Intrinsically Motivated RL

- Intrinsic motivation
- Previous computational approaches
- Barto, Singh & Chentanez, ICDL 2004
- Şimşek & Barto, ICML 2006
- What constitutes a useful skill?

Motivation

- "Forces" that energize an organism to act and that direct its activity
- Extrinsic Motivation: being moved to do something because of some external reward ($$, a prize, etc.)
- Intrinsic Motivation: being moved to do something because it is inherently enjoyable (curiosity, exploration, manipulation, play, learning itself…)

A classic

Robert White, Motivation Reconsidered: The Concept of Competence, Psyc. Rev. 1959

- Competence: an organism's capacity to interact effectively with its environment
- Critique of Freudian and Hullian view of motivation: reducing drives related to the biologically primary needs, e.g. food
- "The motivation needed to obtain competence cannot be wholly derived from sources of energy currently conceptualized as drives or instincts."
- Made a case for exploratory motive as an independent primary drive

Another classic

D. E. Berlyne, Curiosity and Exploration, Science, 1966

- "As knowledge accumulated about the conditions that govern exploratory behavior and about how quickly it appears after birth, it seemed less and less likely that this behavior could be a derivative of hunger, thirst, sexual appetite, pain, fear of pain, and the like, or that stimuli sought through exploration are welcomed because they have previously accompanied satisfaction of these drives."
- Novelty, surprise, incongruity, complexity

Computational Curiosity

Jurgen Schmidhuber, 1991, 1991, 1997

- "The direct goal of curiosity and boredom is to improve the world model. The indirect goal is to ease the learning of new goal-directed action sequences."
- "Curiosity Unit": reward is a function of the mismatch between model's current predictions and actuality. There is positive reinforcement whenever the system fails to correctly predict the environment.
- "Thus the usual credit assignment process … encourages certain past actions in order to repeat situations similar to the mismatch situation."

Computational Curiosity

Schmidhuber (cont.)

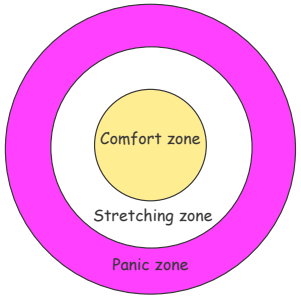
- "The same complex mechanism which is used for 'normal' goal-directed learning is used for implementing curiosity and boredom. There is no need for devising a separate system which aims at improving the world model."
- Problems with rewarding prediction errors
  - Agent will be rewarded even though the model cannot improve. So it will focus on parts of environment that are inherently unpredictable.
  - Agent won't try to learn easier parts before learning hard parts

Computational Curiosity

Schmidhuber (cont.):

- Instead of rewarding prediction errors, reward prediction improvements.
- "My adaptive explorer continually wants … to focus on those novel things that seem easy to learn, given current knowledge. It wants to ignore (1) previously learned, predictable things, (2) inherently unpredictable ones (such as details of white noise on the screen), and (3) things that are unexpected but not expected to be easily learned (such as the contents of an advanced math textbook beyond the explorer's current level)."

From Charlie's 4th grade classroom

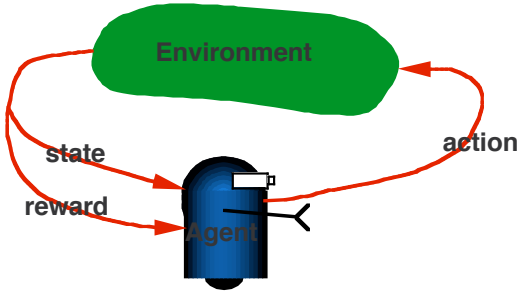


Computational Curiosity

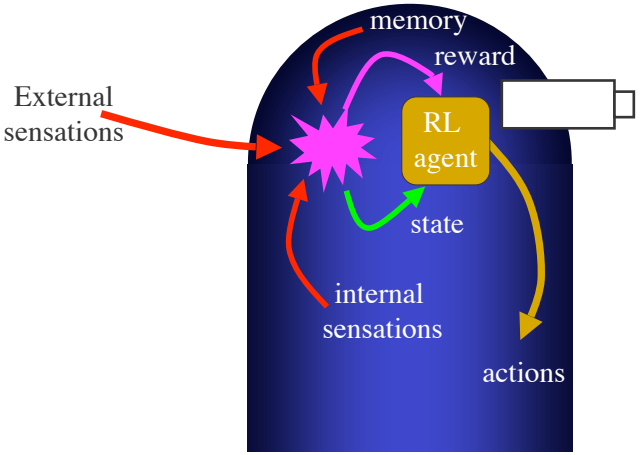
Rich Sutton, Integrated Architectures for Learning, Planning and Reacting based on Dynamic Programming, ICML 1990.

- For each state and action, add a value to the usual immediate reward called the exploration bonus.
- It is proportional to a measure of how uncertain the system is about the value of doing that action in that state.
- Uncertainty is assessed by keeping track of the time since that action was last executed in that state. The longer the time, the greater the assumed uncertainty.
- "…why not expect the system to plan an action sequence to go out and test the uncertain state-action pair?"

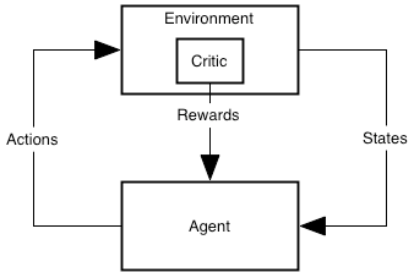
Usual View of RL



A Less Misleading View

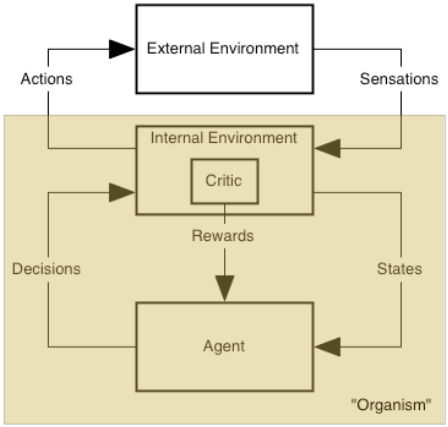


Usual View of RL



Usually represented as a finite MDP.
Reward is extrinsic.

A Less Misleading View



All reward is intrinsic.

So What is IMRL?

- Key distinction
  - Extrinsic reward = problem specific
  - Intrinsic reward = problem independent
- Why important: open-ended learning via acquisition of skill hierarchies

Digression: Skills

- cf: macro: a sequence of operations with a name; can be invoked like a primitive operation
  - Can invoke other macros. . . hierarchy
  - But: an open-loop policy
- Closed-loop macros
  - A decision policy with a name; can be invoked like a primitive control action
  - behavior (Brooks, 1986), skill (Thrun & Schwartz, 1995), mode (e.g., Grudic & Ungar, 2000), activity (Harel, 1987), temporally-extended action, option (Sutton, Precup, & Singh, 1997), schema (Piaget, Arbib)

Options

(Sutton, Precup & Singh 1999)

A generalization of actions to include temporally-extended courses of action

An option is a triple $o = < I, \pi, \beta >$

- I: initiation set: the set of states in which o may be started
- π: is the policy followed during o
- β: termination conditions: gives the probability of terminating in each state

Example: robot docking

I : all states in which charger is in sight

π : pre-defined controller

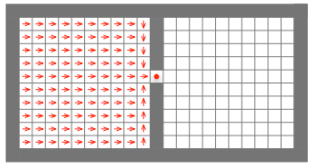
β : terminate when docked or charger not visible

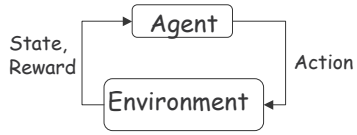
Options (cont.)

- Policies can select from a set of options & primitive actions
- Generalizations of the usual concepts:
  - Transition probabilities ("option models")
  - Value functions
  - Learning and planning algorithms
- Intra-option off-policy learning:
  - Can simultaneously learn policies for many options from same experience

Approach skills

- Skills that efficiently take the agent to a specified set of states, e.g., go-to-doorway
- To learn the skill policy, use pseudo reward, e.g.
  - +1 for transitioning into a subgoal state
  - 0 otherwise





$\ldots\ s_t\ a_t\ \not{r_t}\ s_{t+1}\ a_{t+1}\ \not{r_{t+1}}\ s_{t+2}\ a_{t+2}\ \not{r_{t+2}}\ s_{t+3}\ \ldots$

Use pseudo-reward instead

IMRL Objective

Open-ended learning via acquisition of skill hierarchies

- What skills should the agent learn?
- How can an agent learn these skills efficiently?

Example: Playroom

- Agent has an eye, a hand, and a visual marker
- Actions
  - move eye to hand
  - move eye to marker
  - move eye N, S, E, or W
  - move eye to random object
  - move hand to eye
  - move hand to marker
  - move marker to eye
  - move marker to hand
  - If both eye and hand are on object: turn on light, push ball. etc.

Playroom (cont.)

- Dynamics
  - Switch controls room lights
  - Bell rings and moves one square if ball hits it
  - Press blue/red block turns music on/off
  - Lights have to be on to see colors
  - Monkey cries out if bell and music both sound in dark room
- Salient events: changes in light and sound intensity

Extrinsic reward: Make monkey cry out

- Using primitive actions:
  - Move eye to switch
  - Move hand to eye
  - Turn lights on
  - Move eye to blue block
  - Move hand to eye
  - Turn music on
  - Move eye to switch
  - Move hand to eye
  - Turn light off
  - Move eye to bell
  - Move marker to eye
  - Move eye to ball
  - Move hand to ball
  - Kick ball
- Using skills
  - Turn lights on
  - Turn music on
  - Turn lights off
  - Ring bell

Intrinsic Motivation in Playroom

- What skills should the agent learn?
  - Those that achieve the salient events: Turn-light-on, turn-music-on, make-monkey-cry, etc. All are access skills.
- How can an agent learn these skills efficiently?
  - Augment external reward with “intrinsic” reward generated by each salient event
  - Intrinsic reward is proportional to the error in prediction of that event according to the option model for that event (“surprise”)

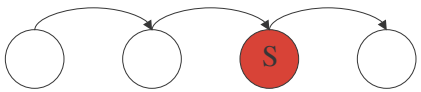
Implementation of Intrinsic Reward

Intrinsic reward = degree of surprise

Of salient stimuli only

(changes in light and sound intensity)

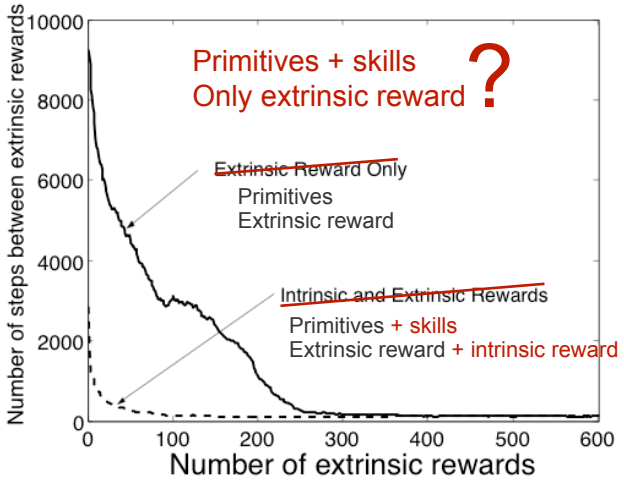
$$r_i = \tau\,[1 - P^o(s_{t+1} \mid s_t)]$$



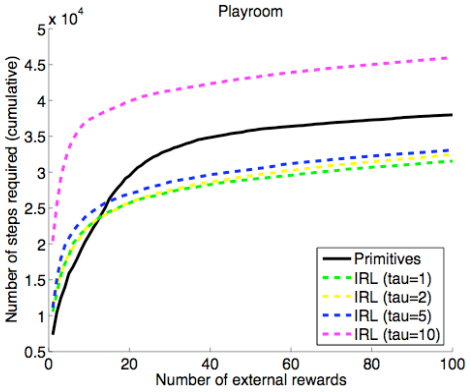
Implementation Details

- Upon first occurrence of salient event: create an option, its pseudo-reward function and initialize:
  - Initiation set
  - Policy
  - Termination condition
  - Option model
- All options and option models updated all the time using intra-option learning (using pseudo-rewards)
- Intrinsic reward added to extrinsic reward, if present, to influence behavior

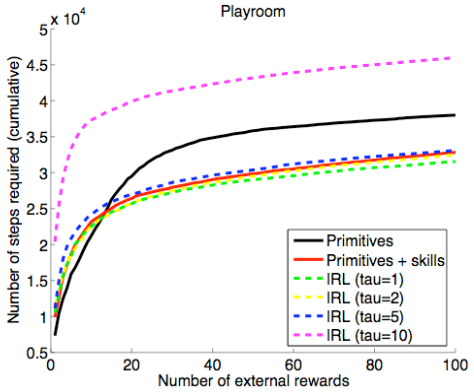
Learning to Make the Monkey Cry Out



A More Informative Experiment

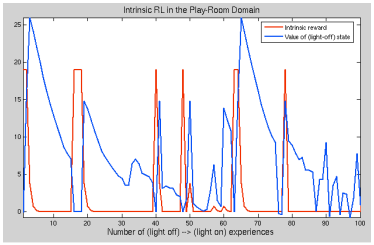


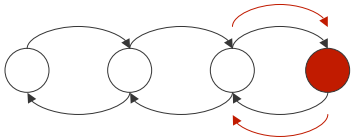
A More Informative Experiment



Behavior of the Algorithm

- Too persistent
- Too local (does not propogate well)
- Will forever chase unpredictable events





IMRL Objective

Open-ended learning via acquisition of skill hierarchies

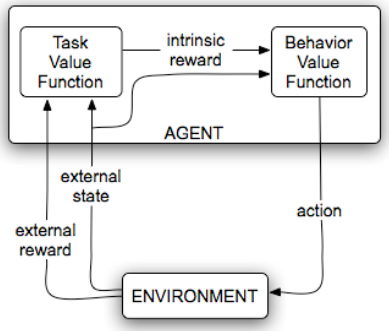
- What skills should the agent learn?
- How can an agent learn these skills efficiently?

An intrinsic reward mechanism for efficient exploration.
Şimşek & Barto, ICML 2006.

Efficient Exploration

How should a reinforcement learning agent act if its sole purpose is to efficiently learn an optimal policy for later use?

Approach

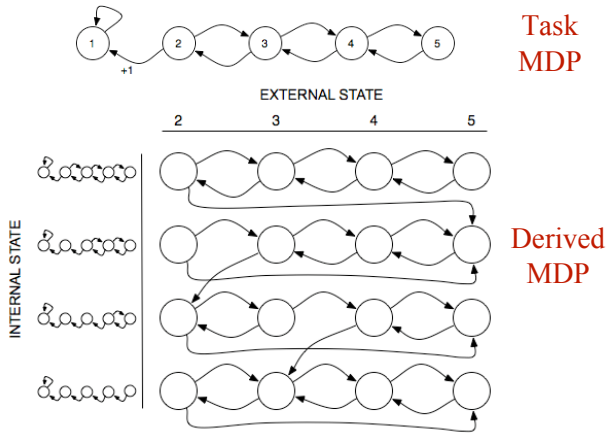


$$r_t^I = \sum_{s \in S} [V_t^T(s) - V_{t-1}^T(s)] - p$$

The Optimal Exploration Problem

- Devise an action selection mechanism such that the policy learned at the end of a given number of training experiences maximizes policy value
- Formulate this problem as an MDP (the derived MDP)
  - State = (external state, internal state)

The Optimal Exploration Problem



Intrinsic Reward

- The reward function of the derived MDP is the difference in policy value of successive states

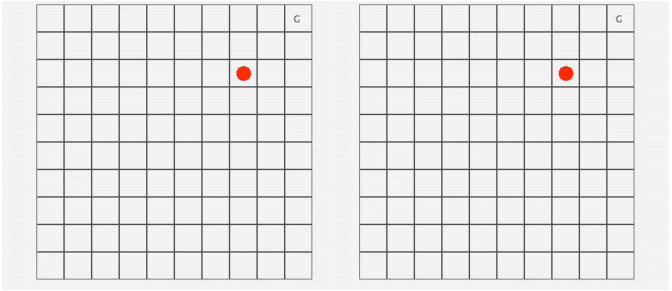
$$V(\pi) = \sum_{s \in S} D(s) V^{\pi}(s)$$

- We estimate this assuming that changes in the agent's value function reflect changes in the actual value of the agent's current policy

$$r_t^I = \sum_{s \in S} [V_t^T(s) - V_{t-1}^T(s)] - p$$

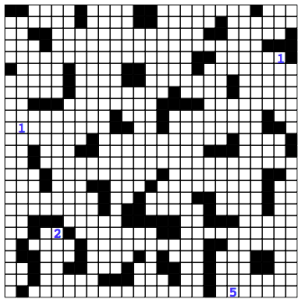
p: a small action penalty

Behavior

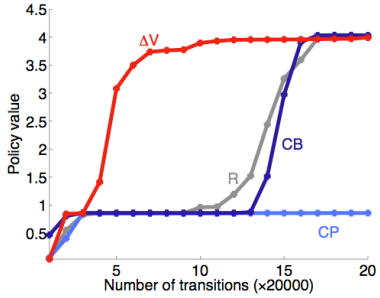


Counter-Based ΔV

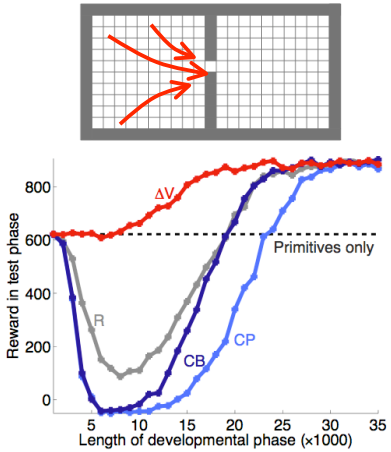
Performance in a Maze Problem



Performance in a Maze Problem



Utility in Skill Acquisition



Some Open Questions

- When should the exploration period terminate?
- What if there are multiple skills to be acquired?
  - Should intrinsic rewards be combined?
  - Or should the agent pursue exploration in service of a single skill at a time?

IMRL Objective

Open-ended learning via acquisition of skill hierarchies

- What skills should the agent learn?
- How can an agent learn these skills efficiently?

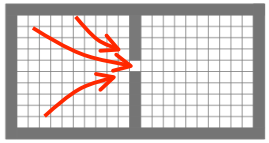
Access skills

Şimşek & Barto, ICML 2004

Şimşek, Wolfe & Barto, ICML 2005

Access Skills

- Access states: allow the agent to transition to a part of the state space that is otherwise unavailable or difficult to reach from its current region

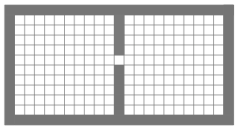


- Doorways, airports, elevators
- Completion of a subtask
- Building a new tool

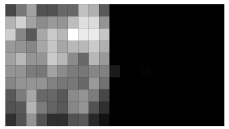
- Closely related to subgoals of
  - McGovern & Barto (2001)
  - Menache et al. (2002)
  - Mannor et al. (2004)

How Do We Identify Access States?

1. Using Relative Novelty



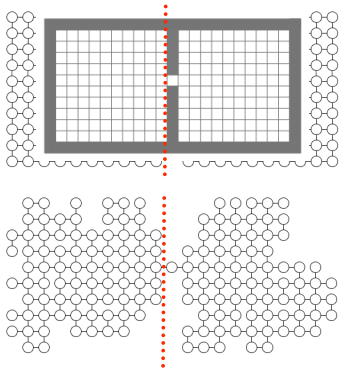
Intuition: Access states are likely to introduce short-term novelty.



… 29 36 32 48 33 16 16 4 1 1 1 1 1 1

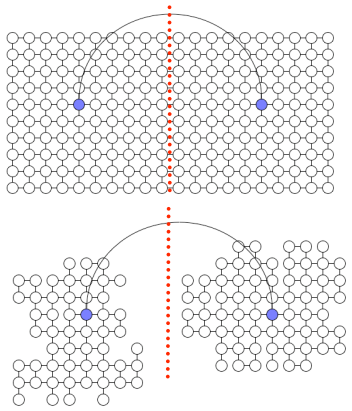
How Do We Identify Access States?

2. By local graph partitioning

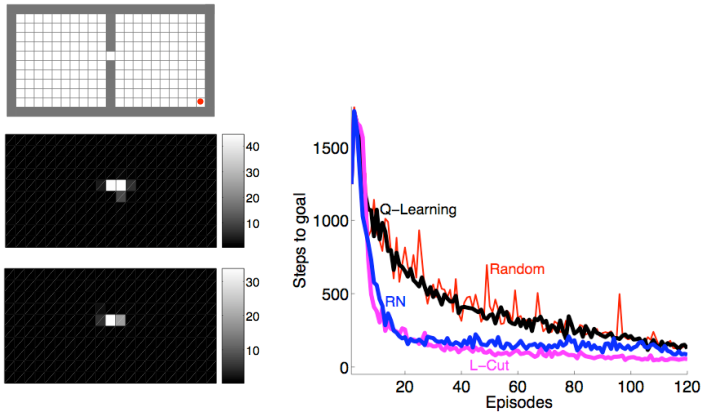


How Do We Identify Access States?

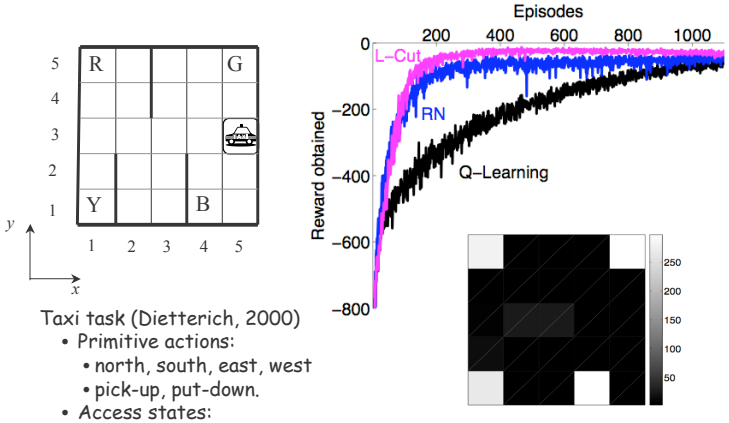
2. By local graph partitioning



Utility of Access Skills



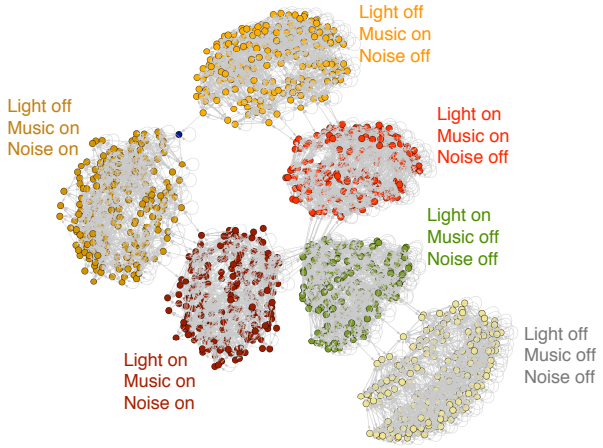
Utility of Access Skills (cont.)



Taxi task (Dietterich, 2000)

- Primitive actions:
  - north, south, east, west
  - pick-up, put-down.
- Access states:
  - picking up the passenger
  - navigational

Playroom State Transition Graph



This Lecture

- Barto, Singh & Chentanez. Intrinsically motivated learning of hierarchical collections of skills. ICDL 2004.
- Singh, Barto & Chentanez. Intrinsically motivated reinforcement learning. NIPS 2005.
- Barto and Şimşek, Intrinsic motivation for reinforcement learning systems. In Proceedings of the Thirteenth Yale Workshop on Adaptive and Learning Systems (2005).
- Şimşek & Barto. An intrinsic reward mechanism for efficient exploration. ICML 2006.

This Lecture (cont.)

- Şimşek, Wolf, & Barto, Identifying useful subgoals in reinforcement learning by local graph partitioning. ICML 2005.
- Şimşek & Barto, Using relative novelty to identify useful temporal abstractions in reinforcement learning. ICML 2004.
- Slides from Andy Barto's recent talks
- Discussions with other members of the "intrinsic" group at UMass: George Konidaris, Andrew Stout, Pippin Wolfe, Chris Vigorito