On the Impact of Prior Knowledge on Autonomous Agents

Benjamin Rosman

Mobile Intelligent Autonomous Systems
Council for Scientific and Industrial Research
&
School of Computer Science and Applied Maths
University of the Witwatersrand
South Africa

8 September 2015

our future through science

Long-Lived Agents

- Agents deployed in some environment over a long duration
  - Multiple tasks
  - Changing environment
- Continuously learn and adapt
  - Growing task, behaviour sets
- How to maintain knowledge?
  - Behaviour transfer
  - Generalisation

Transfer Learning

1. How can an agent **generalise from previous behaviours** to solve new tasks in the same environment quicker and with less risk?
    1. Accelerate policy learning
    2. Model of external agent behaviour

2. Given a set of previously learnt behaviours, what is the optimal way to **select the best one to be re-used** in a new environment or interaction?

CSIR
our future through science

Chapter 0:
A Brief Intro to Reinforcement Learning

our future through science

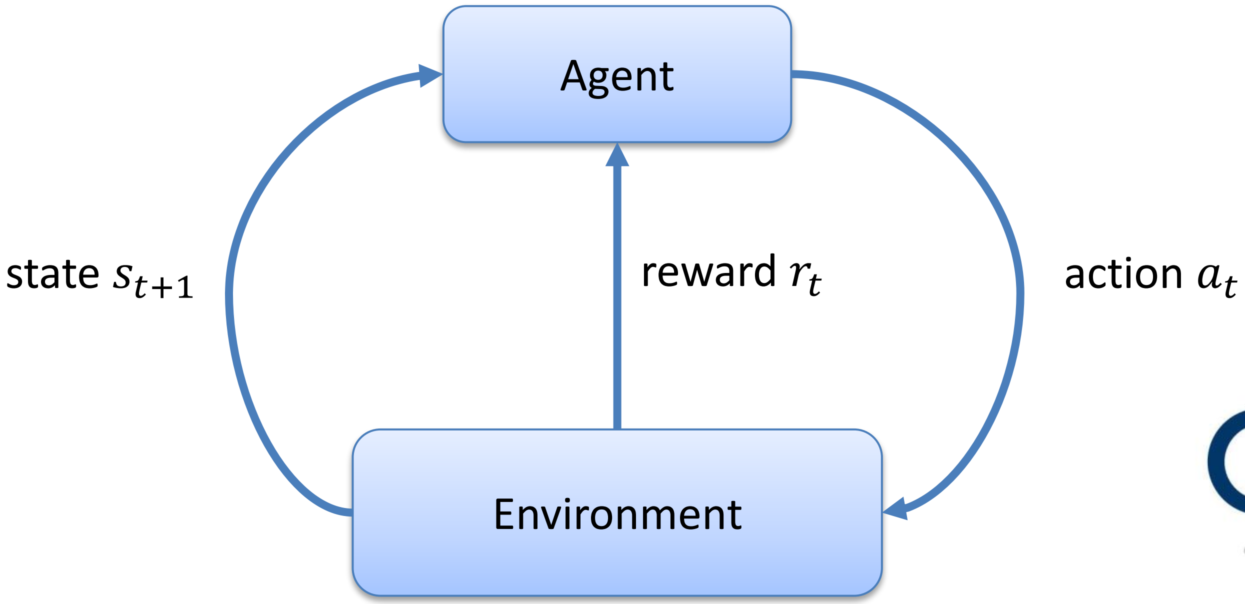
What is reinforcement learning?

- How to learn behaviours under stochasticity and uncertainty?
  - Unsupervised?
  - Supervised?
  - Something else entirely…

our future through science

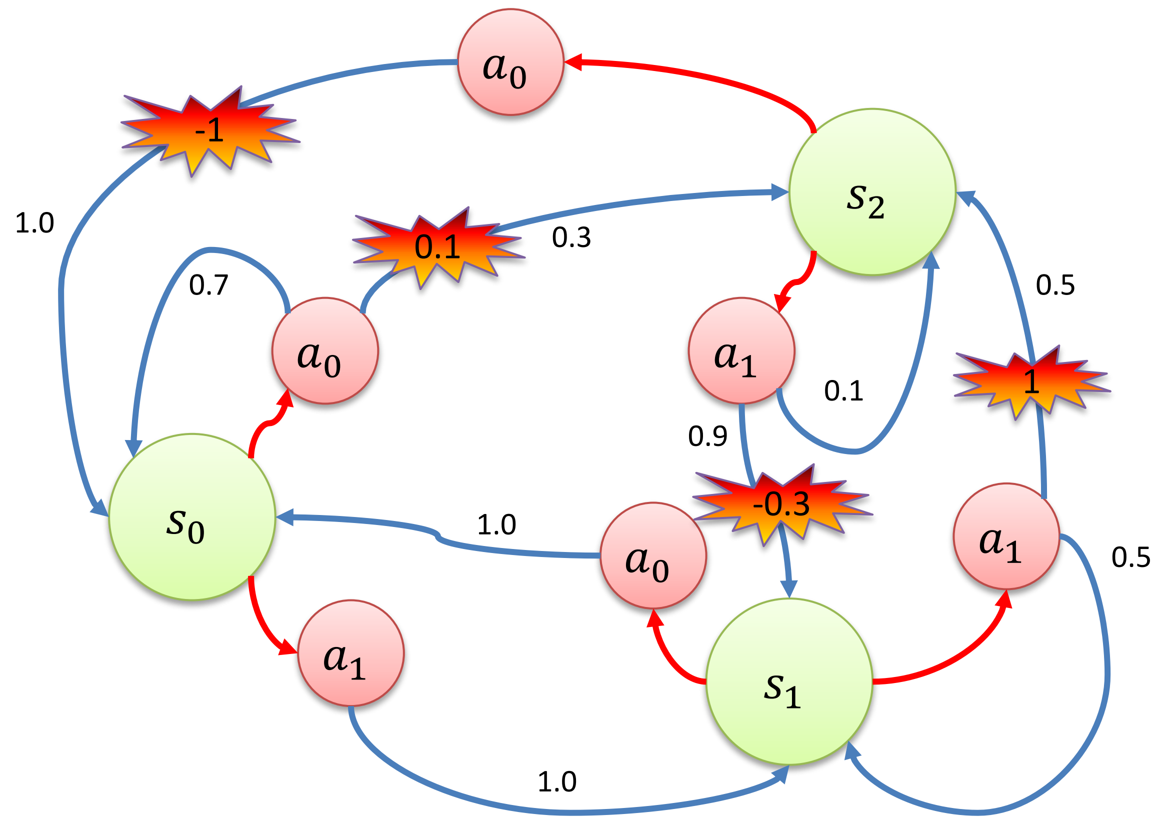
Operating in an environment

- Rewards as a weak, delayed learning signal
  - Goal-directed learning
- Learn from repeated interaction
- Learn to **map situations to actions** so as to **maximise numerical reward** (which may be delayed)

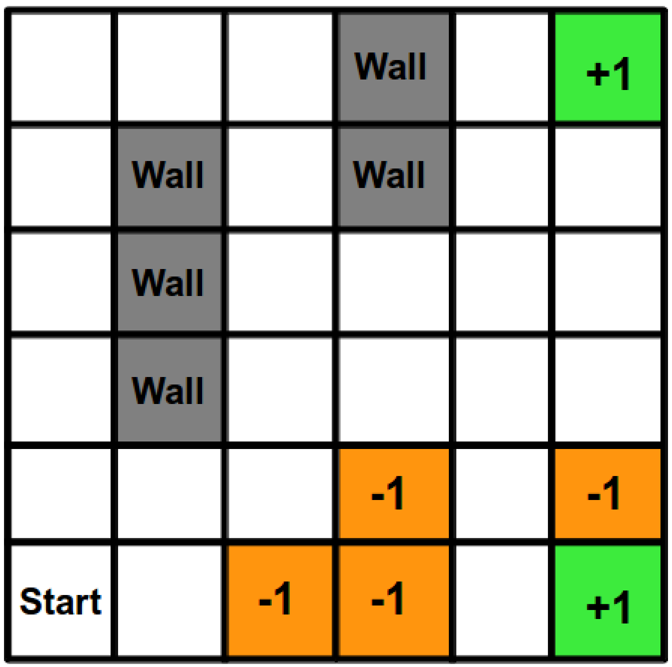


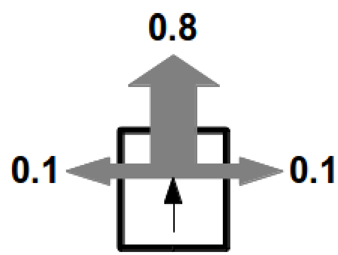
Markov Decision Processes (MDPs)

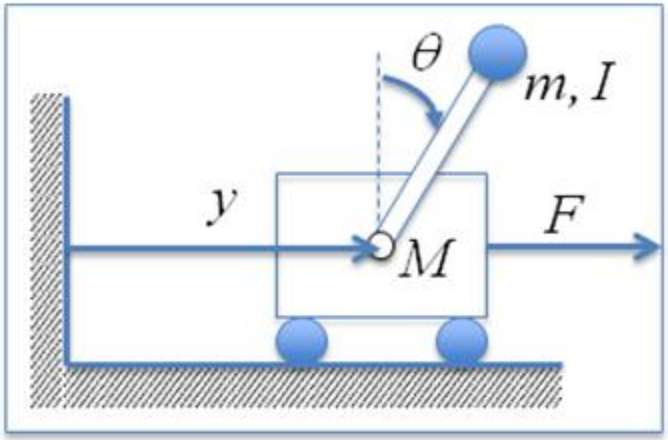
- Model a decision problem
- $M = \langle S, A, T, R, \gamma \rangle$
- Observable
- Markov
- Policy π

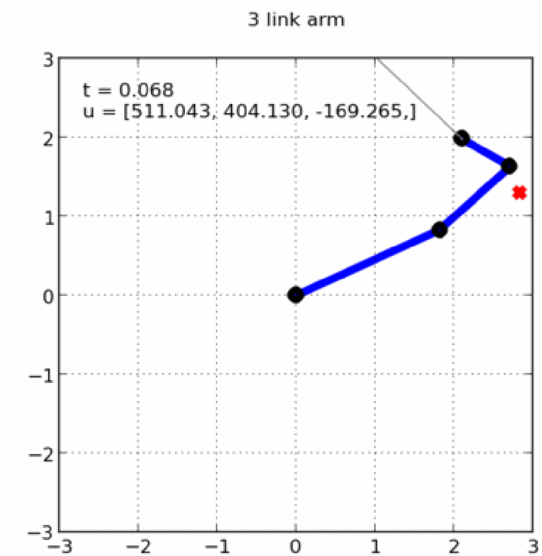


Examples


Wall
+1
Wall
Wall
Wall
Wall
-1
-1
Start
-1
-1
+1


0.8
0.1
0.1

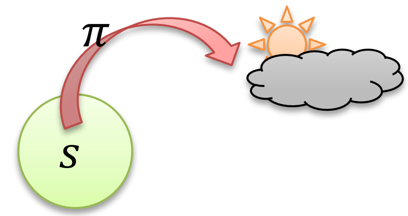

θ
m, I
y
F
M


3 link arm
t = 0.068
u = [511.043, 404.130, -169.265,]

Value functions

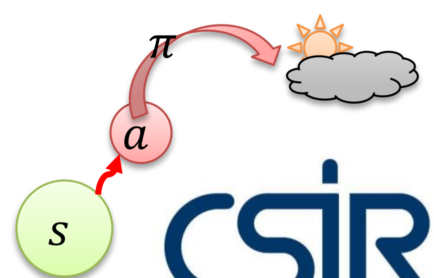
- Value of a state:
  - Expected return starting from that state and following a particular policy
  - $V^{\pi}(s) = E_{\pi}\{R_t | s_t = s\}$

$$= E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s\right\}$$



- Value of an action in a state:
  - Expected return of starting in that state, taking that action, and then following a particular policy
  - $Q^{\pi}(s,a) = E_{\pi}\{R_t | s_t = s, a_t = a\}$

$$= E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s, a_t = a\right\}$$



CSIR
our future through science

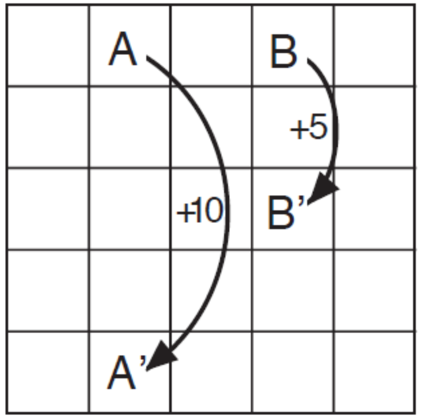
Why value functions?

- Optimal value functions:
  - $V^*(s) = \max_\pi V^\pi(s)$
  - $Q^*(s,a) = \max_\pi Q^\pi(s,a)$
  - These are the value functions given by the optimal policy π^*
- Any policy that is greedy w.r.t V^* (or Q^*) is optimal
  - So, $\pi^*(s) = \arg\max_{a \in A} Q^*(s,a)$

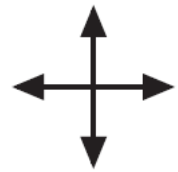
our future through science

Example

- Random policy:



(a)

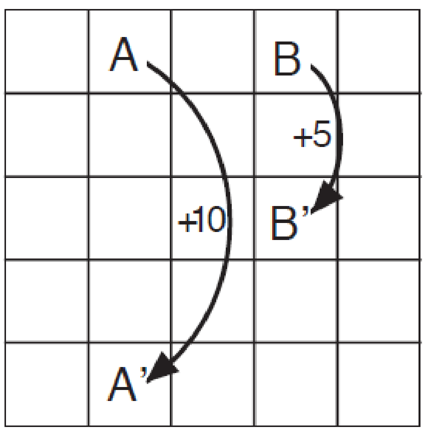


Actions

3.3	8.8	4.4	5.3	1.5
1.5	3.0	2.3	1.9	0.5
0.1	0.7	0.7	0.4	-0.4
-1.0	-0.4	-0.4	-0.6	-1.2
-1.9	-1.3	-1.2	-1.4	-2.0

(b)

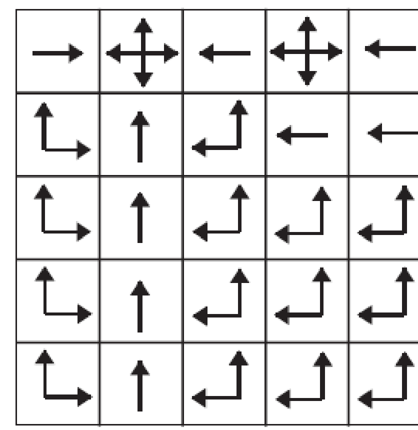
- Optimal:



a) gridworld

22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

b) v_*



c) π_*

RL Algorithms

- RL learning is **trial-and-error learning** to find a good policy from experience
- So as not to solve a large system of value function equations

$$
\begin{aligned}
V^{\pi'}(s) &= \max_a E\left\{r_{t+1} + \gamma V^{\pi'}(s_{t+1}) \mid s_t = s, a_t = a\right\} \\
&= \max_a \sum_{s'} \mathcal{P}^a_{ss'} \left[\mathcal{R}^a_{ss'} + \gamma V^{\pi'}(s')\right].
\end{aligned}
$$

  - Which aren't even known!
- Exploration vs exploitation
- Model free vs model based algorithms

CSIR
our future through science

Q-Learning

- Initialise $Q(s,a)$ arbitrarily
- Repeat (for each episode):
  - Initialise s
  - Repeat (for each step of episode):
    - Choose a from s using ϵ-greedy policy from Q
      - $a \leftarrow \begin{cases} \arg\max_a Q(s,a) & w.p.\,\epsilon \\ random & w.p.\,1-\epsilon \end{cases}$ exploit explore
    - Take action a, observe r, s'
    - Update Q
      - $Q(s,a) \leftarrow Q(s,a) + \alpha\left[r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right]$ learn
    - $s \leftarrow s'$
  - Until s is terminal

our future through science

Chapter 1:
Safe Behaviour Generalisation
(Action Priors)

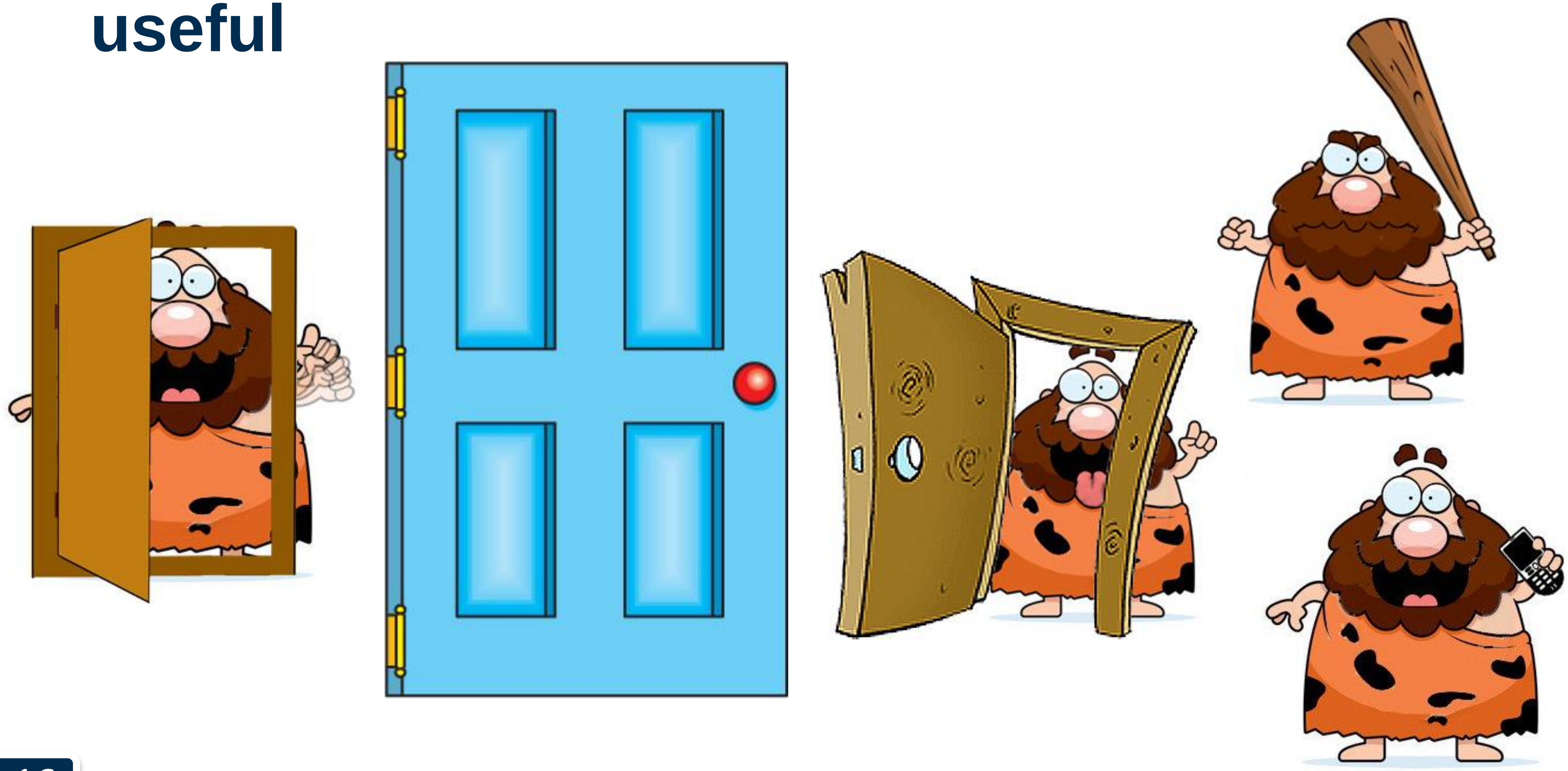
our future through science

Learning Domain Knowledge

- Agent performing multiple tasks in the same environment
  – Improve over time, across tasks
- Lifelong learning: what to learn over an agent's lifetime?
  – Task independent regularities (structure) in the domain
  – Structure: general "common sense" behaviours

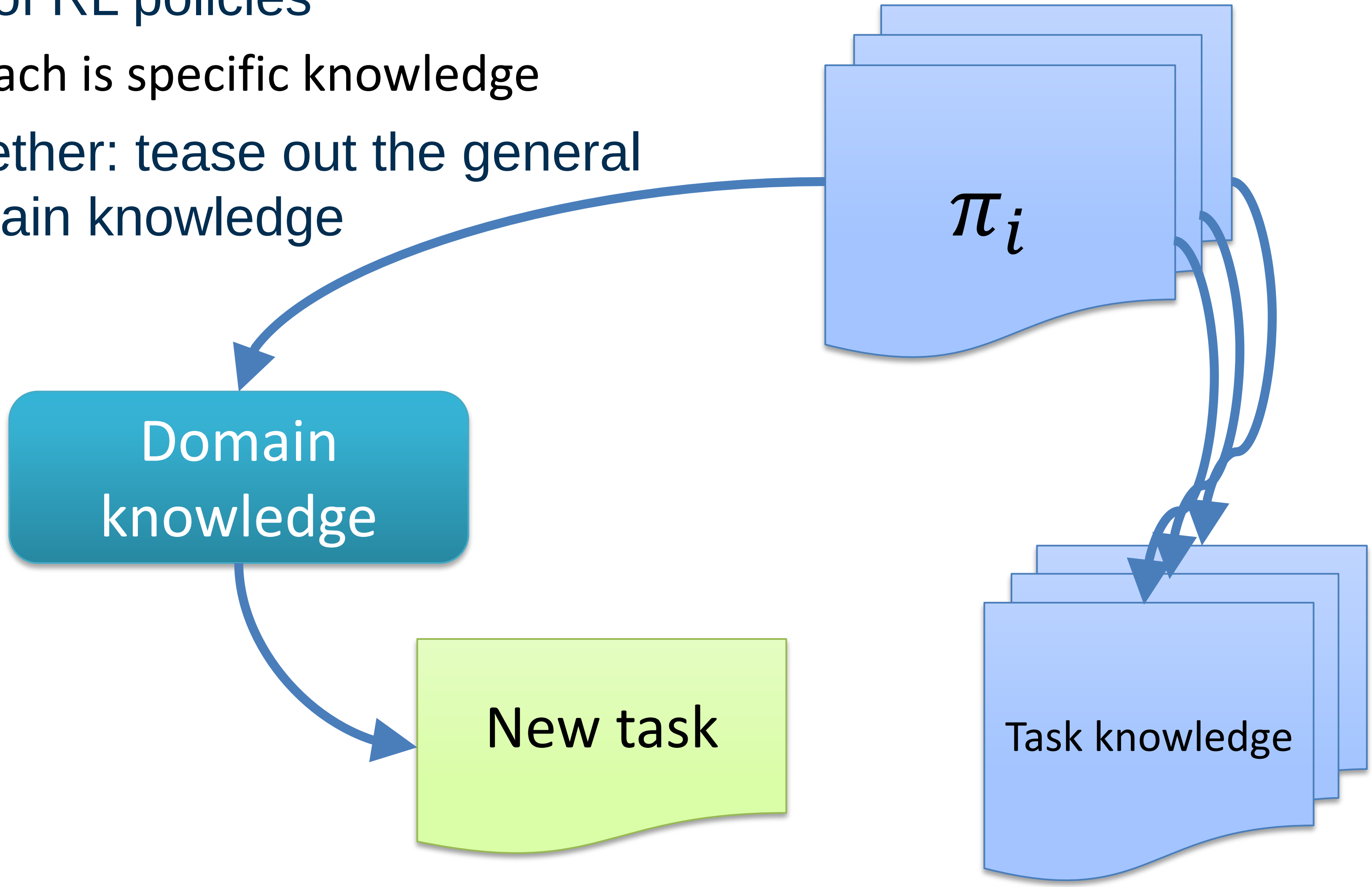
An Intuition

- Although many actions may be possible in some context, **only a small number are typically useful**



The Benefits of Multiple Tasks

- Set of RL policies
  - Each is specific knowledge
- Together: tease out the general domain knowledge

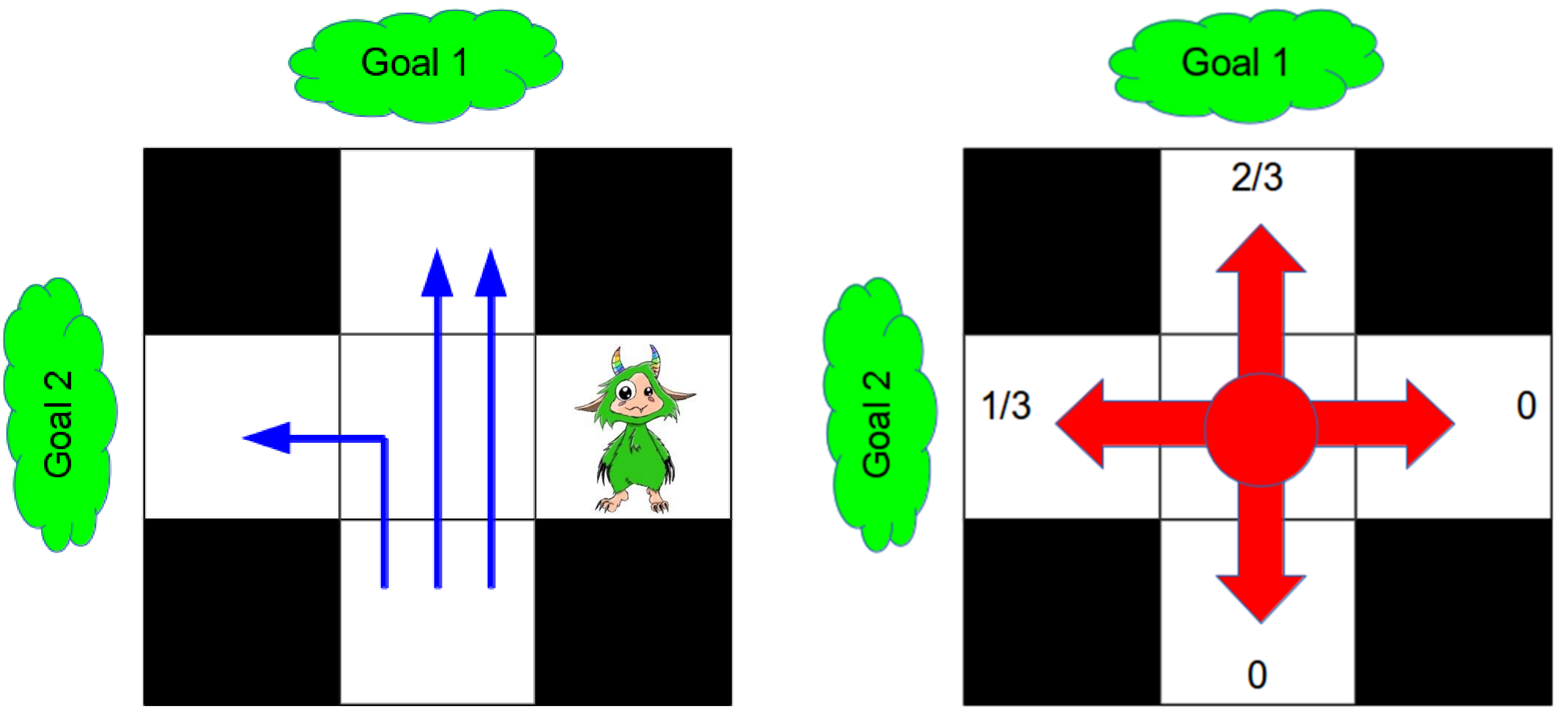


Learning Domain Knowledge

- Tasks are drawn from a domain
  - Differ in goal: reward R
  - (In general: states S, transitions T)
- Learn model of behavioural invariances across domain
  - Task independent
  - From optimal policies
- Model: Context based distributions over action set
  - Action “usefulness” = reasonable behaviour choices
  - Condition on state (observations $\varphi(s)$)

our future through science

An Illustration



(a) (b)

A Model of Domain Knowledge

- Action priors $\theta_{\varphi(s)}(A)$ [Rosman and Ramamoorthy, 2012, 2015]
  - Dirichlet distribution over A
  - Conditioned on $\varphi(s)$

- $\theta_{\varphi(s)}(A) \sim Dir(\alpha_{\varphi(s)}(A))$
  - Parameters $\alpha_{\varphi(s)}(A)$

our future through science

A Model of Domain Knowledge

- Notion of “action usefulness”
- Formally:
  - For each policy π, define a weight $w(\pi)$ Measure of confidence/skill
  - Action utility under a policy:
  - $U^{\pi}_{\varphi(s)}(a) = \delta(\pi(\varphi(s), a), \max_{a' \in A} \pi(\varphi(s), a'))$

    Utility = 1 iff action optimal
  - Action utility under a policy set:
  - $\alpha_{\varphi(s)}(a) = \sum_{\pi \in \Pi} w(\pi)\, U^{\pi}_{\varphi(s)}(a) + \alpha^{0}_{\varphi(s)}(a)$

    Weighted sum of action utilities

    Hyperprior

our future through science

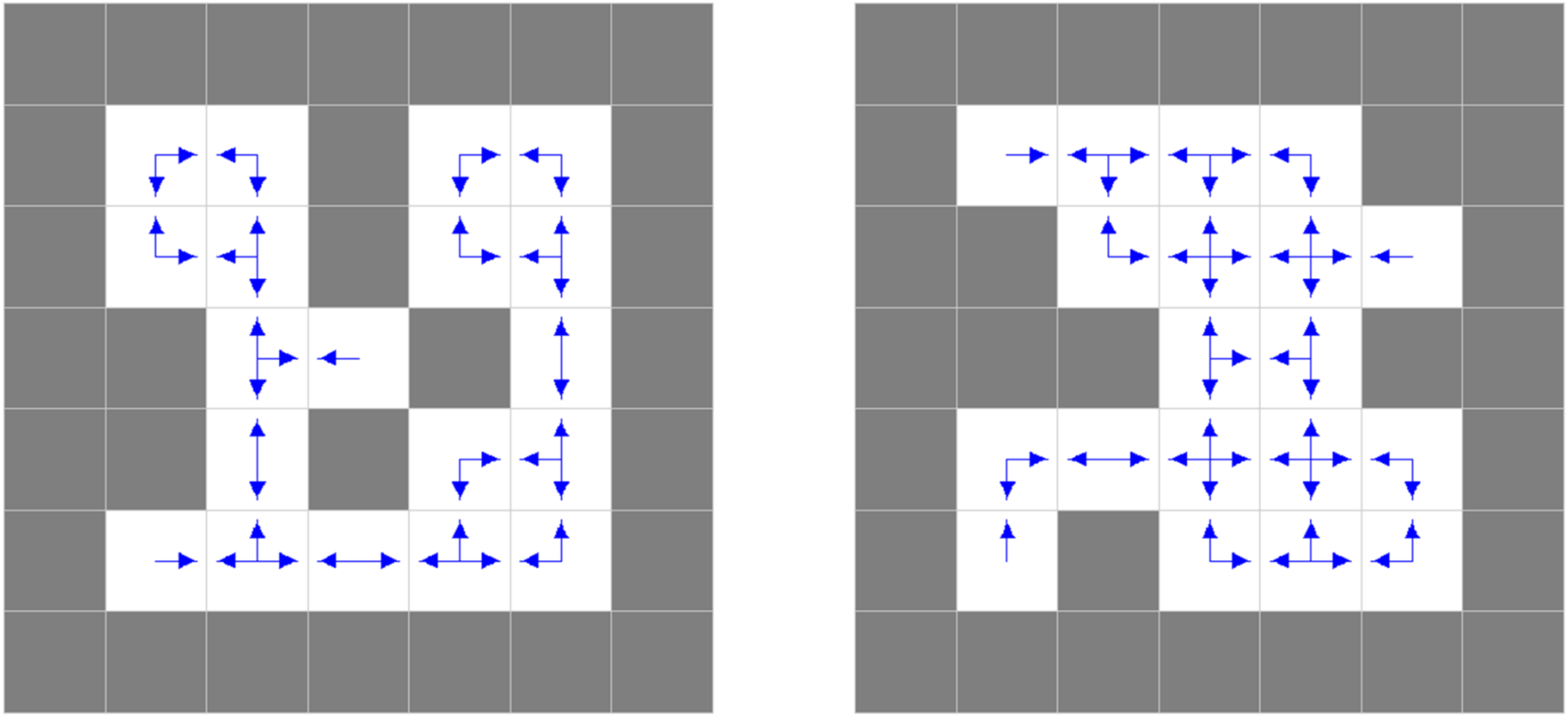
A Model of Domain Knowledge

- Online update:
  - Counts for each $\varphi(s)$
  - For each policy π, define a weight w(π) Measure of confidence/skill
  - $\alpha_\varphi^0(a) \leftarrow \alpha_\varphi(a), \;\; \forall \varphi, a$ Initialise to some hyperprior
  - $\alpha_{\varphi(s)}^{t+1}(a) \leftarrow \begin{cases} \alpha_{\varphi(s)}^t(a) + w(\pi^t), & \pi^t(s,a) = \max_{a'} \pi^t(s,a') \\ \alpha_{\varphi(s)}^t(a), & otherwise \end{cases}$

Given a new policy π^t, update counts of optimal actions

our future through science

Example Priors



How to Use? Guided Exploration

- Action selection:
  - $\theta_{\varphi(s)}(A) \sim Dir\left(\alpha_{\varphi(s)}\right)$ Draw distributions from a Dirichlet
  - $a \sim \theta_{\varphi(s)}(A)$

- Exploration in Q-learning (a twist on ϵ-greedy):
  - $a \leftarrow \begin{cases} \arg\max_{a} Q(s,a), & w.p.\ \ 1-\epsilon \\ a \in A, & w.p.\ \ \epsilon\theta_{\varphi(s)}(a) \end{cases}$

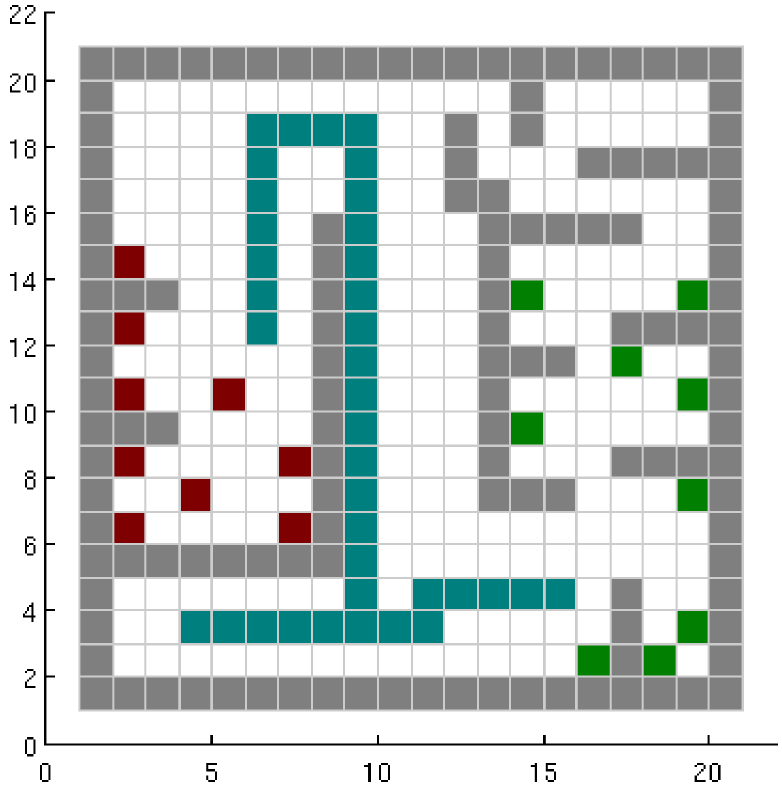
Let action prior bias exploration

Note: standard Q-learning uses uniform priors

our future through science

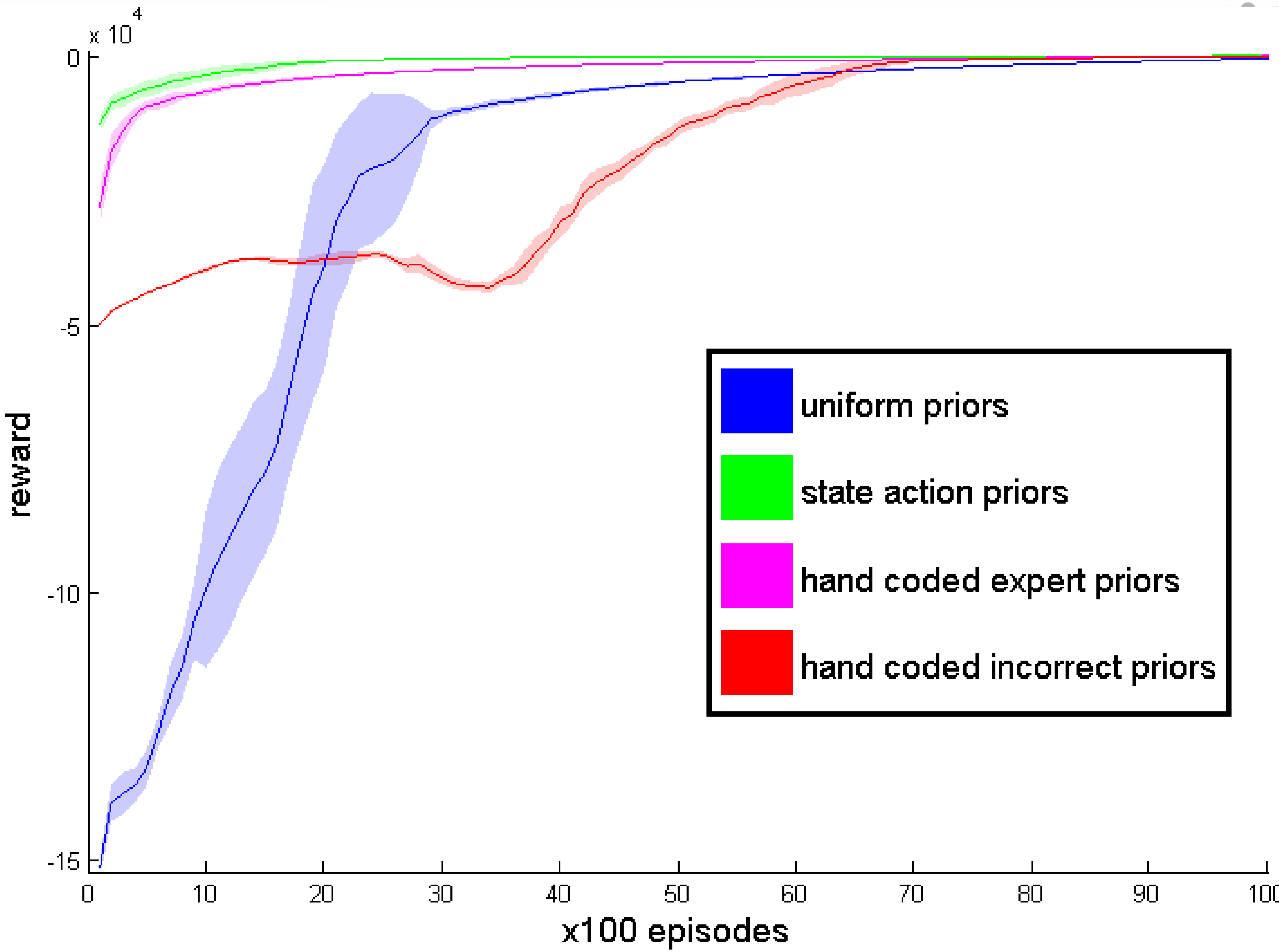
Example: The Factory Domain

- The factory domain
  - Extended navigation domain
  - Task: procure and assemble a list of items
  - Assembly/procurement points, express route
  - Actions: $N, E, S, W, Procure, Assemble$



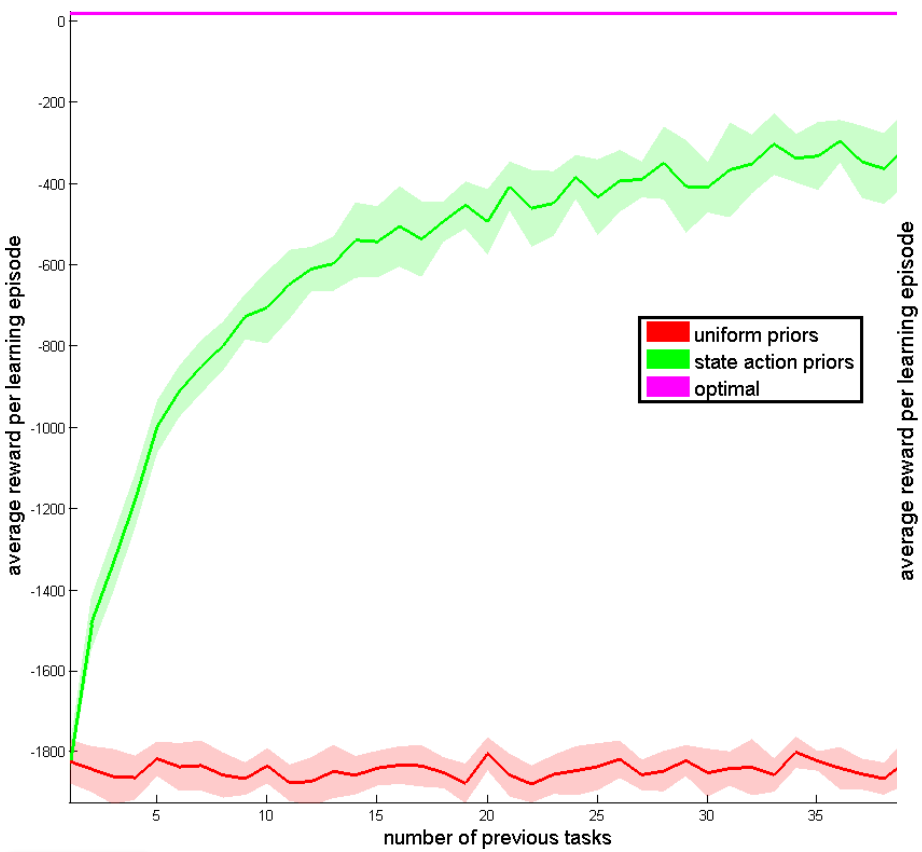
our future through science

The Effect of Priors

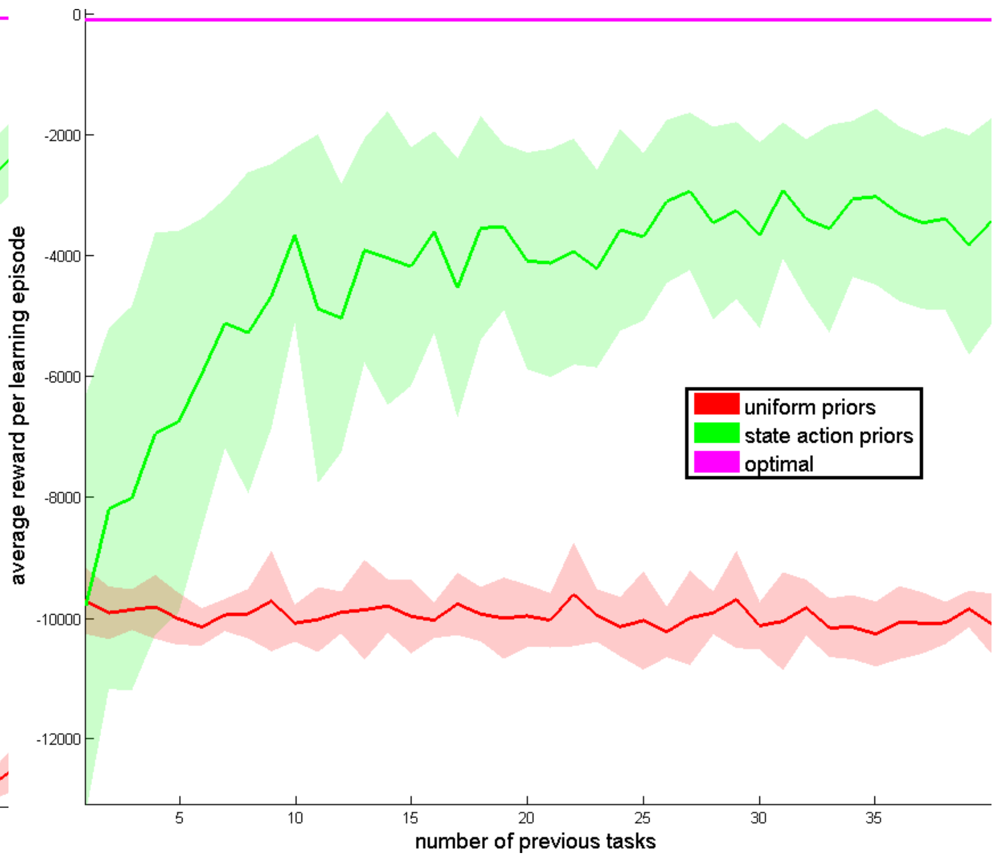


Learning Across Multiple Tasks

- Assemble 1 item

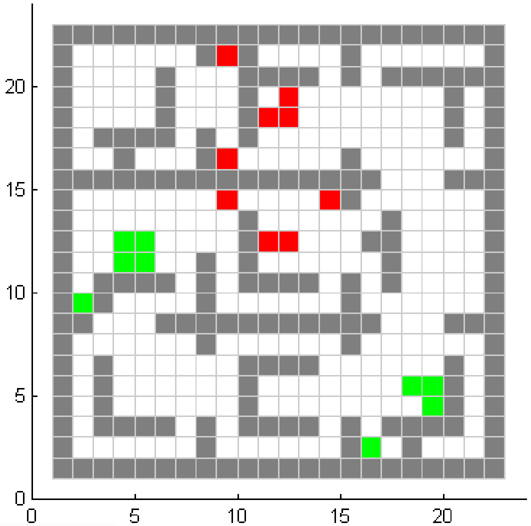


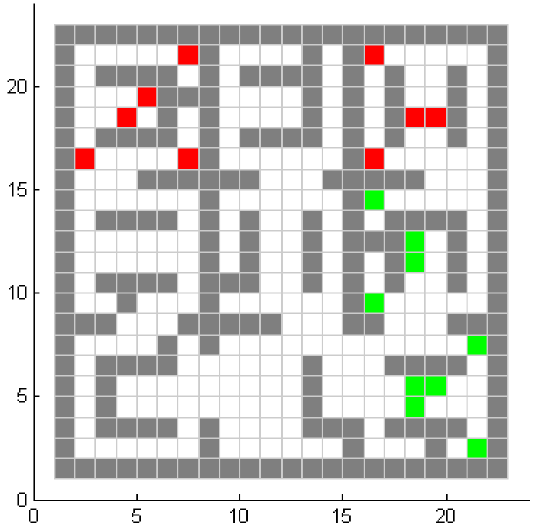
- Assemble 4 items

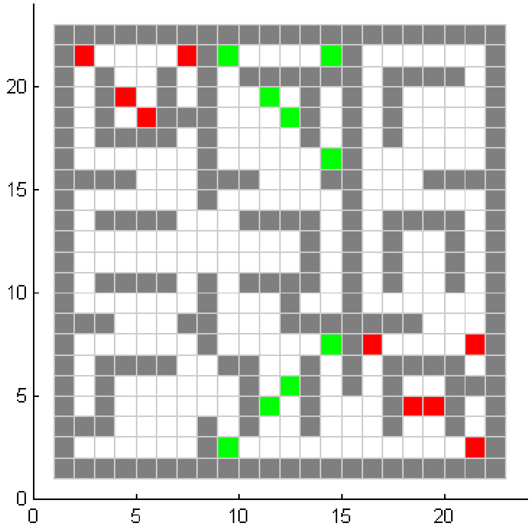


The Factory Domain 2.0

- The extended factory domain
  - **Each instance different**
    - Assembly, procurement regions
    - Semi-random structure
    - States are not useful for transfer!

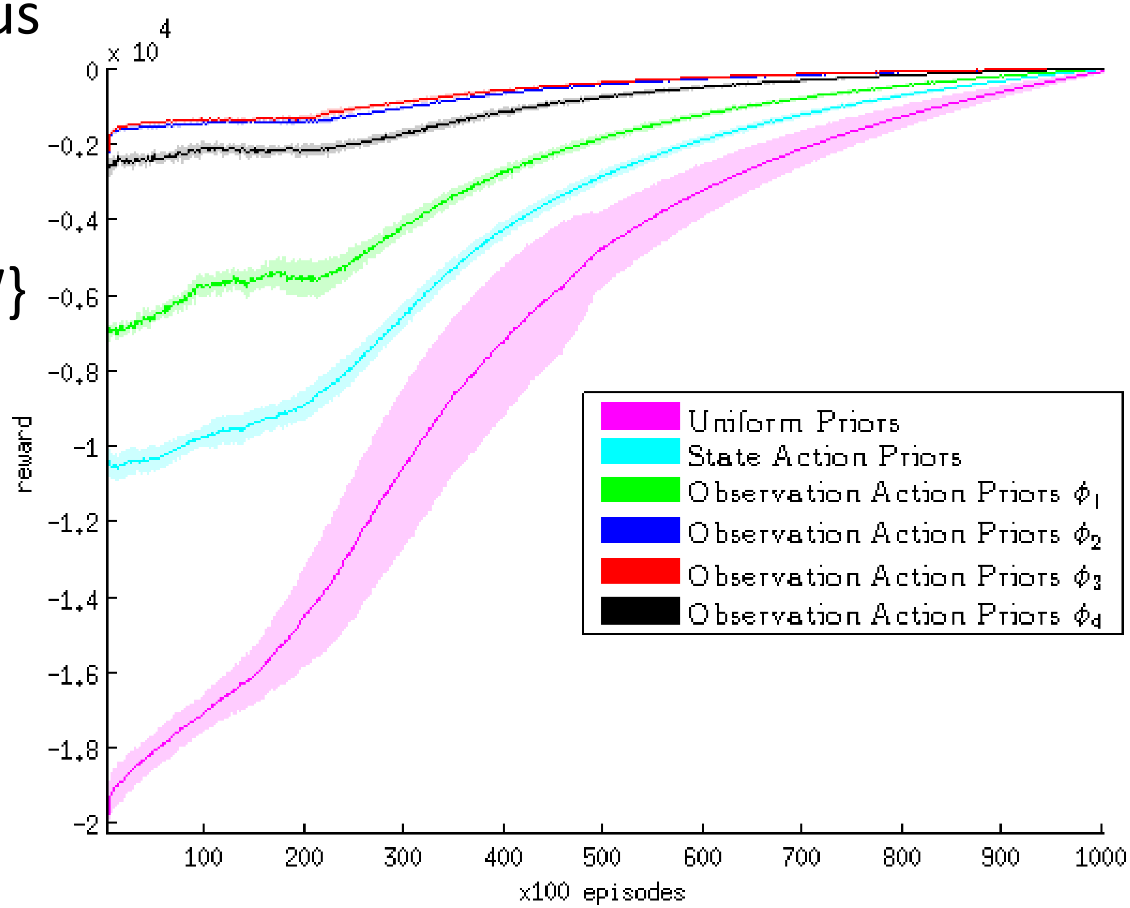






Results: Effect of Different Features

- Different feature sets in action prior
  - φ_1: current cell, item status
  - φ_2: cells N, S, E, W
  - φ_3: $\varphi_1 \cup \varphi_2$
  - φ_4: $\varphi_3 \cup$ {NE, NW, SE, SW}
- Trade-off:
  - Under- vs over-representation
  - Feature learning [Rosman 2014]



An Application: Autonomous Caregivers

Goal

Autonomous agent: enable novice agents to safely learn in a self-directed manner

[Rosman, Hayes, Scassellati, 2015]

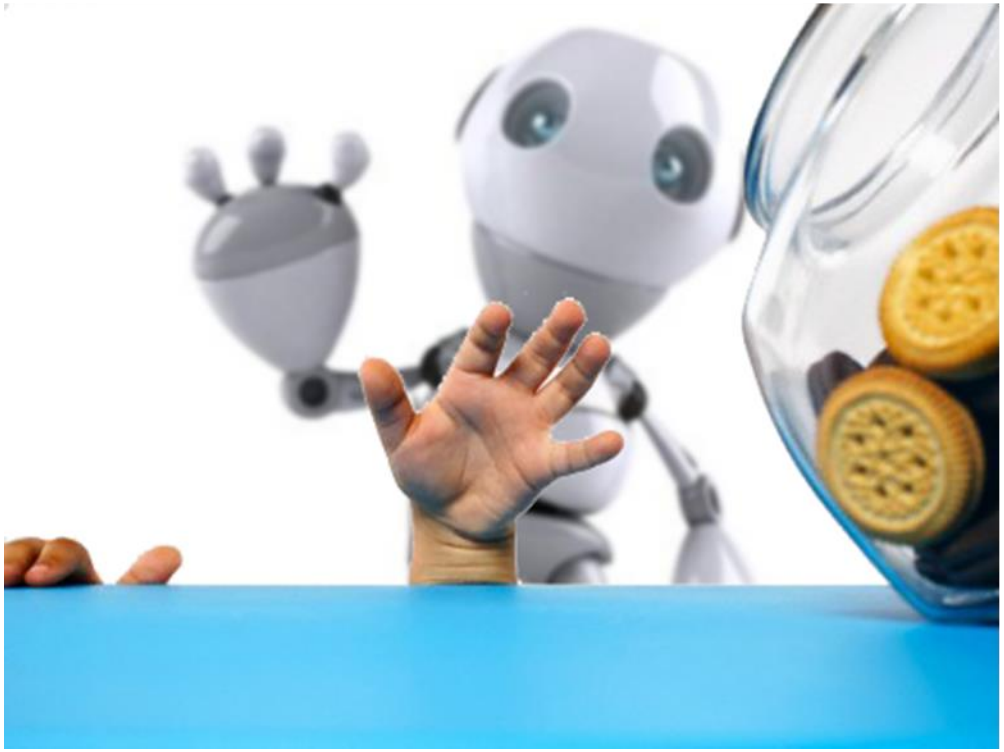
our future through science

Caregivers Perform Risk Mitigation

Approach

Adapt the environment to promote safety

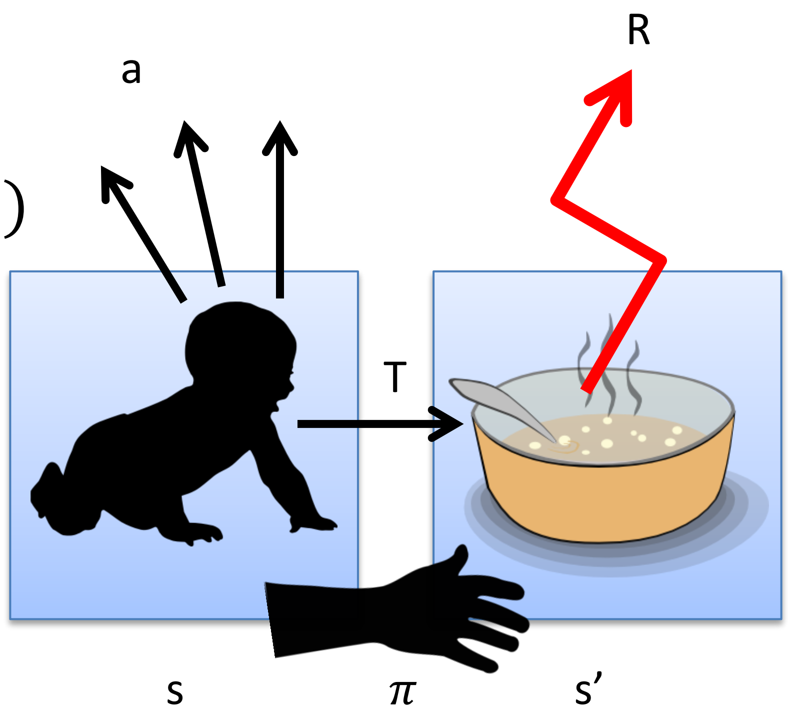
- Without sacrificing quality of learning experience
- Assist through indirect communication and manipulation



CSIR
our future through science

Model of Novice Behaviour

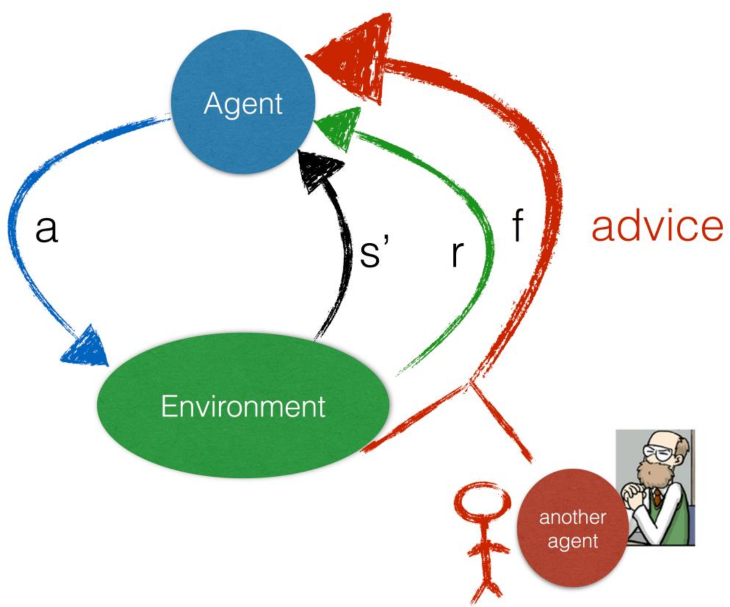
- Novice modelled as an MDP
  - Environment states $s \in S$
  - Actions $a \in A$
  - Environment dynamics $T(s, a, s')$
  - Rewards $R(s, a)$
  - Policy $\pi(s, a)$



our future through science

Caregivers Are A Shaping Mechanism!

- Corrective signals are provided by caregivers
  - Informed by an internal model of reasonable behaviour to assess how risk prone a novice agent is
  - Provided selectively (when necessary)
  - Provided with foresight (before harm is inevitable)



our future through science

A Model of Safety

- **Goal: Determine if novice is behaving safely**
  - Estimate policy similarities between novice and expert
- Current trajectory: $\tau = s^{t+1}, a^{t}, s^{t}, a^{t-1}, s^{t-1}, \ldots$
- Safe behaviour: expert policies → $\theta_s(A)$
- $P(safe\,|\tau) = \dfrac{P(\tau|safe)P(safe)}{P(\tau)}$

Prior prob. of behaving safely

Normalisation factor

- $P(\tau|safe) = \prod_{k=1}^{t} \theta_{s^k}(a^k)$

Safe (reasonable) transition probs.

A Model of Danger

- **Goal: Estimate potential future dangers**
  – Expected environmental harm of likely future actions
- Evaluate expectation for each potential source of harm o:

$$P(collision\,|\tau) \times d_o$$

Extrinsic damage caused by collision with o

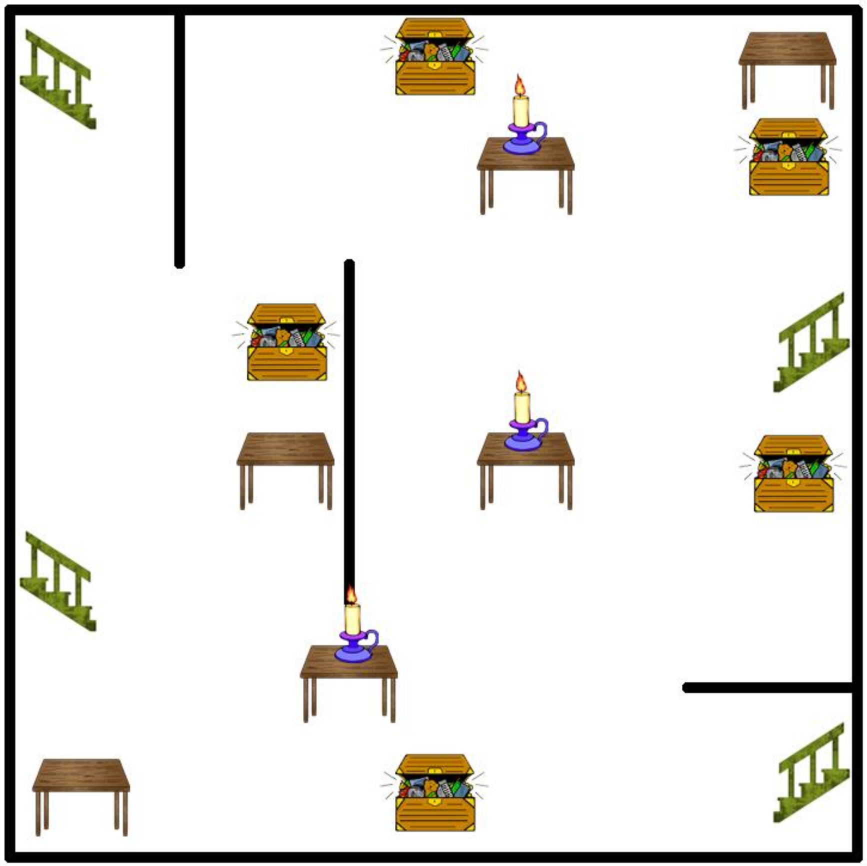
$$= (1 - P(safe\,|\tau)) \times P(reach_o|\,\tau) \times d_o$$

Prob. of behaving safely

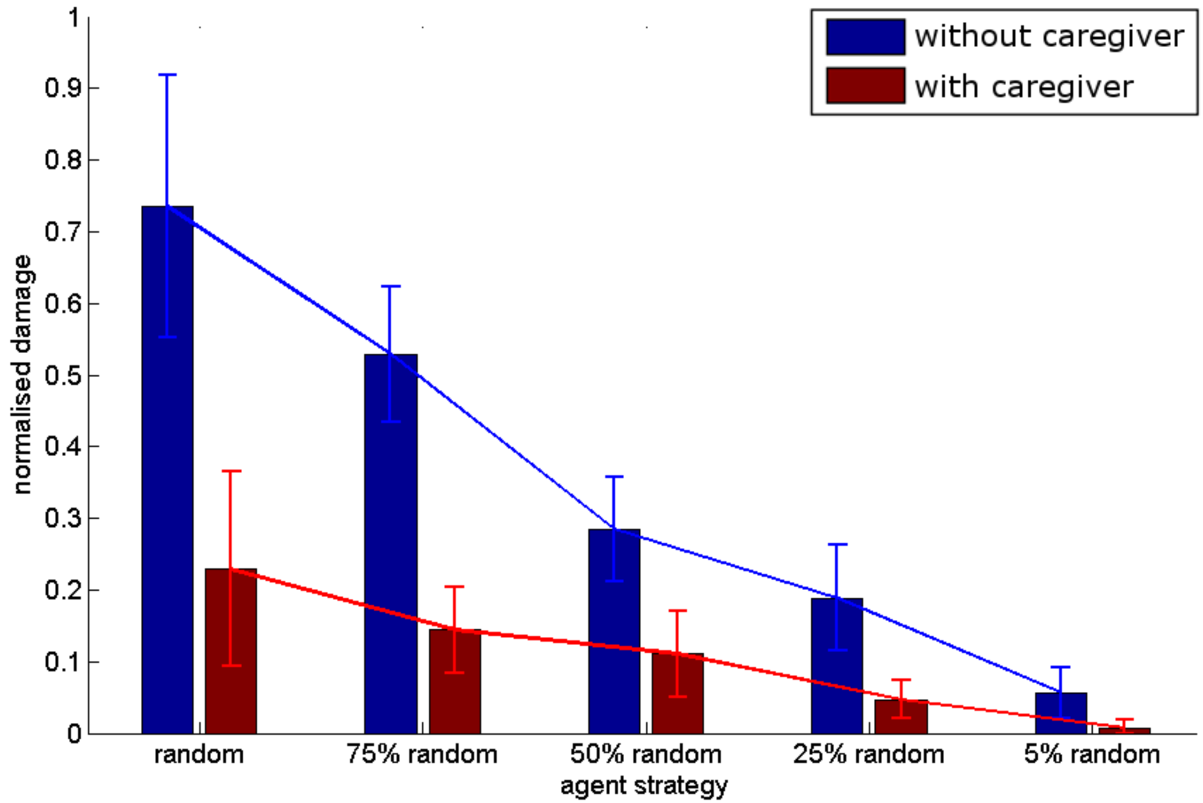
Prob. of reaching o $\propto$ distance to o from current position

Toybox World

- Exploring to reach toy boxes [ICDL 2015]
- Hazards:
  - Major damage: candles, stairs
  - Minor damage: tables
- Novice agent:
  - ϵ-greedy
  - 'Play' for 200 time steps
- Caregiver agent:
  - Trained on 1,000 expert steps
  - Moves 3x faster than novice
- Interventions:
  - Move candle between tables
  - Block stairwell



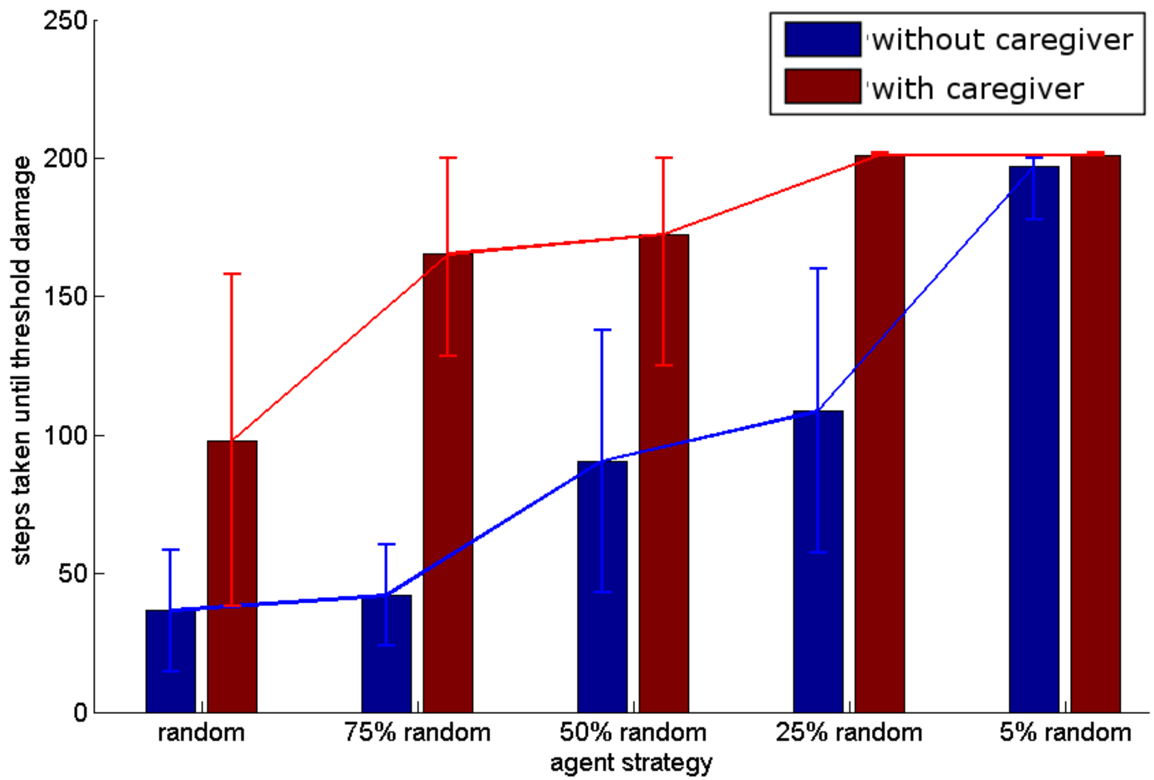
Results: Reducing Harm



Agent	No caregiver	With caregiver
random motion	0.7346	0.2291
75% random	0.5295	0.1438
50% random	0.2846	0.1106
25% random	0.1887	0.0466
5% random	0.0565	0.0072

our future through science

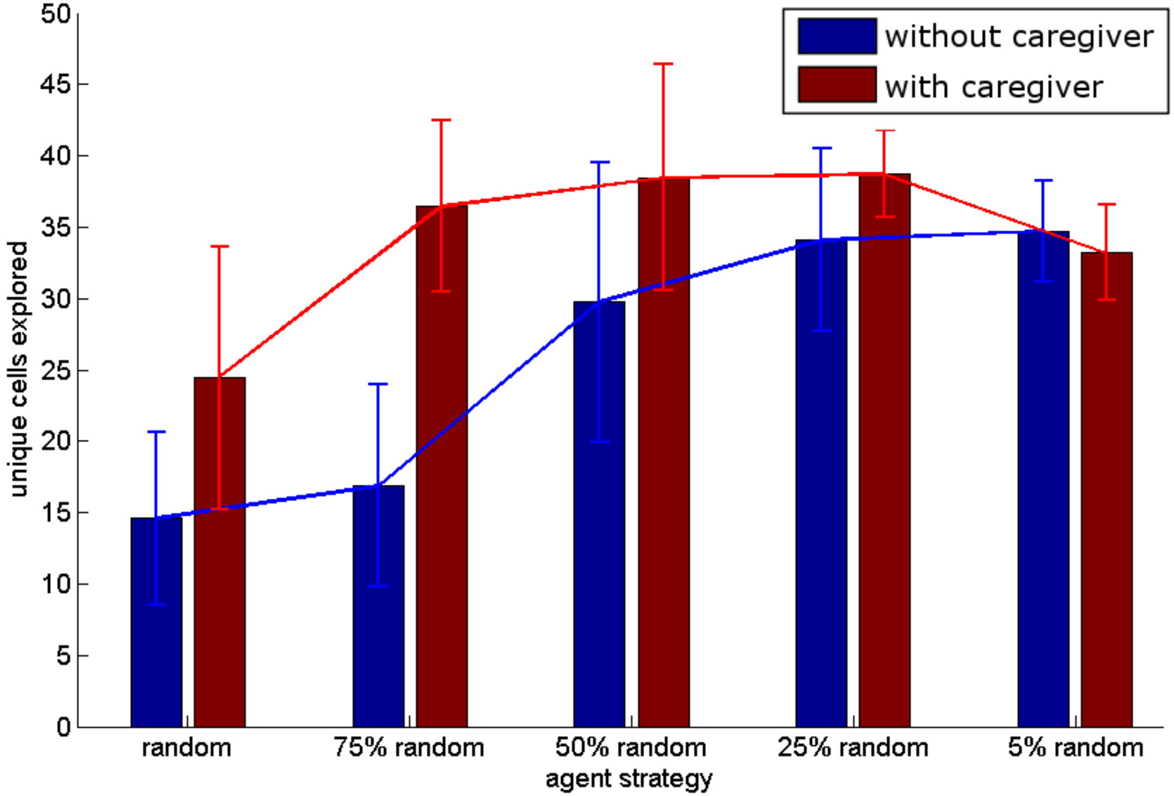
Results: Exploration Time



Agent	No caregiver	With caregiver
random motion	35.50	97.05
75% random	41.05	164.20
50% random	89.50	171.50
25% random	107.70	200.00
5% random	195.75	200.00

our future through science

Results: Environment Coverage



Agent	No caregiver	With caregiver
random motion	14.6000	24.4500
75% random	16.8500	36.4500
50% random	29.7500	38.4500
25% random	34.1000	38.7000
5% random	34.7000	33.2000

CSIR
our future through science

Conclusion

- Action priors
  – Behavioural domain invariances
  – Task independent
  – “Common sense” knowledge
- Improve learning speed
  – Use as exploration bias in RL
- Identify safe/normal behaviour
- General paradigm for multi-task decision making agents
  – If learning multiple tasks in the same domain, **learn from previous tasks!**

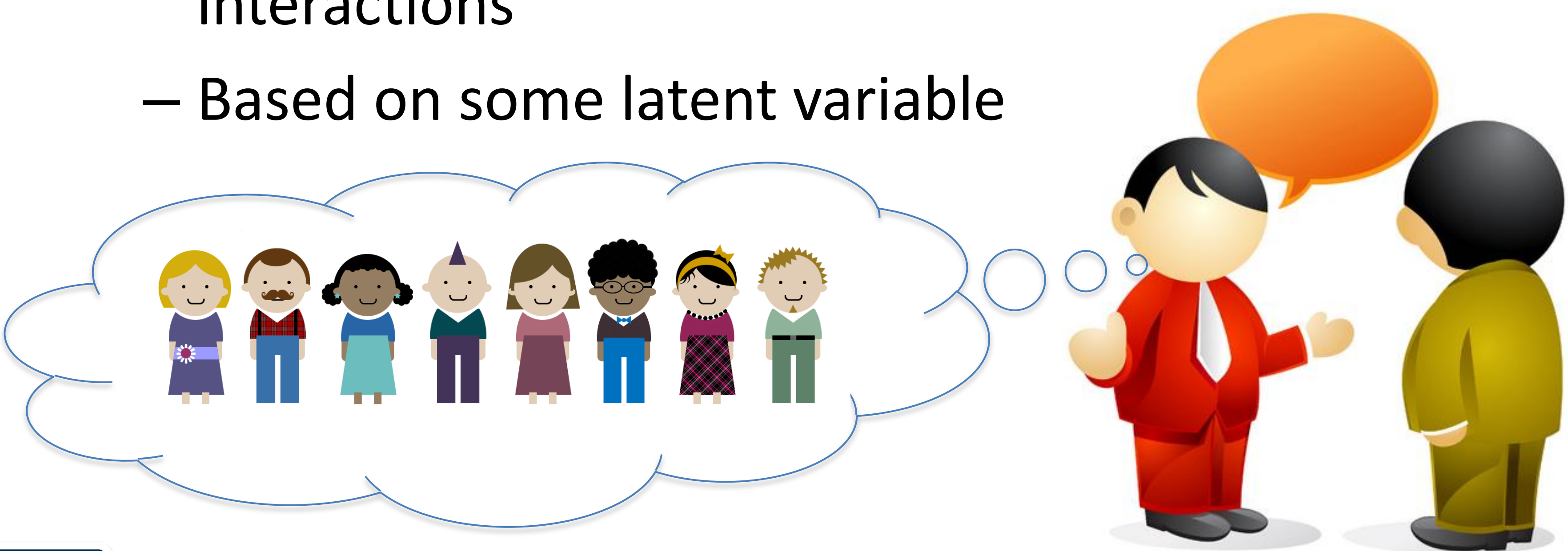
CSIR
our future through science

Chapter 2:
Efficient Skill Selection
(Bayesian Policy Reuse)

our future through science

Responding Online to New Situations

- Engaged in a task
  - Not enough time to learn a policy
- Previous experience of tasks
  - Choose the best policy in a sequence of interactions
  - Based on some latent variable

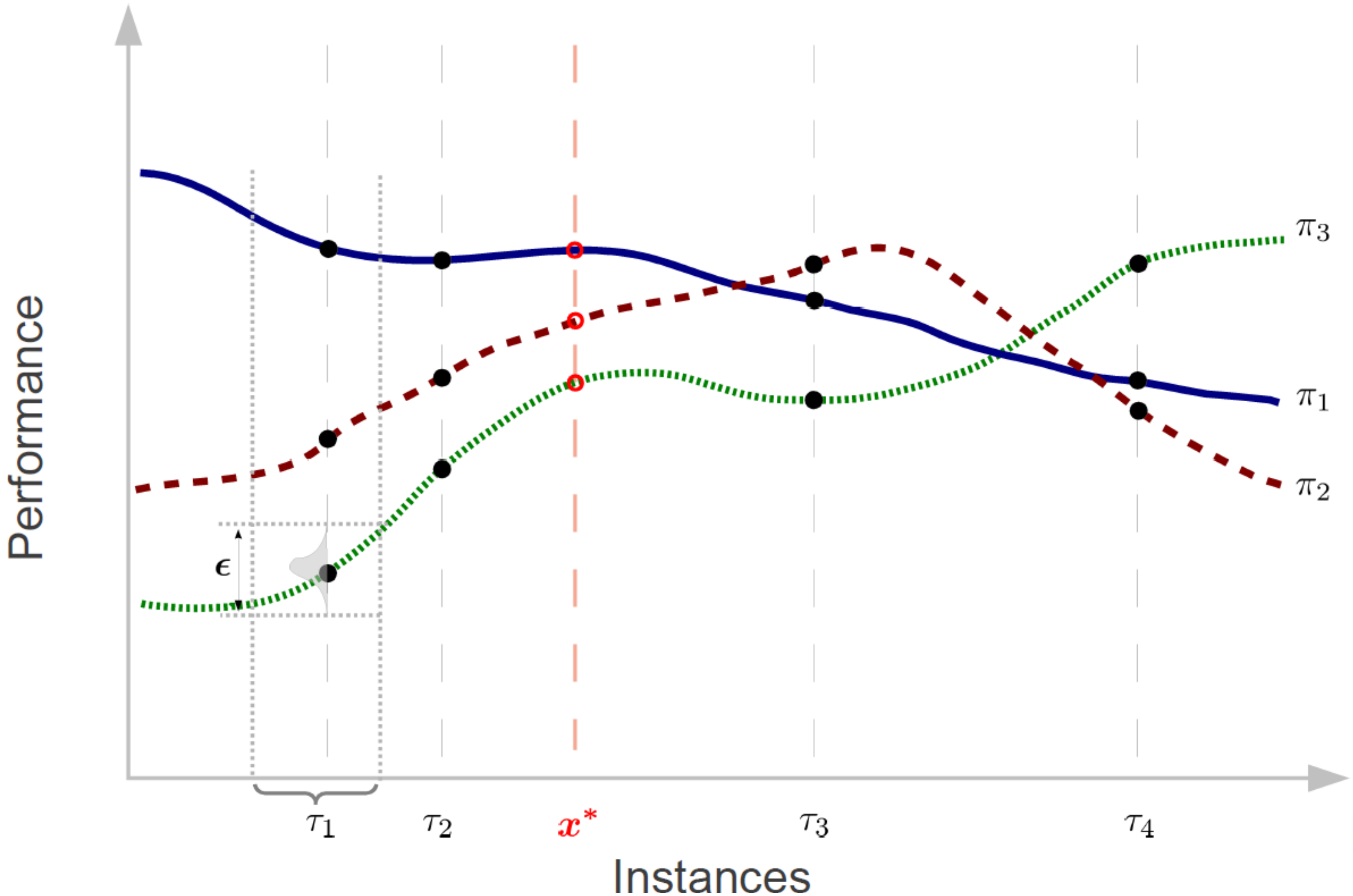


The Policy Reuse Problem

- Given:
  - Exposure to previous task instances
  - A policy library trained on those tasks
- Experience a new task
- Goal:
  - Select policies for new task to minimise total regret
- Assume: limited task duration
  - Cannot learn from scratch

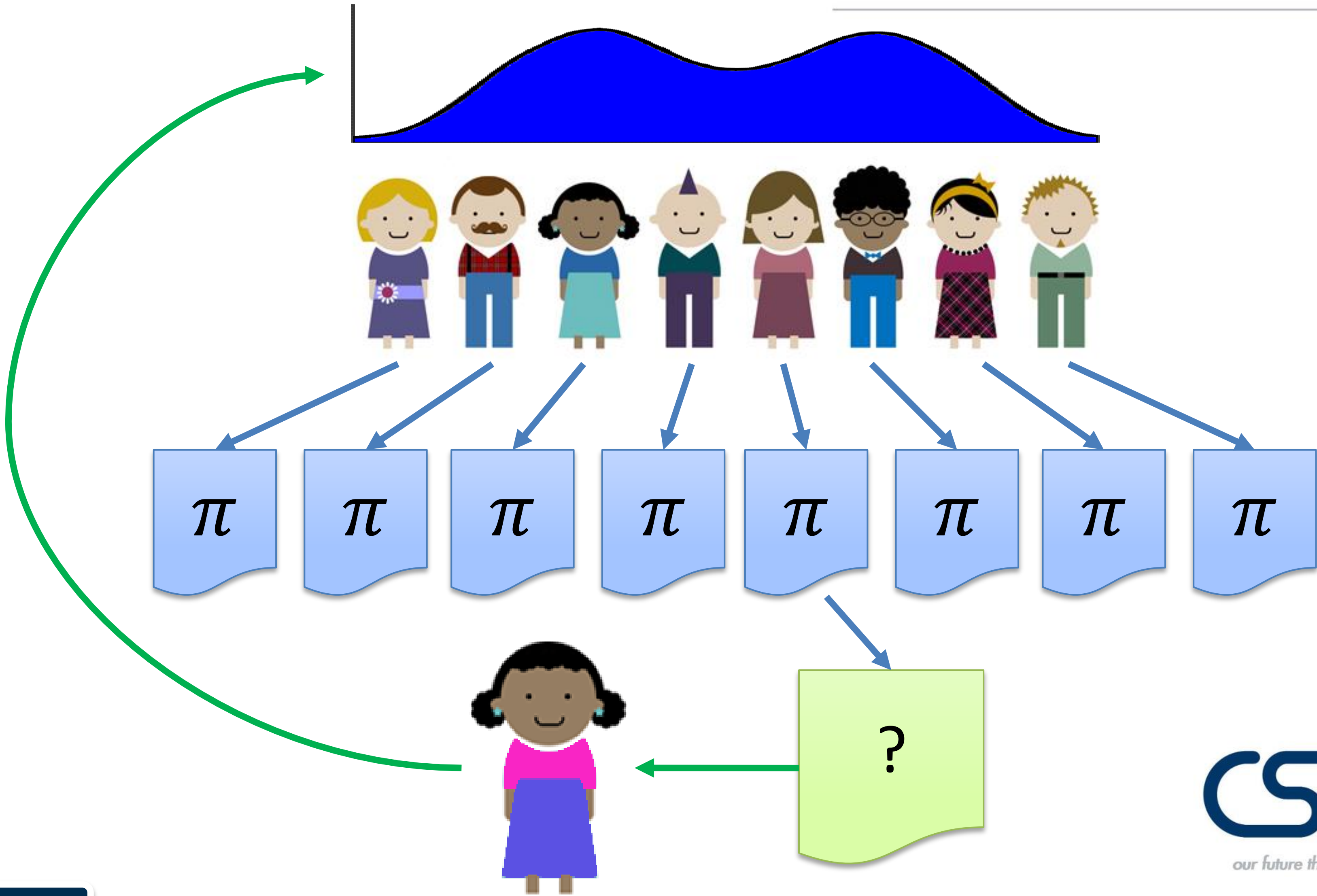
CSIR
our future through science

Insight



(unknown embedding)

Bayesian Policy Reuse Overview



Ingredient 1: Performance

- Performance U:
  - Returns achieved by a policy on a task
- Performance models:
  - $P(U|\tau, \pi)$
  - Maintain for each experienced task and policy
- Use to estimate performance of a policy on an unknown task

Ingredient 2: Signals

- Signals σ: information correlated with task performance, provided during task execution
  - E.g. rewards, (partial) states
- Signal/observation models:
  - $P(\sigma|\tau,\pi)$
  - Maintain for each task and policy
- Use as feedback signal for identifying task

Belief Models

- Maintain belief over set of task instances τ
- Update
  - Based on signals after playing a policy
  - Over ALL known tasks!
  - Notion of task similarity

Signal model

$$\beta^t(\tau) = \frac{P(\sigma^t|\tau, \pi^t)\beta^{t-1}(\tau)}{\sum_{\tau' \in \mathcal{T}} P(\sigma^t|\tau', \pi^t)\beta^{t-1}(\tau')}$$

Bayesian Policy Reuse

1. Select policy
2. Apply policy
3. Observe signal
4. Update belief

Algorithm 1 Bayesian Policy Reuse (BPR)

Require: Problem space $\mathcal{X}$, Policy library Π, observation space Σ, prior over the problem space $\mathrm{P}(\mathcal{X})$, observation model $\mathrm{P}(\Sigma|\mathcal{X},\Pi)$, performance model $\mathrm{P}(U|\mathcal{X},\Pi)$, number of episodes K.

Initialise beliefs: $\beta^0(\mathcal{X}) \longleftarrow \mathrm{P}(\mathcal{X})$.
for episodes $t = 1 \ldots K$ **do**
    Select a policy $\pi^t \in \Pi$ using the current belief β^{t-1} and the performance model $\mathrm{P}(U|\mathcal{X},\pi^t)$.
    Apply π^t on the task instance.
    Obtain an observation signal σ^t from the environment.
    Update the belief $\beta^t(\mathcal{X}) \propto \mathrm{P}(\sigma^t|\mathcal{X},\pi^t)\beta^{t-1}(\mathcal{X})$.
end for

[Rosman, Hawasly, Ramamoorthy, MLJ, to appear 2015]

Policy Selection

- Selection heuristics (based on Bayesian optimisation):

- Probability of Improvement (PI):

$$\hat{\pi} = \arg\max_{\pi \in \Pi} \sum_{\tau \in \mathcal{T}} \beta(\tau) \mathrm{P}(U^+ | \tau, \pi)$$

- Expected Improvement (EI):

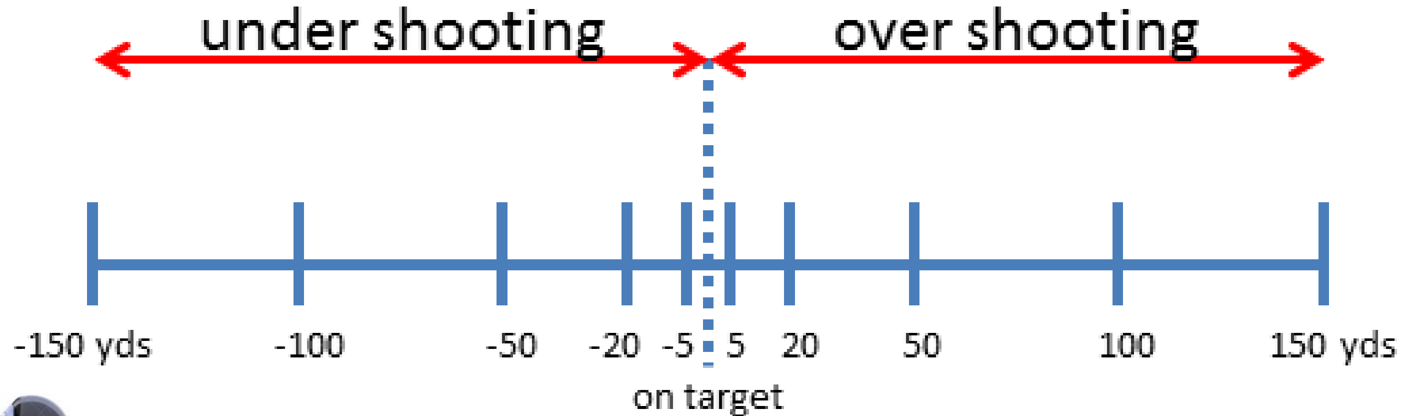
$$\hat{\pi} = \arg\max_{\pi \in \Pi} \int_{\bar{U}}^{U^{max}} \sum_{\tau \in \mathcal{T}} \beta(\tau) \mathrm{P}(U^+ | \tau, \pi) \mathrm{d}U^+$$

our future through science

Illustrative Example – The Golf Range

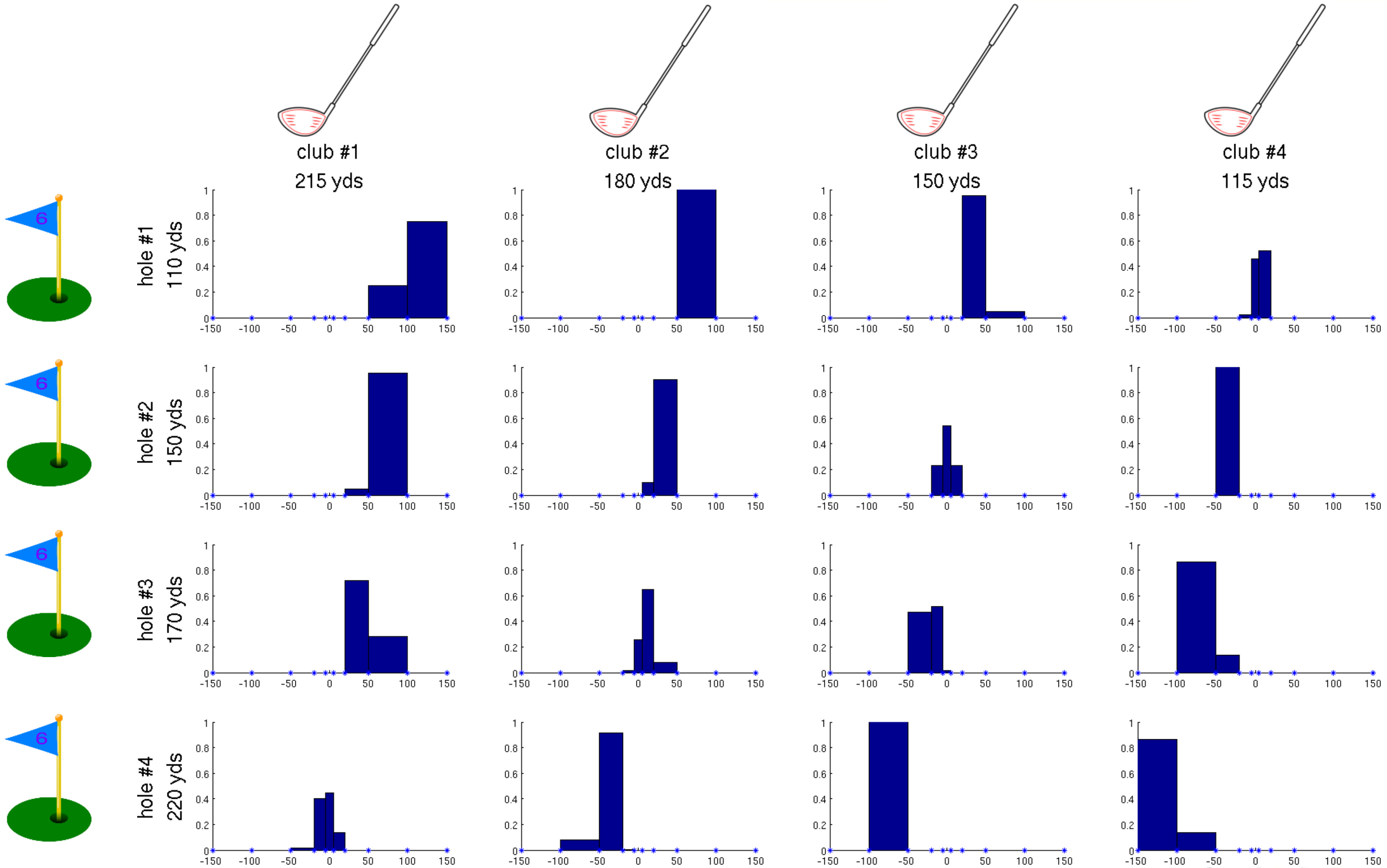
Ground truth:

Club	Average Yardage	Standard Deviation of Yardage
π_1 = 3-wood	215	8.0
π_2 = 3-iron	180	7.2
π_3 = 6-iron	150	6.0
π_4 = 9-iron	115	4.4



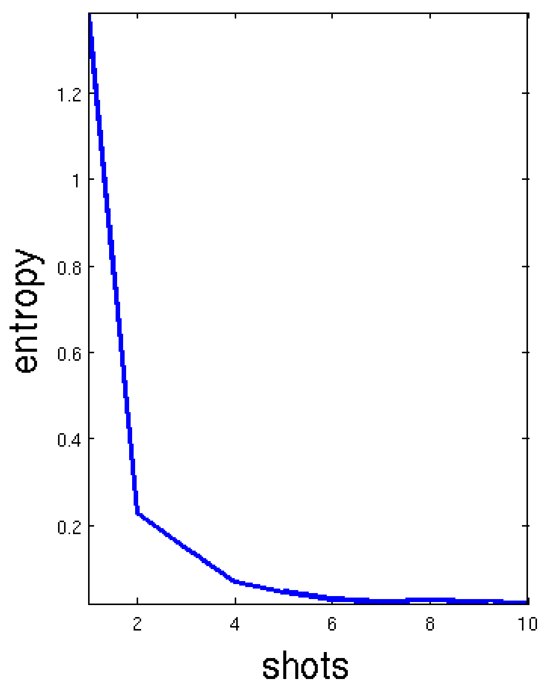
CSIR
our future through science

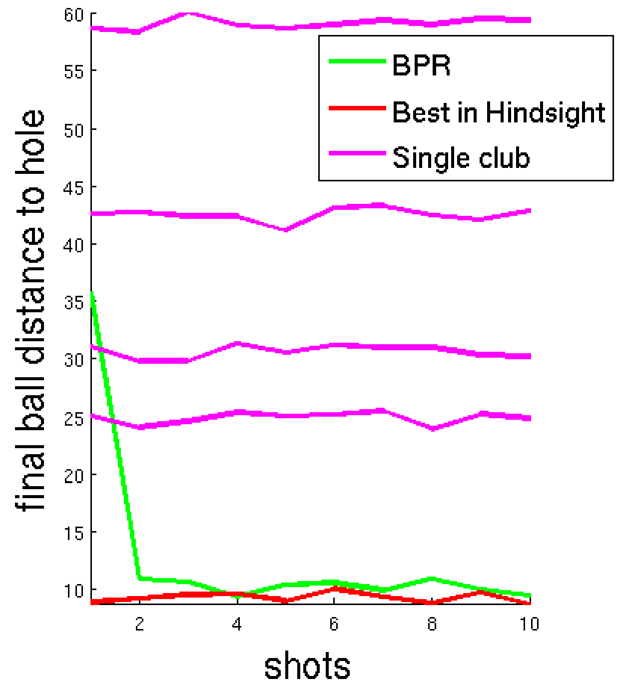
Illustrative Example – Signal Models



Results on New Task

Shot	1	2	3	4	5	6	7	8
Club	1	2	2	2	2	2	2	2
Error	35.3657	13.1603	4.2821	6.7768	2.0744	11.0469	8.1516	2.4527
Signal	20–50	5–20	-5–5	5–20	-5–5	5–20	5–20	-5–5
β entropy	1.3863	0.2237	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

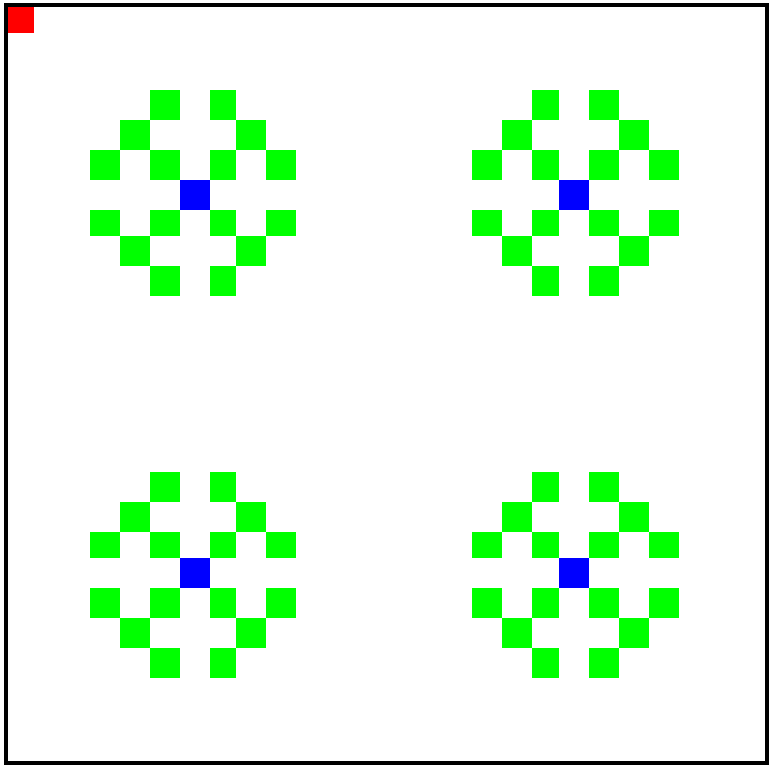


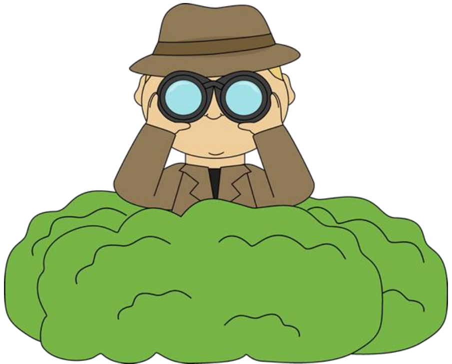


our future through science

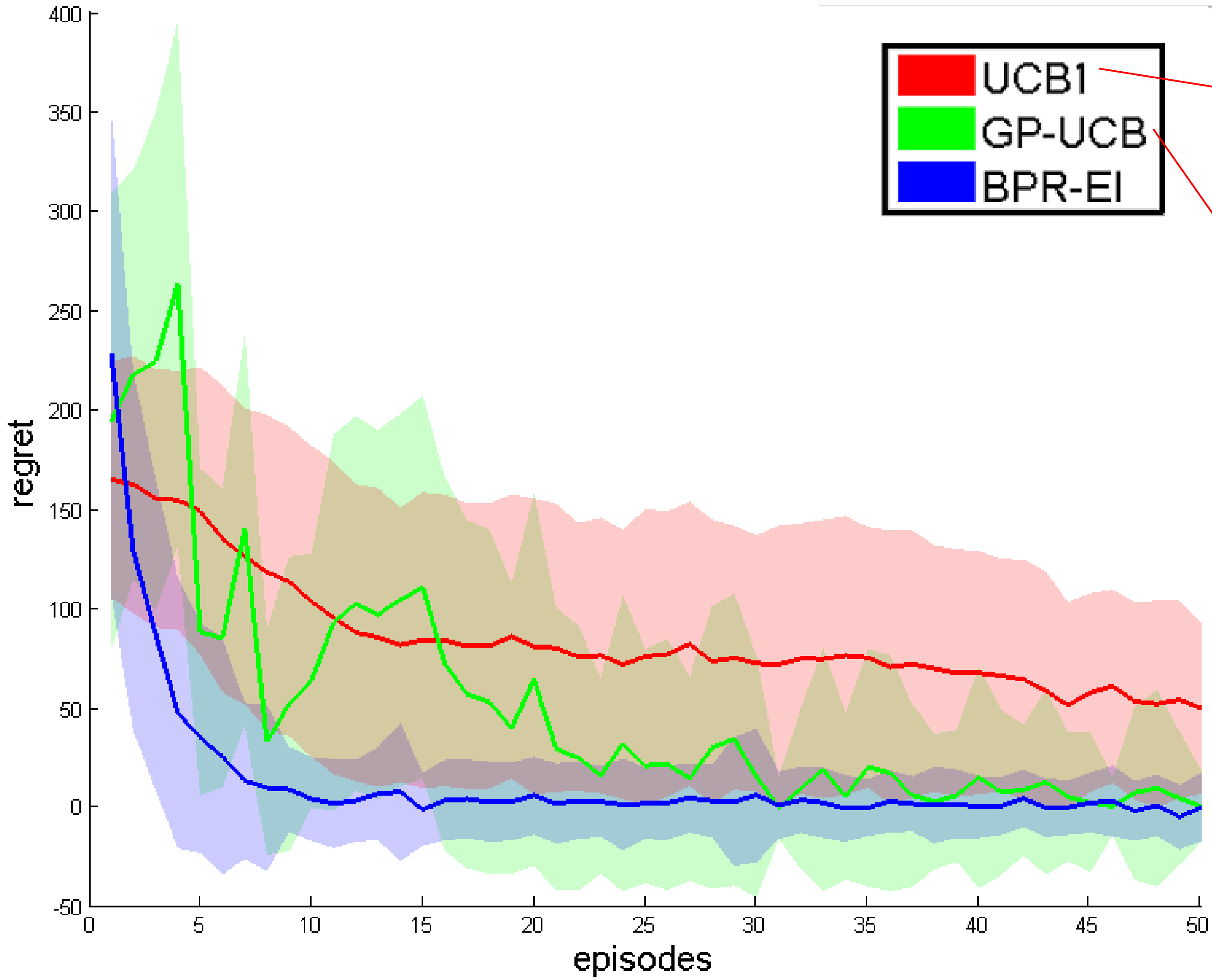
Surveillance Domain

- Watching for intruders, from hills
  - Connected visibility
- 68 tasks





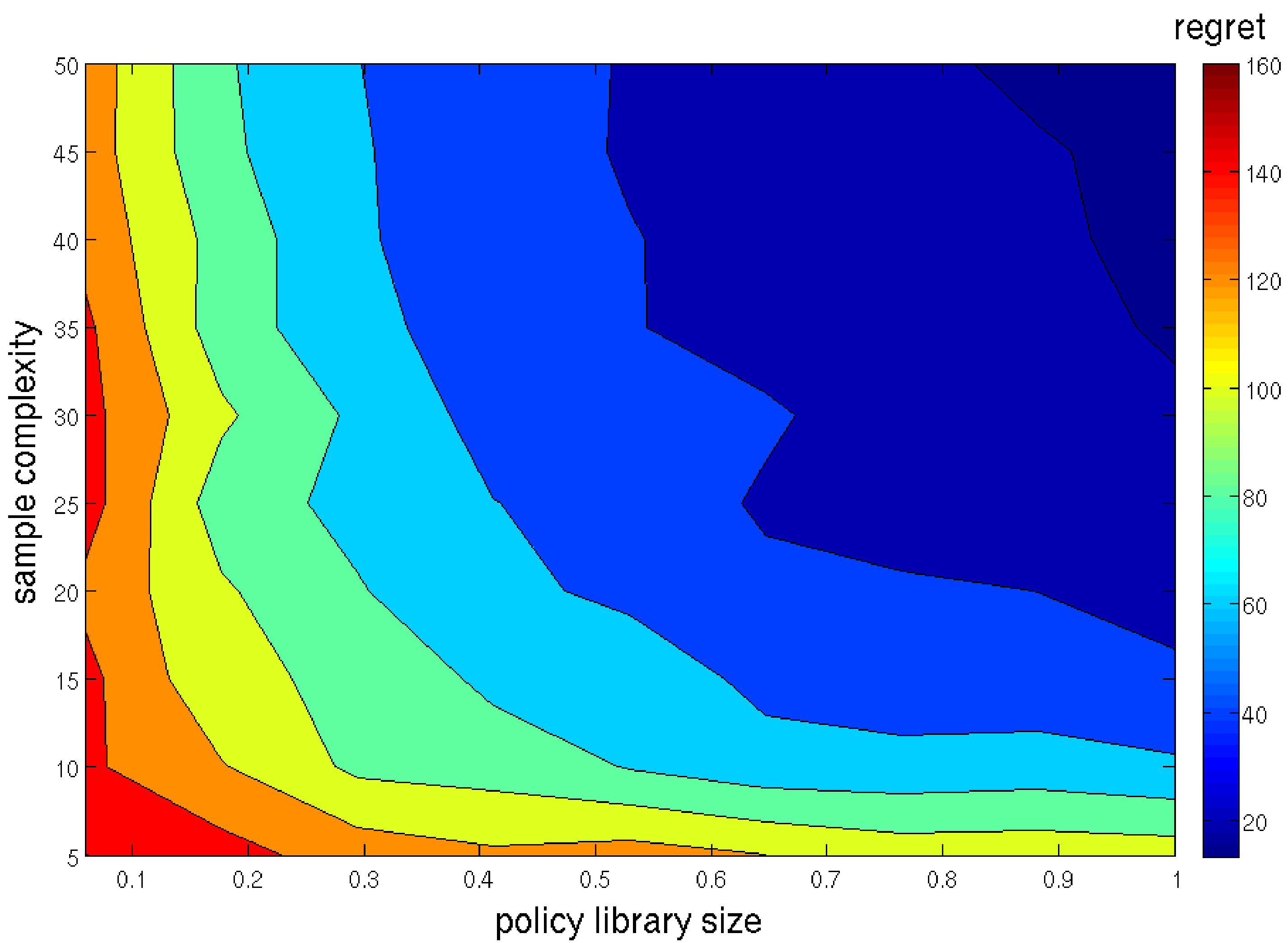
Rapid Identification



Bandits (with cheating)

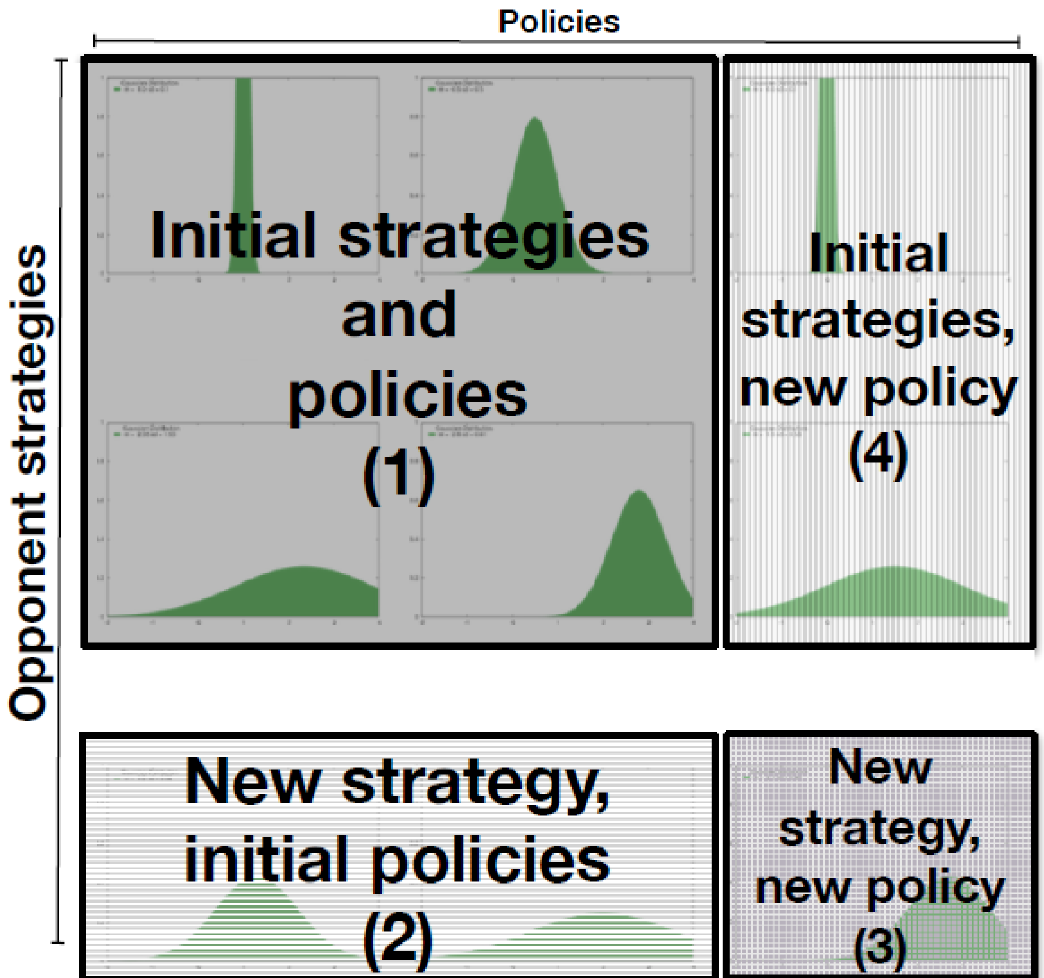
Bayesian optimisation

Library Size-Episodes-Regret Trade-off



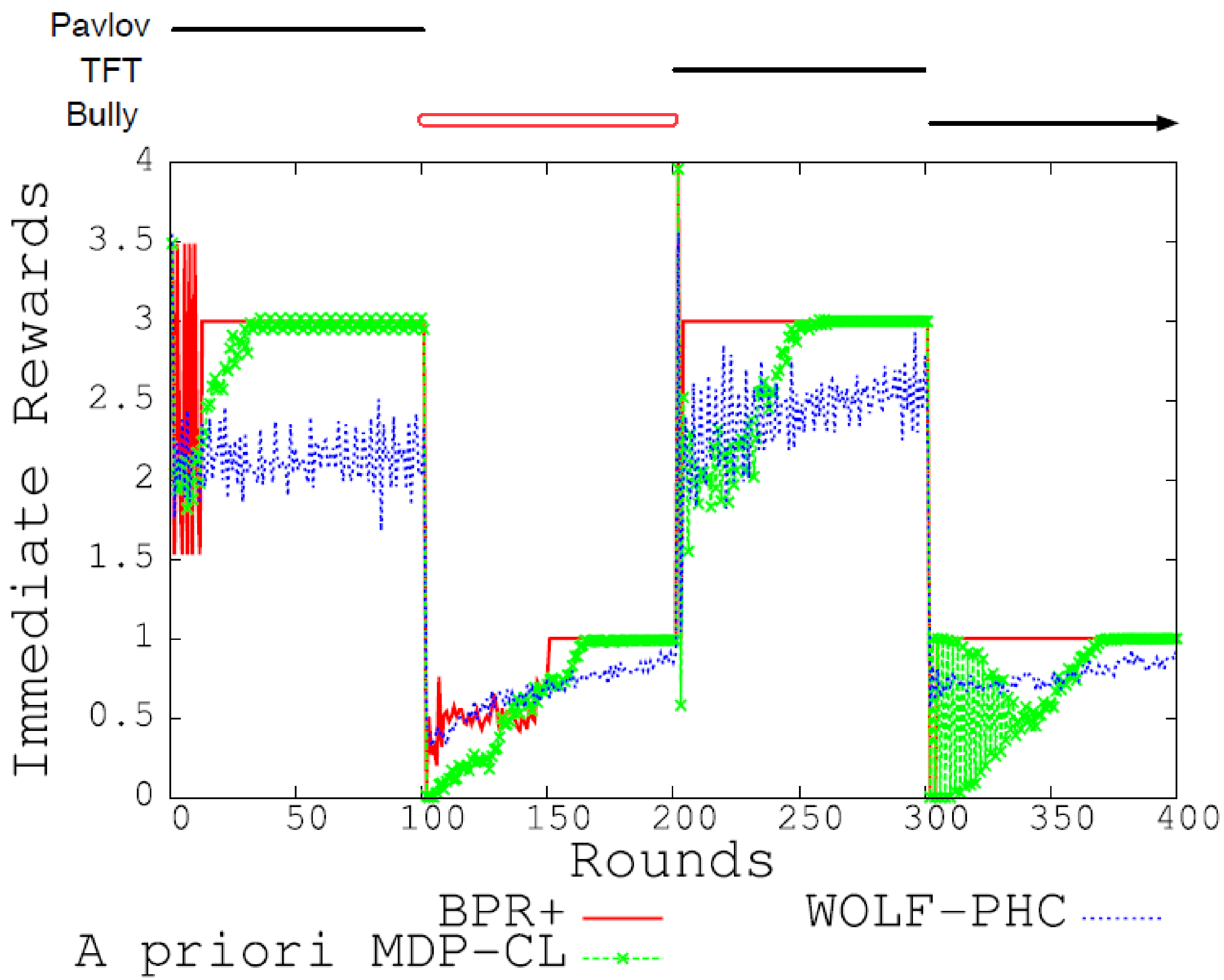
Non-stationarity and Adversity

- Changing opponents:
  – Keep all beliefs non-zero
- New strategies:
  – Unlikely reward sequence
  – Enable learning



[Hernandez-Leal, Taylor, Rosman, *submitted*]

Multi-agents: Tracking Changes



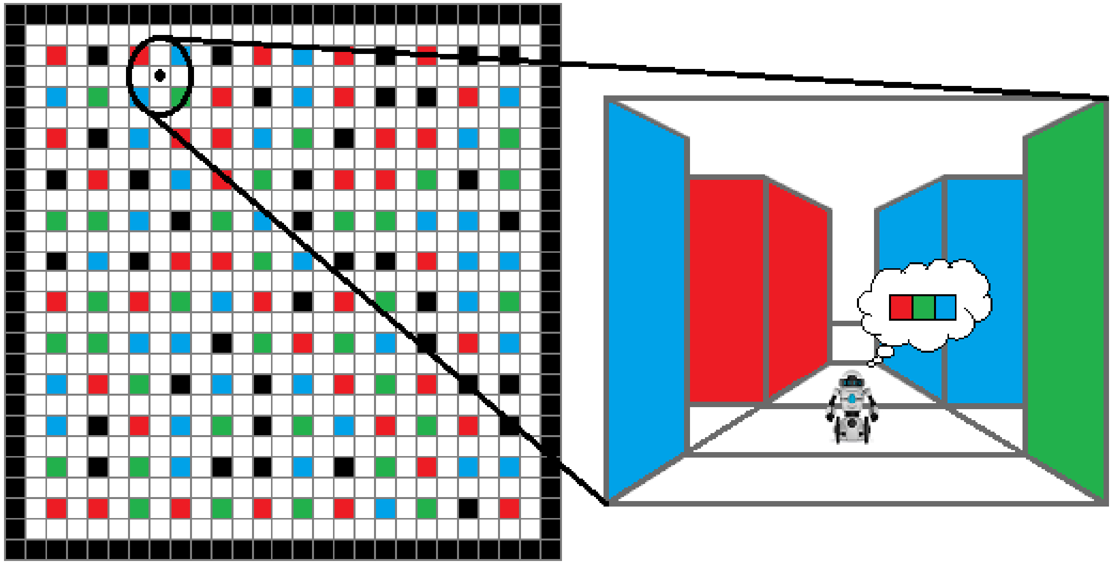
Summary

- Bayesian Policy Reuse: general framework for rapid policy selection
  - Maintain beliefs over tasks
  - Update with observation models
  - Select according to performance models
- Interact efficiently with unknown tasks and agents

our future through science

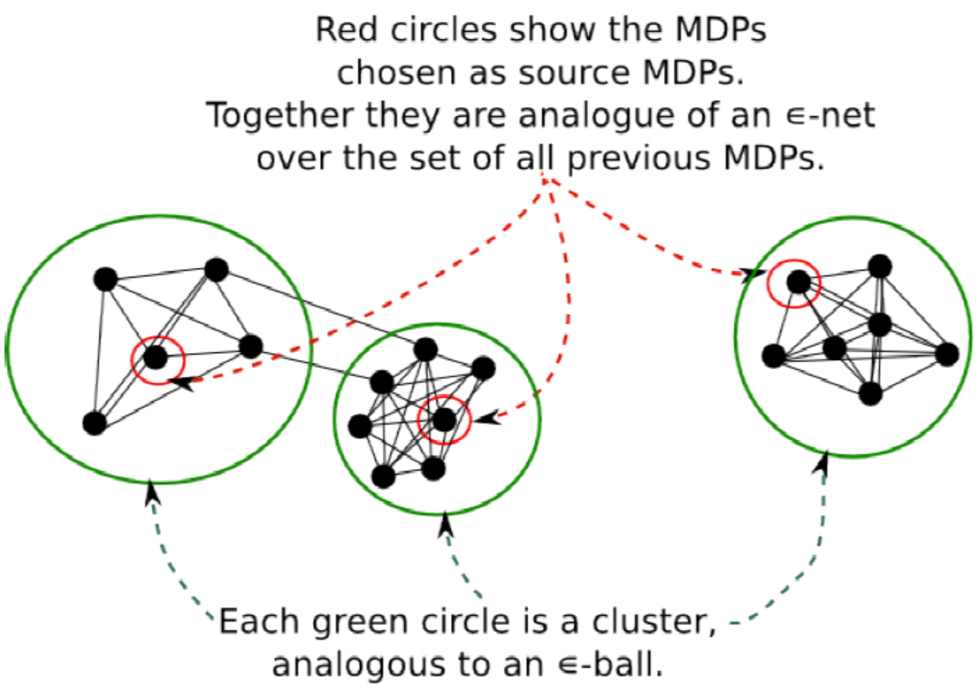
Future Work

- Extensions:
  - Continuous action/task sets
    - Distributions over parameter space
  - Different decision making paradigms
    - Classical planning
    - POMDPs
    - MCTS



Future Work

- Structure in task space?
  - Non-parametric:
    - Clustering MDPs
  - Parametric:
    - Hidden parameter MDPs
  - Compositionality and hierarchy of behaviours



CSIR
our future through science

[Mahmud, Hawasly, Rosman, Ramamoorthy, *under review*]

Thank you!

And thanks to all these great people:

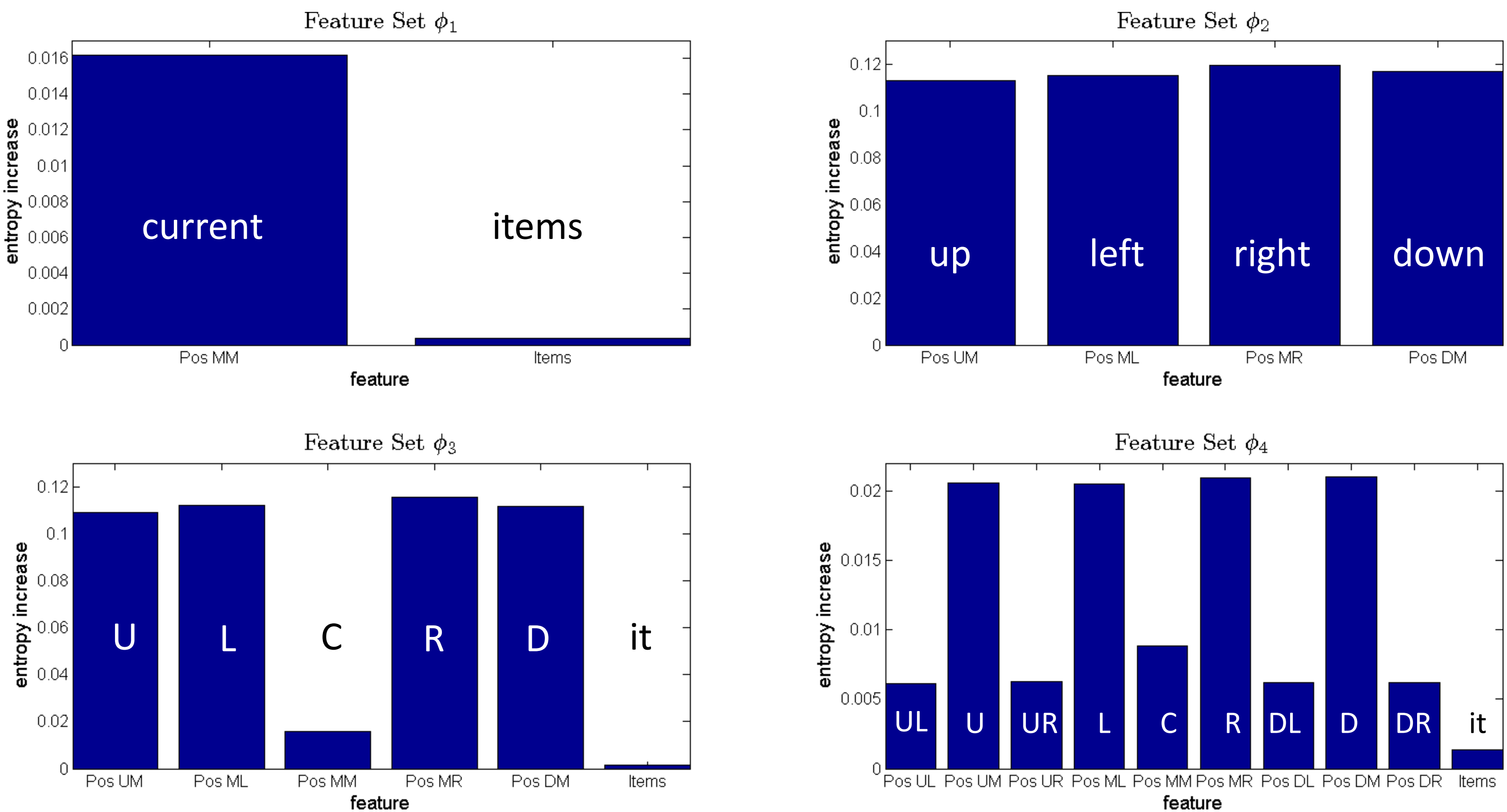
Dr Subramanian Ramamoorthy (U. of Edinburgh)
Dr Majd Hawasly (U. of Edinburgh)
Dr Hassan Mahmud (U. of Edinburgh)
Bradley Hayes (Yale University)
Pablo Hernandez-Leal (INAOE)
Prof George Konidaris (Duke University)
Prof Brian Scassellati (Yale University)
Prof Matt Taylor (Washington State University)

CSIR our future through science

Benjamin Rosman (brosman@csir.co.za)

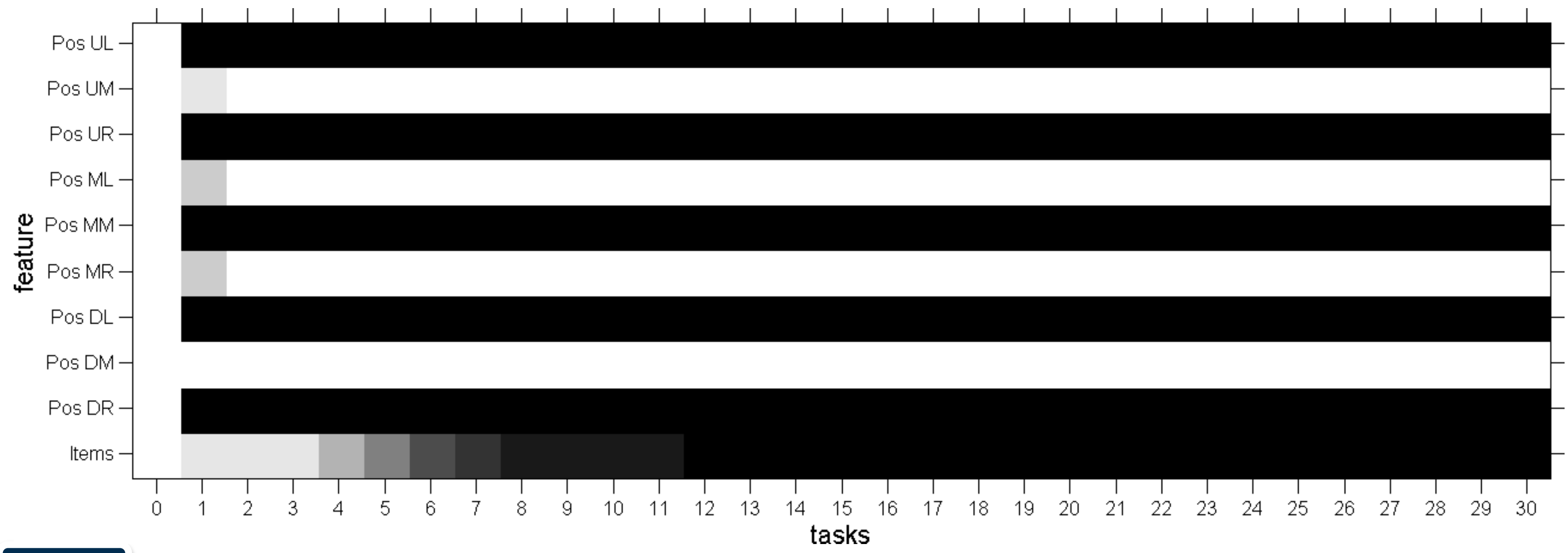
Action Priors: Feature Entropy

Effect of removing a feature:



Adaptive Feature Sets

- Features selected as a function of number of tasks
- Initial features: 10 (values: $4^9 \times 3$)
- Final features: 4 (values: 4^4)



Results: Online Feature Selection

- Effect of priors: episodes 1 and convergence