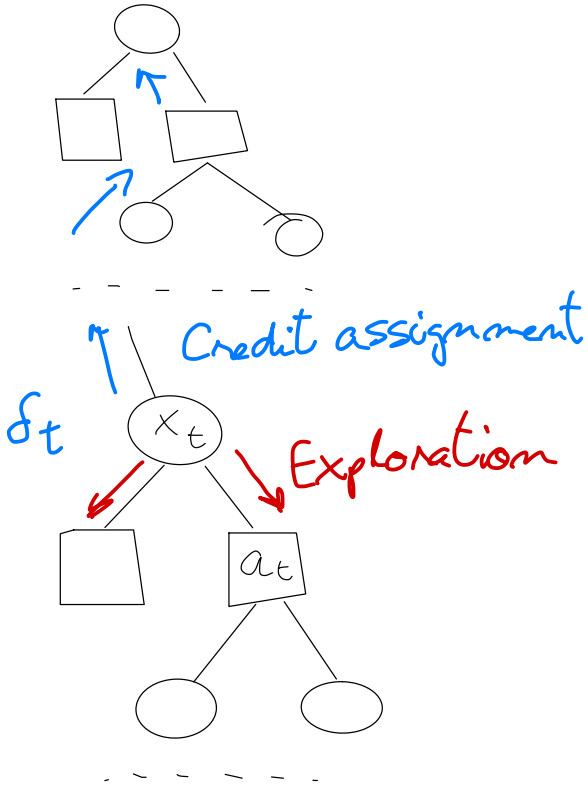
Continual (Never-Ending) Reinforcement Learning - Part 2

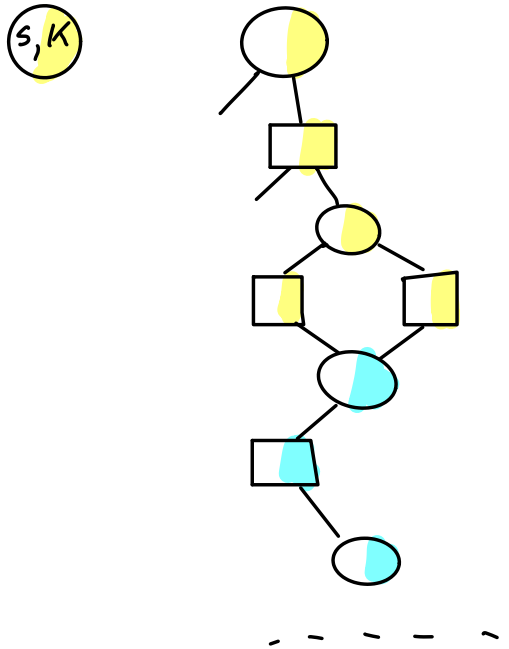
COMP579, Lecture 24

Giving the agent control over its own thinking process



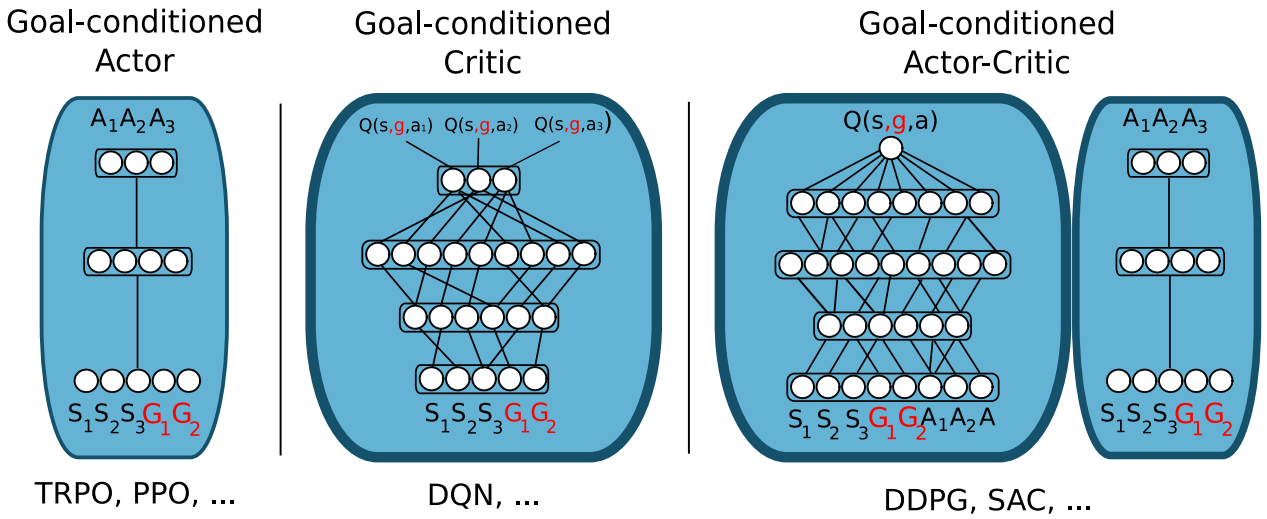
The agent can reshape the tree to help its decision making process

Imagining multiple tasks or goals



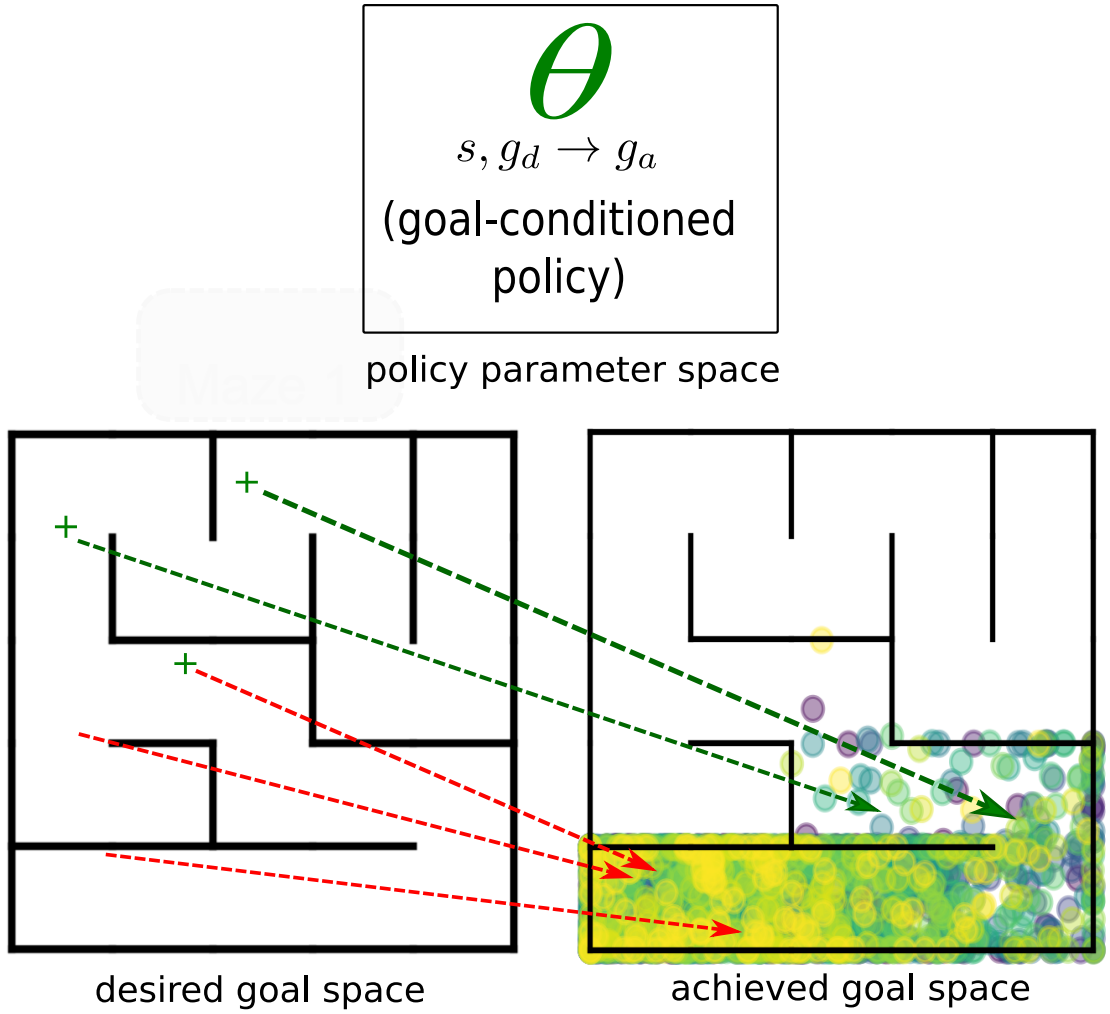
- $x_t = \langle s_t, k_t \rangle$ where k_t is some way to specify a task at time t and s_t is the state inside the task
- *Task structure exists only in the agent's head, in order to make credit assignment and action choice easier*

Goal-conditioned RL



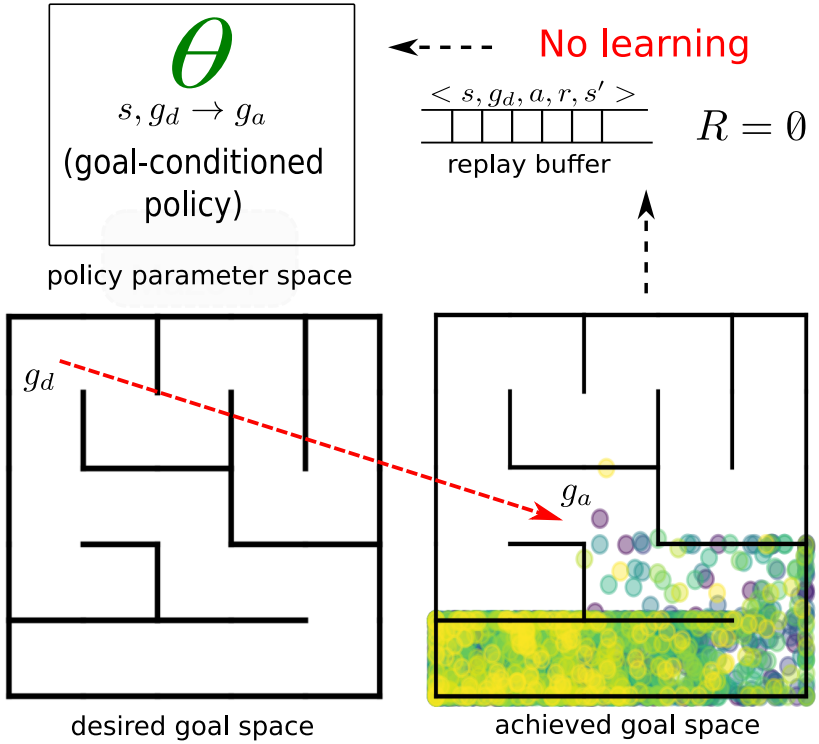
- Agent's goal becomes an input for RL
- Goal can be a state, or an embedding
- A function tells us whether goal has been achieved or not
- Can potentially learn to generalize over multiple goals!

Desired vs achieved goals



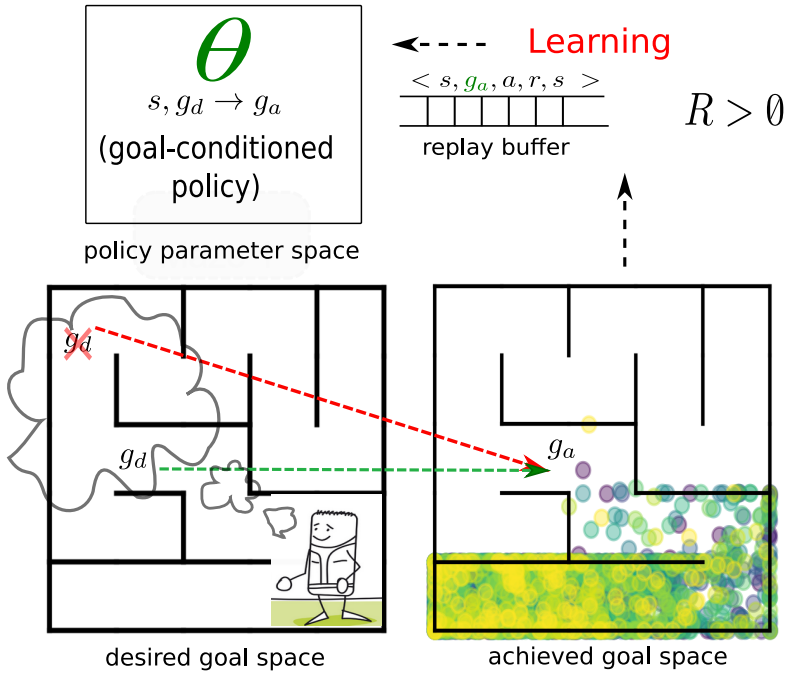
- Goal may be very hard to achieve!
- But maybe we can figure out what *was* achieved and learn from that?

Goal relabelling



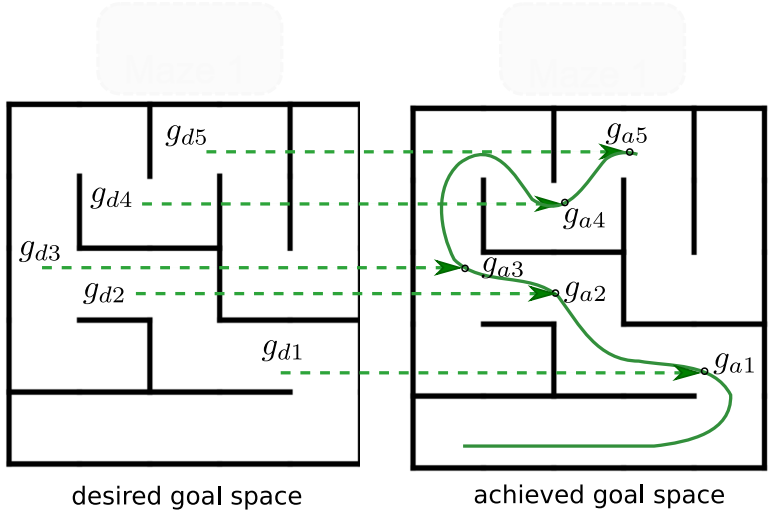
- The agent targets a desired goal g_d
- The policy π_θ produces an achieved goal g_a
- The trajectory is stored (it may produce no reward)

Hindsight experience replay (HER)



- The agent pretends it was targetting g_a
- HER relabels the stored trajectory with g_a instead of g_d
- This propagates value in the (state, action) space through generalization
- And the agent competence increases over unseen goals

When goals are states

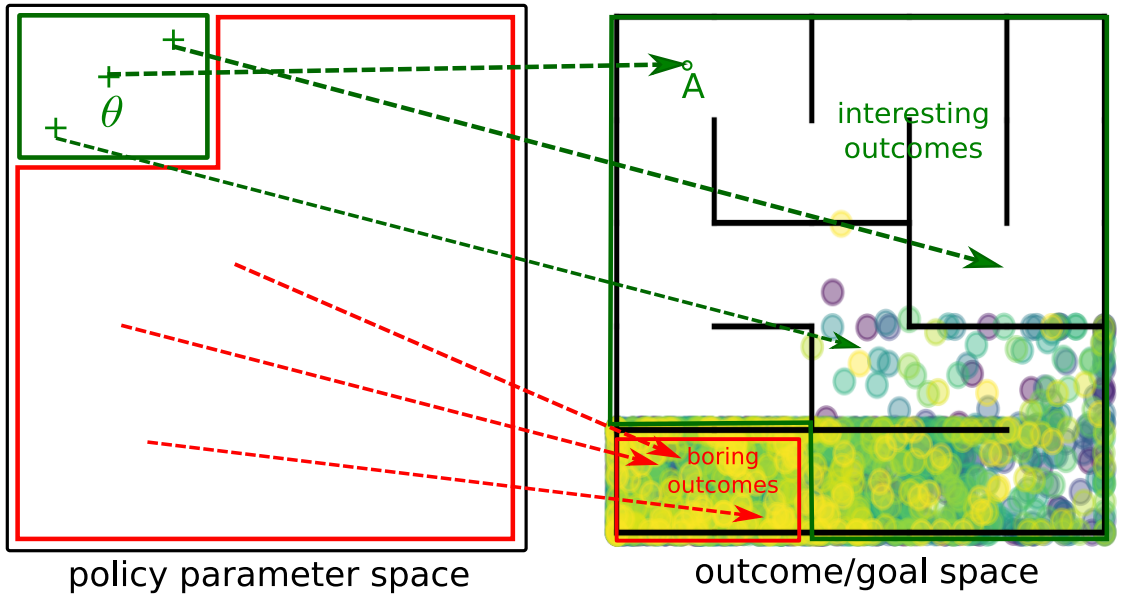


- If goal space = state space, HER may set as goal any state along the trajectory
- Trade-off between replaying more and trying more new actions (risk of over-fitting to replay)
- Variants of HER: CHER (Curriculum + HER), DHER (dynamic goals), MCHER (multi-criteria)...

From HER to curriculum learning

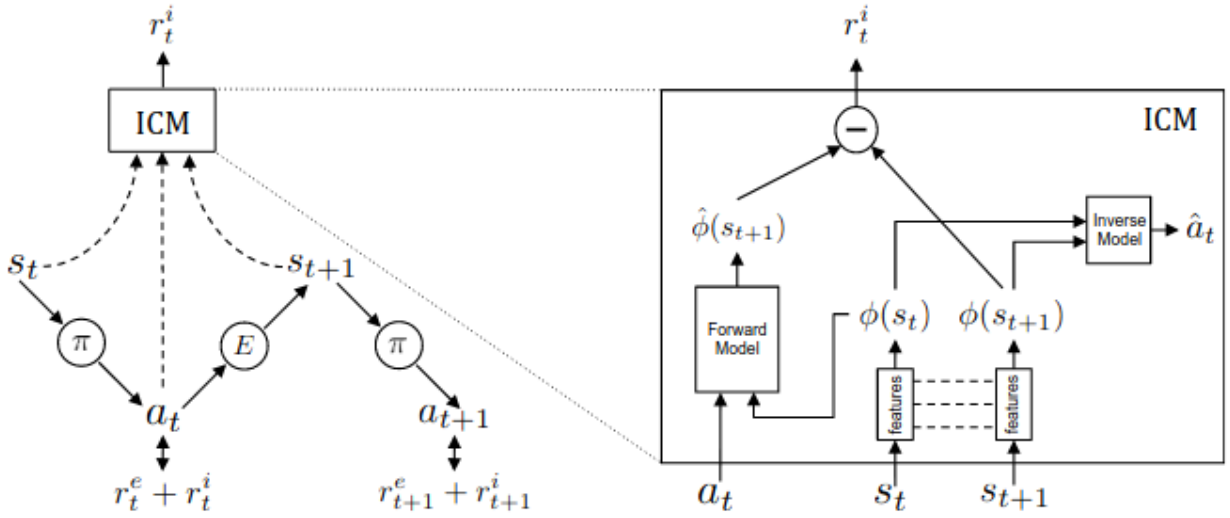
- HER relabels achieved goals as fake desired goals
- It does NOT tell you which goals the agent should strive to achieve
- *Curriculum learning* selects desired goals; can be combined with HER
- How should we select goals:
  - Coverage/Exploration: pick goals that let you get to more places
  - Performance: pick goals that you can actually accomplish

Goal exploration process



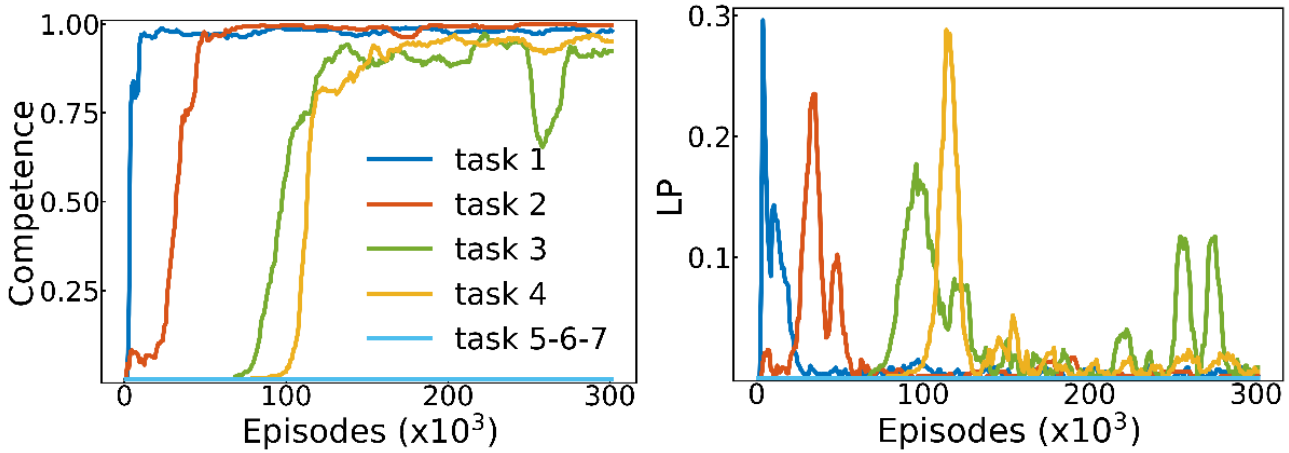
- Very often, few parameter vectors map to interesting achieved goals
- The GEP algorithm favors sampling these interesting achieved goals
  - Sample a random desired goal
  - Find the nearest achieved goal A' and select the corresponding θ
  - Perturb θ into θ' and get a new achieved goal A'
- Results in sampling "at the border" of currently achieved goals

Using surprise/novelty for curriculum



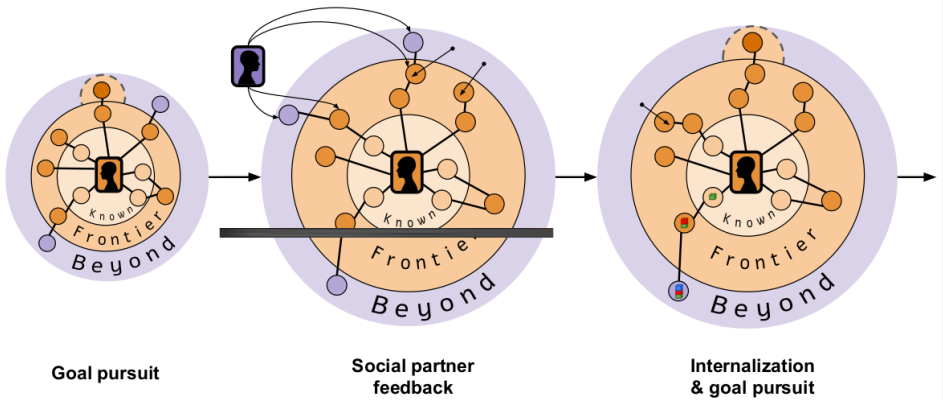
- Intrinsic motivation: reward states for which the forward model predicts poorly
- Target goals corresponding to rewarded states
- Results in visiting poorly visited states
- White noise problem: the agent may get stuck on what it cannot predict

Using learning progress

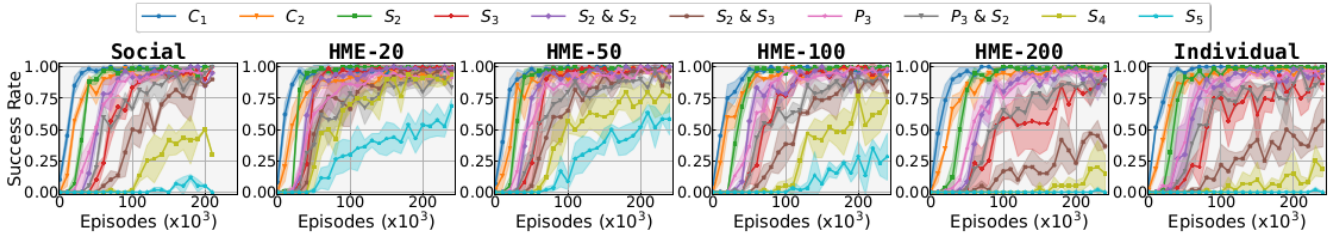


- Competence raises in order of growing difficulty
- LP generates more training in order of growing difficulty
- Catastrophic forgetting generates new training

Goal space can be learned



- Key property: the tutor has a model of the learner's knowledge
- It proposes Frontier + Beyond goals (HME)
- The learner internalizes tutor's goals, it can train on them and on its own goals

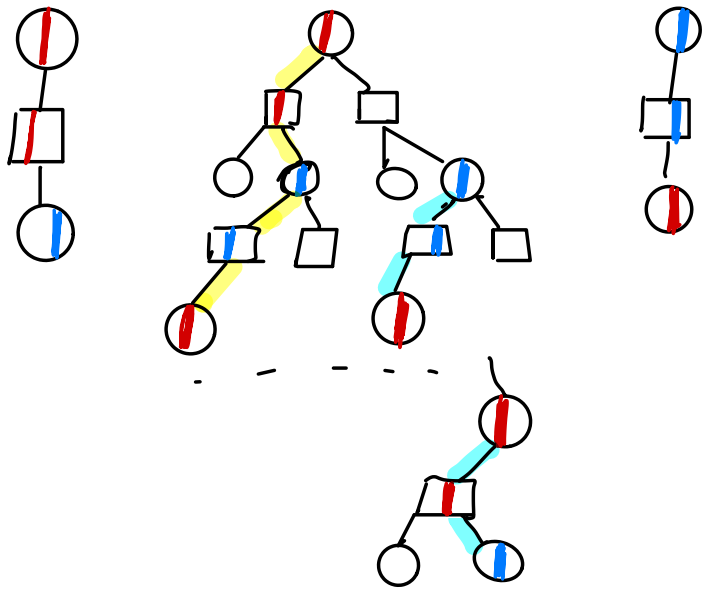


- Guided play is more efficient than learning on its own and full guidance

Summary so far

- HER allows the agent to learn from what it accomplishes
- Curriculum learning provides a good succession of goals
- Goals can help exploration and/or learning

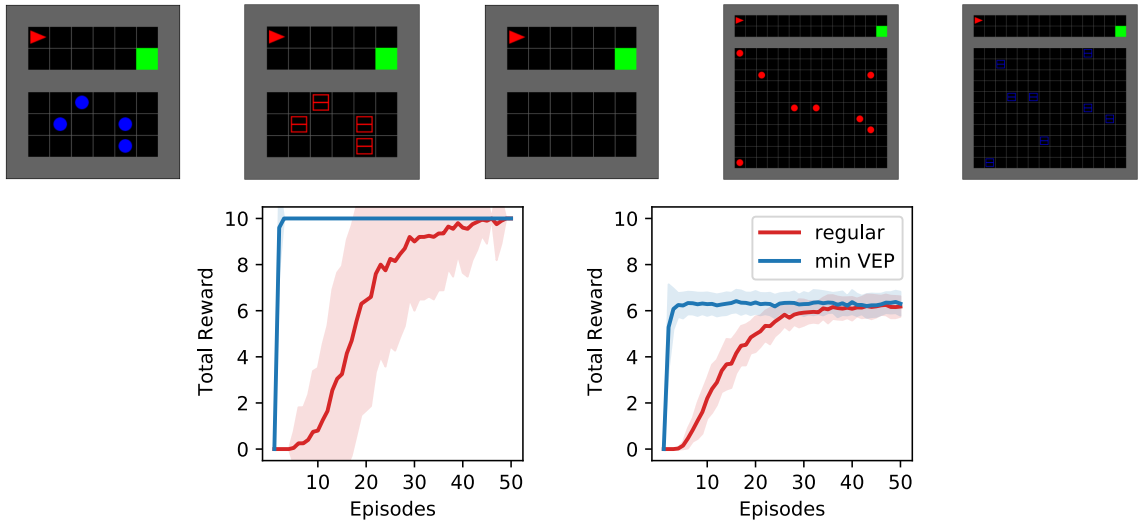
Partial Models



- Predict only specific features / cumulants
- Apply only in specific circumstances

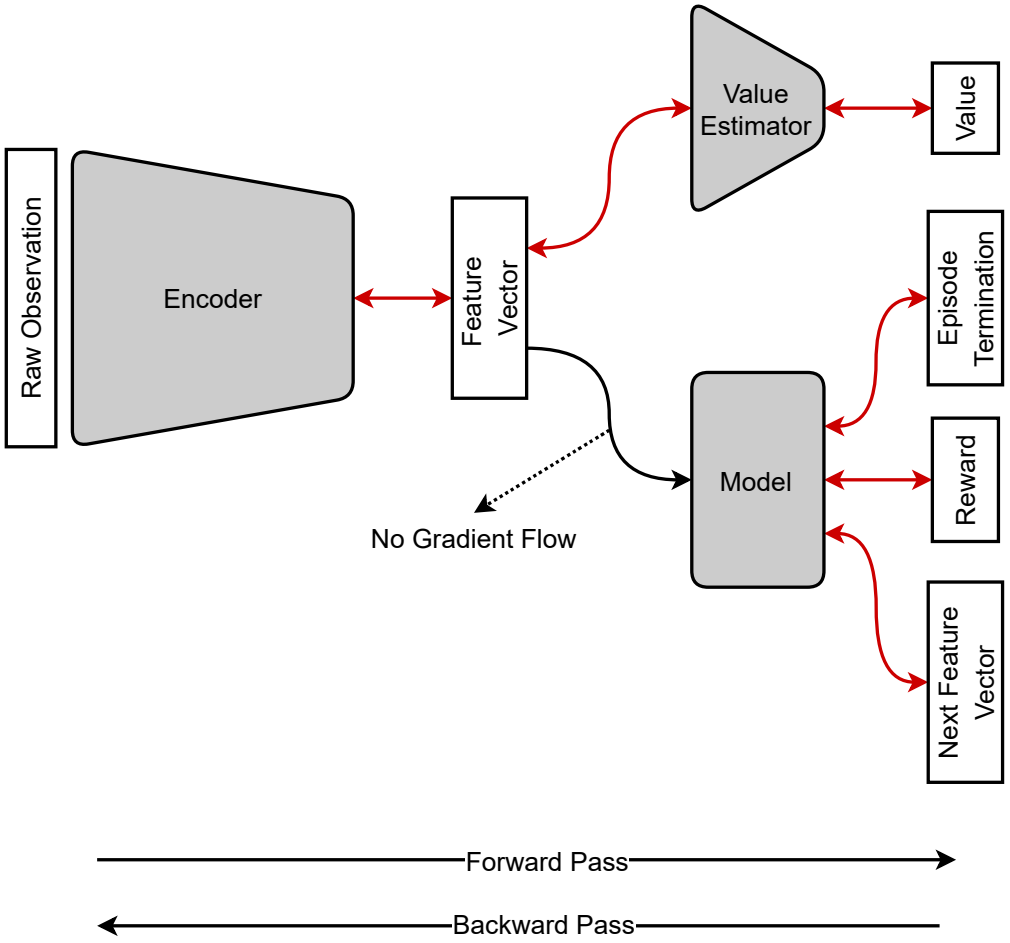
Partial Value-Equivalent Models

- Model only predicts a subset of features (not the entire observation) (cf. Talvitie & Singh, 2008)
- Goal is to obtain correct value estimates (a la MuZero), not to maximize likelihood
- Example: minigrid

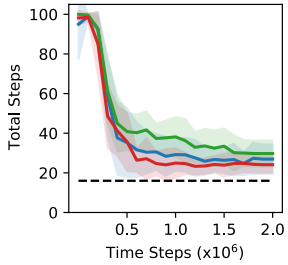


Partial models drastically improve solution speed! (cf Alver & Precup. 2023)

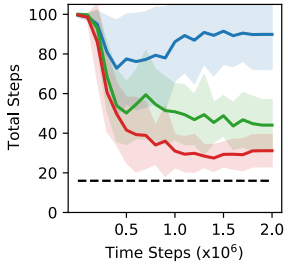
Learning Partial Value-Equivalent Models



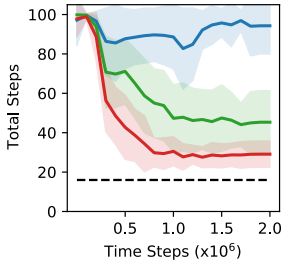
Learned partial models improve generalization



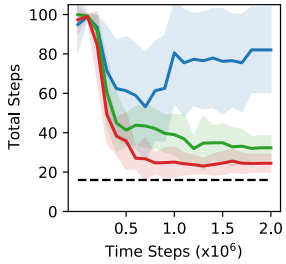
(c) 16x16 BlueBalls-R



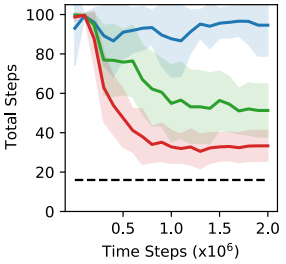
(d) 16x16 NoObstacles-R



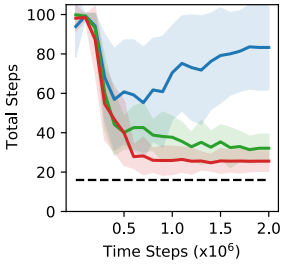
(i) 16x16 RedBalls-R



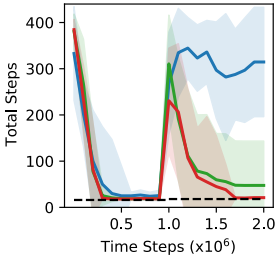
(j) 16x16 GreyBalls-R



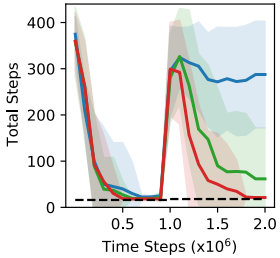
(k) 16x16 RedBoxes-R



(l) 16x16 GreyBoxes-R



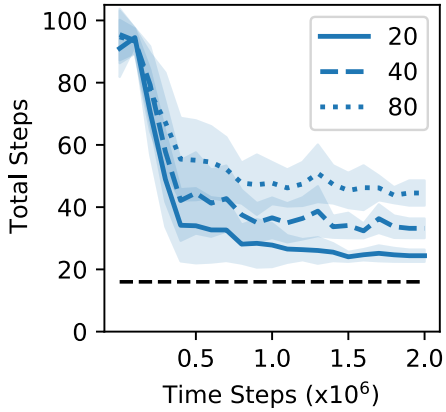
(e) 16x16 RedBalls-K



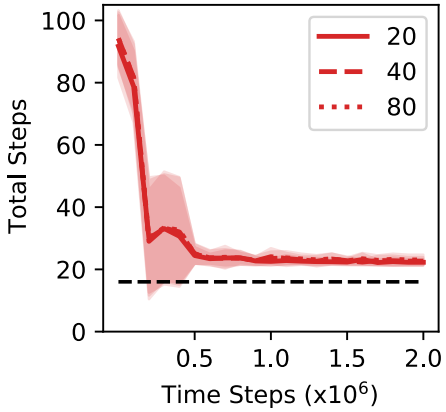
(f) 16x16 GreyBalls-K

Blue: Regular, Green: Value-Equivalent, Red: Value equivalent + models

Partial models allow deeper planning



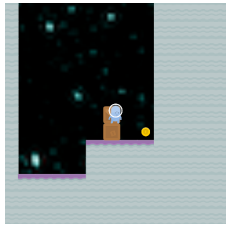
(g) The A_{REG} agent



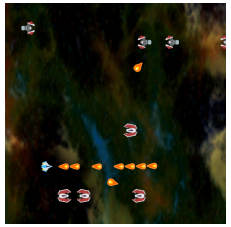
(h) The A_{VES+ME} agent

- Regular models (left) lead to worse performance when doing more planning steps, due to error propagation
- Partial models have better error propagation properties (see Alver & Precup. 2023, for details on the theory)

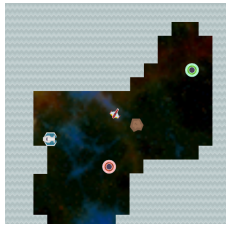
Scaling up: ProcGen



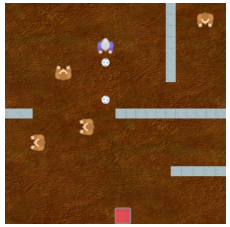
(a) CoinRun



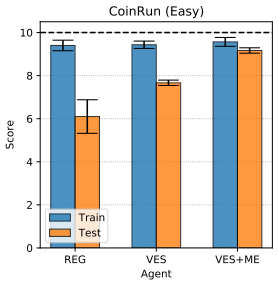
(b) StarPilot



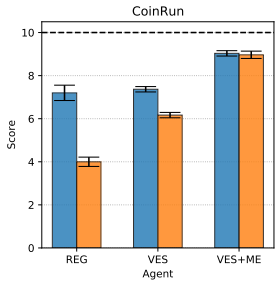
(c) CaveFlyer



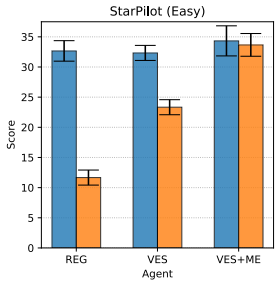
(d) DodgeBall



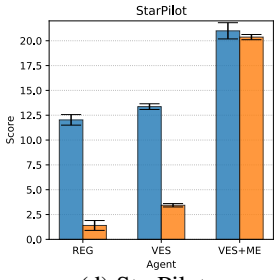
(a) CoinRun (Easy)



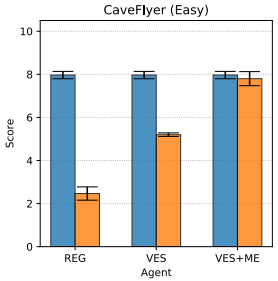
(b) CoinRun



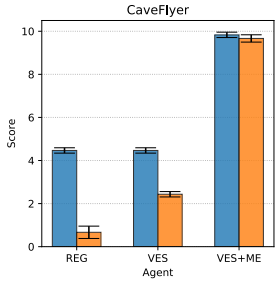
(c) StarPilot (Easy)



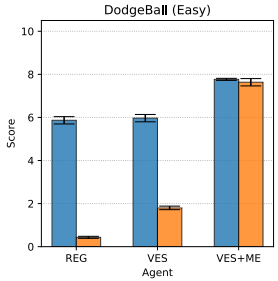
(d) StarPilot



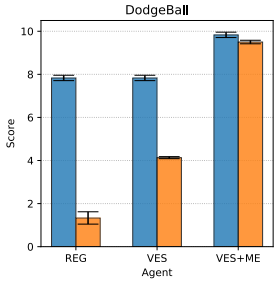
(e) CaveFlyer (Easy)



(f) CaveFlyer



(g) DodgeBall (Easy)



(h) DodgeBall

Partial models improve generalization!

Conclusion

- An agent that is much smaller than its environment will be pressured to find structure on its current trajectory: continually, online, not striving for optimality but for gradual improvement.
- The structure it builds drives two important computations: exploration decisions and credit assignment
- While agent implementations often link these two computations, they can and perhaps should be more decoupled
- Many of the ingredients needed already exist (information-directed sampling, GVFs, options, affordances, partial models)

Some challenges

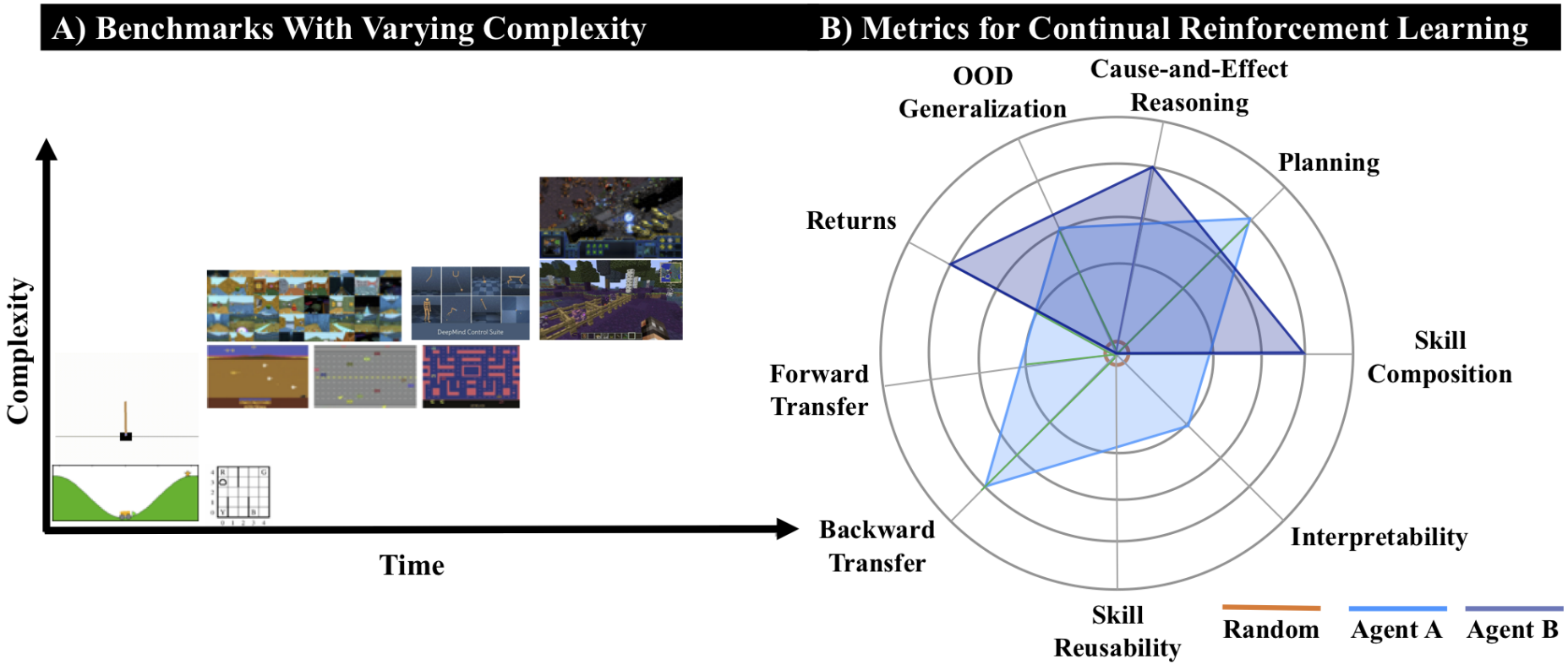
- From a theoretical point of view, we need to formalize the problem further

  Moving away from usual stationarity/recurrence assumptions to fully transient agents

- From an empirical point of view, we should think of the appropriate environments and metrics

  Reconsider reward sparsity as a mark of interesting problems?

Evaluation for continual RL



Cf. Khetarpal, Riemer, Rish and Precup, 2022