Hierarchical POMDP Controller Optimization by Likelihood Maximization

Marc Toussaint

Machine Learning & Robotics Group – TU Berlin

mtoussai@cs.tu-berlin.de

UAI 2008, Helsinki, July 11th, 2008

joint work with

Laurent Charlin, University of Toronto

&

Pascal Poupart, University of Waterloo

POMDPs

- Partially Observable Markov Decision Processes

$\mathcal{S}$	state space
$\mathcal{A}$	action space
$\mathcal{O}$	observation space
$P(S_0)$	start distribution
$P(S_{t+1} \mid S_t, A_t)$	transition probabilities
$P(O_t \mid S_t, A_{t-1})$	observation probabilities
$R(A_t, S_t)$	reward function

POMDPs

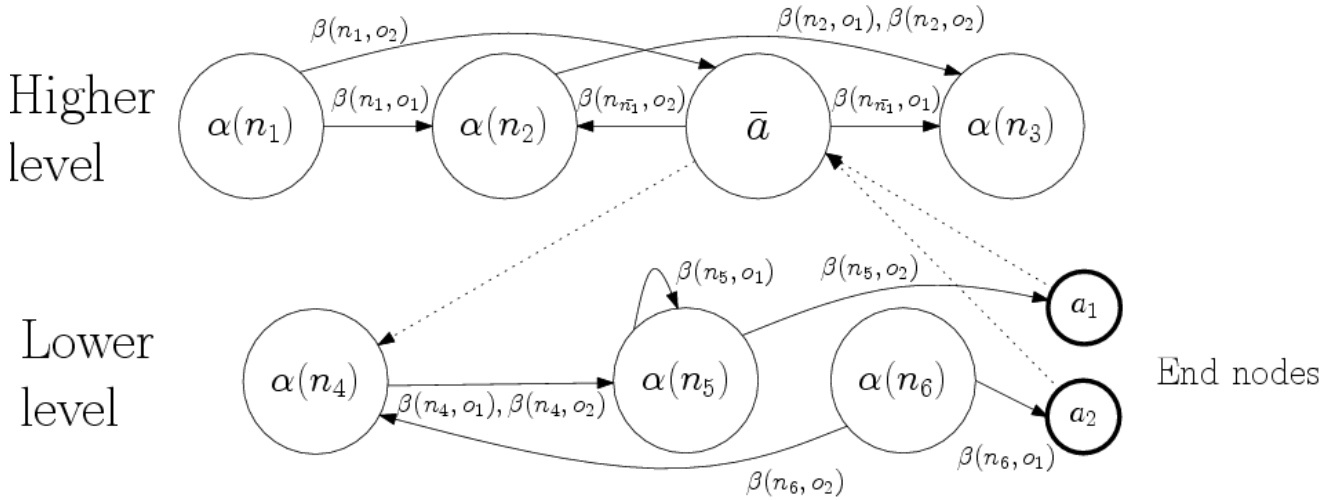
- Partially Observable Markov Decision Processes

$\mathcal{S}$	state space
$\mathcal{A}$	action space
$\mathcal{O}$	observation space
$P(S_0)$	start distribution
$P(S_{t+1} \mid S_t, A_t)$	transition probabilities
$P(O_t \mid S_t, A_{t-1})$	observation probabilities
$R(A_t, S_t)$	reward function

- goal: find controller that maximizes $\mathsf{E}\{\sum_{t=0}^{\infty} \gamma^t r_t\}$

hierarchical FSCs

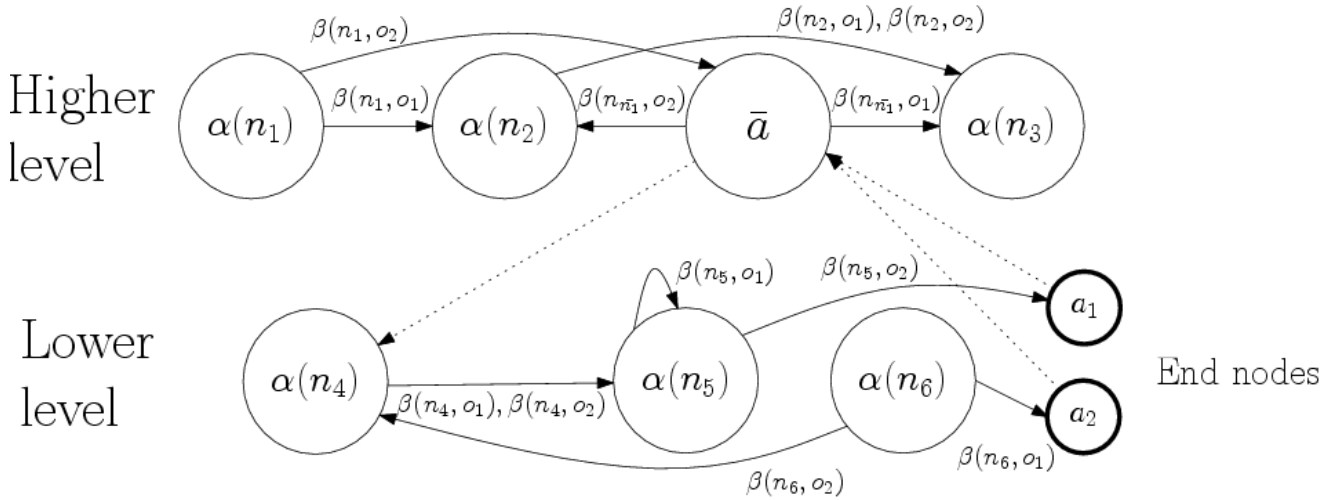
- hierarchical finite state controller



– Hansen & Zhou (ICAPS, 2003)

hierarchical FSCs

- hierarchical finite state controller

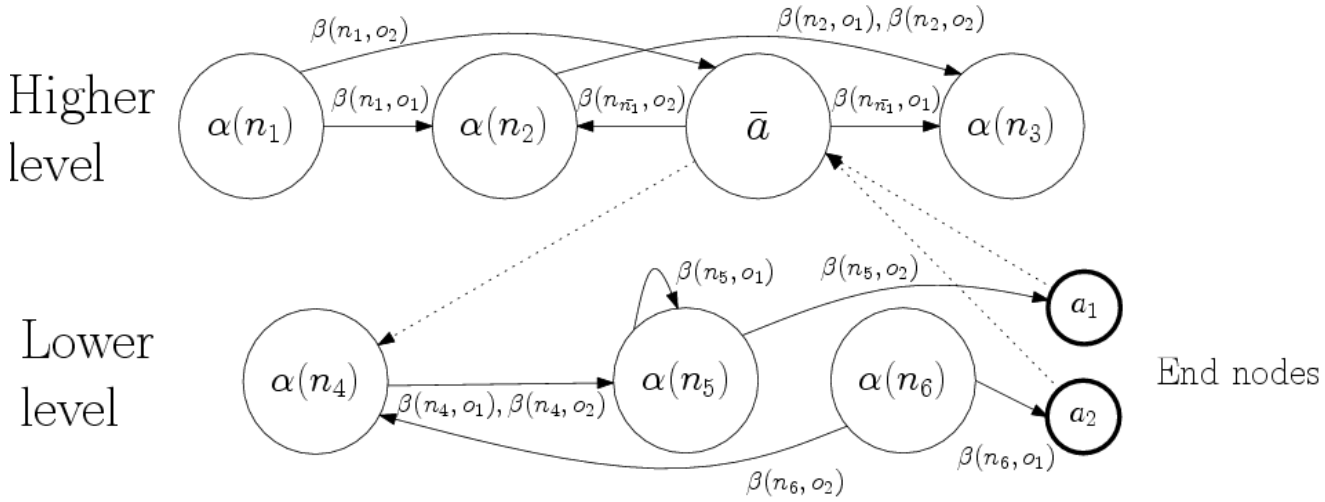


  – Hansen & Zhou (ICAPS, 2003)

- decomposing the problem in sub-tasks
  – problem: discovering the hierarchical decomposition in POMDPs

hierarchical FSCs

- hierarchical finite state controller



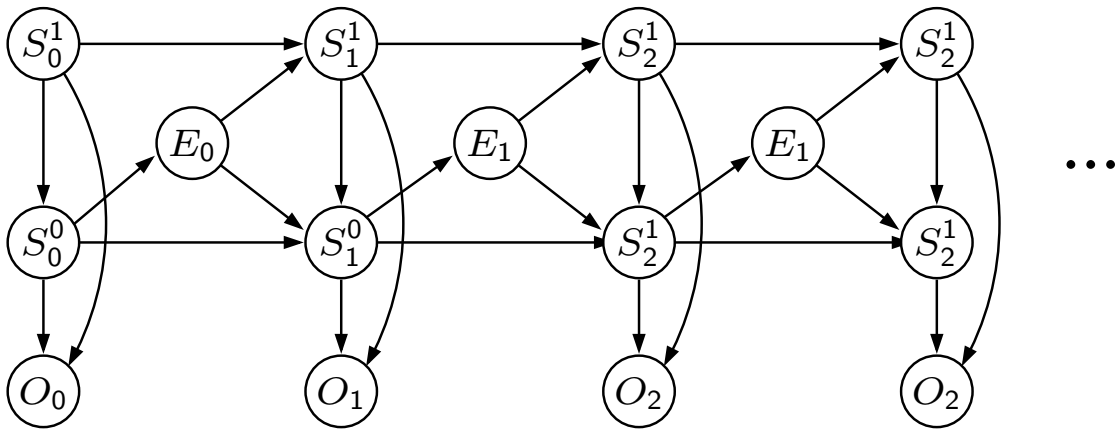
  – Hansen & Zhou (ICAPS, 2003)

- decomposing the problem in sub-tasks
  – problem: discovering the hierarchical decomposition in POMDPs

- Charlin, Poupart & Shioda (NIPS 2007): Hierarchy Discovery
  – general recursive controllers
  – non-convex quartically constrained optimization problem

representing hierarchies in DBNs

- hierarchical HMMs

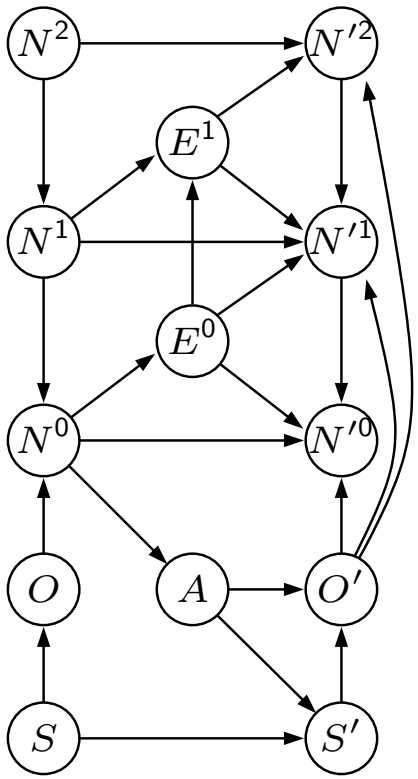


– example: A B C D E F D E F A B C D E F A B C A B C ...

– Murphy & Paskin (NIPS 2001): Linear time inference in hierarchical HMMs

representing hierarchies in DBNs

- hierarchical finite state controller for a POMDP



given (POMDP):

$P(S' \mid A, S)$

$P(O' \mid S', A)$

to be learnt (controller):

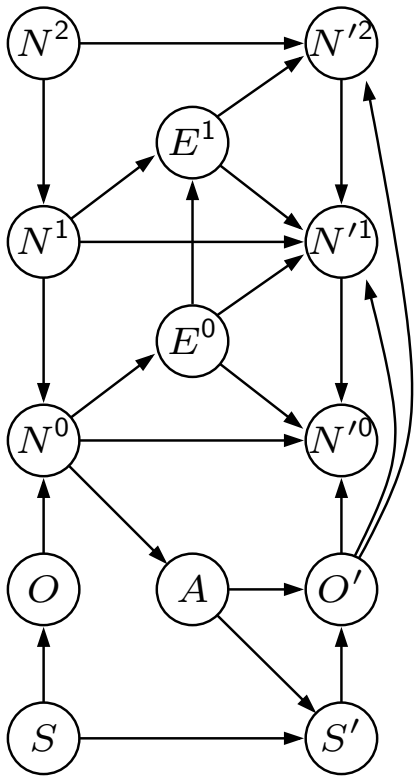
$P(N'^i \mid N^i, \dots)$

$P(E^i \mid N^i, \dots)$

$P(A \mid N^0)$

representing hierarchies in DBNs

- hierarchical finite state controller for a POMDP



given (POMDP):

$P(S' \mid A, S)$

$P(O' \mid S', A)$

to be learnt (controller):

$P(N'^i \mid N^i, \dots)$

$P(E^i \mid N^i, \dots)$

$P(A \mid N^0)$

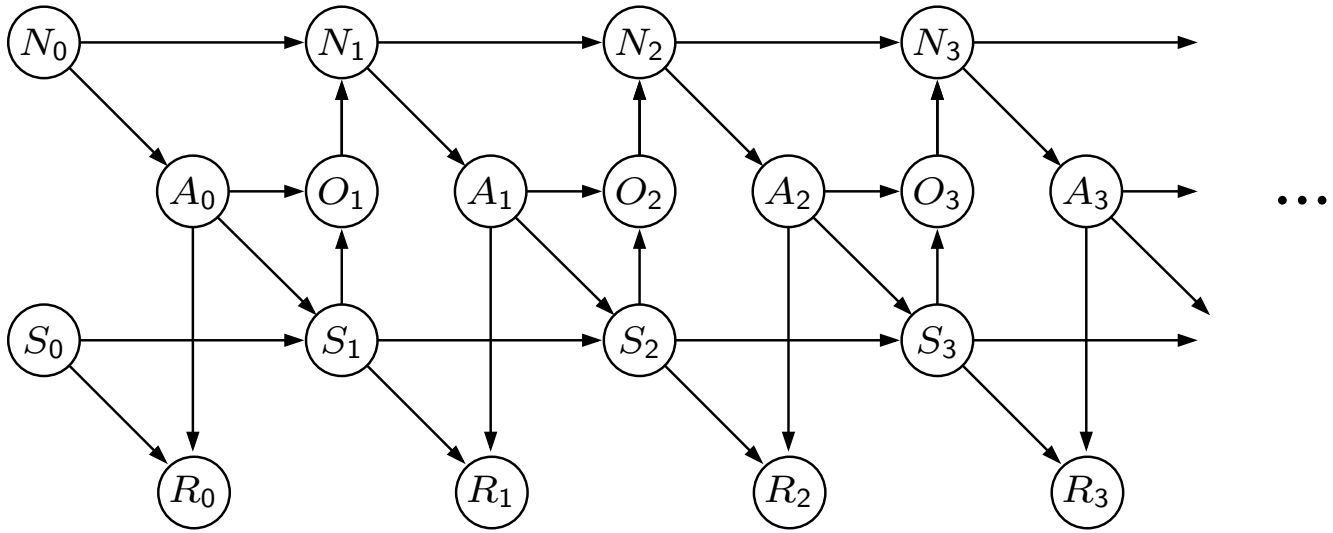
- previous hierarchical controllers are special case (except recursiveness)

outline

- POMDPs & hierarchical controllers
- Expectation-Maximization for controller optimization 
- results

Expectation Maximization for controller optimization

- consider the POMDP with (flat) FSC

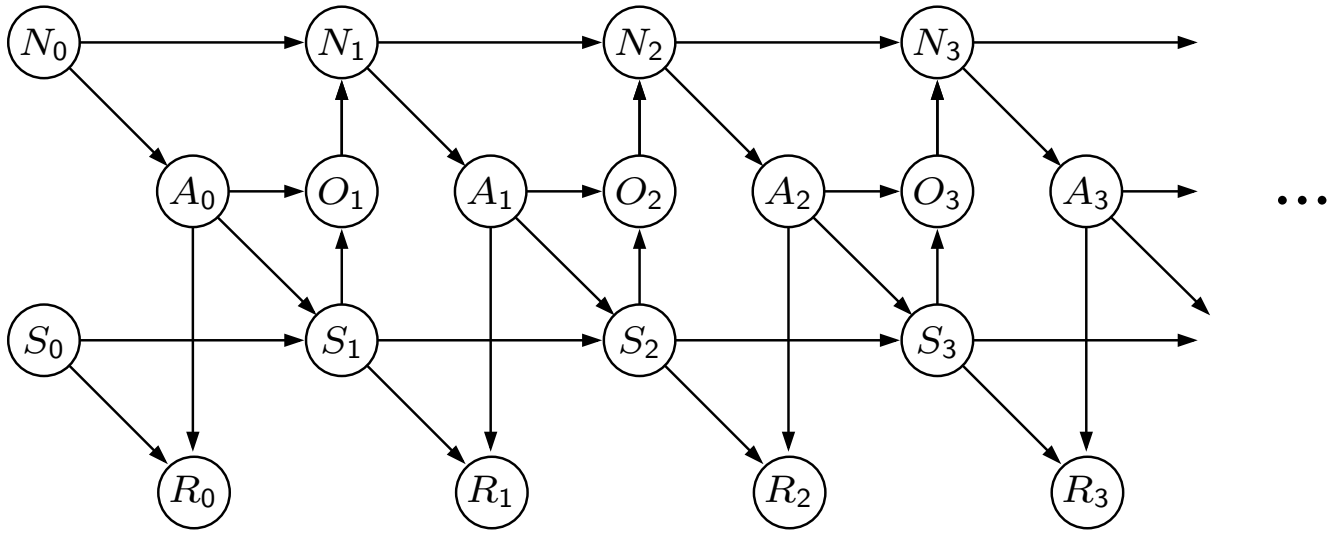


  – rewards in every time step

  – maximize $\mathrm{E}\{\sum_{t=0}^{\infty} \gamma^t r_t\}$

Expectation Maximization for controller optimization

- consider the POMDP with (flat) FSC



  – rewards in every time step

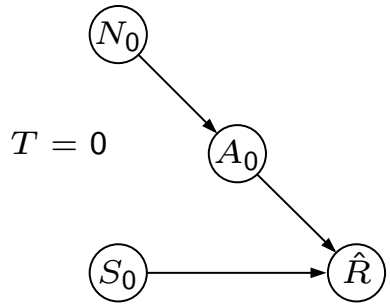
  – maximize $\mathrm{E}\{\sum_{t=0}^{\infty} \gamma^t r_t\}$

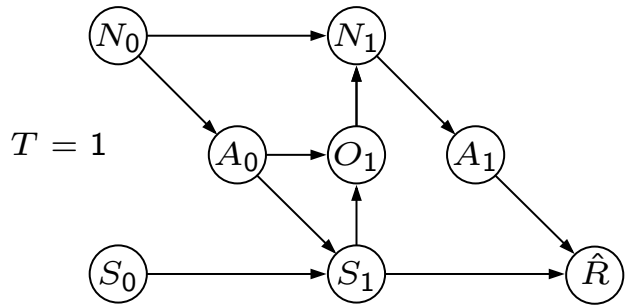
- rough idea:

  compute parameters that maximize the likelihood of observing rewards

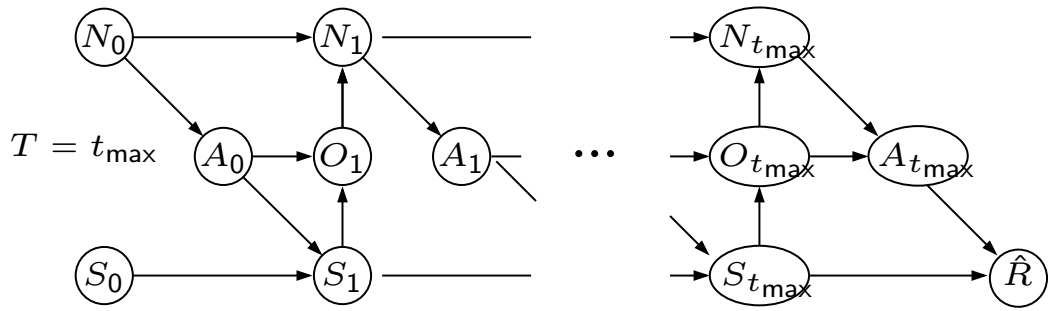
Expectation Maximization for controller optimization

- mixture of finite DBNs





...



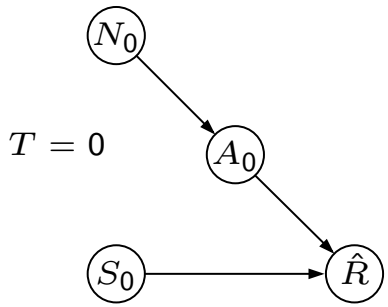
– mixture $P(T=t) = \gamma^t(1-\gamma)$
mimics the discounting

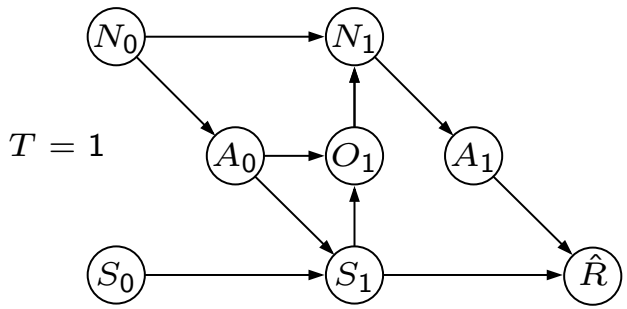
$$\mathsf{E}\{\hat{R}\} = \sum_t P(T=t)\mathsf{E}\{\hat{R} \mid T=t\}$$
$$\propto (1-\gamma)\ \mathsf{E}\{\sum_t \gamma^t r_t\}$$

if $\hat{R}$ s.t. $\mathsf{E}\{\hat{R} \mid a, s\} \propto R(a, s)$

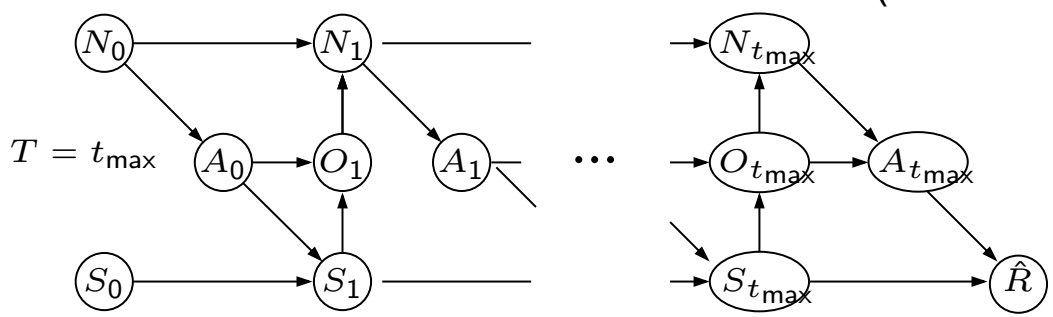
Expectation Maximization for controller optimization

- mixture of finite DBNs





...



– mixture $P(T=t) = \gamma^t(1-\gamma)$
mimics the discounting

$$\mathsf{E}\{\hat{R}\} = \sum_t P(T=t)\mathsf{E}\{\hat{R} \mid T=t\}$$
$$\propto (1-\gamma)\ \mathsf{E}\{\sum_t \gamma^t r_t\}$$

if $\hat{R}$ s.t. $\mathsf{E}\{\hat{R} \mid a, s\} \propto R(a,s)$

maximizing likelihood $\hat{R}=1$ in mixture
$\Longleftrightarrow$ maximizing expected return

(cf. Toussaint & Storkey, ICML 2006)

Expectation Maximization for controller optimization

- *policy optimization is framed as maximizing likelihood* $\hat{R} = 1$

Expectation Maximization for controller optimization

- *policy optimization is framed as maximizing likelihood* $\hat{R}=1$

- EM algorithm

  E-step: expectations (conditioned on $\hat{R}=1$) over hiddens

  $E_n = \Pr(N_0=n \mid \hat{R}=1)$

  $E_{an} = \sum_t \Pr(A_t=a, N_t=n \mid \hat{R}=1)$

  $E_{n'no'} = \sum_t \Pr(N_{t+1}=n', N_t=n, O_{t+1}=o' \mid \hat{R}=1)$

  M-step: relative frequency

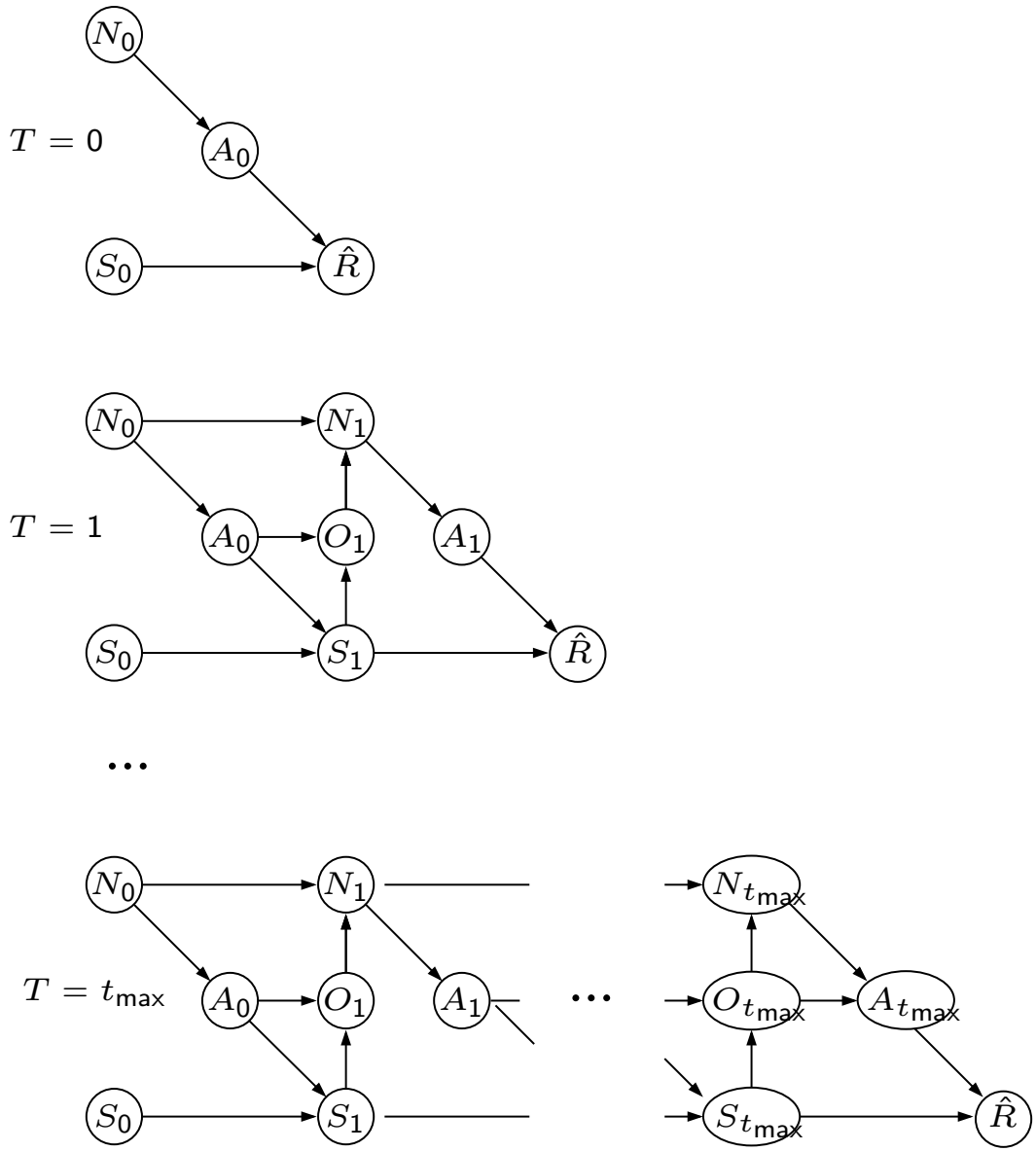
  $p_n = E_n / \sum_n E_n$

  $p_{a|n} = E_{an} / \sum_a E_{an}$

  $p_{n'|no'} = E_{n'no'} / \sum_{n'} E_{n'no'}$

inference in the mixture of DBNs

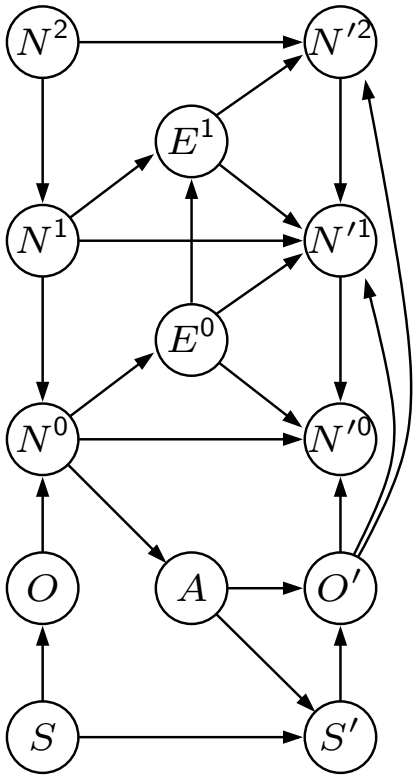
- only single fwd and bwd sweep for inference in all DBNs



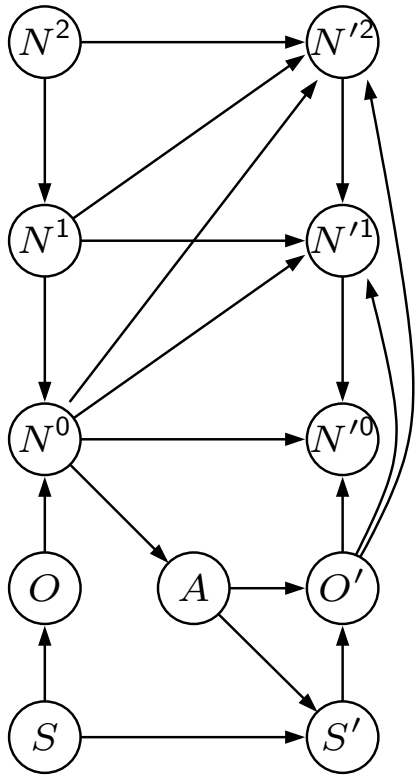
inference in the mixture of DBNs

- exploiting the controller's structure

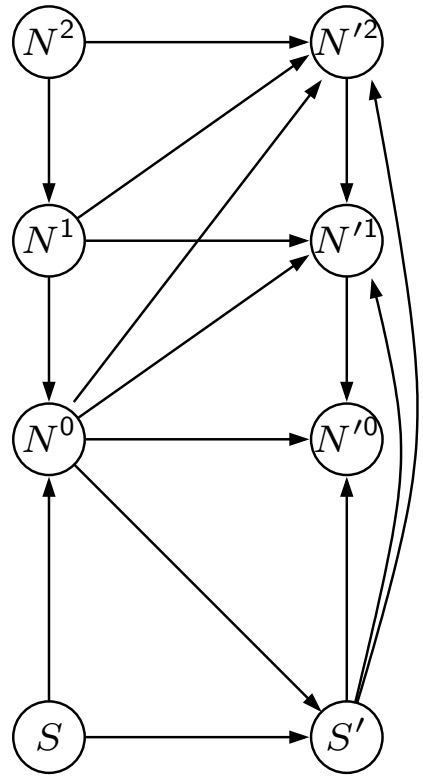
strict hierarchical



factored



junction tree

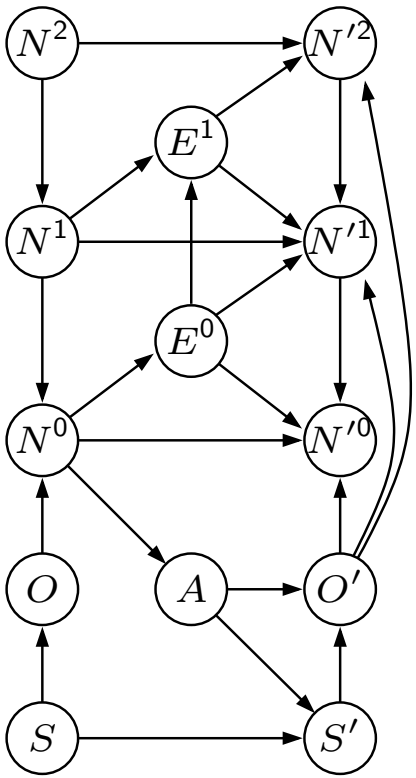


- complexity scales with the largest clique size
- we use modified (more greedy) M-step (Neal & Hinton, 1998)

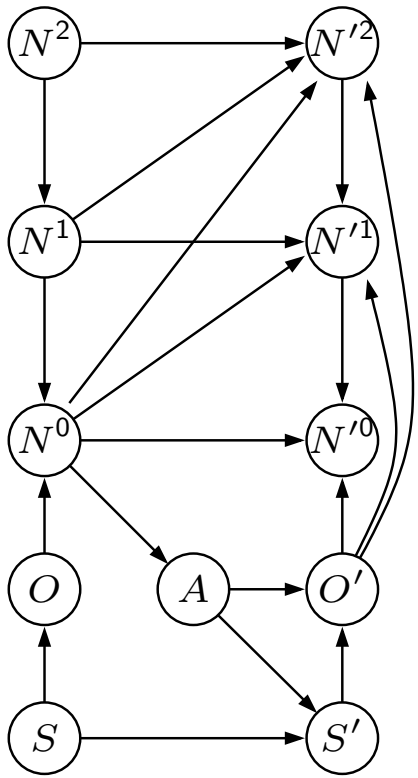
inference in the mixture of DBNs

- exploiting the controller's structure

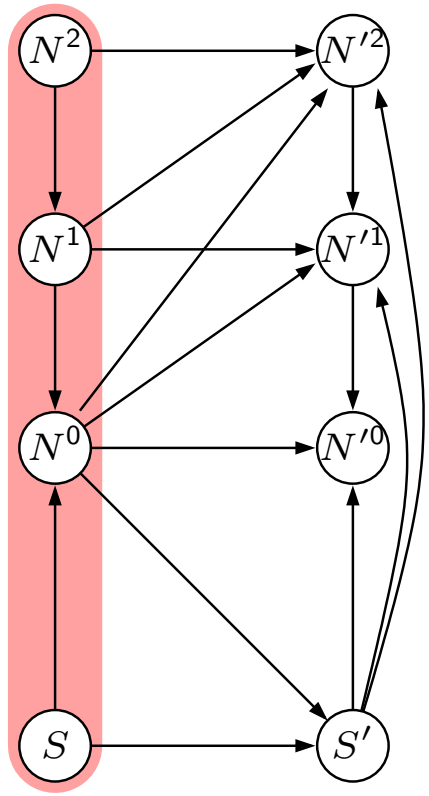
strict hierarchical



factored



junction tree

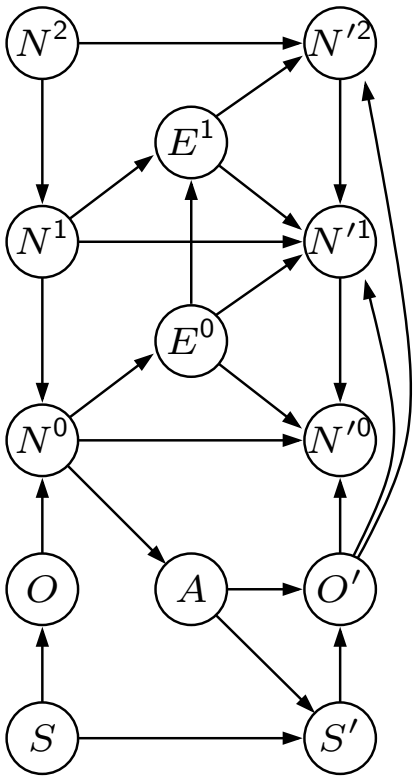


- complexity scales with the largest clique size
- we use modified (more greedy) M-step (Neal & Hinton, 1998)

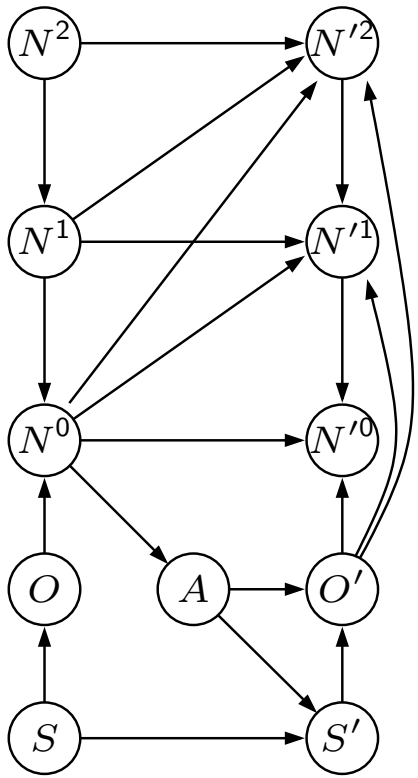
inference in the mixture of DBNs

- exploiting the controller's structure

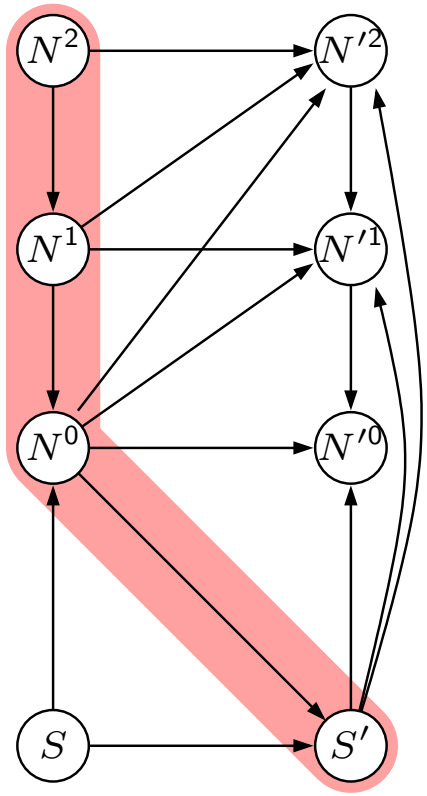
strict hierarchical



factored



junction tree

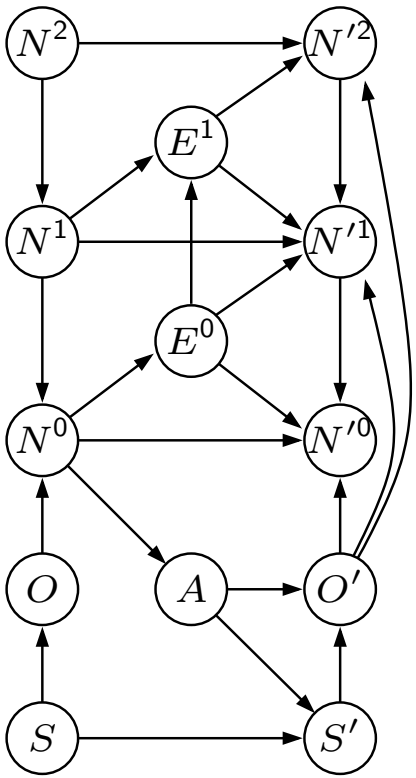


- complexity scales with the largest clique size
- we use modified (more greedy) M-step (Neal & Hinton, 1998)

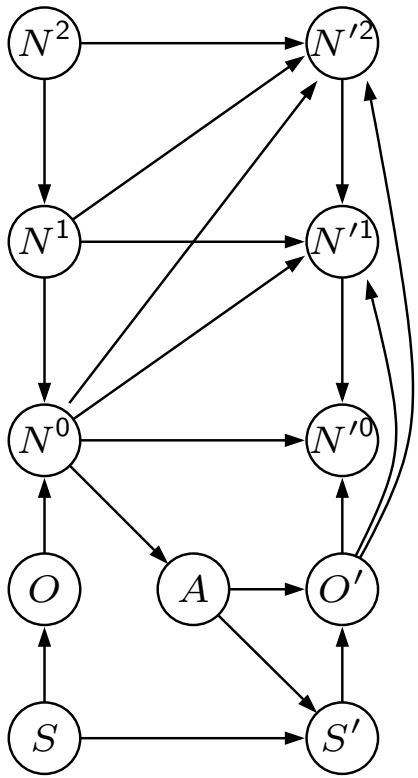
inference in the mixture of DBNs

- exploiting the controller's structure

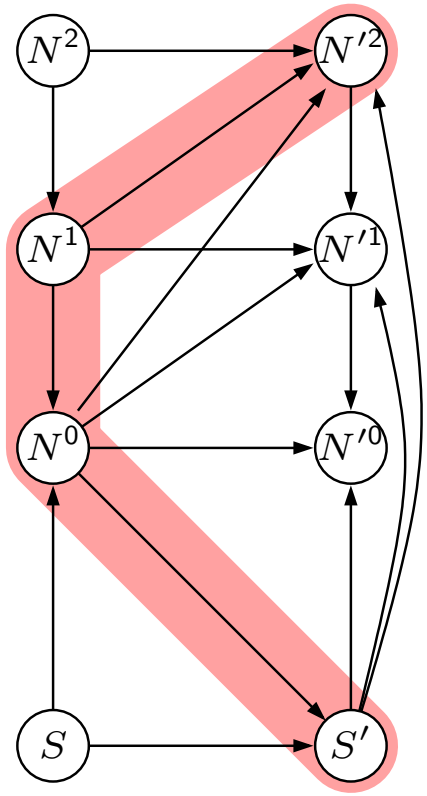
strict hierarchical



factored



junction tree

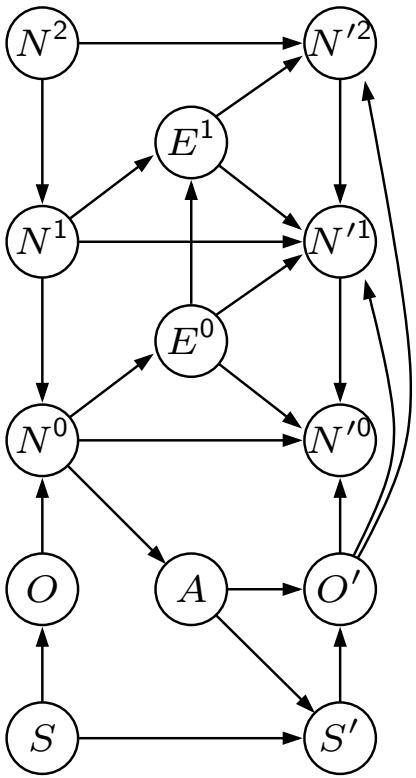


- complexity scales with the largest clique size
- we use modified (more greedy) M-step (Neal & Hinton, 1998)

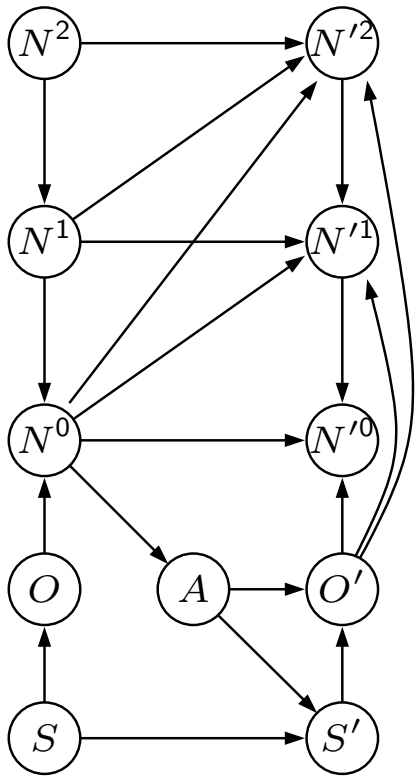
inference in the mixture of DBNs

- exploiting the controller's structure

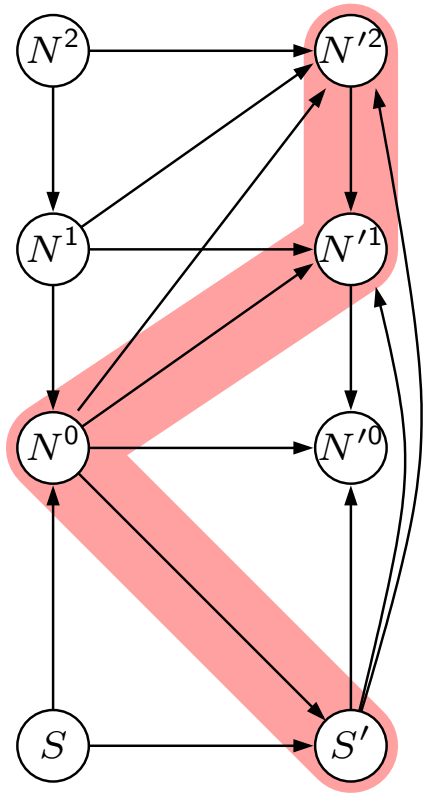
strict hierarchical



factored



junction tree

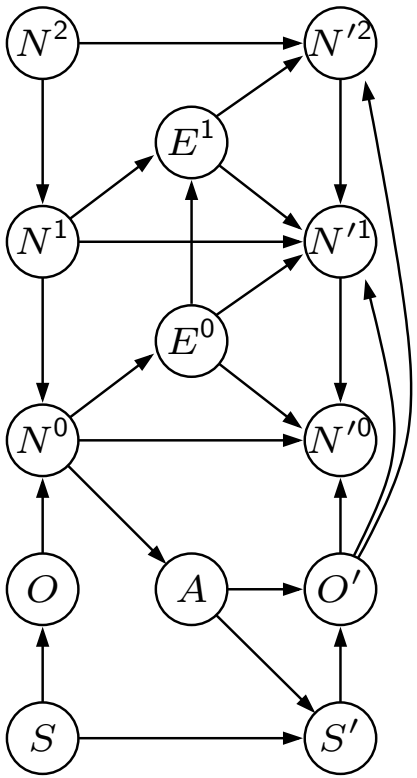


- complexity scales with the largest clique size
- we use modified (more greedy) M-step (Neal & Hinton, 1998)

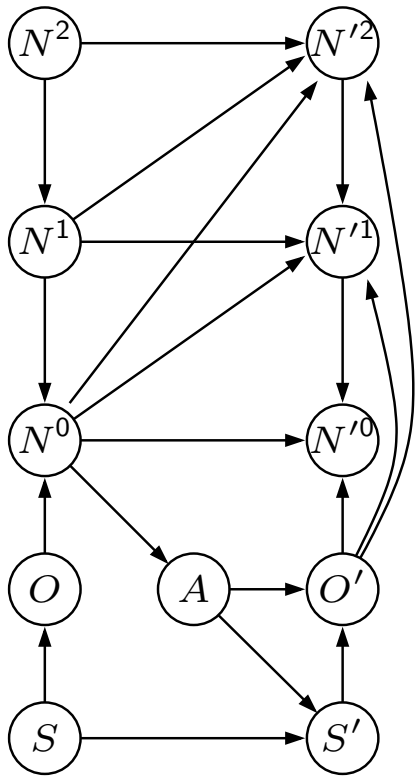
inference in the mixture of DBNs

- exploiting the controller's structure

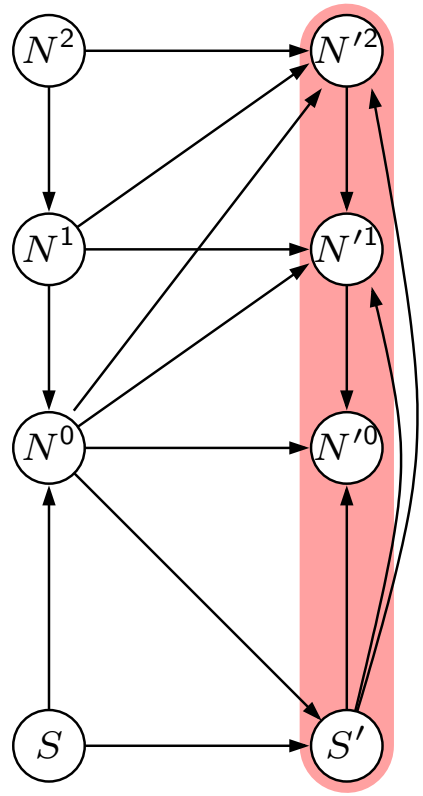
strict hierarchical



factored



junction tree



- complexity scales with the largest clique size
- we use modified (more greedy) M-step (Neal & Hinton, 1998)

outline

- POMDPs & hierarchical controllers
- Expectation-Maximization for controller optimization
- results 

results

- ML approach
  - 2-level controllers
  - tested various controller sizes ($|\mathcal{N}^{base}|$, $|\mathcal{N}^{top}|$), each 10 runs

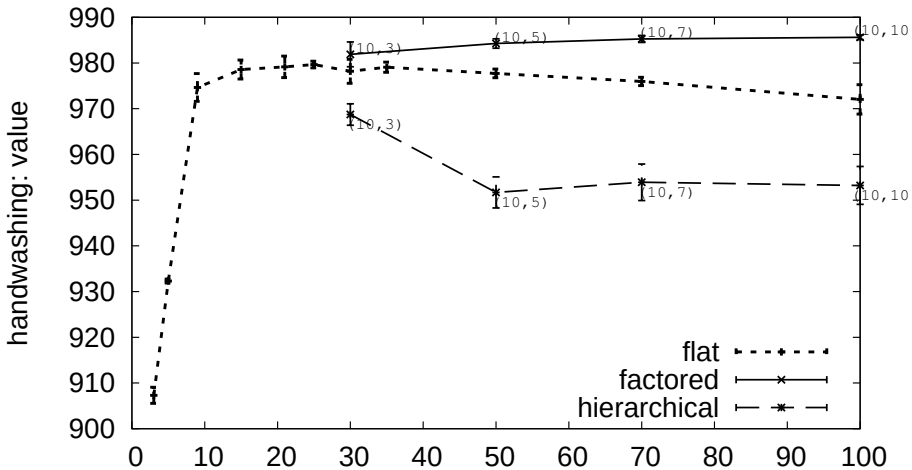
Problem ($\|\mathcal{S}\|$,$\|\mathcal{A}\|$,$\|\mathcal{O}\|$)	V^*	**HSVI2 [2]**	**best from [1]**			**ML approach** (avg. 10 runs)		
		V	$\|\mathcal{N}\|$	t(s)	V	$\|\mathcal{N}\|$	t(s)	V
paint (4, 4, 2)	3.28	3.29 ± 0.04	(1,3)	< 1	3.29	(5,3)	0.96 ± 0.3	3.26 ± 0.004
shuttle (8, 3, 5)	32.7	32.9 ± 0.8	(1,3)	2	31.87	(5,3)	2.81 ± 0.2	31.6 ± 0.5
4x4 maze (16, 4, 2)	3.7	3.75 ± 0.1	(1,2)	30	3.73	(3,3)	2.8 ± 0.8	$3.72 \pm 8e-5$
chain-of-ch. (10, 4, 1)	157.1	157.1 ± 0	(3,3)	10	0.0	(10,3)	6.4 ± 0.2	151.6 ± 2.6
handwash. (84, 7, 12)	$\leqslant 1052$	N/A			N/A	(10,5)	655 ± 2	984 ± 1
cheese-taxi (33, 7, 10)	$\leqslant 5.3$	2.53 ± 0.3			N/A	(10,3)	311 ± 14	$-9 \pm 11 (2.25^*)$

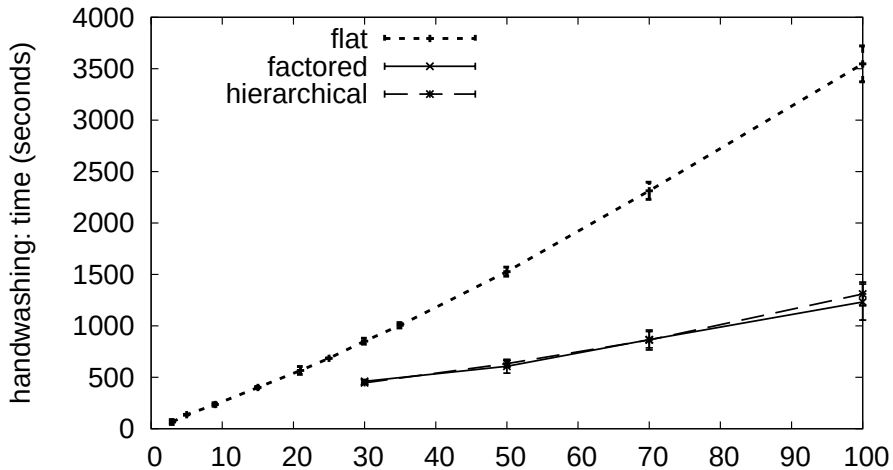
[1] Charlin, Poupart & Shioda (NIPS 2007): Automated Hierarchy Discovery for Planning in Partially Observable Environments

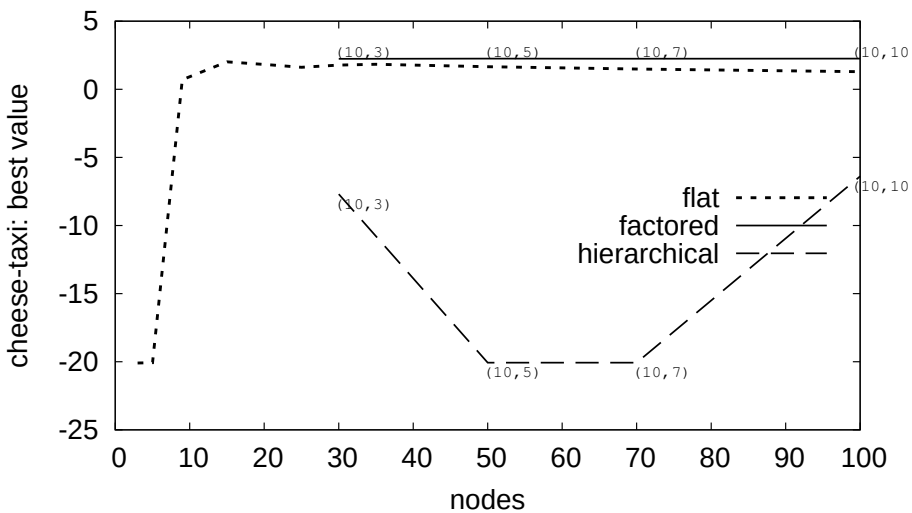
[2] Smith & Simmons (UAI 2004): Heuristic search value iteration for POMDPs

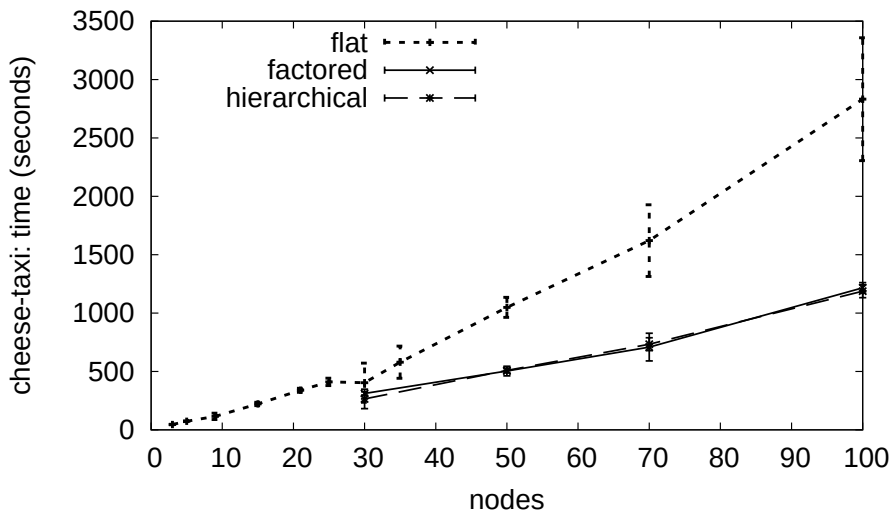
results

- value & runtime for various $(|\mathcal{N}^{base}|, |\mathcal{N}^{top}|)$
  – also comparing strict hierarchical vs. factored controllers







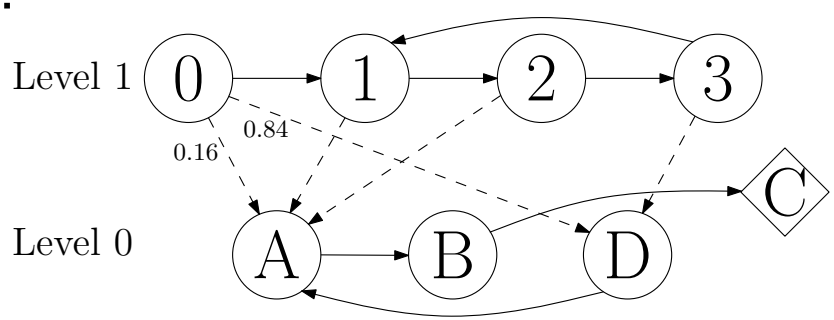


chain-of-chains

- test problem:
  - no observations
  - reward after executing n times the same chain of n actions followed by a submit action

- for $n = 3$: ABC ABC ABC D ABC ABC ABC D ABC ...

- found controller:



conclusions

- *apply inference methods for solving POMDPs*

conclusions

- *apply inference methods for solving POMDPs*
  - maximize likelihood of $\hat{R} = 1$
  - inference in mixture of variable-length DBNs
  - inference methods exploit internal structure

conclusions

- *apply inference methods for solving POMDPs*
  – maximize likelihood of $\hat{R} = 1$
  – inference in mixture of variable-length DBNs
  – inference methods exploit internal structure

- hierarchy discovery framed as DBN optimization

conclusions

- *apply inference methods for solving POMDPs*
  – maximize likelihood of $\hat{R} = 1$
  – inference in mixture of variable-length DBNs
  – inference methods exploit internal structure

- hierarchy discovery framed as DBN optimization

thanks!

  – code will be available on my webpage