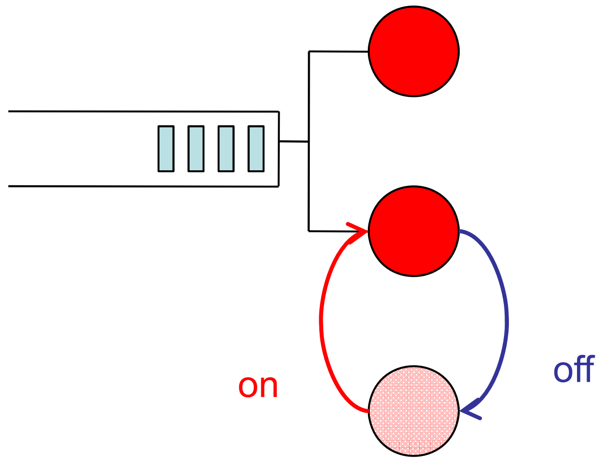
Policy π is ε-optimal iff $\left\| \mathbf{g}_{0,T}^{\pi^{\pm}} - \mathbf{g}_{0,T}^{\pi} \right\| \leq \epsilon$

Examples (queuing networks)

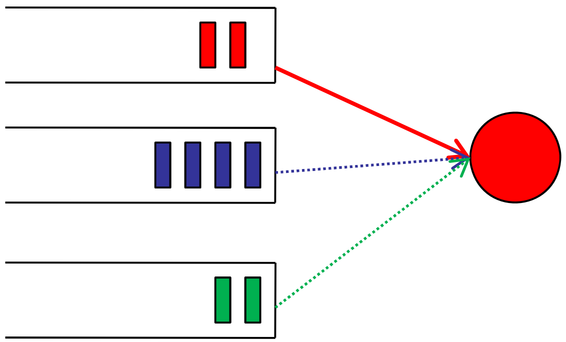
Routing



Resource allocation/deallocation

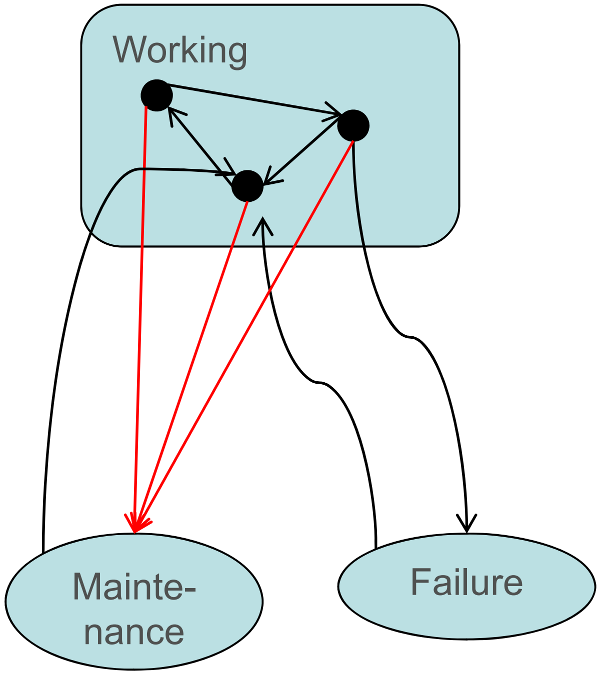


Scheduling

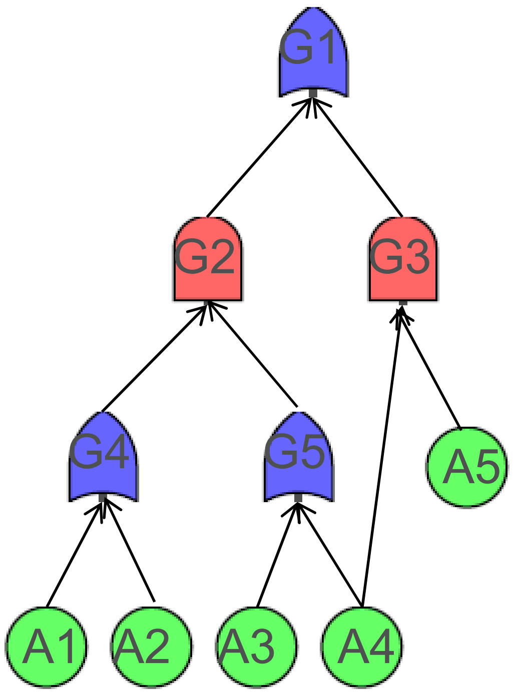


Examples (reliability, availability)

Components



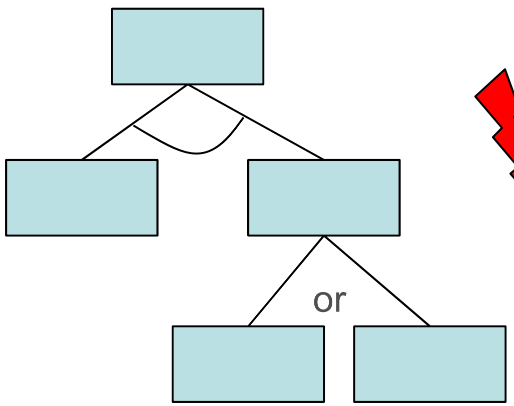
System



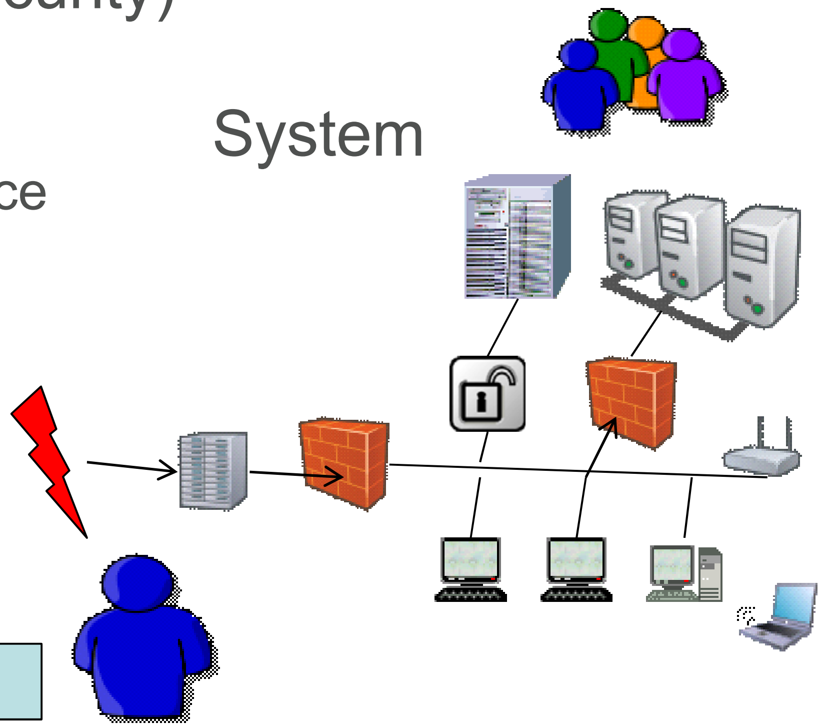
Find a maintenance policy to minimize system down time

Examples (security)

Attack tree
(specifies sequence
of attack steps)



System

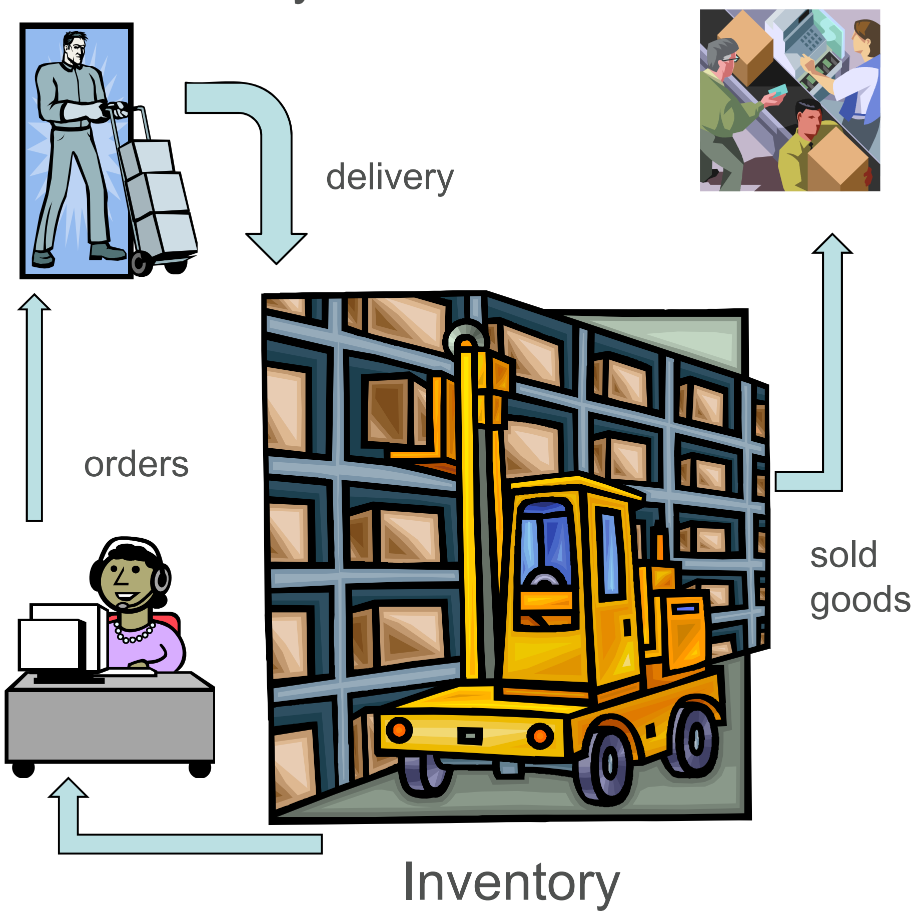


Adversary: Find attack steps to reach goal

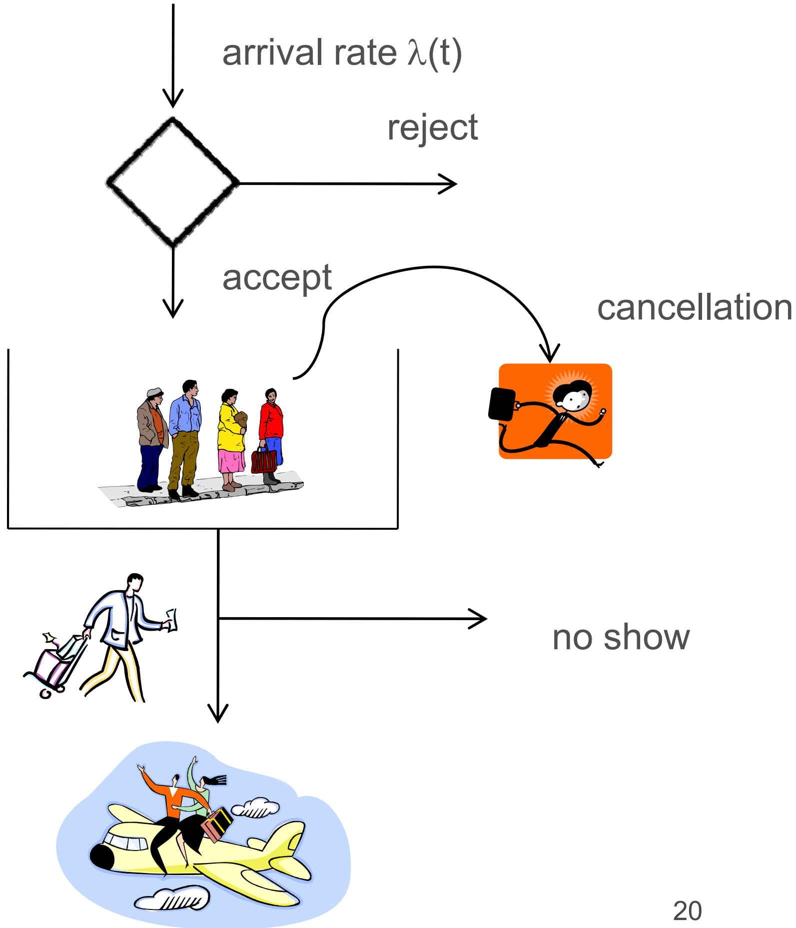
Defender: Find mechanisms to defend the system

Examples (OR)

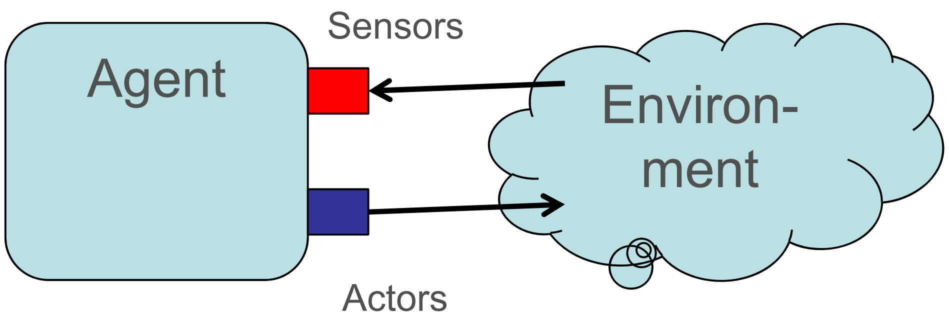
Inventory control



Airline yield management



Examples (AI)



Agent and environment are modeled as CTMDPs

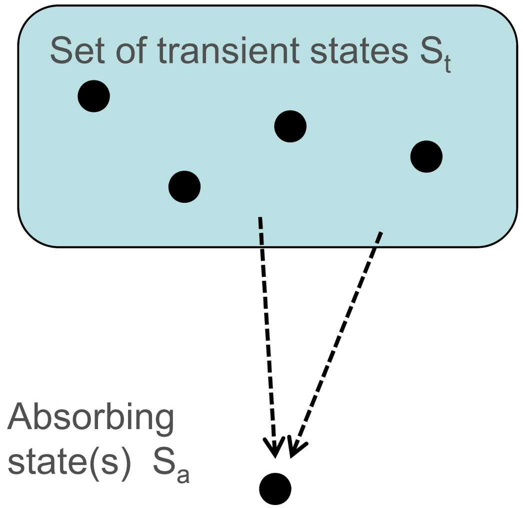
Optimal policy corresponds to an “optimal” behavior of the agent

CTMDPs on Inifinite Horizons

We consider a CTMDP in the interval $[0,\infty)$

- Result Measures
  - Reward to Absorption
  - Average Reward
  - Discounted Reward
- Optimal Policies
- Computational Methods

Reward to Absorption



Assumptions:

- $\mathbf{w}^{\mathbf{d}}(i) \geq 0$ for $i \in S_t$ and 0 for $i \in S_a$ and all $\mathbf{d} \in D$
- $\lim_{t \to \infty} \sum_{j \in S_a} \mathbf{V}_t^{\pi}(i,j) = 1$ for all $i \in S$ and all $\pi \in \mathbb{M}$

Goal:

Find a policy π such that $\mathbf{g}^{\pi} = \int_0^{\infty} \mathbf{V}_t^{\pi} \mathbf{h}^{\pi} dt$ minimal/maximal in all components

Average Reward

Find a policy π such that $\mathbf{g}^\pi = \lim_{T\to\infty} \frac{1}{T}\left(\int_0^T \mathbf{V}_t^\pi \mathbf{w}^\pi dt\right)$

is maximal/minimal in all components

No further assumptions necessary ($\lim_{t\to\infty} (\mathbf{V}_t^\pi)$ exists in our case!)

Discounted Reward

Find a policy π such that $\mathbf{g}^\pi = \lim_{T\to\infty} \left(\int_0^T e^{-\beta t}\mathbf{V}_t^\pi \mathbf{w}^\pi dt\right)$

is maximal/minimal in all components

(discount factor β ≥ 0)

Optimal Policies:

Stationary policies π_1 and π_2 exist such that

$$\mathbf{g}^{\pi_1} = \sup_{\pi \in \mathcal{M}} (\mathbf{g}^{\pi}) \text{ and } \mathbf{g}^{\pi_2} = \inf_{\pi \in \mathcal{M}} (\mathbf{g}^{\pi})$$

in the cases we consider here

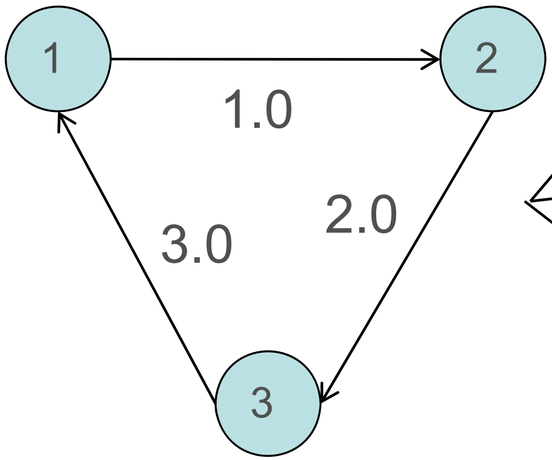
The optimal policies might not be unique such that other criteria can be applied to rank policies with identical gain vectors
(we do not go in this direction here)

Computational Methods

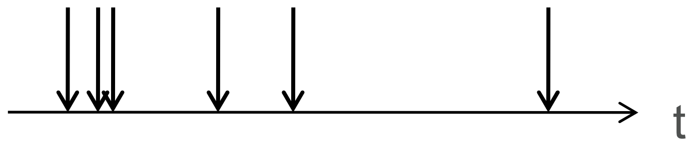
- Basic step is the transformation of the CTMDP into an equivalent DTMDP (plus a Poisson process) using uniformization
- Afterwards methods for computing the optimal policy/gain vector in DTMDPs can be applied (Poisson process is not needed)

The uniformization approach:

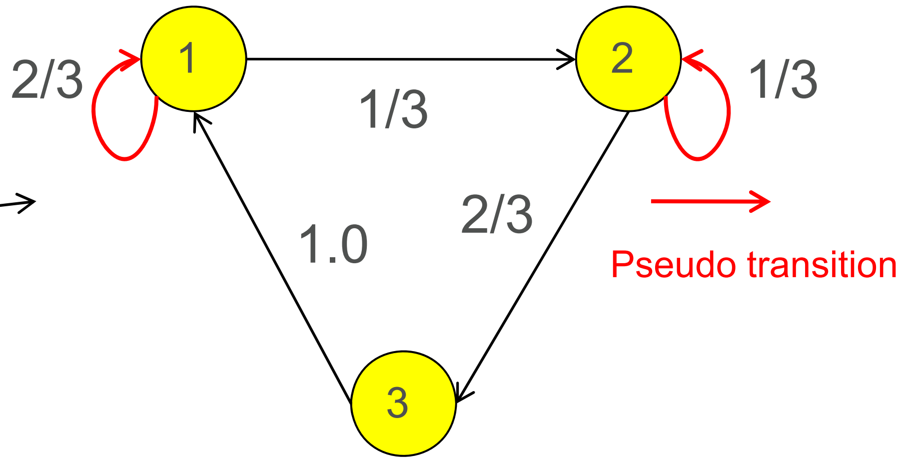
CTMC for a fixed decision vector **d**



1) Poisson process with rate $\alpha \geq \max_{\mathbf{d} \in D} \max_{i \in S} (-\mathbf{Q}^{\mathbf{d}}(i,i))$ for the timing



2) DTMC $\mathbf{P}^{\mathbf{d}}=\mathbf{Q}^{\mathbf{d}}/\alpha + \mathbf{I}$ for transitions



3) Transformed rewards

$$\mathbf{w'}^{\mathbf{d}}(i) = \mathbf{w}^{\mathbf{d}}(i)/\alpha \text{ or } \mathbf{w'}^{\mathbf{d}}(i)/(\alpha+\beta)$$

discount factor $\beta' = \alpha / (\alpha+\beta)$

Uniformization transforms a CTMDP = ($\mathbf{Q}^{\mathbf{d}}$, $\mathbf{p}_0$, $\mathbf{w}^{\mathbf{d}}$, S, D) into a

DTMDP = ($\mathbf{P}^{\mathbf{d}}$, $\mathbf{p}_0$, $\mathbf{w}'^{\mathbf{d}}$, S, D) with an identical

- optimal gain vector **g*** and
- optimal stationary policy π* (described by vector **d***)

The optimal gain vector is the solution of the following equation for the discounted and total reward

$$\mathbf{g}^*(i) = \max_{u \in \mathcal{D}_i} \left(\mathbf{w'}^u(i) + \beta' \sum_{j=1}^{n} \mathbf{P}^u(i,j)\mathbf{g}^*(j) \right) \text{ for } i = 1, \ldots, n$$

Vector **d*** results from setting **d***(i) to *argmax* in the equation

For the optimization of average rewards, we restrict the model class to *unichain* models :

A DTMDP is unichain if for every **d** $\in$ D matrix $\mathbf{P}^{\mathbf{d}}$ contains a single recurrent class of states (plus possibly some transient states)

For unichain DTMDPs the average reward is constant for all states and observes the following equations

$$\rho^* + \mathbf{h}^*(i) = \max_{u \in \mathcal{D}_i} \left(\mathbf{w'}^u(i) + \sum_{j=1}^{n} \mathbf{P}^u(i,j)\mathbf{h}^*(j) \right) \text{ for } i = 1, \ldots, n$$

Numerical methods to compute the optimal gain vector + policy

- linear programming
- value iteration
- policy iteration

+ combinations of value and policy iteration

Practical problems

- Curse of dimensionality, state space explosion
- Slow convergence, long solution times

Linear Programming

Discounted Reward Case

$$\mathbf{g} \leq \max_{\mathbf{d}\in D} \left(\mathbf{w'}^{\mathbf{d}} + \beta' \mathbf{P}^{\mathbf{d}} \mathbf{g} \right) \leq$$

$$\mathbf{g}^* = \mathbf{w'}^{\mathbf{d}^*} + \beta' \mathbf{P}^{\mathbf{d}^*} \mathbf{g}^*$$

g* is the largest **g** that satisfies

$$\mathbf{g} \leq \max_{\mathbf{d}\in D} \left(\mathbf{w'}^{\mathbf{d}} + \beta' \mathbf{P}^{\mathbf{d}} \mathbf{g} \right)$$

$$\text{LP} \;\; \max \left(\sum_{i\in S} x_i \right)$$

subject to

$$x_i \leq \mathbf{w'}^{u}(i) + \beta' \sum_{j=1}^{n} \mathbf{P}^u(j,i) x_j$$

for all $i \in S, u \in D_j$

Average Reward Case

$$\max \left(\sum_{i\in S} \sum_{u\in D_i} x_i^u \mathbf{w'}^{u}(i) \right)$$

subject to

$$\sum_{u\in D_i} x_i^u - \sum_{j\in S} \sum_{u\in D_i} \mathbf{P}^u(j,i) x_j^u = 0$$

$$\sum_{i\in S} \sum_{u\in D_i} x_i^u = 1$$

for all $i \in S, u \in D_i$

- **d*** and **g*** (ρ* and **h***) can be derived from the results of the LP x_i^u
- An LP with
  - n or $\sum_{i=1}^{n} |D_i|$ variables and
  - $\sum_{i=1}^{n} |D_i|$ or n+1 constraints

  has to be solved (this can be time and memory consuming!)

Usually Linear Programming is not the best choice to compute optimal policies for MDP problems!

Value iteration

Initialize $k=0$ and $\mathbf{g}^{(k)} \geq 0$, then iterate until convergence

$$\mathbf{g}^{(k+1)}(i) = \max_{u \in \mathcal{D}_i} \left(\mathbf{w}'(i) + \beta' \sum_{j=1}^{n} \mathbf{P}^u(i,j) \mathbf{g}^{(k)}(j) \right)$$

or

$$\mathbf{h}^{(k+1)}(i) = \max_{u \in \mathcal{D}_i} \left(\mathbf{w}'(i) + \sum_{j=1}^{n} \mathbf{P}^u(i,j) \mathbf{h}^{(k)}(j) \right) - \max_{u \in \mathcal{D}_n} \left(\mathbf{w}'(n) + \sum_{j=1}^{n} \mathbf{P}^u(n,j) \mathbf{h}^{(k)}(j) \right)$$

Repeated vector matrix products

for all i = 1,….,n

- Easy to implement
- Convergence towards the optimal gain vector and policy

  stop if policy does not change and $||\mathbf{g}^{(k)} - \mathbf{g}^{(k-1)}|| < \varepsilon$
- Effort per iteration: $n \cdot nz \cdot \sum_{i=1}^{n} m_i$

  where *nz* is the average number of non-zeros in a row of $\mathbf{P}^{\mathbf{d}}$
- Often slow convergence and a huge effort even for CTMDPs of a moderate size

Policy iteration

Initialize $k=0$ and initial policy $\mathbf{d}_0$, then iterate until convergence

$$\mathbf{g}^{(k)} = \mathbf{w'}^{\mathbf{d}_k} + \beta' \mathbf{P}^{\mathbf{d}_k} \mathbf{g}^{(k)} \quad \text{or}$$

$$\mathbf{h}^{(k)} + \rho^k \mathbf{e} = \mathbf{w'}^{\mathbf{d}_k} + \beta' \mathbf{P}^{\mathbf{d}_k} \mathbf{h}^{(k)} \text{ and } \mathbf{h}^{(k)}(n) = 0$$

Repeated solution of sets of linear equations

and

$$\mathbf{d}_{k+1} = \arg\max_{\mathbf{d} \in \mathcal{M}} \left(\mathbf{w'}^{\mathbf{d}} + \beta' \mathbf{P}^{\mathbf{d}} \mathbf{g}^{(k)} \right) \quad \text{or}$$

$$\mathbf{d}_{k+1} = \arg\max_{\mathbf{d} \in \mathcal{M}} \left(\mathbf{w'}^{\mathbf{d}} + \mathbf{P}^{\mathbf{d}} \mathbf{h}^{(k)} \right)$$

until $\mathbf{d}_{k+1} = \mathbf{d}_k$

- Optimal policy and gain vector is computed after finitely many steps
- Effort per iteration: $O(n^3) + O(n \cdot nz \cdot \sum_{i=1}^{n} m_i)$
- For larger state spaces huge effort

  (solution of a set of linear equations of order n)

Improvements by combining policy and value iteration

Initialize *k=0* and initial policy $\mathbf{d}_0$, then iterate until convergence

$$\mathbf{g}^{(k)} \Leftarrow iterate\left(\left(\mathbf{I} - \beta'\mathbf{P}^{\mathbf{d}_k}\right)\mathbf{g} = \mathbf{w}'^{\mathbf{d}_k}\right) \quad \text{or}$$

$$\mathbf{h}^{(k)} + \rho^k \Leftarrow iterate\left(\left(\mathbf{I} - \mathbf{P}^{\mathbf{d}_k}\right)\mathbf{h} = \mathbf{w}'^{\mathbf{d}_k} \text{ subject to } \mathbf{h}^{(k)}(n) = 0\right)$$

where iterate is some advanced iteration techniques like SOR, ML, GMRES….and

$$\mathbf{d}_{k+1} = \arg\max_{\mathbf{d}\in\mathcal{M}}\left(\mathbf{w}'^{\mathbf{d}} + \beta'\mathbf{P}^{\mathbf{d}}\mathbf{g}^{(k)}\right) \quad \text{or}$$

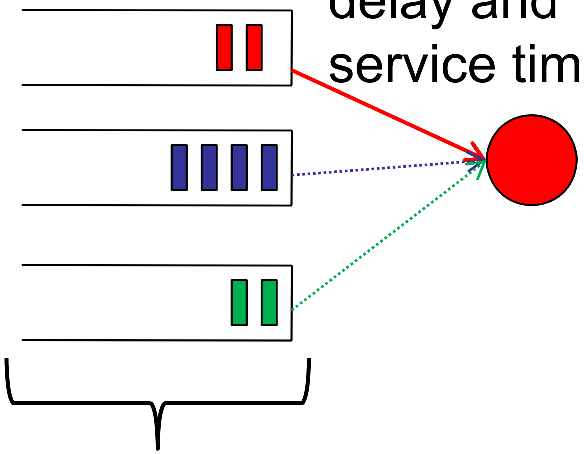
$$\mathbf{d}_{k+1} = \arg\max_{\mathbf{d}\in\mathcal{M}}\left(\mathbf{w}'^{\mathbf{d}} + \mathbf{P}^{\mathbf{d}}\mathbf{h}^{(k)}\right)$$

until $\mathbf{d}_{k+1} = \mathbf{d}_k$ and $||\mathbf{g}^{(k)} - \mathbf{g}^{(k-1)}|| < \varepsilon$ or $||\mathbf{h}^{(k)} - \mathbf{h}^{(k-1)}|| < \varepsilon$

Example:

Exponential class switching delay and service times

Poisson or IPP arrivals



Finite buffer capacity over all classes

Nonlinear reward function:

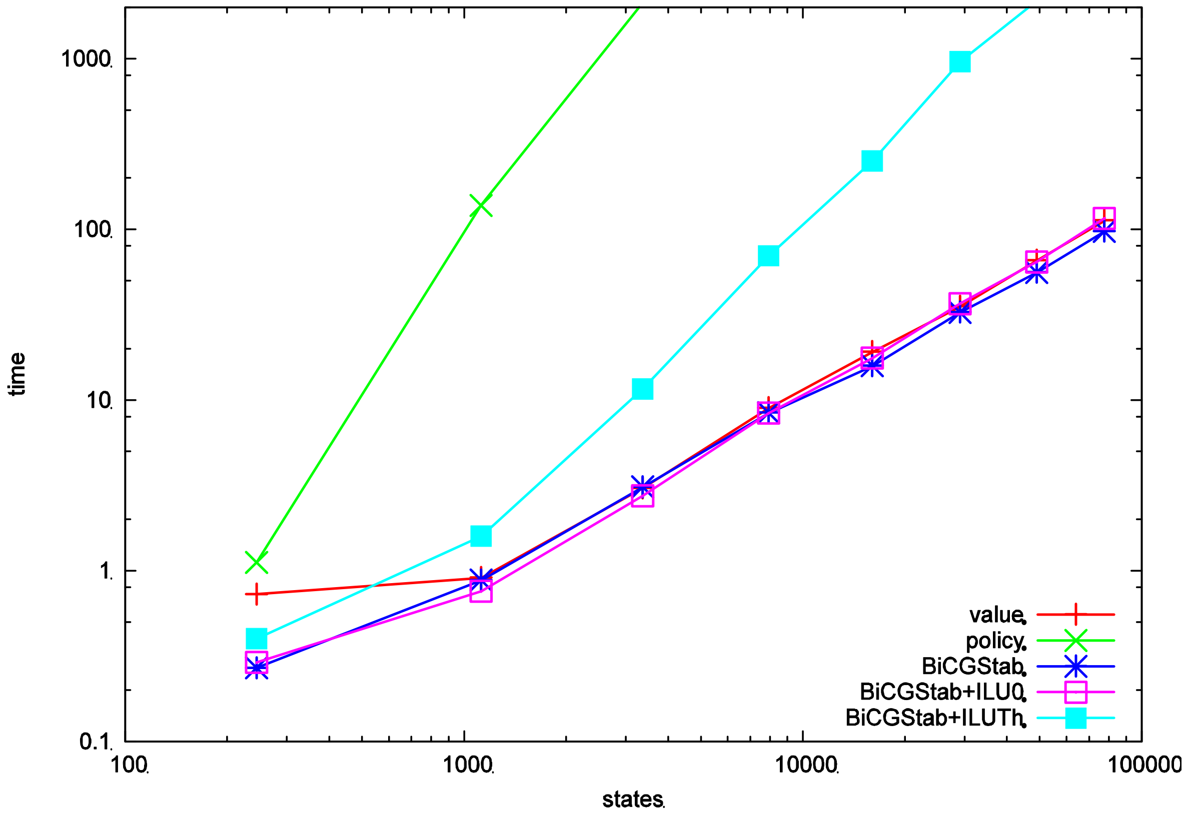
$$(n_1 + n_2 + n_3)^{1.2} + n_1 + 1.5n_2 + 2n_3$$

Determination of the best non-preemptive scheduling strategy in steady state

Methods

- Value iteration
- Policy iteration
- Combined approach with BiCGStab or GMRES + ILU preconditioning

Exponential interarrival times (fast convergence of all solvers)



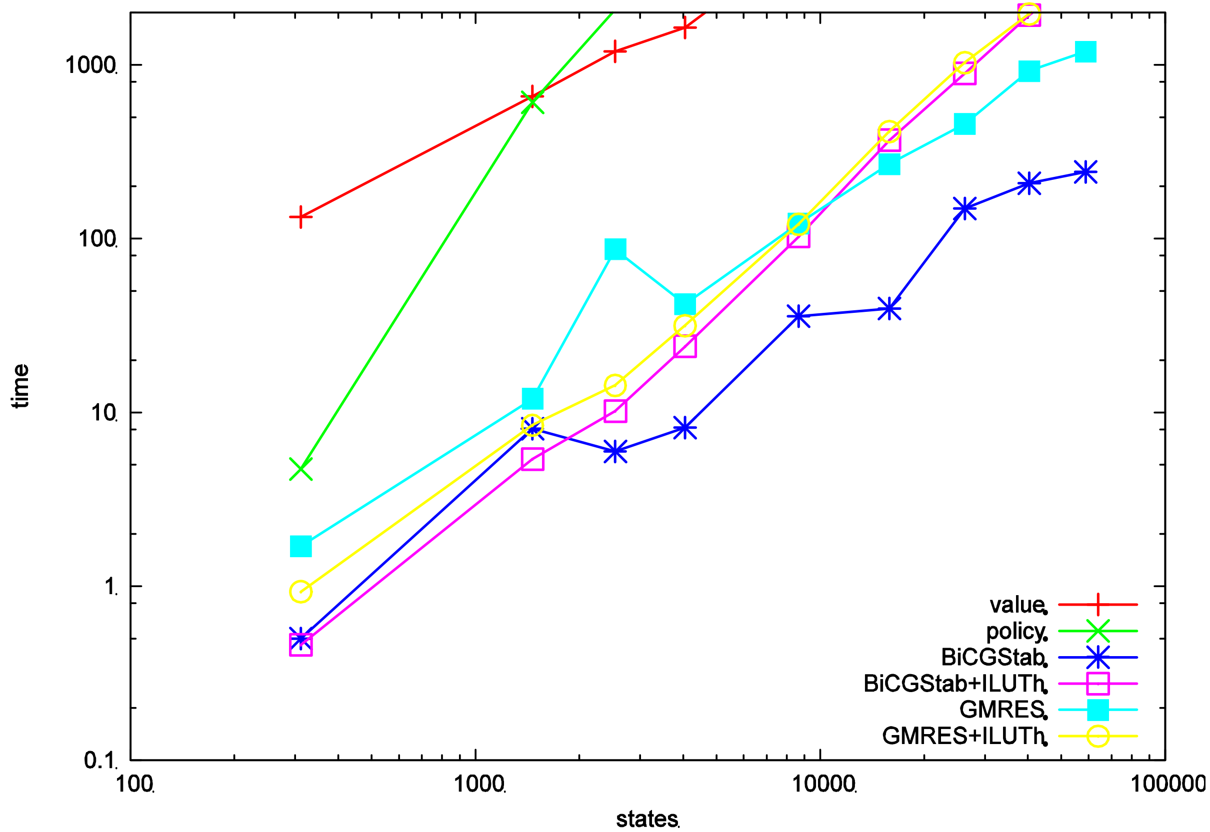
No need for

advanced solvers

value iteration

works fine!

IPP interarrival times (systems are much harder to solve)



Advanced solvers are much more efficient for larger configurations!

Some remarks:

- Analysis for infinite horizons in principle well understood
- Complex models with larger state spaces require sophisticated solution methods
- Work on numerical methods less developed than in CTMC/DTMC analysis
    - Which method to use?
    - Which preconditioner?
    - How many iterations?
    - …….

CTMDPs on Finite Horizons

We consider a CTMDP in the interval $[0,T]$ $(T < \infty)$

- Result Measures
  - Accumulated Reward
  - Accumulated Discounted Reward
- Optimal Policies
- Computational Methods

Many problems are defined naturally on a finite horizon, e.g.,

- yield management (for a specific tour)
- scheduling (for a specific set of processes)
- maintenance (for a system with finite operational periods)
- …

use of the optimal stationary policy is suboptimal

(e.g., maintenance for a machine just before it is shut down)

Optimal policies for DTMDPs on a horizon of T steps

Compute iteratively

$$\mathbf{g}^{(k+1)}(i) = \max_{u \in \mathcal{D}_i} \left(\mathbf{w}^u(i) + \beta \sum_{j=1}^{n} \mathbf{P}^u(i,j)\mathbf{g}^{(k)}(j) \right)$$

for k=1,2,…,T starting with $\mathbf{g}^{(0)}(i) = 0$ for all $i=1,\ldots,n$

where $\beta = 1.0$ for the case without discounting

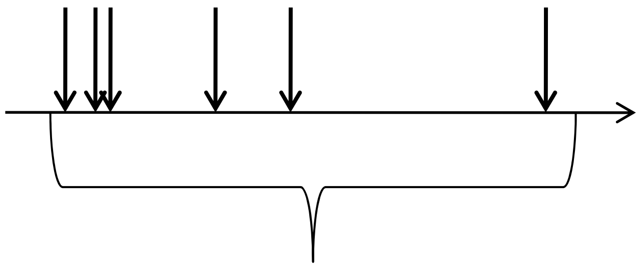
Effort $O(T \cdot n \cdot nz \cdot (\Sigma_{i=1..n} m_i))$

How about applying the DTMC resulting from uniformization for the transient analysis of CTMDPs?

- Times between transitions in the uniformized DTMC are exponentially distributed with rate a rather than constant
- Substitution of the distribution by the mean as done in the stationary case does not work
- It has been shown that the approach computes the optimal policy for uniformization parameter $\alpha \rightarrow \propto$
  (but this results in $O(\alpha T)$ steps to cover the interval $[0,T]$)

Uniformization revisited

Poisson process with rate
$\alpha \geq \max_{\mathbf{d} \in D} \max_{i \in S} (-\mathbf{Q}^{d}(i,i))$
for the timing



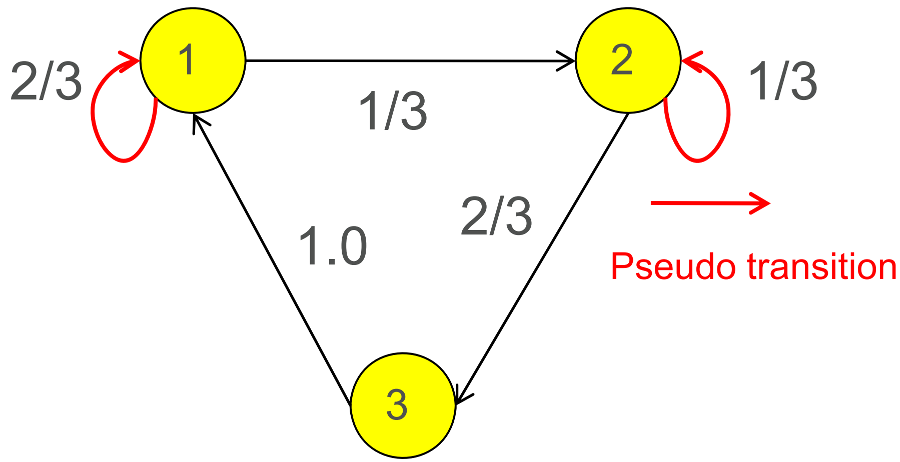
Probability for k events in $[t-\delta,t]$

$\gamma(t, k) = e^{-\alpha t} t^{k} / k!$

Probability for >k events

$1 - \Sigma_{l=0..k} \gamma(t, k)$

Each event equals a transition in the DTMC



Known results for CTMDPs on finite horizons

(by Miller 1968, Lembersky 1974, Lippman 1976 all fairly old!)

- A policy is optimal if it maximizes for almost all $t \in [0,T]$

$$\max_{\pi \in \mathcal{M}} \left(\mathbf{Q}^{\pi} \mathbf{g}_t + \mathbf{w}^{\pi} \right) \text{ where } -\frac{d}{dt}\mathbf{g}_t = \mathbf{Q}^{\mathbf{d}_t} \mathbf{g}_t + \mathbf{w}^{\mathbf{d}_t} \text{ and } \mathbf{g}_T \geq \mathbf{0}$$

- There exists a piecewise constant policy π^* which results in vector $\mathbf{g}^*_t$ and maximizes the equation
  (a policy is piecewise constant, if $m<\infty$ and
  $0=t_0<t_1<\ldots<t_m=T$ exist and $\mathbf{d}_i$ is the decision vector in $[t_{i-1},t_i)$)
  i.e., the optimal policy depends on the time and the state but changes only finitely often in [0,T]

Selection of an optimal policy (using results of Miller 1968)

Assume that $\mathbf{g}^*_t$ is known, then the following selection procedure select $\mathbf{d}^*$ which is optimal in $(t-\delta^*, t)$ for some $\delta^*>0$

Define the sets

$F_1(\mathbf{g}^*_t) = \{\mathbf{d} \in D \mid \mathbf{d} \text{ maximizes } \mathbf{q}^{(1)}(\mathbf{d})\}$

$F_2(\mathbf{g}^*_t) = \{\mathbf{d} \in F_1(\mathbf{g}^*_t) \mid \mathbf{d} \text{ maximizes } -\mathbf{q}^{(2)}(\mathbf{d})\}$

…

$F_{n+1}(\mathbf{g}^*_t) = \{\mathbf{d} \in F_n(\mathbf{g}^*_t) \mid \mathbf{d} \text{ maximizes } (-1)^n\mathbf{q}^{(n+1)}(\mathbf{d})\}$

where $\mathbf{q}^{(1)}(\mathbf{d}) = \mathbf{Q}^{\mathbf{d}}\mathbf{g}^*_t + \mathbf{w}^{\mathbf{d}}$

$\mathbf{q}^{(j)}(\mathbf{d}) = \mathbf{Q}^{\mathbf{d}}\mathbf{q}^{(j-1)}$ where $\mathbf{q}^{(j-1)} = \mathbf{q}^{(j-1)}(\mathbf{d})$ for any $\mathbf{d} \in F_{j-1}(\mathbf{g}^*_t)$

Select the lexicographically smallest vector from $F_{n+1}(\mathbf{g}^*_t)$

Constructive proof in Miller 1968 defines a base for an algorithm:

1. Set t' =T and initialize $\mathbf{g}_T$;
2. Select $\mathbf{d}_{t'}$ as described ;
3. Obtain $\mathbf{g}_t$ for $t \leq t' \leq T$ by solving

$$-\frac{d}{dt}\mathbf{g}_t = \mathbf{Q}^{\mathbf{d}_t}\mathbf{g}_t + \mathbf{w}^{\mathbf{d}_t}$$

   with terminal condition $\mathbf{g}_{t'}$;
4. Set t'' = inf{t | $\mathbf{d}_t$ satisfies the selection procedure} ;
5. If $t'' \leq 0$ terminate, else goto 2. with t' = t'' ;

Not implementable!

From exact to approximate optimal policies:

A policy π is ε-optimal if $||\mathbf{g}^*_t - \mathbf{g}^{\pi}_t||_\infty \leq \varepsilon$ for all $t \in [0,T]$

<u>Discretization approach</u>:

Define for h: $\mathbf{P}_h^{\mathbf{d}} = \mathbf{I} + h\mathbf{Q}^{\mathbf{d}}$

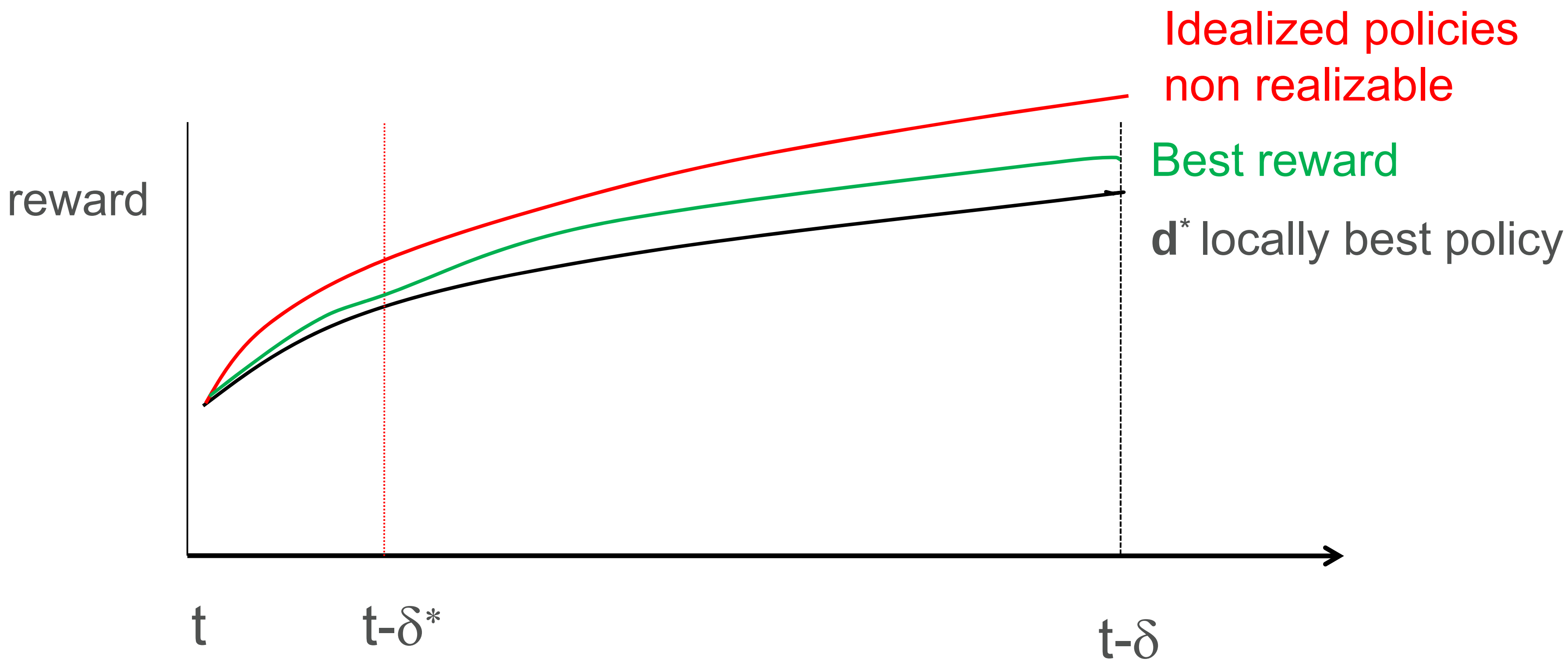
for *h* small enough this defines a stochastic matrix

Let $\mathbf{g}_{t-h} = \mathbf{P}_h^{\mathbf{d}}\mathbf{g}_t + h\mathbf{w}^{\mathbf{d}} + o(h)$

Representation of a DTMDP for policy optimization

- For $h \to 0$ the computed policy is ε-optimal with $\varepsilon \to 0$
- Value of ε is unknown, effort $O(h^{-1} \cdot n \cdot nz \cdot (\Sigma_{i=1..n} m_i))$

Idea of the uniformization based approach



A uniformization based approach

Basic steps

1. Start with t=T and $\mathbf{g}_T=\mathbf{g}^-_T=\mathbf{g}^+_T=\mathbf{0}$;
2. Compute an optimal decision vector **d** based on $\mathbf{g}^-_t$;
3. Compute a lower bound for $\mathbf{g}^-_{t-\delta}$ using decision **d** in $(t-\delta,t)$;
4. Compute an upper bound $\mathbf{g}^+_{t-\delta}$ for any policy in $(t-\delta,t)$;
5. Choose δ such that $\| \mathbf{g}^+_{t-\delta} - \mathbf{g}^-_{t-\delta} \| < \varepsilon (t - \delta) / T$;
6. Store **d** and $t-\delta$;
7. If $t-\delta > 0$ then set $t = t-\delta$ and goto 2; else terminate ;

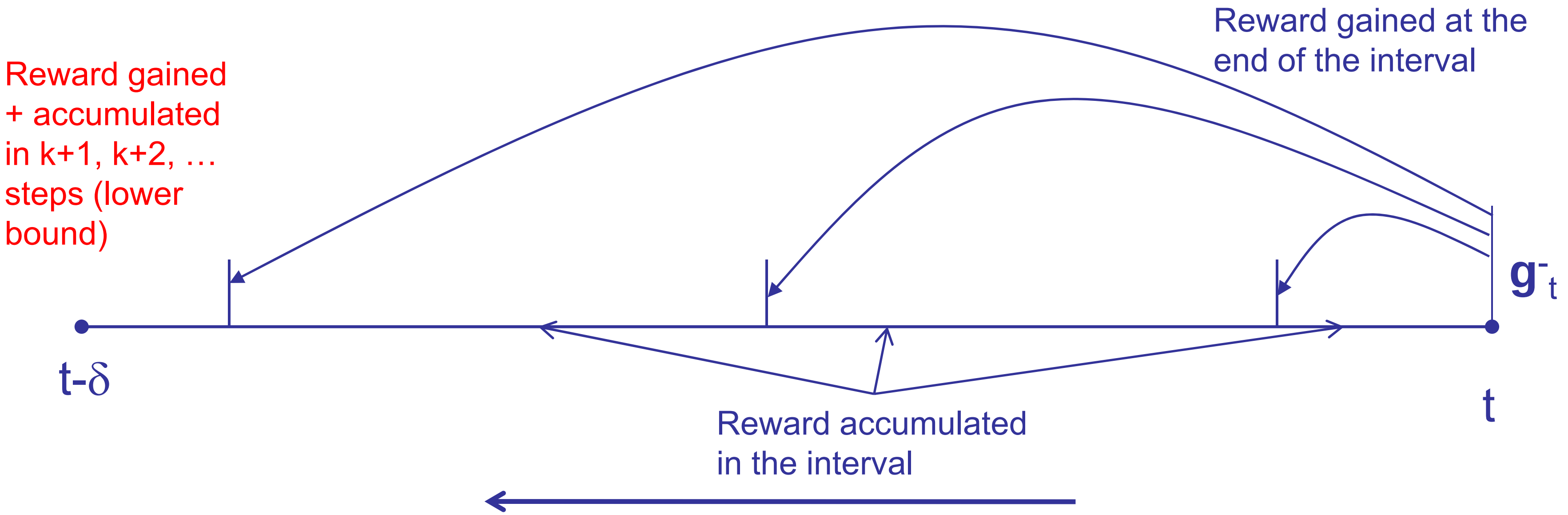
All this has to be computable!

Computation of the lower bound $\mathbf{g}^-_{t-\delta}$ using decision **d** in $[t-\delta,t)$;

d is the optimal decision at t based on $\mathbf{g}^-_t$

Since **d** is constant, this equals the transient analysis of a CTMC which can be computed using unformization

Consider up to K Poisson steps (Prob. known!)



Formally, we get:

$$\mathbf{v}_{-}^{(k)} = \mathbf{P}^{\mathbf{d}}\mathbf{v}_{-}^{(k-1)} \text{ with } \mathbf{v}_{-}^{(0)} = \mathbf{g}_t^{-} \text{ and } \mathbf{w}_{-}^{(k)} = \mathbf{P}^{\mathbf{d}}\mathbf{w}_{-}^{(k-1)} \text{ with } \mathbf{w}_{-}^{(0)} = \mathbf{w}^{\mathbf{d}}$$

Then

$$\mathbf{g}_{t-\delta}^{-} = \sum_{k=0}^{K} \gamma(\alpha\delta, k)\mathbf{v}_{-}^{(k)} + \frac{1}{\alpha}\sum_{k=0}^{K}\left(1 - \sum_{l=0}^{k}\gamma(\alpha\delta, l)\right)\mathbf{w}_{-}^{(k)} + \eta(\alpha\delta, \mathbf{w}^{\mathbf{d}}, \mathbf{g}_t^{-})$$

where $\eta(\alpha\delta, \boldsymbol{w}^{\boldsymbol{d}}, \boldsymbol{g}^{-}{}_{t})$ bounds the missing Poisson probabilities

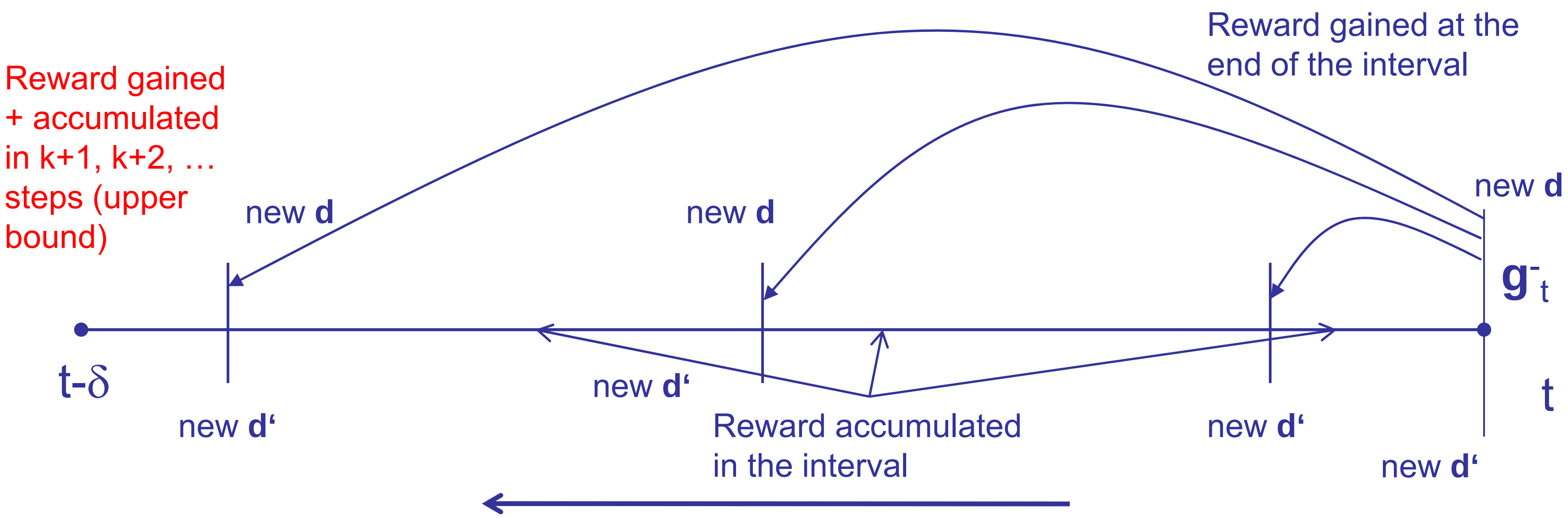
Effort

- for vector computation in O(K·n·nz)
- for evaluation of a new δ in O(K·n)

Computation of the upper bound $\mathbf{g}^{+}_{t-\delta}$ based on two ideas:

1. Compute separate policies for accumulated reward and reward gained at the end
2. Assume that we can choose at every jump of the Poisson process a new policy

Compute for k = 0, 1,…, K steps in the interval



Formally

$$\mathbf{v}_+^{(k)} = \max_{\mathbf{d}\in\mathcal{D}} \left(\mathbf{P}^{\mathbf{d}}\mathbf{v}_+^{(k-1)}\right) \text{ with } \mathbf{v}^{(0)} = \mathbf{g}_t^+$$

and

$$\mathbf{w}_+^{(k)} = \max_{\mathbf{d}\in\mathcal{D}} \left(\mathbf{P}^{\mathbf{d}}\mathbf{w}_+^{(k-1)}\right) \text{ with } \mathbf{w}^{(0)} = \max_{\mathbf{d}\in\mathcal{D}} \left(\mathbf{w}^{\mathbf{d}}\right)$$ (identical for all intervals)

For $\mathbf{g}^+_t \geq \mathbf{g}^*_t$ we obtain

$$\mathbf{g}_{t-\delta}^+ = \sum_{k=0}^{K} \gamma(\alpha\delta, k)\left(\mathbf{v}_+^{(k)}\right) + \left(1 - \sum_{l=0}^{k}\gamma(\alpha\delta, l)\right)\mathbf{w}_+^{(k)} + \nu(\alpha\delta, \mathbf{w}^{\mathbf{d}}, \mathbf{g}_t^+) \geq \mathbf{g}_{t-\delta}^*$$

$\nu(\alpha\delta, \mathbf{w}^d, \mathbf{g}^+_t)$ bounds the truncated Poisson probabilities

Policy is better than any realizable policy!!

Effort

- for vector computation in $O(K \cdot n \cdot nz \cdot (\Sigma_{i=1..n} m_i))$
- for evaluation of a new δ in $O(K \cdot n)$
  can be applied in a line search to find an appropriate δ

Error proportional to the length of the subinterval

Choose δ such that $|| \mathbf{g}^{+}_{t-\delta} - \mathbf{g}^{-}_{t-\delta} || < \varepsilon (t - \delta) / T$;

If δ is known and ***s*** is independent of the decision vector, the upper bound can be improved
by computing a non-realizable bounding policy for accumulated and gained reward

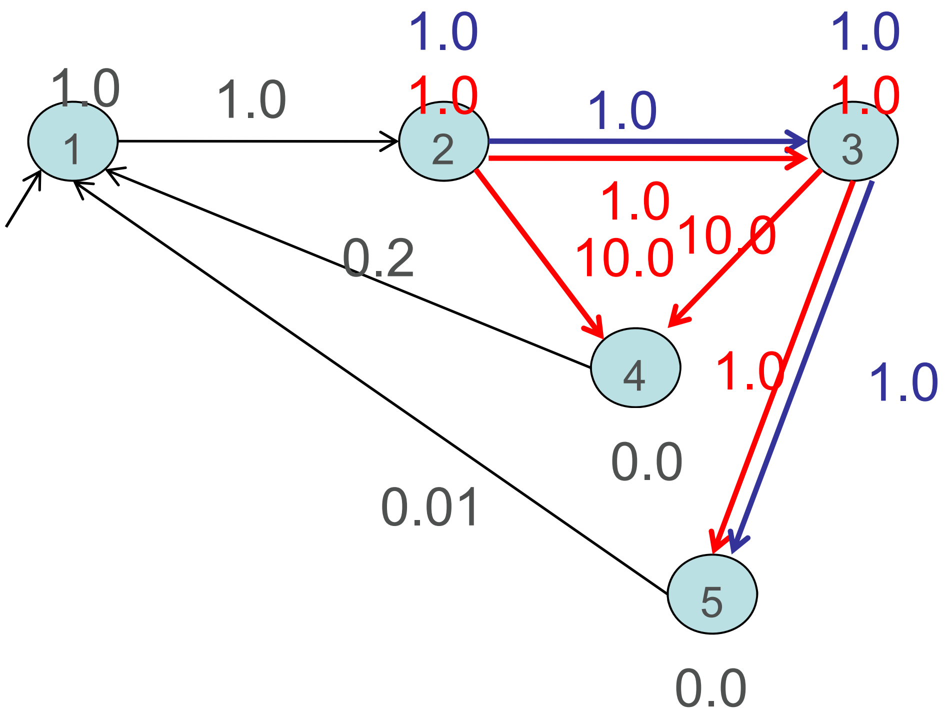
$$\mathbf{x}_+^{(k)} = \max_{\mathbf{d}_1^k,\ldots,\mathbf{d}_k^k \in \mathcal{D}} \left(\prod_{i=1}^{k} \mathbf{P}^{\mathbf{d}_i^k} \left(\eta_k \mathbf{g}_t^+ + \zeta_k \mathbf{w} \right) \right)$$

where $\eta_k = \gamma(\alpha\delta, k)$ and $\zeta_k = \alpha^{-1}\left(1 - \sum_{l=0}^{k} \gamma(\alpha\delta, l)\right)$

We have then $\sum_{k=0}^{\infty} \mathbf{x}_+^{(k)} \geq \mathbf{g}_{t-\delta}^*$

Effort $O(K^2 \cdot n \cdot nz \cdot (\Sigma_{i=1..n} m_i))$

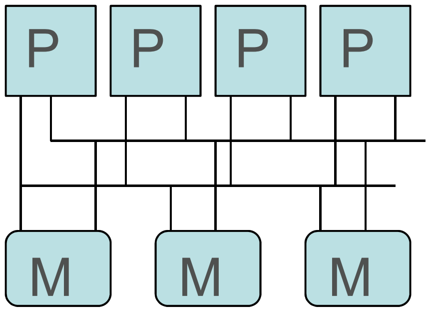
- Overall effort $O(K \cdot n \cdot nz \cdot (\Sigma_{i=1..n} m_i))$ if the optimal policy is selected from $F_i(\mathbf{g}^{-}_t)$ for some small i (usually the case)
- Local error $O(\delta^2) \Rightarrow$ Global error $O(\delta) \Rightarrow$ for any $\varepsilon>0$ (theoretically) the appropriate policy can be computed



For T ≥ 70.5058:

- b,b in [T - 4.11656, T]
- b,r in (T – 70.5058, T – 4.11656]
- r,r in [0, T - 70.50558)

g_0^T = (20.931, 20.095, 19.138, 20.107, 8.6077)
Bounds with ε = 1.0e-6

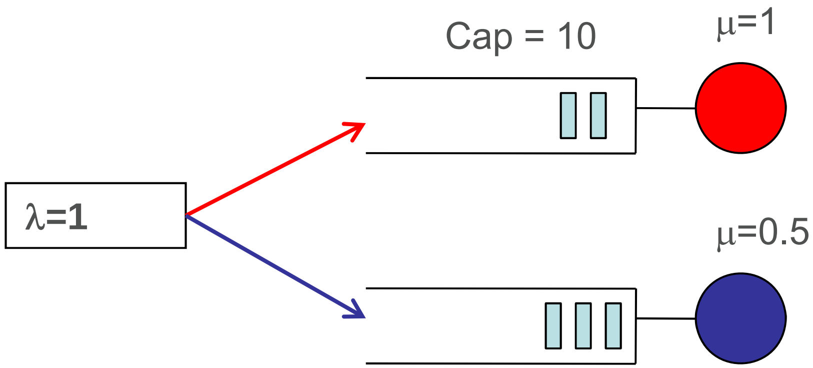


4 processors, 2 buses, 3 memories, 1 repair unit

Prioritized repair $\Rightarrow$ Computation of the availability

CTMDP with 60 states

	Availability in [0,100]			Availabilty at T=100		
ε	iter	Lower bound	Upper bound	iter	Lower Bound	Upper Bound
1.0e-2	1080	0.986336	0.995790	872	0.987109	0.995386
1.0e-4	2026	0.995633	0.995722	1419	0.995188	0.995282
1.0e-6	5896	0.995721	0.995721	21089	0.995279	0.995280
1.0e-8	34273	0.995721	0.995721	1513361	0.995280	0.995280



CTMDP with 121 states

Goal maximization of the throughput in [0,100]

Troughput in [0,100]				
ε	iter	Lower bound	Upper bound	Sw. Time
1.0e+0	6098	97.4599	98.4556	68.2561
1.0e-1	45866	97.4878	97.5875	68.3037
1.0e-2	334637	97.4879	97.4979	68.3099
1.0e-3	2981067	97.4881	97.4891	68.3102

Effort

- Linear in αt and in ε^{-1} (and in $n \cdot nz$)
- Influence of K depends on the model
  - K too small results in small time steps due to the truncated Poisson probabilities
  - K too large results in many unneccessarry computations that are truncated due to the difference in the policy bounds
  - Adaptive approach to choose K such that fraction of the error due to the truncation of the Poisson probabilities remains constant

Extensions

- Method can be applied to discounted rewards after introducing small modifications
- Method can be applied to CTMDPs with time dependent but piecewise constant rates and rewards
- Method can be extended to CTMDPs with time dependent rates and rewards
- Method can be extended to countable state spaces

Advanced Topics

- Model Checking CTMDPs
- Inifinite State Spaces
- CTMDPs with bounds on the transition rates
- Equivalence of CTMDPs
- Partially observable CTMDPs
- ……

CSL model checking of CTMDPs

joint work with Holger Hermanns, Ernst Moritz Hahn, Lijun Zhang

- Model checking of CTMCs is very popular
- Extension for CTMDPs define formulas that hold for a set of states and all schedulers / some scheduler
- Model checking means to compute for every state whether a formula holds/does not hold
- Validation of path formulas requires computation of minimal/maximal gain vectors for finite or infinite horizons

Syntax of CSL for CTMDPs:

$$\Phi := a \mid \neg\Phi \mid \Phi \wedge \Phi \mid \mathbb{P}_{\mathtt{J}}(\Phi\ \mathtt{U}^{\mathtt{I}}\ \Phi) \mid \mathbb{S}_{\mathtt{I}}(\Phi) \mid \mathbb{I}^{t}_{\mathtt{J}}(\Phi) \mid \mathbb{C}^{\mathtt{I}}_{\mathtt{J}}(\Phi)$$

where I und J are closed intervals and t is some time point

$a \mid \neg\Phi \mid \Phi \wedge \Phi$ with the usual interpretation

$s \models \mathbb{P}_{\mathtt{J}}(\Phi\ \mathtt{U}^{\mathtt{I}}\ \Psi)$ if the probability of all paths that start in state *s* observe Φ until Ψ in I falls into J for all policies

$s \models \mathbb{S}_{\mathtt{I}}(\Phi)$ if the process starts in state *s* and has a time averaged stationary reward over state observing Φ that lies in I for all policies

$s \models \mathbb{I}_{\mathtt{J}}^{t}(\Phi)$ if the process starts in state *s* and has a instantaneous reward at time t in states observing Φ that lies in J for all policies

$s \models \mathbb{C}_{\mathtt{J}}^{\mathtt{I}}(\Phi)$ if the process starts in state *s* and has an accumulated reward over states observing Φ in the interval I that lies in J for all policies

Validation of formulas, an example:

$$s \models \mathbb{C}_{[p_1,p_2]}^{[t_0,T]}(\Phi) \;\text{ iff } \mathbf{a}^-(s) \geq p_1 \wedge \mathbf{a}^+(s) \leq p_2$$

where $\mathbf{a}^- = \inf_{\pi \in \mathcal{M}} \left(\mathbf{V}^\pi_{0,t_0}\mathbf{g}^\pi_{t_0,T}|_\Phi\right)$ and $\mathbf{a}^+ = \sup_{\pi \in \mathcal{M}} \left(\mathbf{V}^\pi_{0,t_0}\mathbf{g}^\pi_{t_0,T}|_\Phi\right)$

Two step approach to compute $\mathbf{V}^\pi_{0,t_0}$ and $\mathbf{g}^\pi_{t_0,T}|_\Phi$

Computation of $\mathbf{V}^\pi_{0,t_0}$ with the standard approach, i.e.

$$\mathbf{V}^\pi_{t,t} = \mathbf{I} \text{ and } \frac{d}{du}\mathbf{V}^\pi_{t,u} = \mathbf{V}^\pi_{t,u}\mathbf{Q}^{\mathbf{d}_u}$$

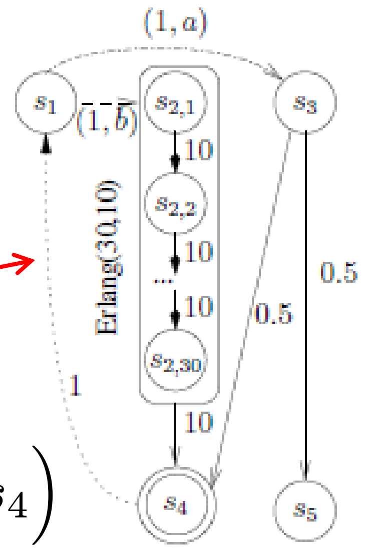
Computation of $\mathbf{g}^\pi_{t_0,T}|_\Phi$ using vectors $\mathbf{s}^\pi|_\Phi$ and $\mathbf{g}_T|_\Phi$ then

$$\mathbf{g}^\pi_{t,T}|_\Phi = \mathbf{V}^\pi_{t,T}\mathbf{g}^\pi_T|_\Phi + \int_t^T \mathbf{V}^\pi_{u,T}\mathbf{w}^\pi|_\Phi du$$

Separate error bounds for both quantities

where $\mathbf{h}|_\Phi(s) = \mathbf{h}(s)$ if $s \models \Phi$ and 0 else

A small example



s_4 is made absorbing to compute the probability of reaching s_4 in the interval [3,7]

$$\sup_{\pi \in \mathcal{M}} \left(P^{\pi}_{s_1} \left(true\ U^{[3,7]} s_4 \right) \right) = P^{\max}_{s_1} \left(\Diamond^{[3,7]} s_4 \right)$$

	$\epsilon = 1.0e-3$				$\epsilon = 6.0e-4$			
ϵ_1	time bounded prob.		$iter_1$	$iter_2$	time bounded prob.		$iter_1$	$iter_2$
$9.0e-4$	0.97170	0.97186	207	90	–	–	–	–
$5.0e-4$	0.97172	0.97186	270	89	0.97176	0.97185	270	93
$1.0e-4$	0.97175	0.97185	774	88	0.97178	0.97185	774	91
$1.0e-5$	0.97175	0.97185	5038	88	0.97179	0.97185	5038	91

Table 1: Bounds for reaching s_4 in $[3, 7]$, i.e., $P^{max}_{s_1}(\Diamond^{[3,7]} s_4)$.

Infinite (countable) state spaces

- Infinite horizon
  - Discounted reward
  - Average reward
- Finite horizon

We assume that the control space per state remains finite, all rewards and transitions rates are bounded

Infinite horizon discounted case

- We assume that the optimality equations (slide 28) for the original model have a solution
- We compute results on $\hat{S}$ = {1,…,n} ⊂ S
- Let $\hat{\mathbf{Q}}^{\mathbf{d}}$ be the *n x n* submatrix of $\mathbf{Q}^{\mathbf{d}}$ and $\hat{\mathbf{w}}^{\mathbf{d}}$ the *n* dimensional subvector of $\mathbf{w}^{\mathbf{d}}$ restricted to states from $\hat{S}$
- Define $\hat{\mathbf{P}}^{\mathbf{d}} = \hat{\mathbf{Q}}^{\mathbf{d}}/\alpha + \mathbf{I}$ and $\hat{\mathbf{w}}'^{\mathbf{d}} = \mathbf{w}^{\mathbf{d}}/(\alpha + \beta)$ (see slide 27)

This defines an DTMDP with n states

Value iteration:

- Modified DTMDP can be analyzed with value iteration (see slide 33)
- If $\lim_{k\to\infty} \|\hat{\mathbf{P}}^{\mathbf{d}_k|_{\hat{S}}} \mathbf{g}^{(k-1)}|_{\hat{S}} - \left(\mathbf{P}^{\mathbf{d}_k}\mathbf{g}^{(k-1)}\right)\Big|_{\hat{S}}\|_\infty < \epsilon$ then
  $\|\hat{\mathbf{g}}^* - \mathbf{g}^*|_{\hat{S}}\|_\infty < \frac{\epsilon}{1-\beta'}$ where
  - $\hat{\mathbf{g}}^*, \mathbf{g}^*$ are the vectors to which value iteration applied to the reduced and original MDP converges
  - $\mathbf{d}_k(i)$ equals the decision in state *i* during the *k*-th iteration of value iteration applied to the finite system if $i \in \hat{S}$ and is arbitrary else
  - $\mathbf{g}^{(k)}(i)$ equals the value in in the *k*-th iteration of value iteration applied to the finite system if $i \in \hat{S}$ and is an upper bound for the value function otherwise.

Policy iteration:

- If for some policy π , vector $\tilde{\mathbf{g}}^{(k)}|_{\hat{S}}$ is the gain vector restricted to states from $\hat{S}$ and $\hat{\mathbf{g}}^{(k)}$ is the approximate solution computed for the finite system, such that $\|\tilde{\mathbf{g}}^{(k)}|_{\hat{S}} - \hat{\mathbf{g}}^{(k)}\|_\infty < \delta$ and
- $\|\hat{\mathbf{P}}^{\mathbf{d}_{k+1}|_{\hat{S}}}\hat{\mathbf{g}}^{(k)}|_{\hat{S}} - \left(\mathbf{P}^{\mathbf{d}_{k+1}}\overline{\mathbf{g}}^{(k)}\right)\Big|_{\hat{S}}\|_\infty < \epsilon$ where $\mathbf{d}_{k+1}$ results from $\hat{\mathbf{g}}_k$ and $\overline{\mathbf{g}}^{(k)}$ is an arbitrary extension of $\hat{\mathbf{g}}^{(k)}$ to *S*,

  then $$\lim_{k\to\infty}\left(\|\hat{\mathbf{g}}^{(k)} - \mathbf{g}^*|_{\hat{S}}\|_\infty\right) < \frac{\epsilon + 2\beta'\delta}{(1-\beta')^2}$$

Infinite horizon average case

- Additional conditions are required to assure existence of an optimal policy (many different conditions exist in the literature)
- Let for some policy π be:

  C_π the expected gain of a cycle that starts in state *1* and ends when entering state *1* again

  N_π the expected number of visited states between two visits of state 1

  if for all policies C_π is finite and the N_π are uniformly bounded, then the Bellman equations have an optimal solution

Solution often via simulation

Finite horizon but infinite state spaces

(not much known from an algorithmic perspective!)

Assumptions:

- $S_0 = \{i \mid p_0(i) > 0\}$, let $|S_0| < \infty$
- Transition rates are bounded by $\alpha < \infty$

Define

- $S_k = \{ j \mid (\Sigma_{\mathbf{d} \in D} \mathbf{P^d})^k(i,j) > 0$ for some $i \in S_0\}$
- $S_T(\varepsilon) = \cup_{k=0,\ldots,K_\varepsilon} S_k$ where $K_\varepsilon = \min_K \Sigma_{k=1\ldots K} \gamma(T,k) \geq 1-\varepsilon$

Algorithm for approximating/bounding the accumulated reward in [0,T]

1. Define some finite subset $\hat{S}$ of the countable state space S (e.g. using $S_T(\varepsilon)$ for some appropriate ε)

2. Define a new CTMDP with state space $\hat{S} \cup \{0\}$

   Matrices
$$\hat{\mathbf{Q}}^{\mathbf{d}} = \begin{cases} \mathbf{Q}^{\mathbf{d}}(i,j) & \text{if } i,j \neq 0 \\ 0 & \text{if } i = 0 \\ \sum_{h \notin \hat{S}} \mathbf{Q}^{\mathbf{d}}(i,h) & \text{if } i \neq 0, j = 0 \end{cases}$$

   Vectors
$$\hat{\mathbf{w}}^{\mathbf{d}\pm} = \begin{cases} \mathbf{w}^{\mathbf{d}}(i) & \text{if } i \neq 0 \\ \max/\min_{j \in S, u \in D_j} \left(\mathbf{w}^{u}(j)\right) & \text{if } i = 0 \end{cases}$$

$$\hat{\mathbf{g}}_T^{\pm} = \begin{cases} \mathbf{g}_T^{\pm}(i) & \text{if } i \neq 0 \\ \max/\min_{j \in S,} \left(\mathbf{g}_T^{\pm}(j)\right) & \text{if } i = 0 \end{cases}$$

3. Solve the resulting CTMDP to obtain bounds for the original one

Bounds on the transition rates

- Transition rates $\mathbf{Q}^{\mathbf{d}}(i,j)$ and reward vectors $\mathbf{s}^{\mathbf{d}}$ are not exactly known but we know $\mathbf{L}^{\mathbf{d}}(i,j) \leq \mathbf{Q}^{d}(i,j) \leq \mathbf{U}^{d}(i,j)$ and $\mathbf{l}^{\mathbf{d}} \leq \mathbf{w}^{\mathbf{d}} \leq \mathbf{u}^{\mathbf{d}}$ (sometimes known as Bounded Parameter MDPs see e.g. Givan et al 2000)

  realistic model if parameters result from measurements
- Goal: Find a policy that maximizes the minimal/maximal gain over an infinite or finite interval

Infinite horizon case:

We assume that the CTMDP is unichain for all $\mathbf{L}^d$

- Uniformization can be used to transform the CTMDP in an equivalent DTMDP (as we did before)
- For a fixed decision vector **d** the minimal/maximal average reward is obtained by a matrix $\mathbf{P} \in [\mathbf{L},\mathbf{U}]$ where in every row all except one elements are equal to the corresponding element in matrix **U** or **L**
  $\Rightarrow$ only finitely many possibilities exist
  $\Rightarrow$ determination of the bounds is again an MDP problem

Infinite horizon case (continued):

- Overall solution is the combination of two nested MDP problems (Markov two person game)
- Some solution algorithms exist (stochastic min/max control) but advanced numerical techniques are rarely used (room for improvements remains!)

Finite horizon case

- Inherently complex to the best of my knowledge almost no results (even for the simpler DTMDP case!)

Thank you!

Bibliography

(incomplete and biased selection but there is too much to be exhaustive!)

Textbooks:

- D. P. Bertsekas. Dynamic Programming and Optimal Control vol I & II, 2nd ed. Athena Scientific 2005, 2007.
- Q. Hu, W. Yue. Markov Decision Processes with their Applications. Springer 2008.
- X. Gua, O. Hernandez-Lerma. Continuous-Time Markov Decision Processes – Theory and Applications. Springer 2009.
- M. L. Puterman. Markov Decision Processes: Discrete Stochastic Dynamic Programming. Wiley 1994.

Papers:

- R. F. Serfozo. An Equivalence Between Continuous and Discrete Markov Decision Processes. Operations Research 37 (3), 1979, 616-620.
- B. L. Miller. Finite State Continuous Time Markov Decision Processes with a Finite Planing Horizon. Siam Journal on Control 6 (2), 1968, 266-280.
- M. R. Lembersky. On Maximal Rewards and ε-Optimal Policies in Continuous Time Markov Decision Processes. The Annals of Statistics 2 (1), 1974, 159-169.
- S. A. Lippman. Countable-State, Continuous-Time Dynamic Programming with Structure. Operations Research 24 (3), 1976, 477-490.
- R. Givan, S. Leach, T. Dean. Bounded-parameter Markov decision processes. Artificial Intelligence 122 (1/2), 2000, 71-109.

Own papers:

- P. Buchholz, I. Schulz. Numerical Analysis of Continuous Time Markov Decision Processes over Finite Horizons. Computers & OR 38, 2011, 651-659.
- P. Buchholz, E. M. Hahn, H. Hermanns, L. Zhang. Model Checking of CTMDPs. Proc. CAV 2011.
- P. Buchholz. Bounding Reward Measures of Markov Models using Markov Decision Processes. To appear in Numerical Linear Algebra and Applications