

Temporal Planning under Uncertainty

Sylvie Thiébaux

National ICT Australia and The Australian National University



Australian Government
Department of Communications,
Information Technology and the Arts
Australian Research Council

NICTA Members



NICTA Partners

Planning with Time and Uncertainty

- **planning with time**

- durative actions
- timed effects
- concurrency

and

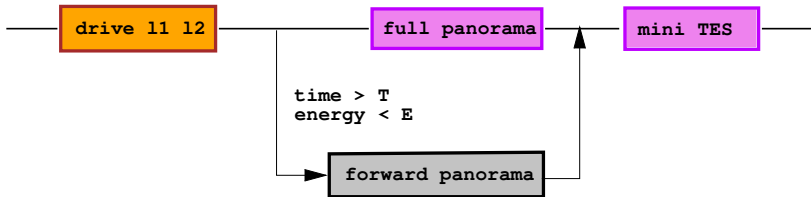
- **uncertainty**

- about the effects
- their timing
- the action duration



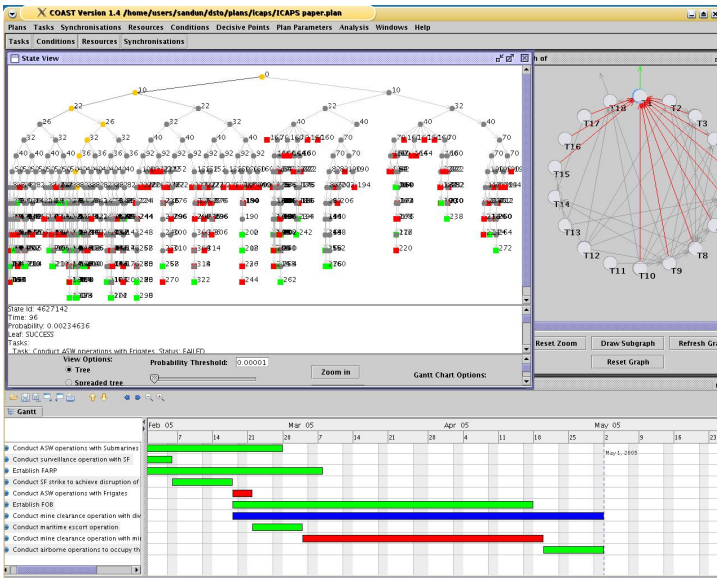
Approaches to Uncertainty

- replanning
- robust planning (conformant, conservative, temp. flexible)
- contingent planning



- benefits of contingent planning for rover applications: see [Bresina *et. al* UAI-02, Meuleau *et. al*, AAAI-04]

Contingent Temporal Plans



Probabilistic Temporal Planning

● temporal planning

- durative actions
- timed effects
- concurrency
- no uncertainty



● probabilistic planning

- probabilistic effects
- no uncertainty about time
- no time
- no concurrency



Probabilistic Temporal Planning

• temporal planning

- planning graph
- heuristic search
- constraints
- partial-order

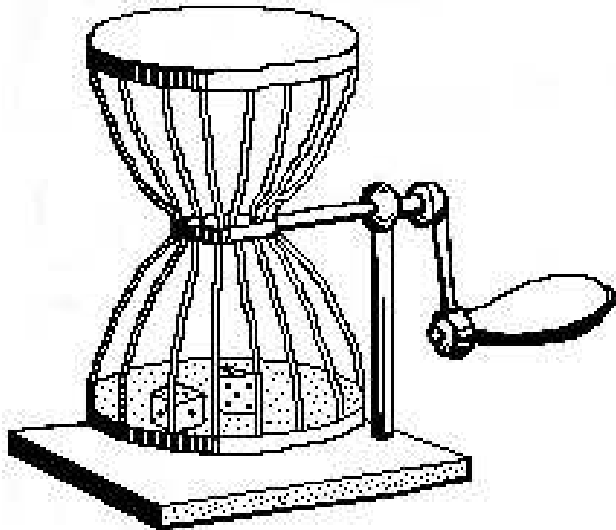


• probabilistic planning

- dynamic programming
- (and-or) heuristic search
- reinforcement learning
(Q,TD,sarsa,grad. ascent)



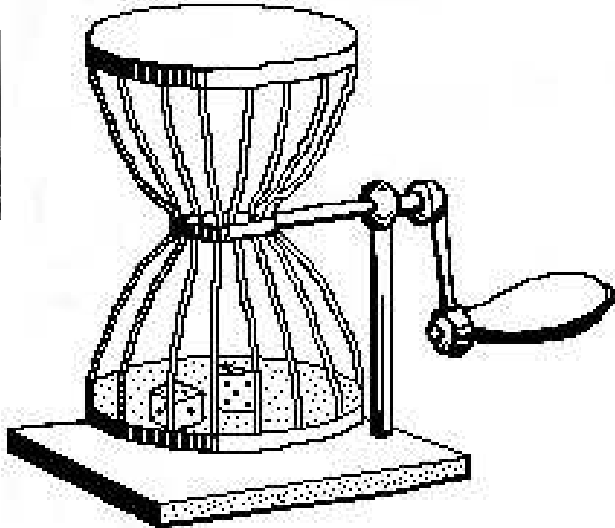
DPOLP Project



DPOLP Project



Australian Government
Department of Defence
Defence Science and
Technology Organisation



DPOLP Project & Friends

definition of the problem approaches and planners

- AI planning
- Markov decision process
- reinforcement learning

knowledge engineering

DPOLP Project & Friends

definition of the problem approaches and planners

- AI planning
- Markov decision process
 - (real-time) dynamic programming + MDP heuristics
 - MOP planner [Aberdeen *et. al.*, ICAPS-04]
 - DUR planner [Mausam & Weld, ICAPS+AAAI 2004-2006]
 - related work by Younes & Simmons
- reinforcement learning

knowledge engineering

DPOLP Project & Friends

definition of the problem approaches and planners

- AI planning
 - heuristic search
 - Prottle planner [Little *et. al*, AAI-05]
 - interesting characterisation of PTP search space
 - generalisation of planning graph heuristics to PTP
 - related work by Ames and JPL
- Markov decision process
- reinforcement learning

knowledge engineering

DPOLP Project & Friends

definition of the problem approaches and planners

- AI planning
- Markov decision process
- reinforcement learning
 - (model-free) policy gradient
 - FPG planner [Aberdeen, NIPS-05]
 - approximate, fast (1000 actions)
 - winner of the probabilistic track of the IPC-5
 - handles both discrete/continuous distributions
 - size of the policy is flexible
 - related work by Peshkin, Meuleau, *et. al*

knowledge engineering

Plan of the Talk

definition of the problem

approaches and planners

- AI planning
- Markov decision process
- reinforcement learning

knowledge engineering

PTP Problem Definition

- **pragmatic definition**
 - set of states
 - initial state
 - set of goal states
 - set of actions [next slide]
 - minimisation criterion (failure probability, expected makespan, expected resource usage, ...)
- **beautiful definition**, see [Younes, ICAPS-PDDL-03]

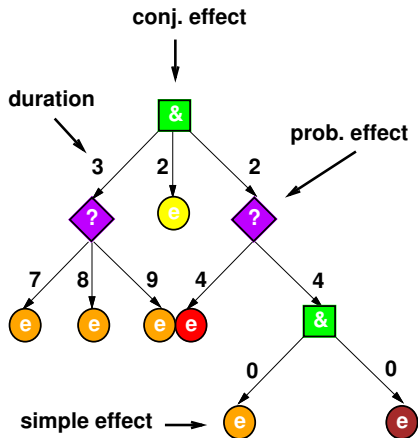
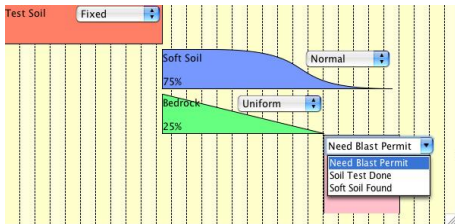
PTP Problem Definition - Actions

(:durative-action jump

:parameters (?p - person ?c - parachute)

:condition (and (at start (and (alive ?p)
(on ?p plane)
(flying plane)
(wearing ?p ?c)))
(over all (wearing ?p ?c)))

:effect (and (at start (not (on ?p plane)))
(at end (on ?p ground))
(at 5 (probabilistic
(0.8 (at 42 (standing ?p)))
(0.2 (at 13 (probabilistic
(0.1 (at 14 (bruised ?p)))
(0.9 (at 14 (not (alive ?p))))))))))))))



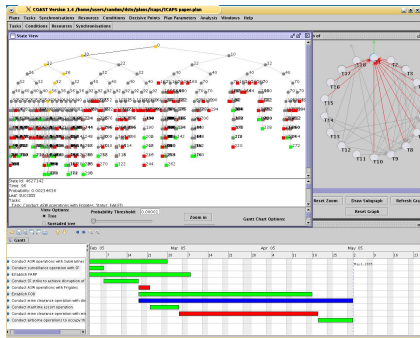
PTP Problem Definition - Plans

- **probabilistic temporal plans**

- set of triples (time,state,action)
- semantics: probability distribution on temporal plans

- **when is a plan “valid”?**

- no commonly adopted definition
- restrictive: all temporal plans are valid (DUR)
- permissive: at least one temp. plan is valid (Prottele, FPG)
- ties in with algorithm and optimisation criterion chosen



Plan of the Talk

definition of the problem

approaches and planners

- AI planning
- Markov decision process
- reinforcement learning

knowledge engineering

Prottle: PTP via Heuristic Search

- **described in** [Little *et. al*, AAAI-05]
- **parents**
 - temporal:
 - decision-epoch planners (Sapa, TLPlan)
 - planning graph heuristics (TGP)
 - probabilistic:
 - and-or heuristic search (LAO*, LRTDP)
 - planning graph heuristics (PGraphplan)
- **plan**
 - search space
 - search algorithm
 - heuristics

Prottle: Search Space

• and-or graph

- and: chance
- or: choice

• node purposes

- selection
- advancement

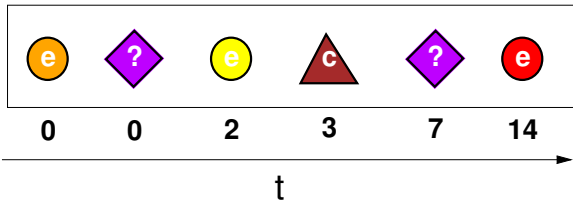
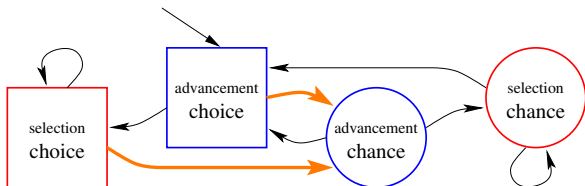
• node contains

- current state
- (current time)
- event queue

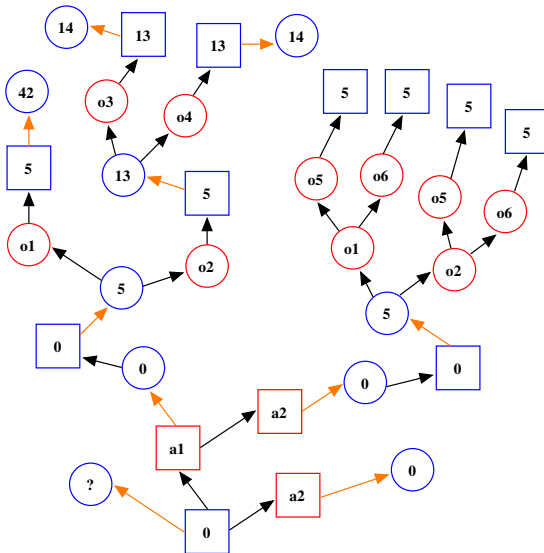
• queue contains

- pairs (time, item)
- effect items
- condition checks

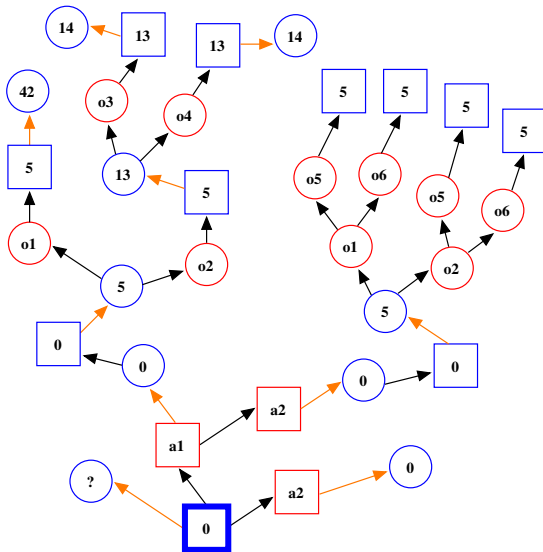
decision epoch planning in an and-or graph



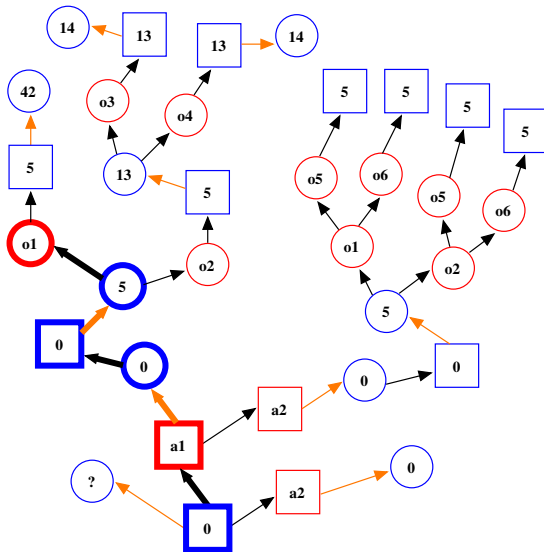
Prottle: Search Space



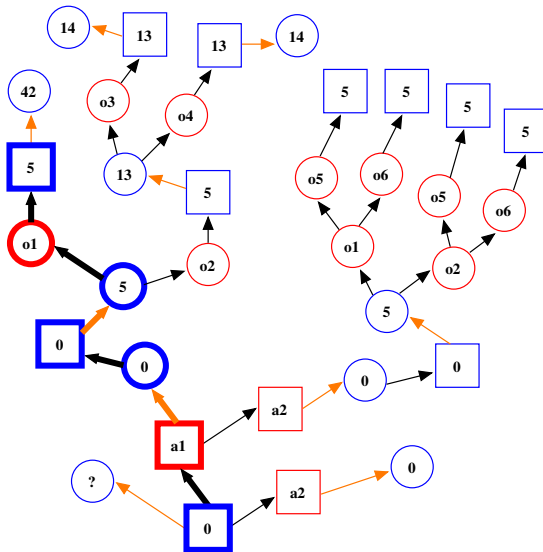
Prottle: Search Space



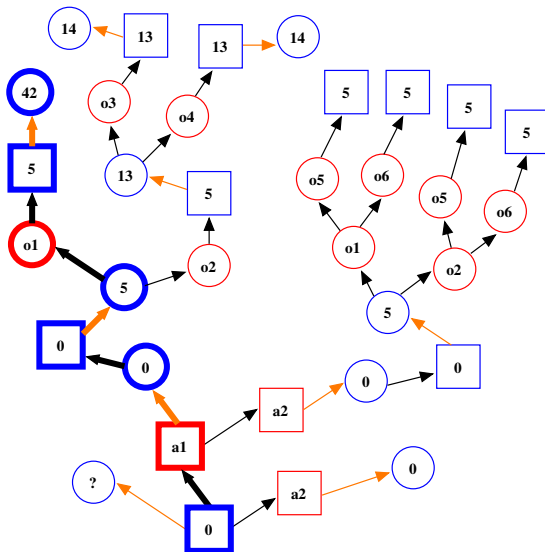
Prottle: Search Space



Prottle: Search Space



Prottle: Search Space



Prottle: Search Algorithm

- **node lower/upper cost bounds**

- cost = probability of failure
- bounds initialised using heuristics
- bounds update rules:

$$L_{\text{choice}}(n) := \max(L(n), \min_{n' \in S(n)} L(n'))$$

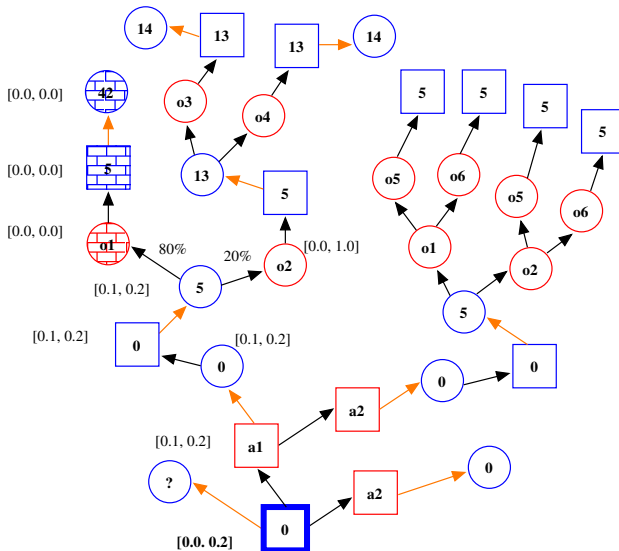
$$U_{\text{choice}}(n) := \min(U(n), \min_{n' \in S(n)} U(n'))$$

$$L_{\text{chance}}(n) := \max(L(n), \sum_{n' \in S(n)} \Pr(n') L(n'))$$

$$U_{\text{chance}}(n) := \min(U(n), \sum_{n' \in S(n)} \Pr(n') U(n'))$$

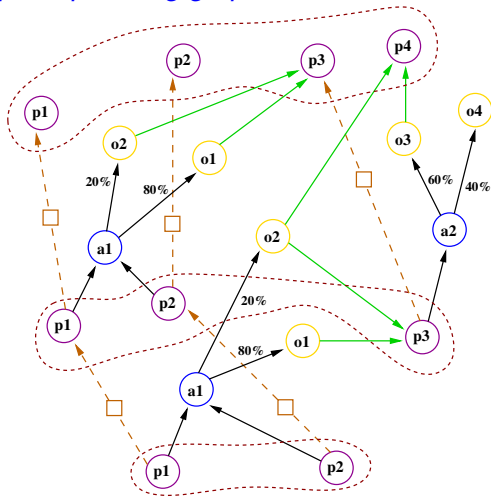
- cost converges when $U(n) - L(n) \leq \epsilon$
- **node labels:** solved, failure (solved with cost 1), unsolved

Prottle: Search Algorithm



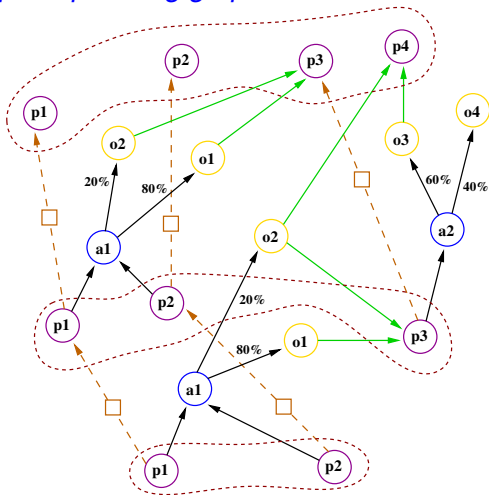
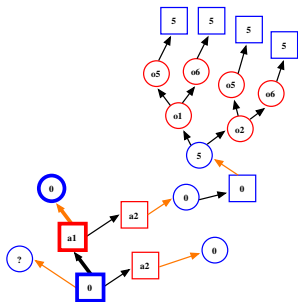
Prottle: Heuristics

probabilistic temporal planning graph



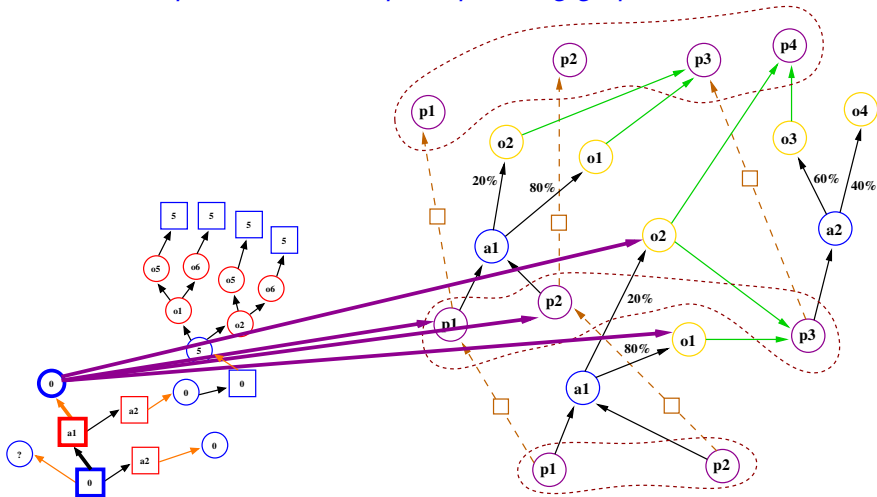
Prottle: Heuristics

probabilistic temporal planning graph



Prottle: Heuristics

probabilistic temporal planning graph



Plan of the Talk

definition of the problem approaches and planners

- AI planning
- Markov decision process
- reinforcement learning

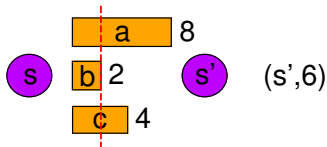
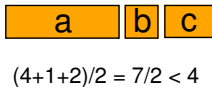
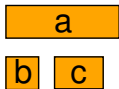
knowledge engineering

DUR: CoMDP Algorithms for PTP

- **described in** [Mausam & Weld, AAI-04, [ICAPS-05](#), AAI-06]
- **assumptions**
 - PPDDL_e^o actions with deterministic durations
 - mutex relation: generalisation of independence
 - minimise expected makespan of proper policy
- **stochastic shortest path problem**
 - $\langle S, s_0, G, A, \Pr(s' | a, s), c \rangle$
 - processing of queue events and actions in one step
- **real-time dynamic programming (RTDP)**
 - lower bound (makespan)
 - SSPP relaxation heuristics
 - sample state space according to the “greedy” policy

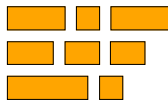
DUR: Heuristics

- **maximum concurrency heuristic**
 - serial SSPP cost/max nb. of parallel actions at any point
 - non-admissible average concurrency heuristic
 - **eager effects heuristic**
 - effects are realised when the fastest executing action ends
 - time advances accordingly
 - SSPP state = (world state after effects, latest end time of any executing action)
- 1 more information in the relaxed problem
 - 2 mutex action combinations are allowed (lost track of time)

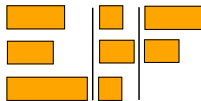


DUR: Hybrid Algorithm

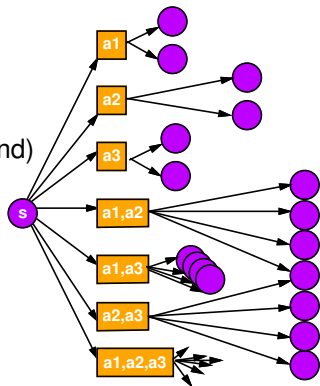
- **exponential blowup in action space**
- **hybridisation**
 - run RTDP for a number of trials (lower bound)
 - run RTDP with aligned epochs on low frequency states (upper bound)
 - repeat until performance ratio reached



interwoven epochs



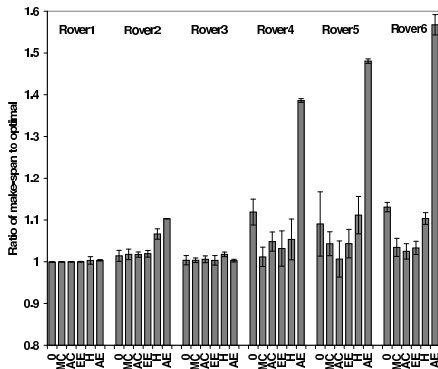
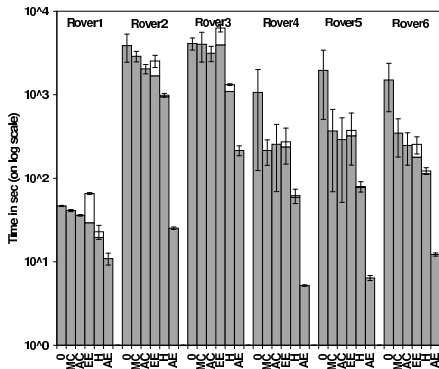
aligned epochs



DUR: Results

- **probabilistic version of the Rovers benchmark**

- 17-21 variables, 12-18 actions, durations in 1...20
- 15K-700K reachable SSPP states
- RTDP with 0, MC, AC, EE heuristics, hybrid, aligned epoch



Plan of the Talk

definition of the problem approaches and planners

- AI planning
- Markov decision process
- reinforcement learning

knowledge engineering

FPG: Reinforcement Learning for PTP

- produce good policies in real domains
- **described in** [Aberdeen, NIPS-05]
- **parents**
 - temporal: decision epoch planners
 - probabilistic: policy gradient (GPOMDP)
- **plan**
 - reinforcement learning
 - policy gradient algorithms
 - PTP as a factored RL problem
 - comparative experimental results



Reinforcement Learning

● reinforcement learning

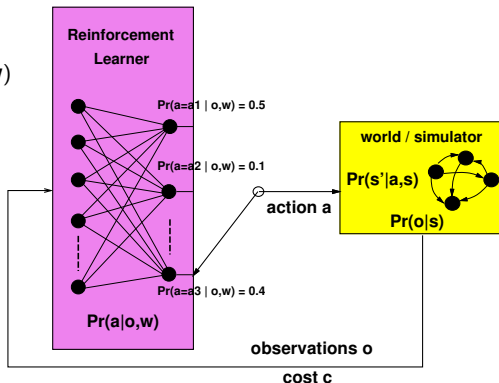
- partially observable MDPs
- model-free: $\Pr(s' | a, s)$ and $\Pr(o | s)$ are unknown
- learn a policy from observations & costs

● direct policy search

- parametrised policy $\Pr(a | o, w)$
- no value function
- flexible memory reqs

● policy gradient

- $$C(\mathbf{w}) = \lim_{T \rightarrow \infty} \frac{1}{T} E_{\mathbf{w}}[\sum_{t=0}^T c(s_t)]$$
- gradient descent (wrt \mathbf{w})
- reaches a local optimum
- continuous/discrete spaces



FPG: Policy Gradient Algorithm

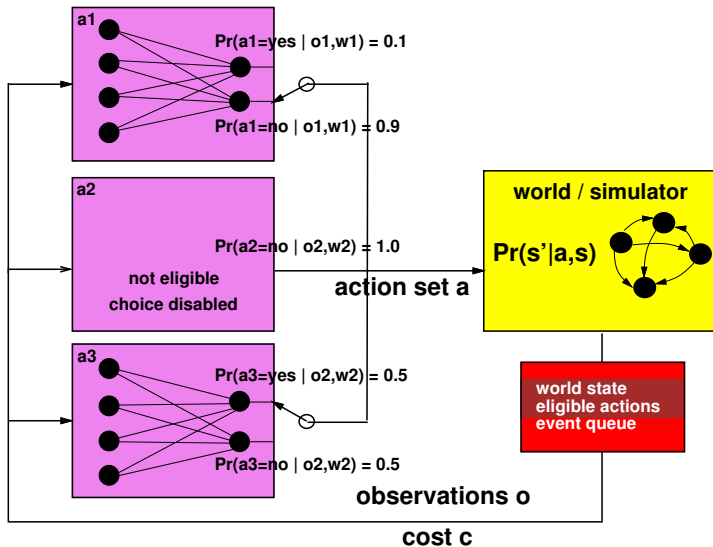
- **policy gradient**

- $C(\mathbf{w}) = \lim_{T \rightarrow \infty} \frac{1}{T} E_{\mathbf{w}} \left[\sum_{t=0}^T c(s_t) \right]$ (failure prob., makespan)
- minimise C by
 - 1 computing its gradient: $\nabla C(\mathbf{w}) = \left[\frac{\partial C}{\partial w_1}, \dots, \frac{\partial C}{\partial w_k} \right]$, and
 - 2 stepping the parameters away: $\mathbf{w}_{t+1} = \mathbf{w}_t - \alpha \nabla C(\mathbf{w})$
until convergence

- **gradient estimate** [Sutton *et. al*, NIPS-99, Baxter *et. al*, JAIR-01]

- Monte Carlo estimate from trace $\mathbf{o}_1, \mathbf{a}_1, \mathbf{c}_1, \dots, \mathbf{o}_T, \mathbf{a}_T, \mathbf{c}_T$:
 - 1 $\mathbf{e}_{t+1} = \beta \mathbf{e}_t + \nabla_{\mathbf{w}} \log \Pr(\mathbf{a}_{t+1} | \mathbf{o}_{t+1}, \mathbf{w}_{t+1})$
 - 2 $\mathbf{w}_{t+1} = \mathbf{w}_t - \alpha \mathbf{c}_t \mathbf{e}_t$

FPG: PTP as a Factored RL Problem



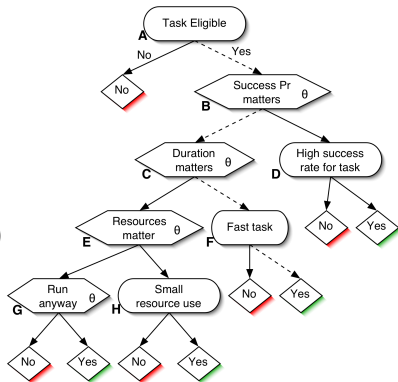
FPG: Individual Action Policies

- linear network policies

$$\Pr(a_{it} = \text{yes} \mid \mathbf{o}_t, \mathbf{w}_i) = \frac{1}{\exp(\mathbf{o}_t^\top \mathbf{w}_i) + 1}$$

$$\Pr(a_{it} = \text{no} \mid \mathbf{o}_t, \mathbf{w}_i) = 1 - \Pr(a_{it} = \text{yes} \mid \mathbf{o}_t, \mathbf{w}_i)$$

- decision tree policies



FPG: Comparative Results

<i>Prob.</i>	<i>Plan.</i>	<i>Fail%</i>	<i>MS</i>	<i>Res</i>	<i>R</i>	<i>Time</i>	<i>Features</i>
MZ	FPG-L	14.7	5.6		100.0	13.0	medium, easy problem 165 actions 207 facts
MZ	FPG-T	15.8	7.0		102.0	2.0	
MZ	Prottle	17.8				10.0	
MZ	MOP	7.2	8.2			72.0	
MZ	Random	76.5	13.0		16.4		
MZ	Naive	90.8	16.0		8.6		
MS	FPG-L	0.1	6.5		89.0	32.0	small, harder problem 28 actions 38 facts
MS	FPG-T	65.9	14.0		16.0	37.0	
MS	Prottle	2.9				272.0	
MS	MOP		out of memory				
MS	Random	93.3	18.0		0.1		
MS	Naive	100.0	20.0		0.0		
R500	FPG-T-16	2.5	158.0	4276	1.56	3345.0	large problem 500 actions, 200 facts 40 res. types, 200 units
R500	Random	76.6	765.0	6194	0.23		
R500	Naive	69.5	736.0	6359	0.10		

Plan of the Talk

definition of the problem approaches and planners

- AI planning
- Markov decision process
- reinforcement learning

knowledge engineering

Brazil: Knowledge Engineering for PTP

- **model PTP actions**

- probabilistic effects/durations
- continuous duration distributions

- **view contingency plan**

- output most likely plan trajectory
- let user perturbate durations/resources
- let user explore contingencies
- update Gantt chart (real-time with FPG)
- statistical view



Brazil: Knowledge Engineering for PTP

DEMO

Conclusion

- **approaches to uncertainty**
 - replanning (no high cost failure)
 - conformant planning (no high cost missed opportunities)
 - contingent planning
- **PTP is a relatively new area**
- **range of contingent planning methods**
 - Prottle or DUR (small-medium problems, a lot of memory)
 - FPG (medium-large problems, no optimality guarantees)

Related & Future Work

● related work

- incremental contingency plans [Dearden *et. al*, ICAPS-03]
- reasoning about robustness [Schaeffer *et. al*, IJCAI-05]
- dynamic programming in the plan graph [Meuleau *et. al*, AAAI-04]
- learning to cooperate via policy search [Peshkin *et. al*, UAI-00]
- planning with GSMDPs [Younes *et. al*, ICAPS-04, AAAI-04]
- coordinators [Musliner *et. al*, 06]

● future work

- trial FPG on significant problems
- constraint-based methods, probabilistic counterpart of LPGP [Fox & Long, ICAPS-03] based on Paragraph [Little & Thiébaux, ICAPS-06]
- handle continuous time and resources
- integrating replanning, conformant, and contingent planning

Finally

- **thanks to**

- Australian Defense Science and Technology Organisation
- National ICT Australia
- DPOLP@NICTA: D. Aberdeen, O. Buffet, A. Gabaldon
P. Haslum, I. Little, J. Rintanen, O. Thomas
- Mausam, David Smith

- **job ad**

- 2 positions: researcher or senior researcher level
- area: reinforcement learning, statistical ML, planning
- where: NICTA Canberra Laboratory
- salary range: AUD 70K-130K
- **contact** `douglas.aberdeen@nicta.com.au`