Coping with the Blooming Buzzing World that Surrounds Us

Thomas Dean

Brown University

In collaboration with:

Craig Boutilier
Robert Givan
Milos Hauskrecht
Leslie Kaelbling
Kee-Eung Kim
Nicolas Meuleau

Quantifying the Blooming and Buzzing

"Let me count the ways."

- The propositions used to distinguish states
  - $\Rightarrow$ large number of spatial dimensions
- The units used to describe the granularity of time
  - $\Rightarrow$ seemingly infinite temporal resolution
- The characteristics used to describe our options
  - $\Rightarrow$ daunting number of control parameters

States are many, time is dense, and our options are multitude.

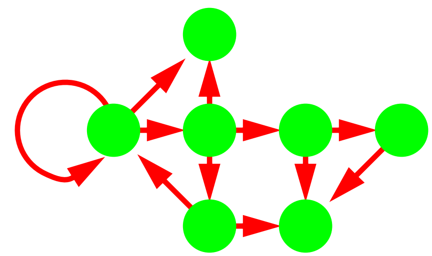
Overview

1. Semantics of time, space, and activity
   - attempt at representational agnosticism
2. Characterizing structure in the world
   - characteristics that *may* provide leverage
3. Classic problems with a combinatorial twist
   - optimal control in high-dimensional spaces
4. Reaching for the whole enchilada
   - reformulating the planning problem

Markov Decision Processes

Let $M = (Q, A, T, R)$ where

- Q is a finite set of *states*
- A is a finite set of *actions*
- T is a *state-transition distribution*
- R is a *reward function*



state-transition diagram

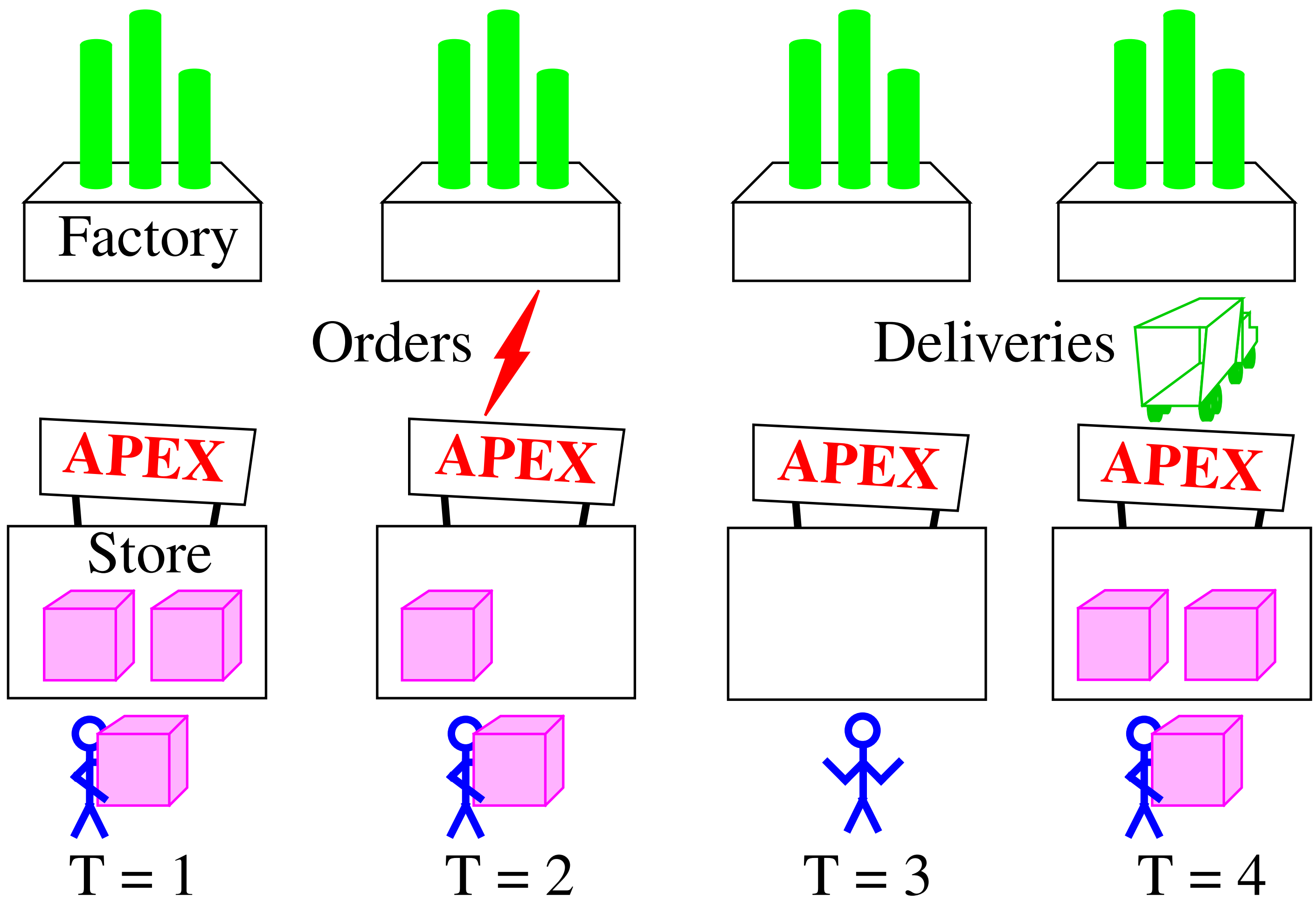
where for $\forall i, j \in Q, \alpha \in A$,

$$T(i, \alpha, j) = \Pr(X_{t+1} = j \mid X_t = i, U_t = \alpha)$$

$$R(i, \alpha) = \text{Reward}(X_t = i, U_t = \alpha) = \boxed{R(i)}$$

simplification

Inventory Example



Markov Decision *Problems*[1]

An MDP together with an *objective function*, *e.g.*,

- *Expected cumulative discounted reward with rate* γ
- *Policies* $\pi \in \Pi$ where $\pi : Q \to A$
- *Value functions* $v \in V$ where $v : Q \to \Re$, in particular

$$v_\pi(i) = R(i) + \gamma \sum T(i, \pi(i), j) v_\pi(j)$$ *value function for* π

$$v^*(i) = \max_\pi v_\pi(i)$$ *optimal value function*

- *Vector notation* $v(i)$ denotes the ith component of v

1. This presentation adopts the terminolgy and notation used in “Markov Decision Processes” [Puterman, 1994].

Computing Optimal Solutions

System of (*Bellman*) equations has a unique solution v^*

$$v(i) = R(i) + \max_{\pi}\gamma\sum T(i, \pi(i), j)v(j)$$

The *nonlinear operator* L defined by

$$Lv(i) = R(i) + \max_{\pi}\gamma\sum T(i, \pi(i), j)v(j)$$

is a *contraction mapping*, *i.e.*, $\exists(0 \le \lambda < 1)$ *s.t.* $\forall(u, v \in V)$

$$\|Lu - Lv\| \le \lambda\|u - v\| \quad \textit{where} \quad \|v\| = \max_i |v(i)|$$

with *fixed point* v^* and the basis for *value iteration*.

Modeling Dynamical Systems I

Explicit models for planning and control

complex environments

requires a clever domain expert

compact, explicit models

equivalent, reduced models

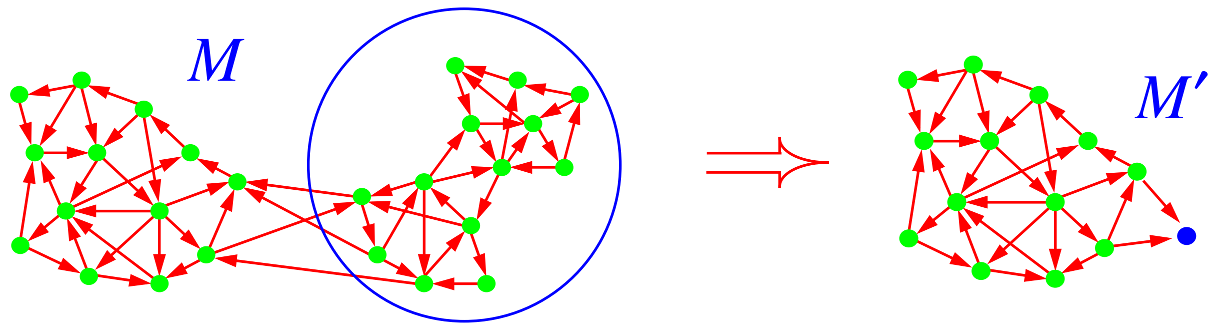
[Hartmanis & Stearns, 1966] [Paz, 1971]

approximations based on explicit reduction

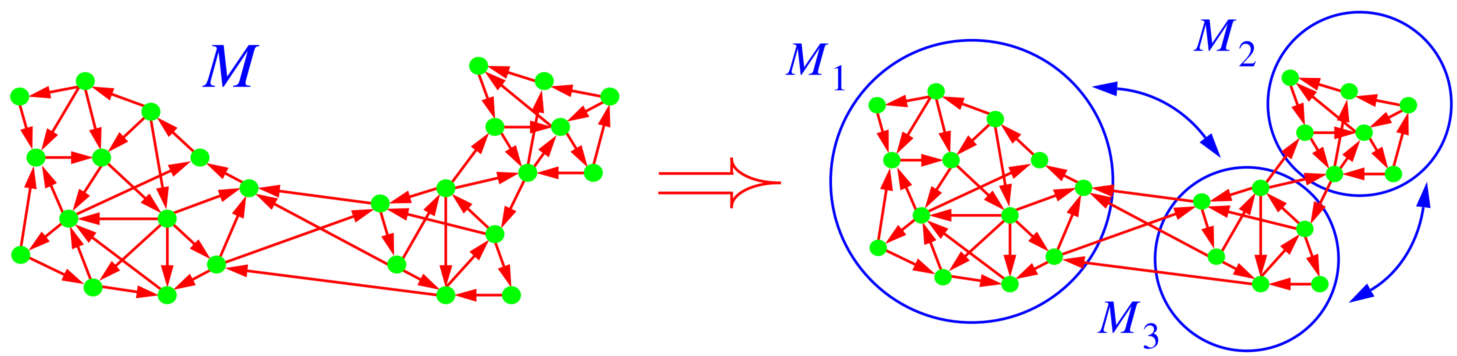
[Schweitzer *et al.*, 1985] [Bertsekas & Castanon, 1989] [Dean *et al.*, 1993]
[Dantzig & Wolfe, 1960] [Kushner & Chen, 1974] [Dean & Lin, 1995]

Aggregation and Decomposition

Aggregating states that behave *or are treated* the same



Decomposing large problems into subproblems



Modeling Dynamical Systems II

Implicit models for planning and control

complex environments

requires a clever domain expert

compact, implicit models

[McCarthy & Hayes, 1969] [Dean & Kanazawa, 1989] [Boutilier *et al.*, 1995]

implicit reduction

[Lee & Yannakakis, 1992] [Bouajjani *et al.*, 1992] [Dean & Givan, 1997]

approximations based on implicit reduction

[Boutilier & Dearden, 1996] [Dean *et al.*, 1997]

minimax policies on sets of models

[Shapley, 1953] [Satia & Lave, 1973] [Givan *et al.*, 1997]

Factorial Models

Parameterized state and action spaces

- M state variables or *fluents*: $\{X_i\}$

$$Q = \prod_{i=1}^{M} \Omega_{X_i} \qquad \vec{X} = X_1, \ldots, X_M$$

- K control parameters or *action fluents*: $\{U_i\}$

$$A = \prod_{i=1}^{K} \Omega_{U_i} \qquad \vec{U} = (U_1, \ldots, U_K)$$

- Factored function representations:

$$R(i, \alpha) = R'(X_1, \ldots, X_M, U_1, \ldots, U_K)$$

$$T(i, \alpha, j) = T'(X_1, \ldots, X_M, U_1, \ldots, U_K, X_1, \ldots, X_M)$$

Example Factored Representation[1]

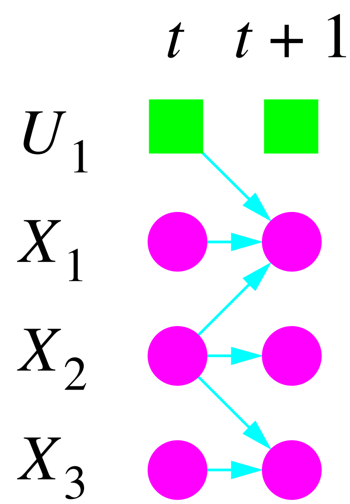
Factored joint distribution for transition probabilities

$$T(i, \alpha, j) = \Pr(\overrightarrow{X_{t+1}} = j \mid \overrightarrow{X_t} = i, \overrightarrow{U_t} = \alpha)$$

$$= \Pr(X_{1,t+1}, \ldots, X_{M,t+1} \mid X_{1,t}, \ldots, X_{M,t}, U_{1,t}, \ldots, U_{K,t})$$

$$= \prod_{i=1}^{M} \Pr(X_{i,t+1} \mid \text{Parents}(X_{i,t+1}))$$

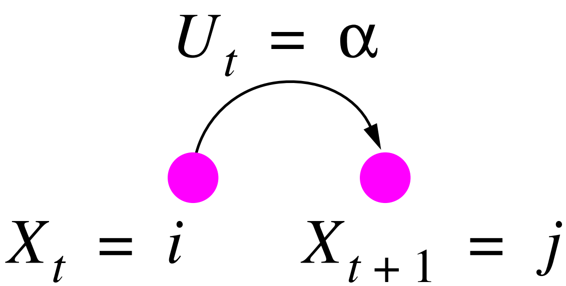
Temporal Bayesian networks

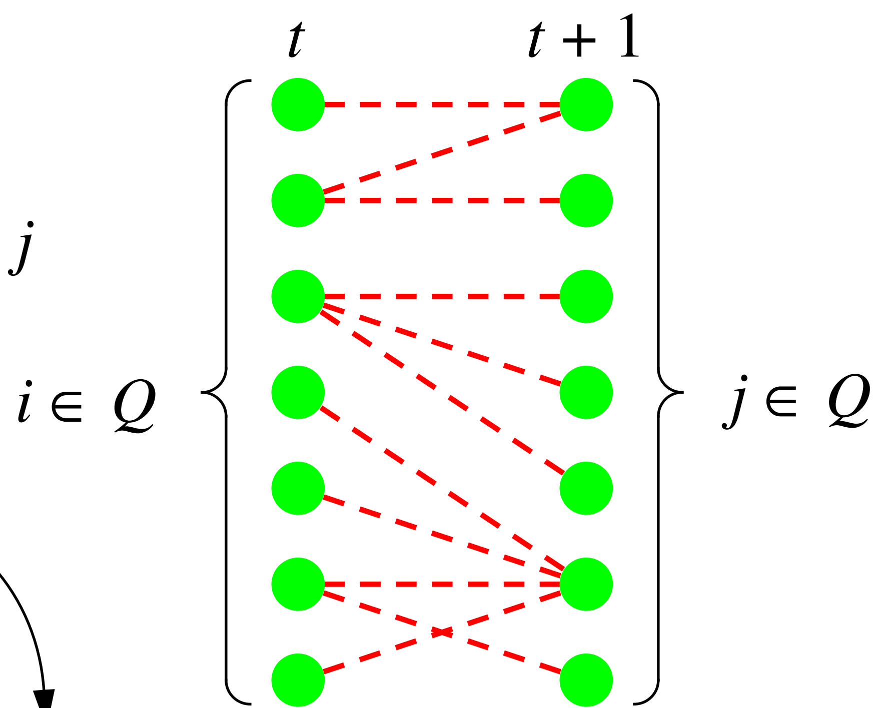


1. Details in [Kanazawa & Dean, 1987], [Kanazawa, 1991] and [Dean & Wellman, 1991].

Curse of Dimensionality

Standard value iteration





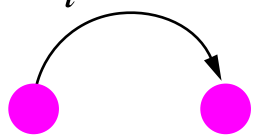
expectation

maximization

$$v(i) = R(i) + \max_{\alpha} \gamma \sum T(i, \alpha, j) v(j)$$

Structure in Factorial Models[1]

$U_t = \alpha$

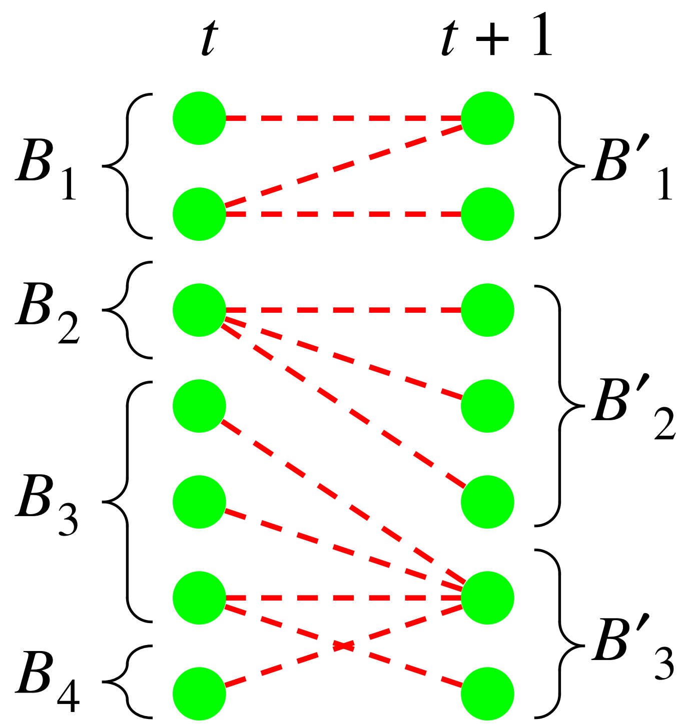


$X_t \in B_i \qquad X_{t+1} \in B'_j$

$P = \{B_1, \ldots, B_4\}$

$P' = \{B'_1, \ldots, B'_3\}$

For which partitions does the following work?



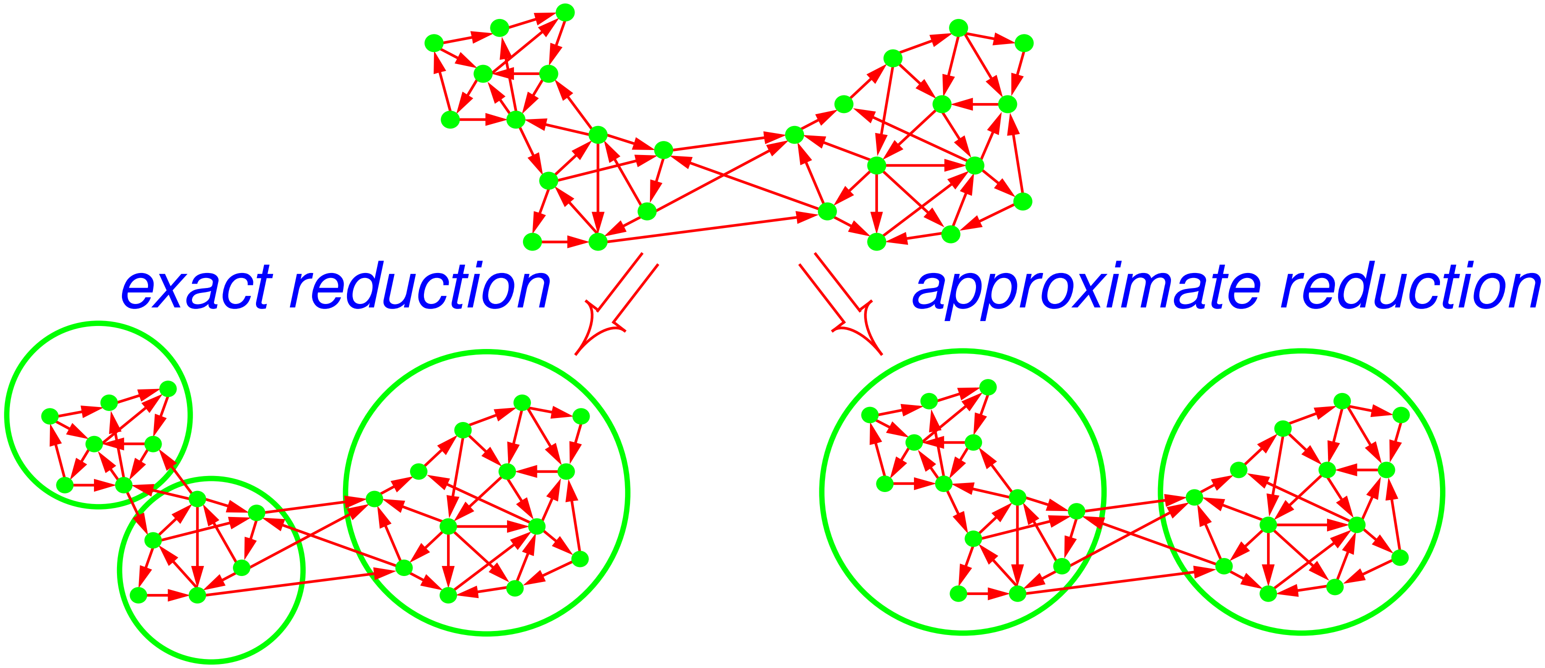
$$v(B_i) = R(B_i) + \max_\alpha \gamma \sum T'(B_i, a, B'_j) v(B'_j)$$

1. Structured value iteration is described in "Exploiting Structure in Policy Construction" [Boutilier *et al.*, 1995].

Partitions Induced by Model Reduction

Replace one MDP by another, smaller MDP

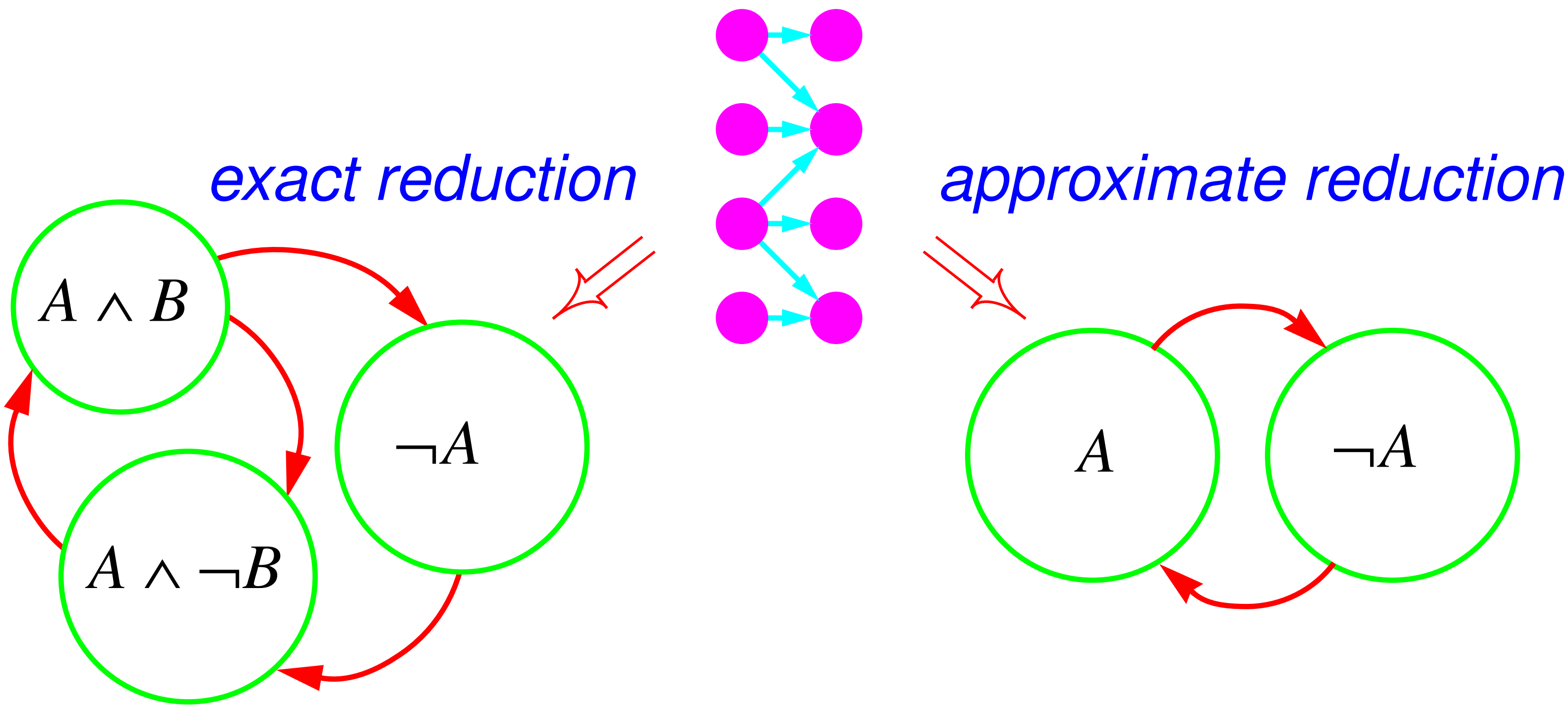
- Ignore distinctions that don't affect performance
- Aggregate states with the same transition behavior



Implicit Model Reduction

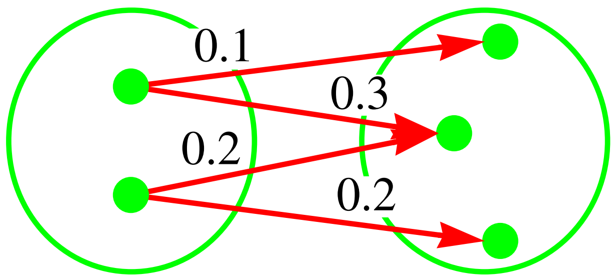
Groups of states are represented by formulas

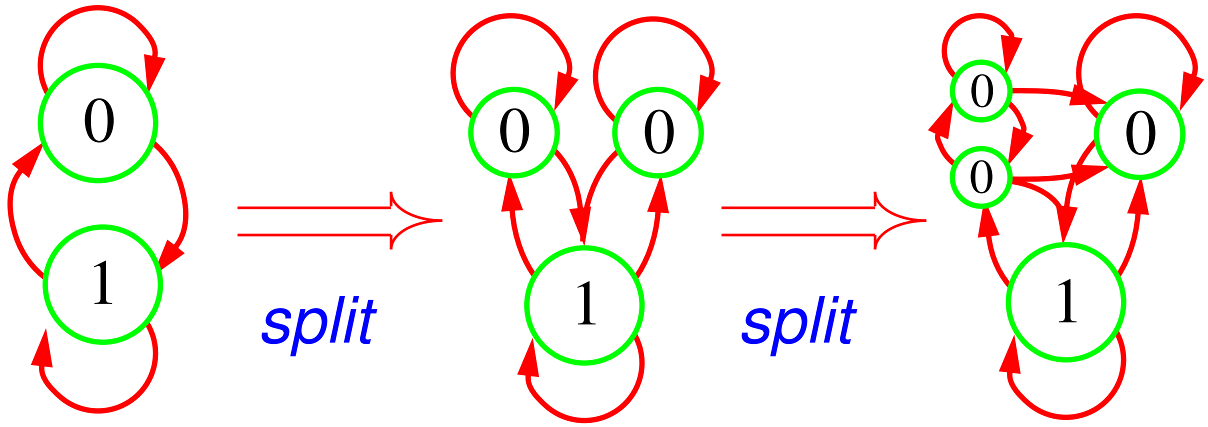
- Reduction proceeds by manipulating formulas



Online Model Reduction[1]

- Partition according to reward
- Split blocks that aren't *stable*
- Repeat until all blocks stable



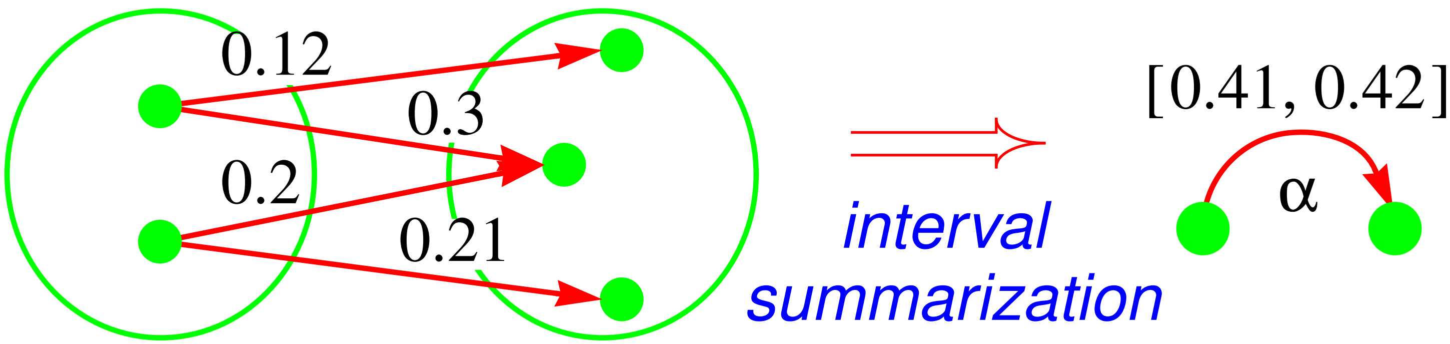


1. Details can be found in "Model Minimization in Markov Decision Processes" [Dean & Givan, 1997] which describes a variant of an algorithm described in "Online Minimization of Transition Systems" [Lee & Yannakakis, 1992].

Approximate Reductions[1]

Markov decision processes with imprecise probabilities

- Capture variation in state-transition probabilities
- Encode *families* of Markov decision processes
- Reduction with ε*-stability* as a stopping criterion



1. Details can be found in "Model Reduction Techniques for Computing Approximately Optimal Solutions for Markov Decision Processes" [Dean, Givan, & Leach, 1997] and "Bounded Parameter Markov Decision Processes" [Givan, Leach, & Dean, 1997].

Modeling Dynamical Systems III

Learning models for planning and control

complex environments

avoids building a model

system identification

[Angluin, 1978][1] [Rivest & Schapire, 1987][1]

[Basye *et al.*, 1995][1] [Dean *et al.*, 1997][1]

adaptive control and reinforcement learning

[Barto *et al.*, 1983][2] [Sutton, 1987][2] [Watkins, 1989][2] [Sutton, 1990]

[Chapman & Kaelbling, 1991][2,3] [Kearns & Singh, 1998]

---

1. System identification is simple if the state and action spaces are small and the system is fully observable.
2. Many reinforcement learning algorithms never even bother to build a model of the environment.
3. Of the above, only [Chapman & Kaelbling, 1991] assumes a factored representation of the state space.

Designing Artifacts

Consider the task of an automotive engineer

- Engineered artifact — midpriced family car
- Environment — determined by the consumer
- Expectations — know your target consumer
  - variations in driving habits but few race drivers
  - most have their vehicles serviced regularly
  - few keep their vehicles more than 200,000 miles
  - watermelon-sized hail is rare, road salt is not
  - an oil leak after ten years won’t nix the next sale

Designing Intelligent Artifacts

Designing janitor robots for office buildings

- Accounting for variations in office layouts
  - grid of rooms linked by axis-aligned corridors
  - five to fifty rooms with central service core
- Fixing reasonable expectations about change
  - office layout may change, but not often
  - position of furniture will change often
- Establishing performance expectations
  - three clean offices is a good night’s work
  - product life over five years is exceptional

Problem Reformulation

Space of environments — $\{M\}$

Space of designed artifacts — $\{\eta\}$

Distribution over environments — $\Pr(\{M\})$

Designer's knowledge of $\Pr(\{M\})$ — $\hat{\Pr}(\{M\})$[1]

We evaluate a particular artifact η according to

$$U_\eta = \sum \Pr(M) \times \mathrm{EV}_M^\eta$$

where EV_M^η is the expected value of η in environment M.

1. This formulation is a special case of the Bayesian formulations for learning such as those suggested by [Buntine, 1991], [Wolpert, 1995], and [Haussler *et al.*, 1991]. Here we focus on the special case in which $\Pr(\{M\}) = \hat{\Pr}(\{M\})$.

Unreasonable Expectations

Omniscient standard— *impossible*

$$G^{**} = \sum \Pr(M) \times \mathrm{EV}_M^{\pi^*}$$

where π^* is the optimal policy for the environment M.

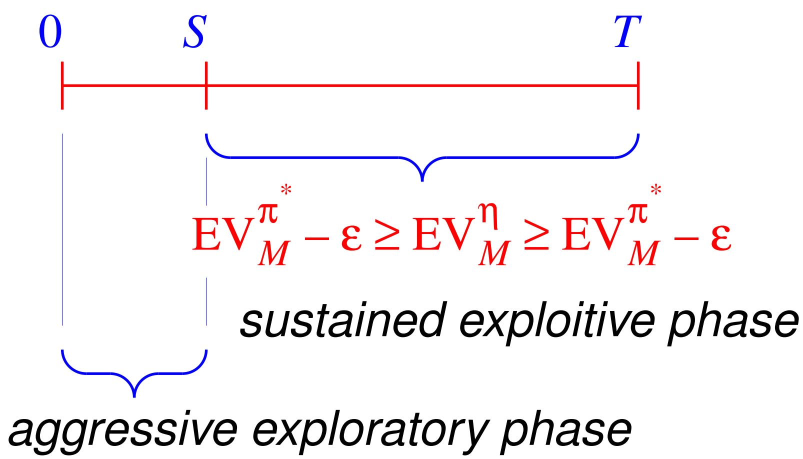
Bayes optimal standard — *impractical*

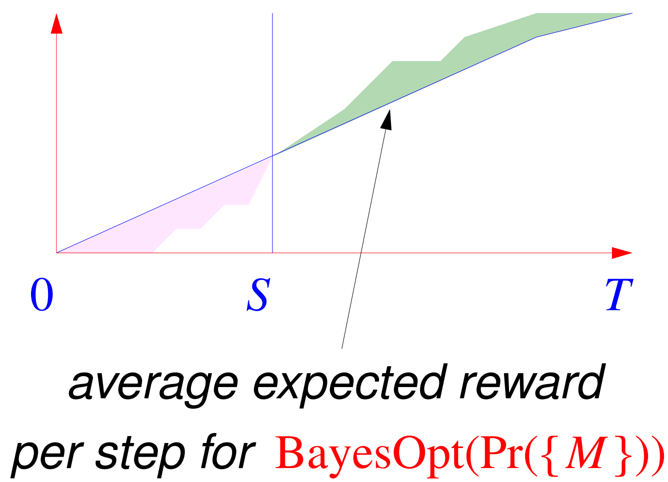
$$G^{*} = \sum \Pr(M) \times \mathrm{EV}_M^{\mathrm{BayesOpt}(\Pr(\{M\}))}$$

where $\mathrm{BayesOpt}(\Pr(\{M\}))$ is the optimal policy for the MDP defined in terms of belief states corresponding to distributions over $\{M\}$ and the initial state $\Pr(\{M\})$.

Reasonable Expectations

Given $G^{**} \geq G^{*} \geq U_{\eta}$ how close to G^{*} might we expect?





Can you make up for the sins of your youth?[1]

---

1. Kearns and Singh [1998] show how to bound the length of the initial exploratory phase as a polynomial function of the mixing time of a policy associated with optimal value function. One would expect that even the Bayes optimal solution will be subject to the constraints imposed by the mixing rates associated with policy-induced Markov chains.

Conclusions

Advantages of factored models

- Compactly represent complex dynamical systems
- Accord well with how we understand such systems
- Encode structure that can assist in decision making

Real interest is in designing intelligent control systems

- Know something about the target environments
- Lends itself nicely to a Bayesian formulation

The objective is to act so as to maximize expected value, but as is so often the case where uncertainty plays a role, planning cannot be separated from learning and information gathering.