Drowning in Degrees of Freedom: One Agent for Every Task

Steven James
School of Computer Science and Applied Mathematics, University of the Witwatersrand
Machine Intelligence and Neural Discovery (MIND) Institute
steven.james@wits.ac.za

Abstract

We describe a research programme aimed at the construction of a single, generally intelligent agent—one competent across all tasks rather than skilled at any one. Such an agent must act in the real world through a rich sensorimotor interface, with sensors and actuators general enough for any task it may face. But that same richness makes any single task nearly impossible to learn directly. We argue that the resolution lies in abstraction, and survey our work towards this goal: discovering skills from experience, deriving compact models from those skills, and transferring both to new tasks. Skills, in particular, can be composed so that new tasks are solved with no further learning. Together, these threads work towards a single agent capable, in principle, of every task its body allows.

1 Introduction

The field of artificial intelligence is premised on the idea that it is possible to build a machine as intelligent as a human [McCarthy *et al.*, 2006]. While a definition of intelligence remains elusive, the goal has always been *general* intelligence, for what makes humans intelligent is not our skill at any one particular task, but our competence across a wide array of them. Intelligence is not a repertoire but a capacity: the ability, when confronted with something new, to learn it. A machine that plays chess today and learns to drive tomorrow—*that* is intelligence.

Historically, the field has rarely attacked this problem directly, focusing instead on narrow (albeit impressive) grand challenges [Chollet, 2019]. It does so through a *preframing* of the problem, so pervasive that it is often invisible. In classical planning, the world arrives already described: a set of symbols stipulated to be complete, and operators specifying how actions transform them [Ghallab *et al.*, 2004]. In reinforcement learning (RL), the agent is handed a decision process whose state and action spaces have already been fixed [Sutton and Barto, 2018]. In both traditions, the hard work—deciding what the world consists of—is done by a human before the agent ever acts. Such preframing is inconsequential if the goal is narrow AI, but fatal if the goal is generality. AlphaGo's action space is the set of legal Go moves [Silver *et al.*, 2016]; it can never drive a car, not for want of intelligence, but because driving was removed from its world before it took its first action. Whatever is excluded from the frame is excluded from the agent's possible competence.

A generally capable agent must therefore be *capable* in the true sense of the word. There can be no preframing on the part of a human; the agent's sensors must be rich enough to perceive anything relevant and its actuators fine-grained enough to accomplish anything useful [Konidaris, 2019]. Generality is a property of the interface between an agent and its environment, before it is a property of the algorithm.

However, this results in an inescapable dilemma. The more general the agent, the more tasks it can accomplish in principle, but the harder it becomes to solve any one of them in practice. Consider a robot whose innate actions are raw motor commands, tasked with making a cup of coffee. The solution is a sequence of tens or hundreds of thousands of primitive actions, rewarded only at the end, with decisions made on the basis of raw pixel input. We posit that no *tabula rasa* learner can possibly solve such a task from trial-and-error feedback. Not even large amounts of data, as is common in the deep learning era, can help, not least because its motors wear out long before the billionth data point arrives. The generality that makes every task possible is the same generality that makes each task intractable.

Despite this, we argue there can be no retreat. Any preframing precludes some tasks, and so the hard version of the problem is the only version worth solving. This paper describes our research programme that takes this predicament seriously.

2 The Anatomy of a General Agent

Our argument so far concerns agents in general, but its natural home is the physical world—the one environment that arrives unframed. An agent embodied in the real world is a robot, and a robot's instantiation defines the envelope of tasks it can, in principle, solve. Ideally, this envelope should be as wide as possible. The framework itself, though, must be agnostic to the embodiment: a robot that cannot locomote is confined to its tabletop, but should be able to accomplish everything possible within it. Humanoids are attractive precisely because their envelope is the widest we know: a machine with a human's body can, within the bounds of its physics, attempt everything a human attempts.

A general agent acting within an environment is modelled by RL's agent-environment loop, and more formally as a Markov decision process (MDP) $\langle \mathcal{S}, \mathcal{A}, P, r \rangle$: the states $\mathcal{S}$ the world can occupy, the actions $\mathcal{A}$ available to the agent, the dynamics P governing how actions change states, and the reward r specifying the task. In nearly every application, $\mathcal{S}$ and $\mathcal{A}$ are gifts: a designer chooses the state features, fixes the action set, and the Markov property holds by construction because a human made it hold. Our robot receives no such gifts. Its state space is whatever its sensors emit, its action space is whatever its motors afford, and nothing guarantees that the former is compact or that the latter is convenient. Three problems immediately arise.

Observations are too rich. The robot perceives through high-dimensional sensors, and reasoning directly in that space is hopeless. The agent requires compact latent representations that retain only what decision-making requires: the long-studied problem of *state abstraction* [Li *et al.*, 2006].

Actions are too small. With raw motor control, even a task as simple as picking up a cup might be a plan thousands of actions long, and credit assignment does not survive such horizons. The agent must shorten its horizon by acting over temporally extended *skills* rather than primitive controls [Sutton *et al.*, 1999].

Tasks are too many. The set of tasks a general agent may face is unbounded, so learning each from scratch is ruled out by the sample argument above. Knowledge must accumulate: a robot that has learned chess and is asked to play checkers should pick it up quickly. Children face the same predicament, yet acquire higher-order concepts from a handful of examples and generalise them far beyond the setting in which they were learned [Gelman and Markman, 1987; Werchan *et al.*, 2015]. Every abstraction the agent builds must, therefore, *transfer* [Taylor and Stone, 2009].

We claim that these threads should not be pursued independently. Representations learned in isolation risk being incompatible, producing states that ignore what the skills need, or skills that ignore what a planner can use. Figure 1 illustrates our proposed approach, where an agent first lifts its raw actions into *skills* (Section 3); its skills then determine the compact *models* it plans in (Section 4); and both are constructed so that they *transfer* to new tasks or environments (Section 5).

3 From Actions to Skills

Section 2 argued that credit assignment cannot survive horizons thousands of primitives long. Episodes in standard RL tasks typically run to a few hundred steps, and tasks that exceed this budget have become challenges in their own right: escaping the first room of *Montezuma's Revenge* resisted agents for years, since it requires a precise sequence of hundreds of primitive actions with only sparse reward feedback [Bellemare *et al.*, 2013]. Robotics benchmarks are scoped similarly, asking for *reach*, *push*, or *pick-and-place* [Yu *et al.*, 2019]; no benchmark yet expects a robot, even in simulation, to succeed in making breakfast. Longer behaviours can be obtained through human demonstration, from classical learning-from-demonstration [Argall *et al.*, 2009] to vision-language-action models trained on large-scale teleoperation

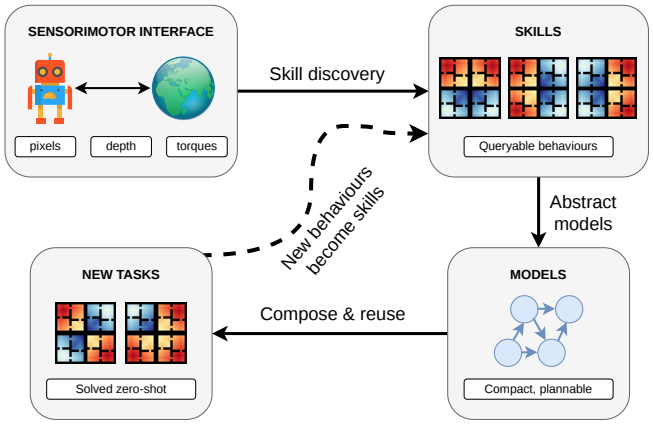


Figure 1: The agent lifts raw sensorimotor experience into skills, derives from those skills the models it plans in, and composes the skills to solve new tasks. Composed behaviours may re-enter as skills at the next level.

[Black *et al.*, 2024], but the task decomposition is supplied by the person rather than discovered by the agent.

3.1 Skill Acquisition

The standard formalism for temporal abstraction is the *option* [Sutton *et al.*, 1999]: an initiation set, a policy, and a termination condition, packaging extended behaviour as a single decision. A rich literature investigates learning options from experience autonomously [Klissarov *et al.*, 2025]. Early work identified bottleneck states and built options to reach them [McGovern and Barto, 2001; Şimşek and Barto, 2004]; graph-theoretic descendants find the same structure spectrally [Machado *et al.*, 2017]. Option-critic architectures learn options end-to-end from task reward [Bacon *et al.*, 2017]; skill chaining grows options backwards from a goal [Konidaris and Barto, 2009]; intrinsic objectives yield diverse behaviours with no reward signal at all, in a flat repertoire [Eysenbach *et al.*, 2019]; and segmentation methods recover skills from trajectories annotated with actions or curated into demonstrations [Ranchod *et al.*, 2015; Kipf *et al.*, 2019].

Our recent work continues this line with the supervision removed entirely: Harvey *et al.* [2026] segment unlabelled observation sequences into skills and induce a hierarchy over them by grammar induction, so that recurring observation segments become low-level skills and recurring skill chains become higher-level ones. In pixel-based environments including full, unmodified *Minecraft*, the discovered hierarchies are semantically meaningful and accelerate downstream learning.

3.2 Skill Representation

Discovery answers where skills come from, but a separate question is what a skill should *be*. Typically, option discovery methods produce skills that are not inherently multitask: each is a policy, and a policy is an answer to exactly one question. Its goal is not an input the agent can vary, but a constant fixed into the skill when it was learned. Portable options [Konidaris and Barto, 2007] relax part of this restriction by defining skills over agent-centric features, so that a skill

learned in one environment transfers to another—the option that makes coffee in one kitchen then makes coffee in another. However, portability only widens where a skill applies, not what it is for; the coffee option says nothing about how to make toast. A general agent needs the complementary generality: skills that carry not only across environments, but also across goals.

Our work represents a skill as an object that the agent can query about goals other than the one it happens to be pursuing. Building on general and goal-conditioned value functions [Sutton *et al.*, 2011; Schaul *et al.*, 2015], which make the goal an input to the skill rather than a constant inside it, we propose the *world value function* (WVF) [Nangue Tasse *et al.*, 2022b]. Given a goal-reaching task $M = \langle \mathcal{S}, \mathcal{A}, P, r \rangle$, with candidate goals $\mathcal{G} \subseteq \mathcal{S}$, we extend the reward function to penalise the agent for terminating at any goal other than the one it pursues:

$$\bar{r}(s, g, a) = \begin{cases} \bar{r}_{\min} & \text{if } s \in \mathcal{G} \text{ and } s \neq g, \\ r(s, a) & \text{otherwise,} \end{cases} \tag{1}$$

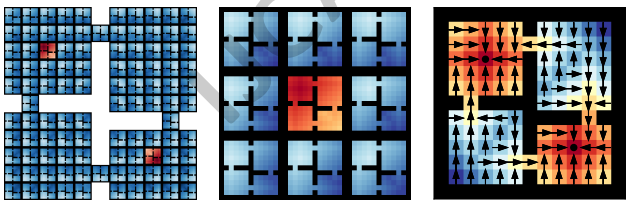
where $\bar{r}_{\min}$ is a large negative penalty. The *world value function* $\bar{Q} : \mathcal{S} \times \mathcal{G} \times \mathcal{A} \to \mathbb{R}$ is then the optimal value function of this extended task,

$$\bar{Q}(s, g, a) = \bar{r}(s, g, a) + \mathbb{E}_{s'}\Big[\max_{a'} \bar{Q}(s', g, a')\Big],$$

estimating the value of each action in pursuit of *each* goal. The penalty in Equation 1 is what compels mastery. The agent is rewarded only for achieving goals deliberately, so $\bar{Q}$ must learn how to reach every goal, not merely which goals lie on the way to the rewarding ones. The skill then answers queries directly. For any goal $g \in \mathcal{G}$, a policy can be extracted as

$$\pi_g(s) \in \arg\max_{a \in \mathcal{A}} \bar{Q}(s, g, a),$$

and the task's own optimal values are recovered by maximising over goals, $Q^*(s, a) = \max_g \bar{Q}(s, g, a)$. When $\mathcal{G} = \mathcal{S}$, the agent has *mastered* its environment: the WVF encodes how to achieve any reachable state. Figure 2 illustrates this in the Four Rooms domain, where a trained WVF encodes knowledge of how to reach all states, even undesirable ones.



(a) (b) (c)

Figure 2: (a) Learned WVF with positive rewards for reaching the middle of the top-left or bottom-right rooms. (b) Close-up view of the WVF for the "top-left" goal. Note that in adjacent cells, an agent can reach other states by following the value gradient. (c) Inferred values and policy for solving the current task, extracted by maximising over goals. Reproduced from Nangue Tasse *et al.* [2022b].

4 From Observations to Models

In practice, a robot cannot reason in the high-dimensional space its sensors provide. It needs a compact representation that retains only what decision-making requires. The dominant approach is to learn such representations through prediction. World models learn latent states in which imagination replaces experience [Hafner *et al.*, 2025], and joint-embedding architectures predict in representation space rather than pixel space [LeCun, 2022; Bardes *et al.*, 2024]. However, such representations serve prediction, not control. They retain whatever helps forecast the next observation, regardless of its relevance to acting.

We argue that state abstraction cannot be separated from action abstraction. The two have largely been studied apart: states aggregated so as to preserve values or dynamics, as in bisimulation and its relaxations [Li *et al.*, 2006; Dean and Givan, 1997], and options learned independently of any representation [Sutton *et al.*, 1999]. Where the two are combined, the state abstraction typically leads: states are partitioned first, and options are constructed to travel between the partitions [Abel *et al.*, 2020; Jothimurugan *et al.*, 2021].

We contend the dependence runs the other way. An option is constrained by the dynamics of the world and the body of the agent; a state abstraction is something the agent may hypothesise freely, so the constrained should dictate the flexible. An agent that cannot open a door has no use for a symbol denoting whether the door is open. A representation is worth maintaining exactly when some skill's success depends on it.

Konidaris *et al.* [2018] formalise this intuition: given a set of skills, the symbols required to plan with them are fully determined—they are the preconditions and effects of the skills themselves, and can therefore be learned rather than designed. Our work demonstrates that such symbols can be acquired autonomously from the agent's own experience, grounded in its egocentric observations and attached to the objects it encounters [James *et al.*, 2020; James *et al.*, 2022]; their most consequential property, portability, is the subject of Section 5. Further, since the models are symbolic, they are open to inspection and formal analysis. On physical robots, we show that learned symbols support the automatic encoding, verification, and repair of reactive tasks [Pacheck *et al.*, 2023].

Symbols are not the only form an abstract model can take. Skill graphs point in the same direction, using the structure of chained options as an abstract state space for planning [Bagaria *et al.*, 2021]. Our recent work derives the abstraction directly from the Bellman equation. Ahmetoglu *et al.* [2025] show that, for a given set of skills, abstract states should represent distributions over ground states, and characterise the conditions under which the resulting abstract MDP is Markov and approximately model-preserving. This is further extended to factored actions and applied in visual domains, including *Montezuma's Revenge*. Together with conditions under which planning in the abstract model incurs only bounded value loss in the original [Rodriguez-Sanchez and Konidaris, 2024], these results give the agent something prediction alone cannot: a guarantee that its abstract plans are grounded in the real world.

5 From One Task to Every Task

The abstractions of the previous two sections are learned within a task, but the general agent faces an unbounded stream of them, and Section 2 has already ruled out relearning from scratch. When the environment changes, what persists is the agent itself—its body travels with it everywhere—and the kinds of objects it repeatedly encounters. Abstractions anchored to these therefore apply in the new setting immediately. When instead the task changes within the same environment, what persists is the world: behaviours learned there can be kept and, as we will see, *composed*, so that new tasks are solved without being learned at all.

5.1 State Abstraction Transfer

On the observation side, this is why our symbols are egocentric and object-centric. James *et al.* [2020] learn symbolic representations defined relative to the agent rather than to the world. As the agent is the one constant across every task it will ever face, symbols acquired once apply immediately in the next task. James *et al.* [2022] attach symbols to object types instead, so that knowledge acquired about one object extends to every other of its kind. Neither representation is complete: the agent must perform some task-specific learning, much as a person entering an unfamiliar kitchen knows what kettles and cupboards afford but must look around to find them. The division between what persists and what must be relearned is made explicit by Rosen *et al.* [2023]. In that work, object-level representations are learned once and carried everywhere, while each new environment demands only a lightweight semantic map recording where the objects are.

5.2 Action Abstraction Transfer

On the action side, the world value functions of Section 3 were built to transfer, and they can be composed to produce new behaviours. Compositionality has long been recognised in special cases [Todorov, 2009; Barreto *et al.*, 2017], and van Niekerk *et al.* [2019] show that, in entropy-regularised RL, optimal value functions for disjunctive task combinations are obtained zero-shot from the value functions of the constituent tasks. We extend this to a full Boolean algebra over tasks [Nangue Tasse *et al.*, 2020]. For tasks M_1 and M_2,

$$\bar{Q}_{M_1 \vee M_2} = \max\left(\bar{Q}_{M_1}, \bar{Q}_{M_2}\right),$$
$$\bar{Q}_{M_1 \wedge M_2} = \min\left(\bar{Q}_{M_1}, \bar{Q}_{M_2}\right),$$
$$\bar{Q}_{\neg M_1} = \left(\bar{Q}_{M_{\sup}} + \bar{Q}_{M_{\inf}}\right) - \bar{Q}_{M_1},$$

where $M_{\sup}$ rewards every goal and $M_{\inf}$ rewards none. An agent with a small basis of learned skills therefore solves any logical combination of them immediately, and a basis of n skills yields doubly exponentially many expressible tasks. In the lifelong setting, each newly learned skill combinatorially enlarges the set of tasks solvable for free [Nangue Tasse *et al.*, 2022a], and the results generalise beyond Boolean algebras to arbitrary lattices over tasks [Nangue Tasse *et al.*, 2025].

Logical composition handles what the agent should achieve, but not in what order. Temporal structure is typically specified through reward machines [Toro Icarte *et al.*, 2018] or linear temporal logic (LTL) [Camacho *et al.*, 2019], but these specify the problem, not its solution. Our *skill machines* [Nangue Tasse *et al.*, 2024] address this by compiling the specification into a machine whose states index compositions of the agent's learned skills. Tasks expressed in regular fragments of LTL can then be provably satisfied zero-shot, and because the specification is formal, the behaviour can be verified before execution. Moreover, the specification need not be written in logic at all: we translate natural-language instructions into Boolean compositions of the agent's skills, using in-context learning refined by the agent's own RL feedback [Cohen *et al.*, 2024], so that tasks specified in English are solved by composition.

6 Future Work and Conclusion

This paper has presented skills, models, and transfer as a pipeline, but an autonomous agent must run them as a cycle. Discovered skills induce a model, and planning in that model composes the skills into higher-level behaviours, which themselves are skills demanding more abstract models still. Exploration belongs in the loop too: the agent should seek out precisely the experience its current abstractions cannot explain. To date, we have built each component in isolation. The clearest test of the framework is an agent that runs the cycle on its own, continually.

Several questions remain open. No single formalism fits every task. PDDL serves deliberative problems well, but robot football demands something else entirely, so the general agent must select or construct the class of abstraction a problem requires. Among other agents, this may extend to abstractions of others' goals and beliefs. Abstract states should factorise into components that can be learned and recombined independently, but this is harder than it appears. In the supervised learning setting, the representations that emerge are often governed by accidents of the data, rather than by any structure the agent cares about [Michlo *et al.*, 2023]. The deeper goal is factors that are causal—components the agent can intervene on, not merely observe [Schölkopf *et al.*, 2021].

Even a good model may not be enough. In our *MinePlanner* benchmark [Hill *et al.*, 2024], an agent equipped with an object-level model of a large *Minecraft* world still fails to plan, because the model represents everything in it. The agent must decide, for each task, which objects are relevant. Finally, safety cannot be an afterthought for an agent intended to do everything. We have shown that unsafe behaviour can be deterred through the reward alone, by learning the smallest penalty that provably suffices [Nangue Tasse *et al.*, 2023]. This guarantee holds for a single skill, and our agent composes them, so whether safety holds is an open question.

We began with a tension that cannot be resolved: an agent general enough to attempt every task is, by that same generality, unable to learn any single one directly. The programme surveyed here is one response—learn the skills, derive the models, and transfer both—so that the cost of a rich sensorimotor space is paid once and amortised over every task that follows. The destination has not changed since the field began: one agent, every task. An agent that constructs its own abstractions need not drown in its degrees of freedom, and our results suggest that such an agent is no longer out of reach.

References

[Abel *et al.*, 2020] David Abel, Nathan Umbanhowar, Khimya Khetarpal, Dilip Arumugam, Doina Precup, and Michael L. Littman. Value preserving state-action abstractions. In *International Conference on Artificial Intelligence and Statistics*, 2020.

[Ahmetoglu *et al.*, 2025] Alper Ahmetoglu, Steven James, Cameron Allen, Sam Lobel, David Abel, and George Konidaris. Skill-driven neurosymbolic state abstractions. In *Advances in Neural Information Processing Systems*, 2025.

[Argall *et al.*, 2009] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5):469–483, 2009.

[Bacon *et al.*, 2017] Pierre-Luc Bacon, Jean Harb, and Doina Precup. The option-critic architecture. In *AAAI Conference on Artificial Intelligence*, 2017.

[Bagaria *et al.*, 2021] Akhil Bagaria, Jason Senthil, and George Konidaris. Skill discovery for exploration and planning using deep skill graphs. In *International Conference on Machine Learning*, 2021.

[Bardes *et al.*, 2024] Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *Transactions on Machine Learning Research*, 2024.

[Barreto *et al.*, 2017] André Barreto, Will Dabney, Rémi Munos, Jonathan J. Hunt, Tom Schaul, Hado van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. In *Advances in Neural Information Processing Systems*, 2017.

[Bellemare *et al.*, 2013] Marc G. Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

[Black *et al.*, 2024] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[Camacho *et al.*, 2019] Alberto Camacho, Rodrigo Toro Icarte, Toryn Q. Klassen, Richard Valenzano, and Sheila A. McIlraith. LTL and beyond: Formal languages for reward function specification in reinforcement learning. In *International Joint Conference on Artificial Intelligence*, 2019.

[Chollet, 2019] François Chollet. On the measure of intelligence. *arXiv preprint arXiv:1911.01547*, 2019.

[Cohen *et al.*, 2024] Vanya Cohen, Geraud Nangue Tasse, Nakul Gopalan, Steven James, Matthew Gombolay, Ray Mooney, and Benjamin Rosman. Compositional instruction following with language models and reinforcement learning. *Transactions on Machine Learning Research*, 2024.

[Dean and Givan, 1997] Thomas Dean and Robert Givan. Model minimization in Markov decision processes. In *AAAI Conference on Artificial Intelligence*, 1997.

[Eysenbach *et al.*, 2019] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *International Conference on Learning Representations*, 2019.

[Gelman and Markman, 1987] Susan A. Gelman and Ellen M. Markman. Young children's inductions from natural kinds: The role of categories and appearances. *Child Development*, 58(6):1532–1541, 1987.

[Ghallab *et al.*, 2004] Malik Ghallab, Dana Nau, and Paolo Traverso. *Automated Planning: theory and practice*. Elsevier, 2004.

[Hafner *et al.*, 2025] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025.

[Harvey *et al.*, 2026] Damion Harvey, Geraud Nangue Tasse, Benjamin Rosman, Branden Ingram, and Steven James. Unsupervised hierarchical skill discovery. In *International Conference on Machine Learning*, 2026.

[Hill *et al.*, 2024] William Hill, Ireton Liu, Anita De Mello Koch, Damion Harvey, Nishanth Kumar, George Konidaris, and Steven James. MinePlanner: A benchmark for long-horizon planning in large Minecraft worlds. In *6th ICAPS Workshop on the International Planning Competition*, 2024.

[James *et al.*, 2020] Steven James, Benjamin Rosman, and George Konidaris. Learning portable representations for high-level planning. In *International Conference on Machine Learning*, 2020.

[James *et al.*, 2022] Steven James, Benjamin Rosman, and George Konidaris. Autonomous learning of object-centric abstractions for high-level planning. In *International Conference on Learning Representations*, 2022.

[Jothimurugan *et al.*, 2021] Kishor Jothimurugan, Osbert Bastani, and Rajeev Alur. Abstract value iteration for hierarchical reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, 2021.

[Kipf *et al.*, 2019] Thomas Kipf, Yujia Li, Hanjun Dai, Vinicius Zambaldi, Alvaro Sanchez-Gonzalez, Edward Grefenstette, Pushmeet Kohli, and Peter Battaglia. CompILE: Compositional imitation learning and execution. In *International Conference on Machine Learning*, 2019.

[Klissarov *et al.*, 2025] Martin Klissarov, Akhil Bagaria, Ziyan Luo, George Konidaris, Doina Precup, and Marlos Machado. Discovering temporal structure: An overview of hierarchical reinforcement learning. *arXiv preprint arXiv:2506.14045*, 2025.

[Konidaris and Barto, 2007] George Konidaris and Andrew Barto. Building portable options: Skill transfer in reinforcement learning. In *International Joint Conference on Artificial Intelligence*, 2007.

[Konidaris and Barto, 2009] George Konidaris and Andrew Barto. Skill discovery in continuous reinforcement learning domains using skill chaining. In *Advances in Neural Information Processing Systems*, 2009.

[Konidaris *et al.*, 2018] George Konidaris, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. From skills to symbols: Learning symbolic representations for abstract high-level planning. *Journal of Artificial Intelligence Research*, 61:215–289, 2018.

[Konidaris, 2019] George Konidaris. On the necessity of abstraction. *Current Opinion in Behavioral Sciences*, 29:1–7, 2019.

[LeCun, 2022] Yann LeCun. A path towards autonomous machine intelligence. *OpenReview preprint*, 2022.

[Li *et al.*, 2006] Lihong Li, Thomas J. Walsh, and Michael L. Littman. Towards a unified theory of state abstraction for MDPs. In *International Symposium on Artificial Intelligence and Mathematics*, 2006.

[Machado *et al.*, 2017] Marlos C. Machado, Marc G. Bellemare, and Michael Bowling. A Laplacian framework for option discovery in reinforcement learning. In *International Conference on Machine Learning*, 2017.

[McCarthy *et al.*, 2006] John McCarthy, Marvin L Minsky, Nathaniel Rochester, and Claude E Shannon. A proposal for the Dartmouth summer research project on artificial intelligence, August 31, 1955. *AI Magazine*, 27(4):12–14, 2006.

[McGovern and Barto, 2001] Amy McGovern and Andrew Barto. Automatic discovery of subgoals in reinforcement learning using diverse density. In *International Conference on Machine Learning*, 2001.

[Michlo *et al.*, 2023] Nathan Michlo, Richard Klein, and Steven James. Overlooked implications of the reconstruction loss for VAE disentanglement. In *International Joint Conference on Artificial Intelligence*, 2023.

[Nangue Tasse *et al.*, 2020] Geraud Nangue Tasse, Steven James, and Benjamin Rosman. A Boolean task algebra for reinforcement learning. In *Advances in Neural Information Processing Systems*, 2020.

[Nangue Tasse *et al.*, 2022a] Geraud Nangue Tasse, Steven James, and Benjamin Rosman. Generalisation in lifelong reinforcement learning through logical composition. In *International Conference on Learning Representations*, 2022.

[Nangue Tasse *et al.*, 2022b] Geraud Nangue Tasse, Steven James, and Benjamin Rosman. World value functions: Knowledge representation for multitask reinforcement learning. In *The 5th Multi-disciplinary Conference on Reinforcement Learning and Decision Making*, 2022.

[Nangue Tasse *et al.*, 2023] Geraud Nangue Tasse, Tamlin Love, Mark Nemecek, Steven James, and Benjamin Rosman. ROSARL: Reward-only safe reinforcement learning. *arXiv preprint arXiv:2306.00035*, 2023.

[Nangue Tasse *et al.*, 2024] Geraud Nangue Tasse, Devon Jarvis, Steven James, and Benjamin Rosman. Skill machines: Temporal logic skill composition in reinforcement learning. In *International Conference on Learning Representations*, 2024.

[Nangue Tasse *et al.*, 2025] Geraud Nangue Tasse, Steven James, and Benjamin Rosman. Composition and zero-shot transfer with lattice structures in reinforcement learning. *Journal of Artificial Intelligence Research*, 82:2325–2388, 2025.

[Pacheck *et al.*, 2023] Adam Pacheck, Steven James, George Konidaris, and Hadas Kress-Gazit. Automatic encoding and repair of reactive high-level tasks with learned abstract representations. *The International Journal of Robotics Research*, 42(4–5):263–288, 2023.

[Ranchod *et al.*, 2015] Pravesh Ranchod, Benjamin Rosman, and George Konidaris. Nonparametric Bayesian reward segmentation for skill discovery using inverse reinforcement learning. In *International Conference on Intelligent Robots and Systems*, 2015.

[Rodriguez-Sanchez and Konidaris, 2024] Rafael Rodriguez-Sanchez and George Konidaris. Learning abstract world models for value-preserving planning with options. *Reinforcement Learning Journal*, 2024.

[Rosen *et al.*, 2023] Eric Rosen, Steven James, Sergio Orozco, Vedant Gupta, Max Merlin, Stefanie Tellex, and George Konidaris. Synthesizing navigation abstractions for planning with portable manipulation skills. In *Conference on Robot Learning*, 2023.

[Schaul *et al.*, 2015] Tom Schaul, Daniel Horgan, Karol Gregor, and David Silver. Universal value function approximators. In *International Conference on Machine Learning*, 2015.

[Schölkopf *et al.*, 2021] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

[Silver *et al.*, 2016] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[Şimşek and Barto, 2004] Özgür Şimşek and Andrew Barto. Using relative novelty to identify useful temporal abstractions in reinforcement learning. In *International Conference on Machine Learning*, 2004.

[Sutton and Barto, 2018] Richard Sutton and Andrew Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2nd edition, 2018.

[Sutton *et al.*, 1999] Richard S. Sutton, Doina Precup, and Satinder Singh. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1–2):181–211, 1999.

[Sutton *et al.*, 2011] Richard S. Sutton, Joseph Modayil, Michael Delp, Thomas Degris, Patrick M. Pilarski, Adam White, and Doina Precup. Horde: A scalable real-time architecture for learning knowledge from unsupervised sensorimotor interaction. In *International Conference on Autonomous Agents and Multiagent Systems*, 2011.

[Taylor and Stone, 2009] Matthew E. Taylor and Peter Stone. Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10:1633–1685, 2009.

[Todorov, 2009] Emanuel Todorov. Compositionality of optimal control laws. In *Advances in Neural Information Processing Systems*, 2009.

[Toro Icarte *et al.*, 2018] Rodrigo Toro Icarte, Toryn Q. Klassen, Richard Valenzano, and Sheila A. McIlraith. Using reward machines for high-level task specification and decomposition in reinforcement learning. In *International Conference on Machine Learning*, 2018.

[van Niekerk *et al.*, 2019] Benjamin van Niekerk, Steven James, Adam Earle, and Benjamin Rosman. Composing value functions in reinforcement learning. In *International Conference on Machine Learning*, 2019.

[Werchan *et al.*, 2015] Denise M. Werchan, Anne G. E. Collins, Michael J. Frank, and Dima Amso. 8-month-old infants spontaneously learn and generalize hierarchical rules. *Psychological Science*, 26(6):805–815, 2015.

[Yu *et al.*, 2019] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-World: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on Robot Learning*, 2019.