Symbolic Perseus: a Generic POMDP Algorithm with Application to Dynamic Pricing with Demand Learning

Pascal Poupart (University of Waterloo)

INFORMS 2009

Outline

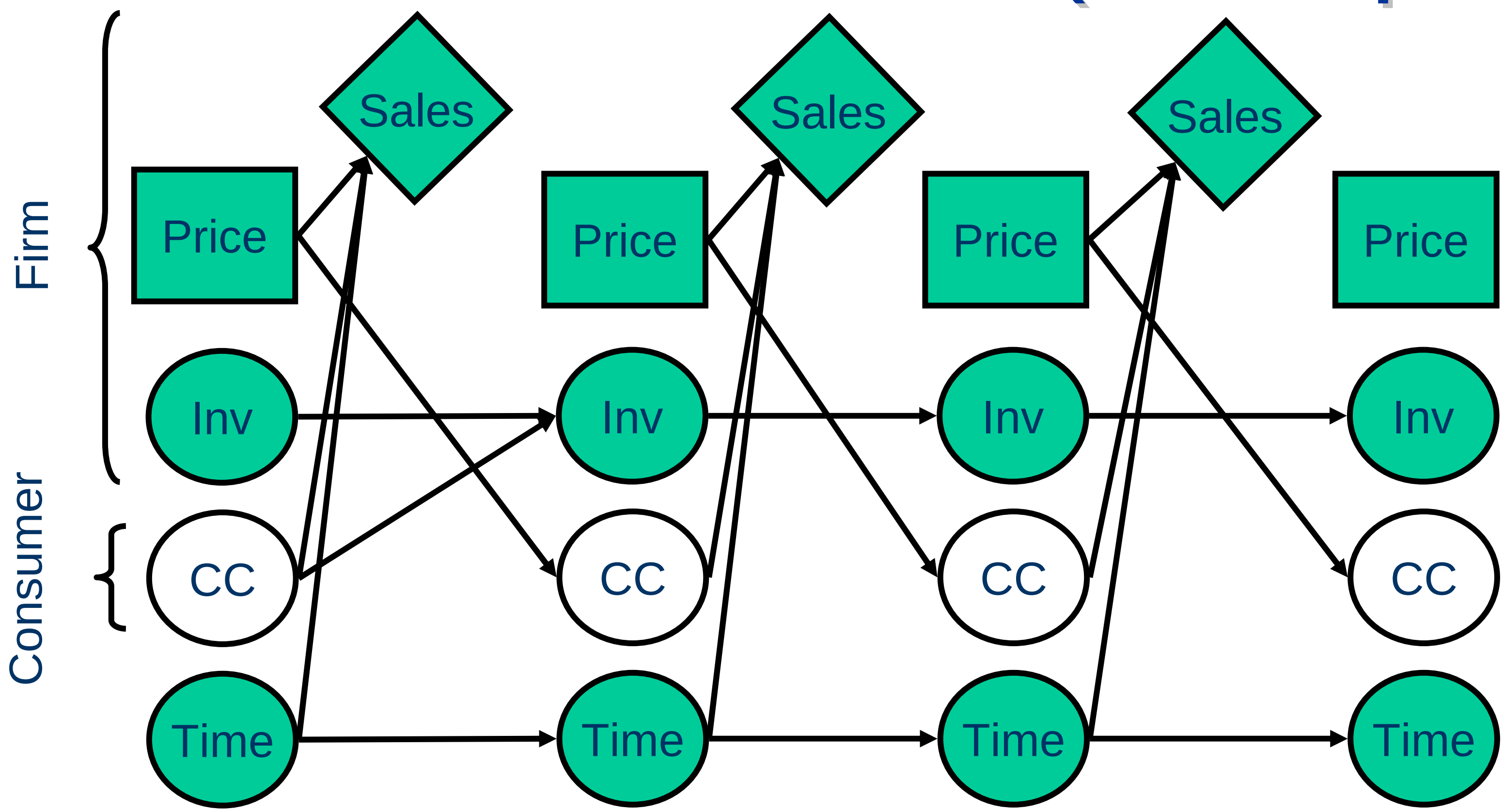
- Dynamic Pricing as a POMDP
- Symbolic Perseus
  - Generic POMDP solver
  - Point-based value iteration
  - Algebraic decision diagrams
- Experimental evaluation
- Conclusion

Setting

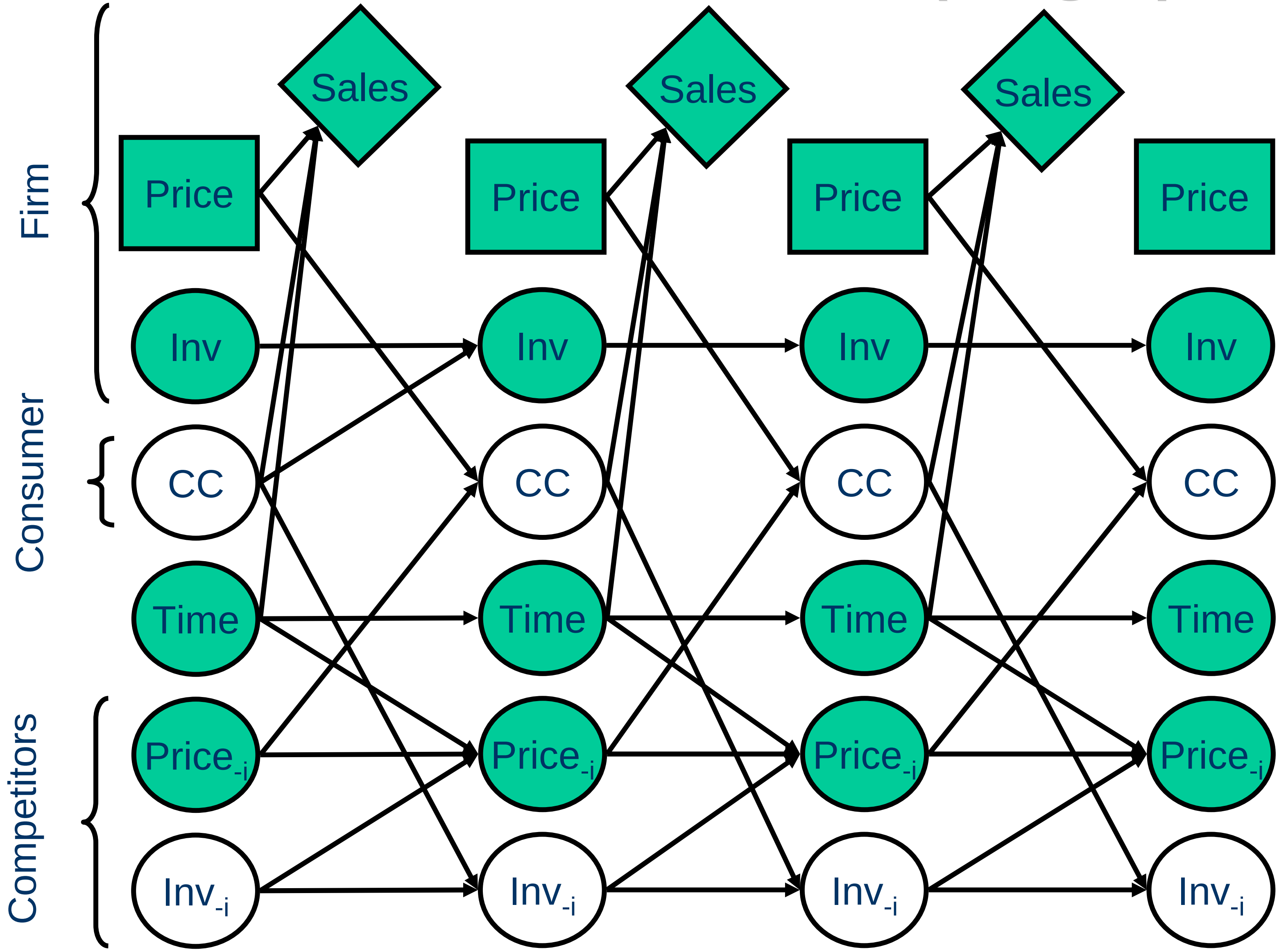
- One or several firms (monopoly or oligopoly)
- Fixed capacity and fixed number of selling rounds (i.e., sale of seasonal items)
- Finite range of prices
- Unknown and varying demand

- Question: how to dynamically adjust prices to maximize sales?

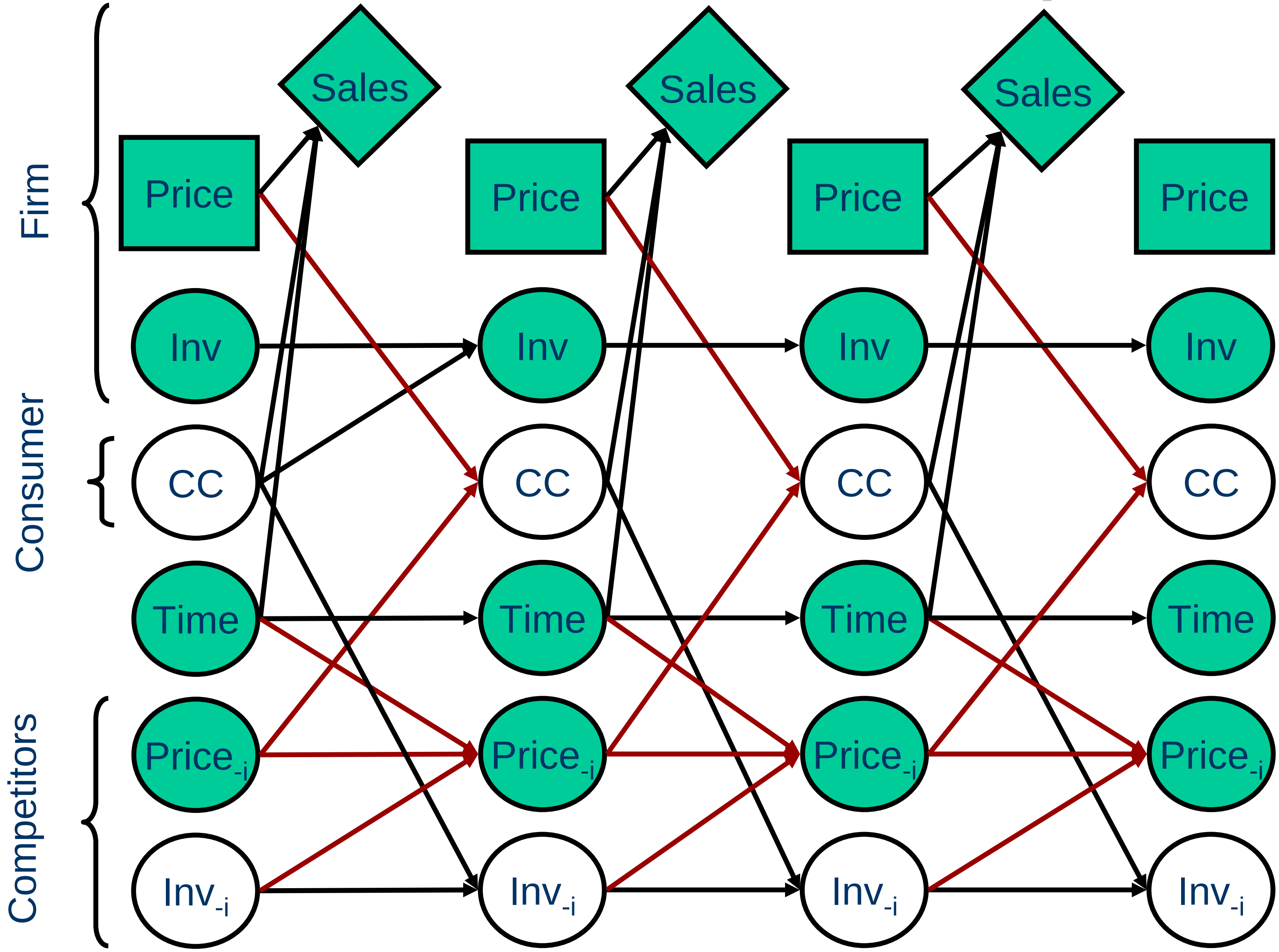
POMDPs Formulation (monopoly)



POMDPs Formulation (oligopoly)



Unknown demand & competitors



Demand Model

- Probability that consumer chooses firm i:

$$\Pr(CC=i) = \frac{e^{a_i+b_ip_i}}{\Sigma_i e^{a_i+b_ip_i} + 1}$$

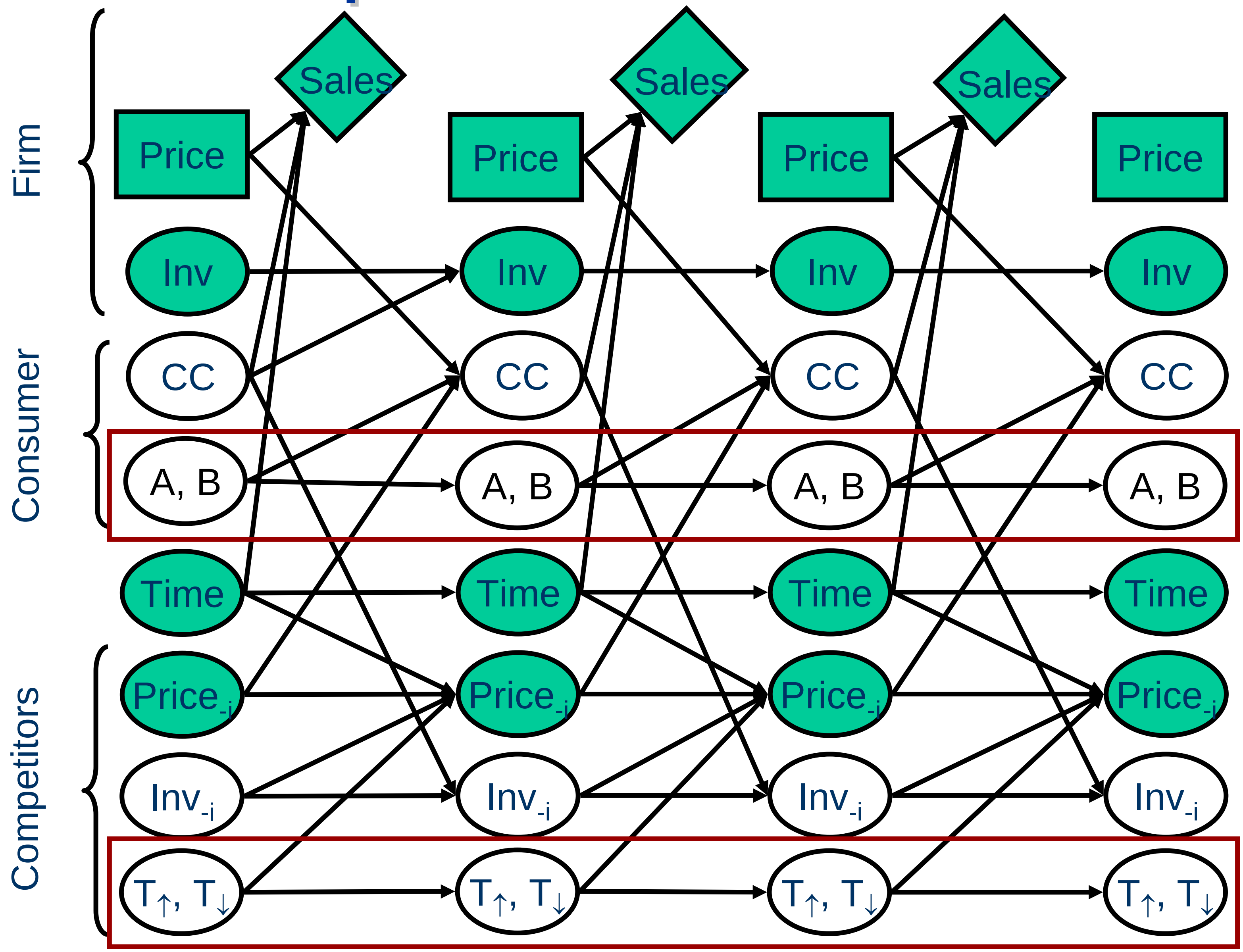
- Parameters a_i and b_i are unknown
- Learn them
  - From historical data
  - As process evolves

Competitors

- Model each competitor:
  - Pricing strategy: inv/time → price
  - Two thresholds: t_{up} and t_{down}
    - If inv/time < t_{up} ➔ price↑
    - If inv/time > t_{down} ➔ price↓

- Learn thresholds
  - From historical data
  - As process evolves

Expanded POMDP



POMDPs

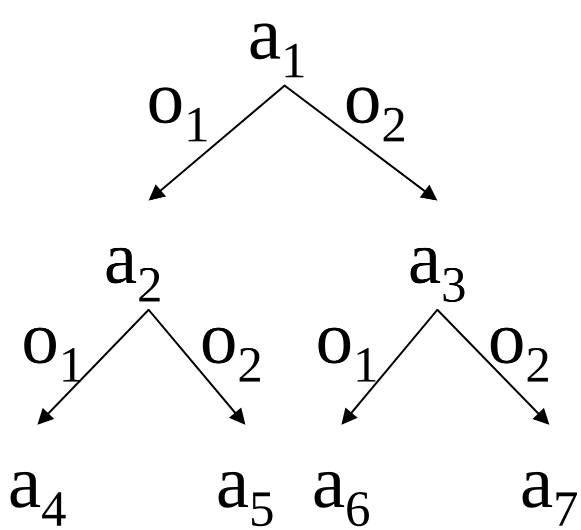
- Partially Observable Markov Decision Processes
  - **S:** set of states
    - Cross product of domain of all variables
    - |**S**| = $\prod_i$ |dom(V_i)| (exponentially large!)
  - **A:** set of actions
    - {price$\uparrow$, price$\downarrow$, price unchanged}
  - **O:** set of observations
    - Cross product of domain of observable variables
  - **T(s,a,s') = Pr(s'|s,a):** transition function
    - Factored rep: Pr(s'|s,a) = $\prod_i$ Pr(V_i|parents(V_i))
  - **R(s,a) = r:** reward function
    - Sale = price x CC

Belief monitoring

- Belief: $b(s)$
  - Distribution over states
- Belief update: Bayes theorem
  - $b_{ao'}(s') = k \sum_{s \in S} b(s) \Pr(s'|s,a) \Pr(o'|a,s')$
  - $b_{ao'} = < o', a, b >$
- Demand learning and opponent modeling:
  - Implicit learning by belief monitoring

Policy trees

- Policy π
  - Mapping from past actions & obs to next action
  - Tree representation



  - Problem: tree grows exponentially with time

Policy Optimization

- Policy $\pi : B \rightarrow A$
  - mapping from beliefs to actions
- Value function $V^{\pi}(b) = \Sigma_t\ \gamma^t\ \mathrm{E}_{b_t|\pi}\ [R]$
- Optimal policy π^*:
  - $V^*(b) \geq V^{\pi}(b)$ for all π, b
- Bellman's Equation:
  - $V^*(b) = \max_a \mathrm{E}_b[R] + \gamma\ \Sigma_{o'} \Pr(o'|s,a)\ V^*(b_{ao'})$

Difficulties

- Exponentially large state space
  - $|\mathbf{S}| = \prod_i |\mathrm{dom}(V_i)|$
  - Solution: algebraic decision diagrams

- Complex policy space
  - Policy $\pi : B \rightarrow A$
  - Continuous belief space
  - Solution: point-based Bellman backups

Symbolic Perseus

Point-based value iteration	+	algebraic decision diagrams

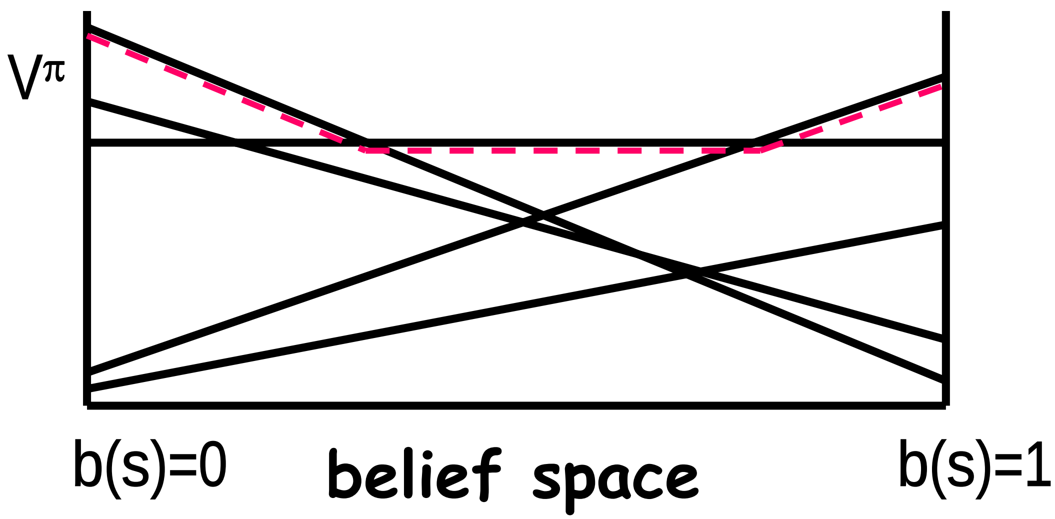
- Publicly available:
  - http://www.cs.uwaterloo.ca/~ppoupart/software.html
- Has been used to solve POMDPs with millions of states
- Currently used by
  - Intel, Toronto Rehabilitation Institute, Univ of Dundee, Technical Univ of Lisbon, Univ of British Columbia, Univ of Manchester, Univ of Waterloo

Piecewise linear & convex val fn

- Value of a policy tree β is linear

  $V^\beta(b_0) = \Sigma_{s \in S}\, b_0(s)\, V^\beta(s)$

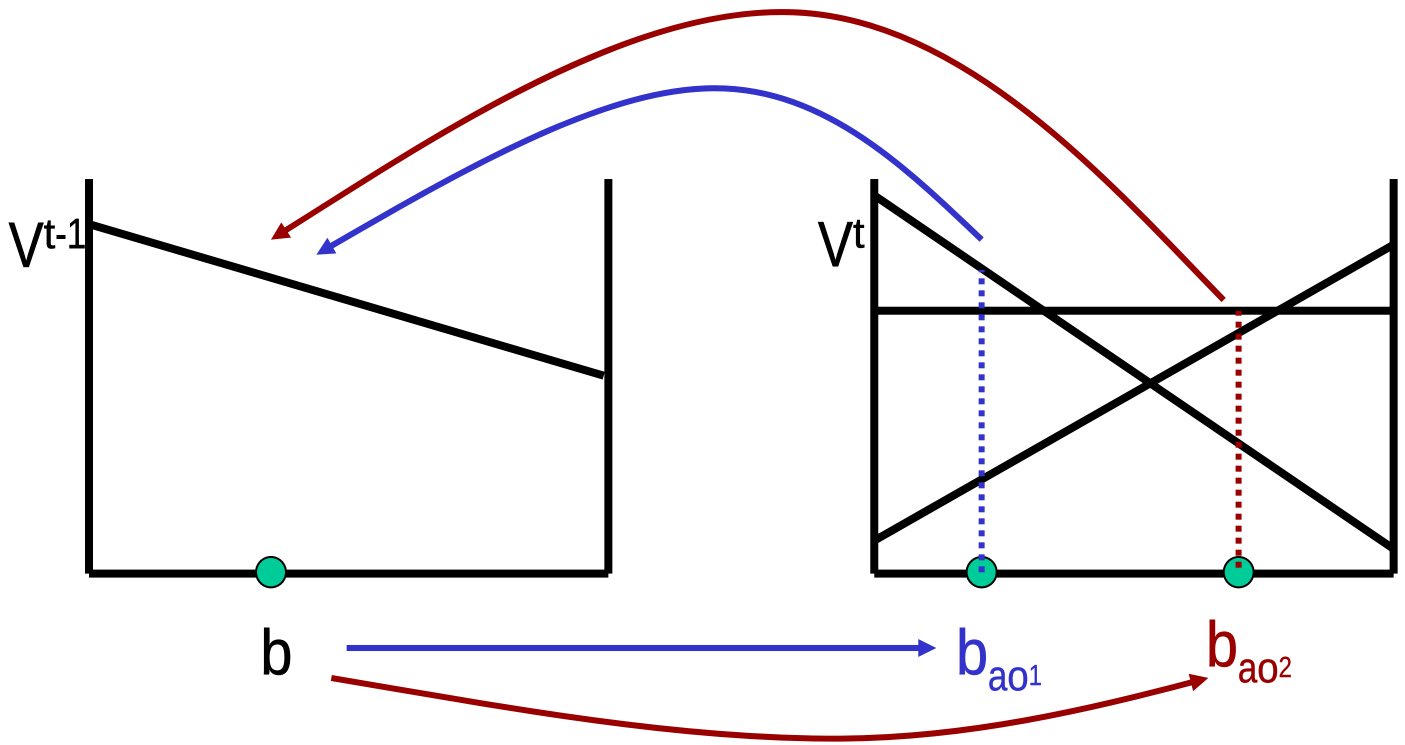
- Value of an optimal finite horizon policy is piecewise-linear and convex [SS73]



Point-based value iteration

- Point-based backup (Pineau & al. 2003)

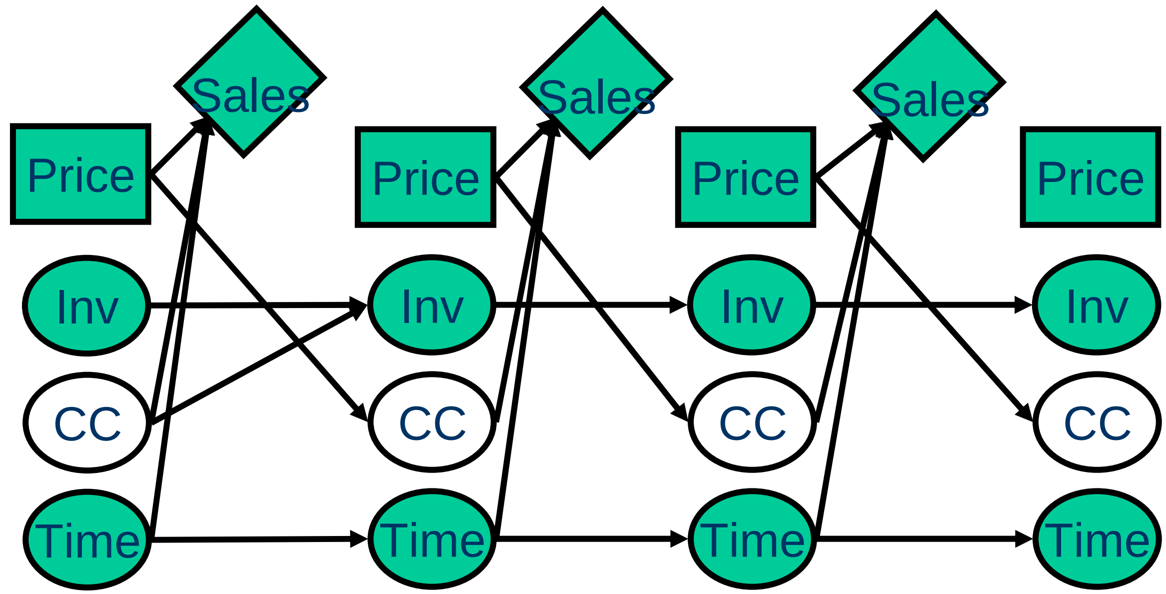
$$\alpha_{t-1}(b) = \max_a E_b[R] + \gamma \Sigma_{o'} \Pr(o'|s,a)\ \alpha_t(b_{ao'})$$



Algebraic Decision Diagrams

- First use in MDPs: Hoey et al. 1999
- Factored Representation
  - Exploit conditional independence
  - $Pr(s'|s,a) = \prod_i Pr(V_i|parents(V_i))$
- Automatic State aggregation
  - Exploit context specific independence
  - Exploit sparsity

Factored Representation



- Transition fn: Pr(s'|s,a)
  - Flat representation: matrix $O(|S|^2)$
  - Factored representation: often $O(\log |S|)$

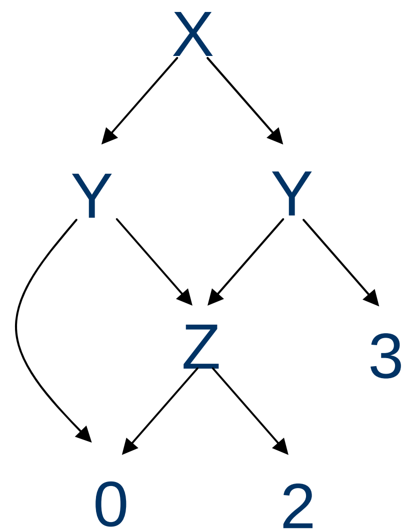
Computation with Factored Rep

- Belief monitoring:
  - $b_{ao'}(s') = k \Pr(o'|a,s') \Sigma_s b(s) \Pr(s'|s,a)$
- Point-based Bellman backup:
  - $\alpha(s) = \max_a R(s,a) + \Sigma_{s'o'} \Pr(s'|s,a) \Pr(o'|a,s') \alpha_{ao'}(s')$

- Flat representation: $O(|S|^2)$
- Factored representation: often $O(|S| \log |S|)$

Algebraic Decision Diagrams

- Tree-based representation
  - Acyclic directed graph
- Avoid duplicate entries
  - Exploit context specific independence
  - Exploit sparsity



xyz	0
xy~z	0
x~yz	0
x~y~z	2

~xyz	0
~xy~z	2
~x~yz	3
~x~y~z	3

Empirical Results

- Monopolistic Dynamic Pricing

Inv / Time	|S|	SP Value	Upper bound	Runtime (min)
10 / 20	73,920	121	133	19
15 / 30	158,720	152	167	48
20 / 40	275,520	171	187	61
25 / 50	424,320	182	198	161
30 / 60	605,120	188	199	350
35 / 70	817,920	192	199	448

COACH project

- Automated prompting system to help elderly persons wash their hands
- IATSL: Alex Mihailidis, Jesse Hoey, Jennifer Boger et al.

Policy Optimization

- Partially observable MDP:
  - Handle noisy HandLocation and noisy WaterFlow
  - Can adapt to user responsiveness
  - 50,181,120 states, 20 actions, 12 observations

- Approximation: fully observable MDP
  - Assume HandLocation, WaterFlow are fully observable
  - Remove responsiveness user variable
  - 25,090,560 states, 20 actions

Empirical Comparison (Simulation)

DL/RE/AW	PO-MDP	Heur-istic	Null	CG	CE	fo-MDP
lo/none/never	3.8 ± 1.2	-1.1 ± 0.9	-2.0 ± 0.0	-75.2 ± 3.2	6.8 ± 0.6	9.1 ± 0.4
lo/max/no	3.0 ± 0.5	2.3 ± 0.7	-0.9 ± 0.1	-92.1 ± 4.2	2.8 ± 1.2	6.3 ± 0.7
lo/med/yes	4.5 ± 1.1	3.9 ± 0.6	0.1 ± 0.5	-117.8 ± 4.0	-0.2 ± 0.7	7.4 ± 0.7
med/max/no	1.1 ± 1.0	1.4 ± 0.7	0.2 ± 0.3	-93.6 ± 3.9	3.4 ± 0.8	6.0 ± 0.8
med/min/yes	5.1 ± 0.9	6.3 ± 0.7	3.1 ± 1.5	-118.4 ± 4.3	0.9 ± 0.9	8.1 ± 0.6
hi/med/no	7.1 ± 0.7	5.6 ± 0.4	0.4 ± 0.3	-95.6 ± 3.9	7.2 ± 0.7	9.3 ± 0.6
hi/min/yes	8.3 ± 0.7	9.8 ± 0.6	9.7 ± 0.9	-118.5 ± 4.3	3.7 ± 1.0	9.1 ± 0.7
ρ_0	4.9 ± 1.1	3.8 ± 2.8	0.9 ± 3.3	-97 ± 16	4.2 ± 2.5	8.3 ± 1.1
ρ_δ	$\mathbf{4.8 \pm 0.6}$	$\mathbf{4.6 \pm 1.0}$	$\mathbf{0.5 \pm 2.1}$	$\mathbf{-105 \pm 13}$	$\mathbf{2.9 \pm 2.4}$	$\mathbf{7.9 \pm 0.8}$

Conclusion

- Natural encoding of Dynamic Pricing as POMDP
  - Demand and competitor learning by belief monitoring
  - Factored model
- Symbolic Perseus (generic POMDP solvers)
  - Point-based value iteration + algebraic decision diagrams
  - Exploit problem specific structure

- Future work
  - Bayesian reinforcement learning
  - Planning as inference