Scalable Bayesian Learning in Factored Partial Observable Environments

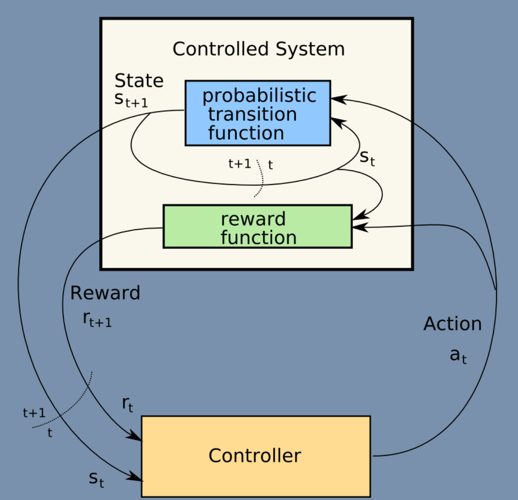
Sammie Katt
Frans Oliehoek
Christopher Amato
University of New Hampshire

Overview

- Robots must often operate in complex domains where the exact dynamics are unknown
- Learning from past experiences is crucial for such domains
- Current approaches rarely exploit structure that systems exhibit, discarding learning opportunities
- We demonstrate an efficient sample-based method of learning both the structure and dynamics of environments

Bayes-Adaptive MDP

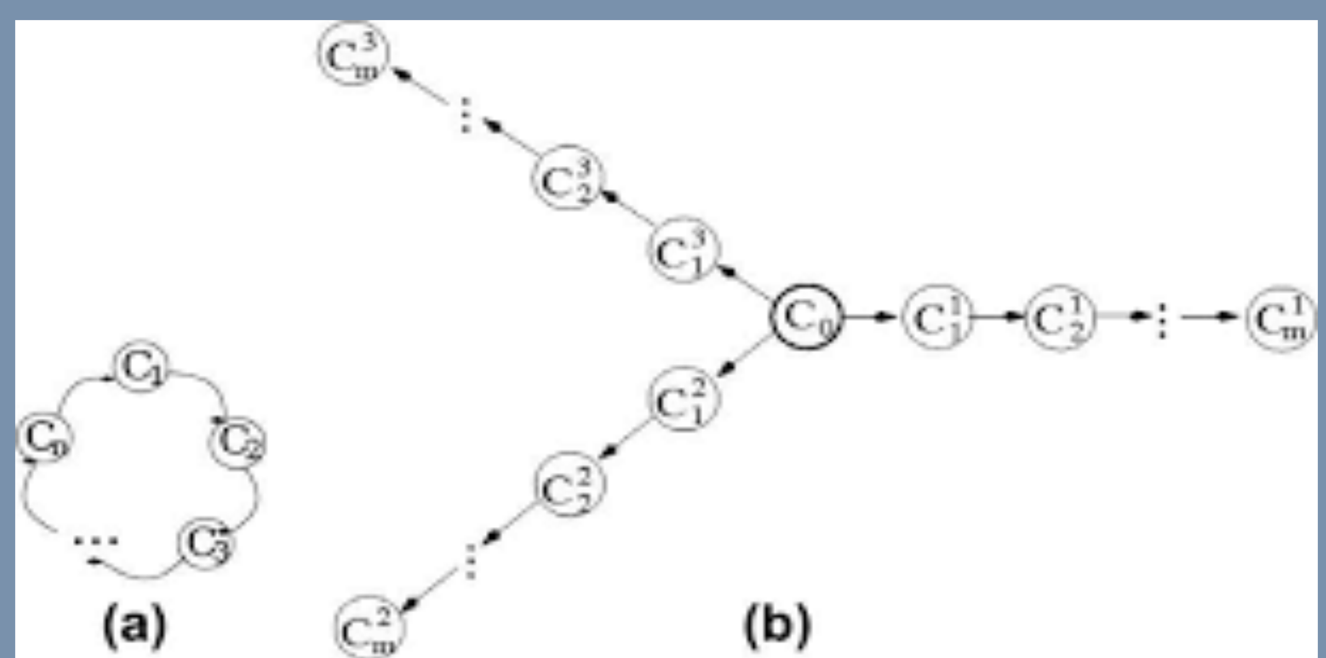
- BAMDP: Bayesian model learning for MDPs
- BAMDP can be solved as a POMDP with a believe over the T
- T - state transition model: P(s'|a,s)
- R - reward function: R(s,a)
- S - state-space ⊗ T
- A - action space
- γ - discount factor



Representation of agent-world interaction.
Credit: Wikipedia

SysAdmin

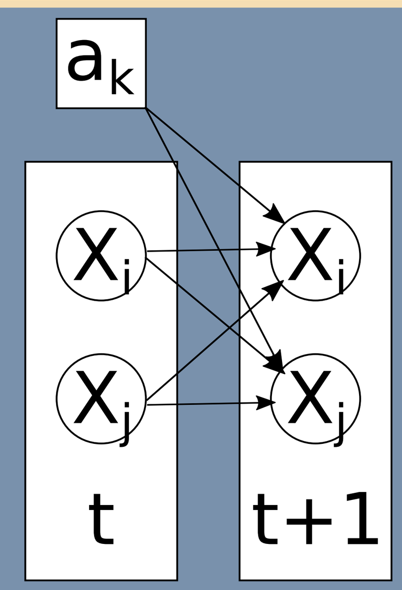
- Structured problem for fully observable MDP
- N computers, either 'on' or 'off' (2^N states)
- Agent can reboot one of the computers per step
- Reward is based on the amount of computers 'on'
- Small chance failing each time step
- Connected 'off' computers are contagious



Two possible structures in the Sysadmin problem.
A) Unidirectional circle and B) Star-connection.
Delgado, Karina Valdivia, et al. (2011)

Factored BAMDP

- State is represented as X: $\{X_1, X_2 \ldots X_n\}$ features
- Models represented as Dynamic Bayesian Networks
- Nodes in the DBN graph G represent features, Θ_G specifies the probabilities
- $P(s'|s,G,\Theta_G,a) = \prod_i \Theta_G^{i,s_i'|ParVal_i(s,G_a)}$



Fully connected temporal DBN

Partial observability

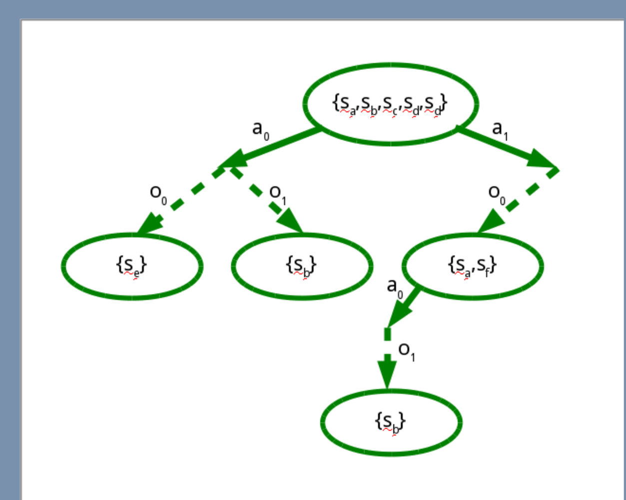
- Partial observability introduces new form of uncertainty
- O - observation model: P(o|a,s'); Ω - observation space
- Maintain a belief over current state, which includes a belief over models in Bayes-Adaptive approach
- Belief maintained as single particle filter, where each particle contains a system state, T and O

Process \ State structure	Flat	Known Structure (G)	Unknown Structure
MDP	s	$\mathbf{X}$	$<\mathbf{X}, b(G)>$
POMDP	$b(s)$	$b(\mathbf{X})$	$b(\mathbf{X}, b(G))$
BAMDP	$<s, \phi_s>$	$<\mathbf{X}, \phi_G>$	$<\mathbf{X}, b(G^T, \phi_G)>$
BAPOMDP	$b(s, \phi_s, \psi_s)$	$b(\mathbf{X}, \phi_G, \psi_G)$	$b(\mathbf{X}, b(G^{T,O}, \phi_G, \psi_G))$

Table of internal representation of belief over system. Up to down increases uncertainty, left to right introduces factorization.

POMCP

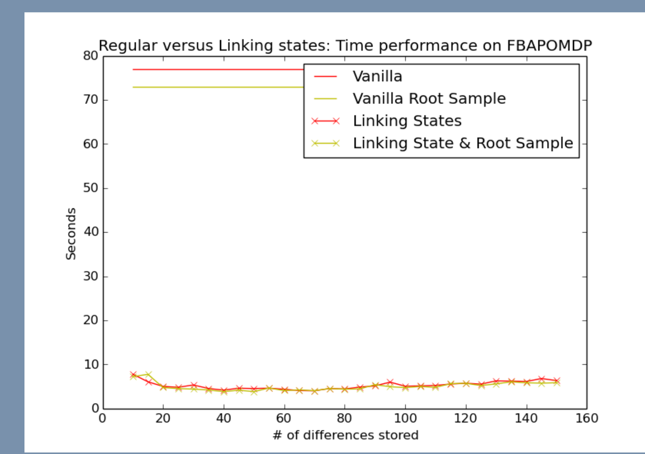
- Online, sampled-based planning method
- Constructs tree of action of observation history
- Root samples (hyper)state from belief
- Extendable to factored representations
- Requires vast amount of hyper state copies

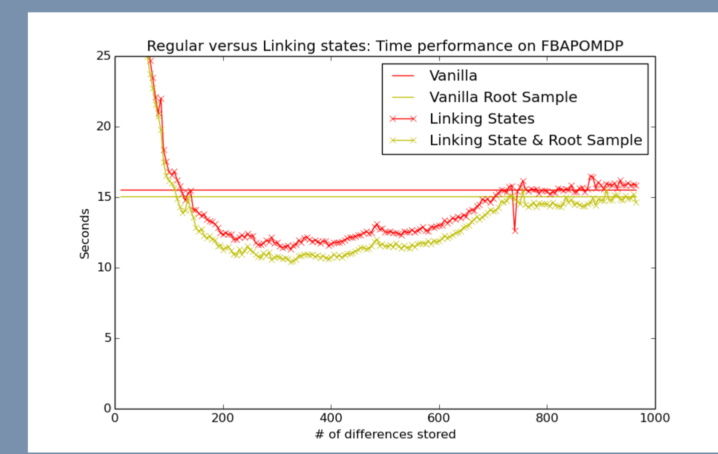


Part of the POMCP tree.

Optimization: Linking States

- Increase size of hyper-state results in large copy-time
- POMCP becomes slow in (generally huge) BAPOMDP
- Linking States store differences rather than full states, making copying and therefor POMCP faster
- Linking States tuple <s,δ,ℓ>
- ℓ: link to models, may be shared
- δ: list of differences between Linking State and ℓ

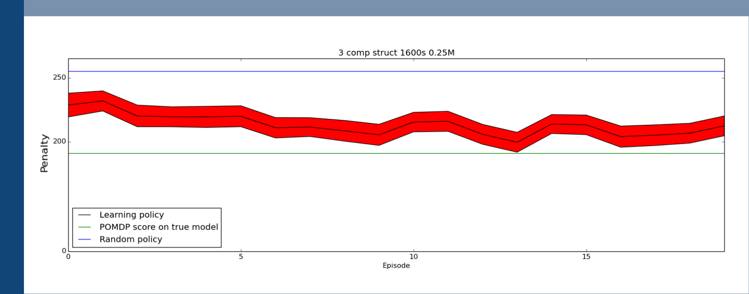


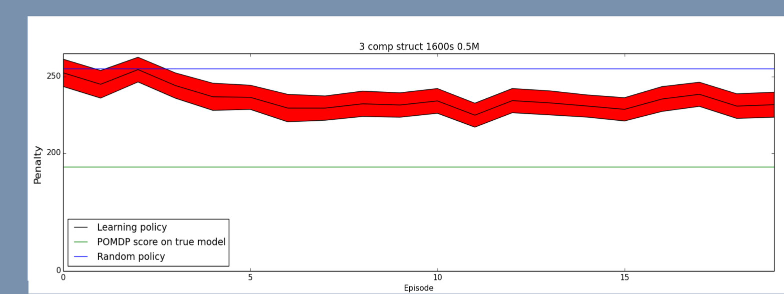


Comparing Linking States and Root Sampling to regular POMCP

Learning Structure

- POSysAdmin with 3 fully connected computers
- 1600 simulations per step, 5000 particles in the believe
- Allow learning for 20 episodes with horizon 20





Performance over episodes with initial uncertain structure.
0.25 uncertainty on the left
0.5 uncertainty on the right

- POMCP on FBAPOMDP significantly improves policy when initialy faced with uncertainty, by learning the structure

Conclusion

- Initial experiments imply POMCP is able to learn structure from an initial uncertain believe
- The increase in hyper-state space slows down POMCP, but Linking States prove to be a generally applicable speed up
- Further research is necessary for better understanding of simultaneously learning structure and dynamics