World Value Functions: Knowledge Representation for Multitask Reinforcement Learning

Geraud Nangue Tasse, Steven James, Benjamin Rosman
School of Computer Science and Applied Mathematics
University of the Witwatersrand
Johannesburg, South Africa
geraud.nanguetasse1@students.wits.ac.za, {steven.james,benjamin.rosman1}@wits.ac.za

Abstract

An open problem in artificial intelligence is how to learn and represent knowledge that is sufficient for a general agent that needs to solve multiple tasks in a given world. In this work we propose world value functions (WVFs), which are a type of general value function with mastery of the world—they represent not only how to solve a given task, but also how to solve any other goal-reaching task. To achieve this, we equip the agent with an internal goal space defined as all the world states where it experiences a terminal transition—a task outcome. The agent can then modify task rewards to define its own reward function, which provably drives it to learn how to achieve all achievable internal goals, and the value of doing so in the current task. We demonstrate a number of benefits of WVFs. When the agent's internal goal space is the entire state space, we demonstrate that the transition function can be inferred from the learned WVF, which allows the agent to plan using learned value functions. Additionally, we show that for tasks in the same world, a pretrained agent that has learned any WVF can then infer the policy and value function for any new task directly from its rewards. Finally, an important property for long-lived agents is the ability to reuse existing knowledge to solve new tasks. Using WVFs as the knowledge representation for learned tasks, we show that an agent is able to solve their logical combination zero-shot, resulting in a combinatorially increasing number of skills throughout their lifetime.

Keywords: reinforcement learning, multitask, transfer, logic, composition

1 Introduction

The ultimate goal of artificial intelligence is to create general agents capable of solving multiple tasks in the world in the same way that humans are able to. To achieve this, we need a general decision-making framework for agents that interact with a world to solve tasks, and a sufficiently general knowledge representation for what those agents learn. Reinforcement learning (RL) [2] is one such framework that has made several major breakthroughs in recent years, ranging from robotics to board games, but these agents typically are narrowly designed to solve only a single task.

In RL, tasks are specified through a reward function from which the agent receives feedback. Most commonly, an agent represents its knowledge in the form of a value function, which represents the sum of future rewards it expects to receive. However, since the value function is directly tied to one single reward function (and hence task), it is definitionally insufficient for constructing agents capable of solving a wide range of tasks. We seek to overcome this limitation by proposing *world value functions* (WVFs), a knowledge representation that encodes how to solve not just the current task, but any other task, in the world. In the literature, agents with such abilities are said to possess *mastery* [5]. Importantly, WVFs are a form of general value function [3] that can be learned from a single stream of experience like standard value functions—no additional information need be provided to the agent.

In this work, we define the WVF and demonstrate how it can be learned using standard RL algorithms. We then demonstrate several advantages of learning such a representation. In particular, we show that (a) WVFs implicitly encode the dynamics of the world and can be used for model-based RL; (b) having learned a WVF, any new task that an agent encounters can be solved by estimating its reward function, reducing the problem to a supervised learning one; and (c) WVFs can be *composed* with Boolean operators to produce novel and human-interpretable behaviours.

2 World Value Functions

We model the agent's interaction with the world as a Markov Decision Process (MDP) $(\mathcal{S}, \mathcal{A}, P, R)$, where (i) $\mathcal{S}$ is the state space, (ii) $\mathcal{A}$ is the action space, (iii) $P(s, a, s')$ are the transition dynamics of the world, and (iv) R is a reward function bounded by $[R_{\text{MIN}}, R_{\text{MAX}}]$, representing the task the agent needs to solve. Note that in this work, we focus on environments with *deterministic* dynamics, but put no restrictions on their complexity.

The agent's aim is to compute a *policy* π from $\mathcal{S}$ to $\mathcal{A}$ that optimally solves a given task. This is often achieved through a value function that represents the expected return obtained under π starting from state s: $V^\pi(s) = \mathbb{E}^\pi\left[\sum_{t=0}^{\infty} r(s_t, a_t, s_{t+1})\right]$. Similarly, a Q-value function $Q^\pi(s, a)$ represents the expected return obtained by executing a from s, and thereafter following π. The optimal Q-value function is given by $Q^*(s, a) = \max_\pi Q^\pi(s, a)$ for all states s and actions a, and the optimal policy follows by acting greedily with respect to Q^* at each state.

Algorithm 1: Q-learning for WVFs

```
Initialise: WVF $\bar{Q}$, goal buffer $\mathcal{G}$
foreach episode do
    Observe an initial state $s \in \mathcal{S}$ and sample a goal $g \in \mathcal{G}$
    while episode is not done do
        $a \leftarrow \begin{cases} \arg\max_{a \in \mathcal{A}} \bar{Q}(s, g, a) & \text{probability } 1 - \varepsilon \\ \text{a random action} & \text{probability } \varepsilon \end{cases}$
        Take action $a$, observe reward $r$ and next state $s'$
        if $s'$ is absorbing then $\mathcal{G} \leftarrow \mathcal{G} \cup \{s\}$
        for $g' \in \mathcal{G}$ do
            $\bar{r} \leftarrow \bar{R}_{\text{MIN}}$ if $g' \neq s$ and $s \in \mathcal{G}$ else $r$
            $\bar{Q}(s, g', a) \xleftarrow{\alpha} \left[\bar{r} + \max_{a'} \bar{Q}(s', g', a')\right] - \bar{Q}(s, g', a)$
        $s \leftarrow s'$
```

2.1 A New Form of General Value Function

We now introduce *world value functions* (WVFs), which provably encode how to reach all achievable goals in a world with arbitrary task rewards. We first define the internal goal space $\mathcal{G}$ of the agent as all states where it experiences a terminal transition. Importantly, the goal the agent wishes to achieve is not specified by the world, but rather chosen by itself. In order to simultaneously learn how to achieve the overall task goal and the agent's internal goals, the agent can define its own reward function $\bar{R}$, which extends the task rewards R to simply penalise itself for achieving goals it did not intend to:

$$\bar{R}(s, g, a, s') \coloneqq \begin{cases} \bar{R}_{\text{MIN}} & \text{if } g \neq s \text{ and } s' \text{ is absorbing} \\ R(s, a, s') & \text{otherwise,} \end{cases}$$

where $\bar{R}_{\text{MIN}}$ is a large negative penalty that can be derived from the bounds of the reward function [1]. We can think of the penalty $\bar{R}_{\text{MIN}}$ as adding one bit of information to the agent's rewards, which we will later prove is sufficient for the agent to learn how to achieve its internal goals and the value of achieving them in the current task. The overall goal of the agent now is to compute a *world policy* $\bar{\pi} : \mathcal{S} \times \mathcal{G} \to Pr(\mathcal{A})$ that optimally reaches its internal goal states. Given a world policy $\bar{\pi}$, the corresponding WVF is defined as follows:

$$\bar{Q}^{\bar{\pi}}(s, g, a) \coloneqq \mathbb{E}_{s'}^{\bar{\pi}}\left[\bar{R}(s, g, a, s') + \bar{V}^{\bar{\pi}}(s', g)\right], \text{ where } \bar{V}^{\bar{\pi}}(s, g) \coloneqq \mathbb{E}^{\bar{\pi}}\left[\sum_{t=0}^{\infty} \bar{R}(s_t, g, a_t, s_{t+1})\right].$$

Since the WVF satisfies the Bellman equations, $\bar{Q}^*(s,g,a)$ can be learned using any suitable RL algorithm like Q-learning (Algorithm 1). To show that a WVF does still encode knowledge of how to solve the task in which it was learned, we prove that the regular task reward function and value function can be recovered by simply maximising over goals (Theorem 1). The task policy can then be obtained as: $\pi^*(s) \in \arg\max_{a\in\mathcal{A}} \left(\max_{g\in\mathcal{G}} \bar{Q}^*(s,g,a)\right)$.

Theorem 1. *Let $M = (\mathcal{S}, \mathcal{A}, P, R)$ be a task with optimal Q-value function Q^* and optimal world Q-value function $\bar{Q}^*$. Then for all (s,a) in $\mathcal{S} \times \mathcal{A}$, we have (i) $R(s,a) = \max\limits_{g\in\mathcal{G}} \bar{R}(s,g,a)$, and (ii) $Q^*(s,a) = \max\limits_{g\in\mathcal{G}} \bar{Q}^*(s,g,a)$.*

Having established WVFs as a type of task-specific GVF, we prove in Theorem 2 that they do indeed have mastery—that is, they learn how to reach all reachable goal states in the world. We formally define mastery as follows:

Definition 1. *Let $\bar{Q}^*$ be the optimal world Q-value function for a task M. Then $\bar{Q}^*$ has mastery if for all $g \in \mathcal{G}$ reachable from $s \in \mathcal{S} \setminus \mathcal{G}$, there exists an optimal world policy*

$$\bar{\pi}^*(s,g) \in \arg\max_{a\in\mathcal{A}} \bar{Q}^*(s,g,a) \text{ such that } \bar{\pi}^* \in \arg\max_{\bar{\pi}} P_s^{\bar{\pi}}(s_T = g),$$

where $P_s^{\bar{\pi}}(s_T = g)$ is the probability of reaching g from s under a policy $\bar{\pi}$.

Theorem 2. *Let $\bar{Q}^*$ be the optimal world Q-value function for a task M. Then $\bar{Q}^*$ has mastery.*

Finally, if the agent's goal space is the state space ($\mathcal{G} = \mathcal{S}$), then we can estimate the transition probabilities for each $s, a \in \mathcal{S} \times \mathcal{A}$ using only the reward function and optimal WVF. That is, $P(s,a,s')$ for all $s' \in \mathcal{S}$ can be obtained by simply solving for them in the system of Bellman optimality equations given by each goal $g \in \mathcal{S}$: $\bar{Q}^*(s,g,a) = \sum_{s'\in\mathcal{S}} p(s,a,s') \left[\bar{R}(s,g,a,s') + \bar{V}^*(s',g)\right]$. This shows that optimal WVFs implicitly encode the dynamics of the world. In practice, if the learned WVF is sufficiently optimal in a neighbourhood $\mathcal{N}(s)$, then we only need to consider $s', g \in \mathcal{N}(s) \times \mathcal{N}(s)$ if the transition probabilities are known to be non-zero only in $\mathcal{N}(s)$, common in most domains.

Experiment 2.1. *Consider a 4-rooms gridworld where an agent may be required to visit various goal positions. The agent can move in any of the four cardinal directions at each timestep (with reward -0.1), and also has a "done" action that it chooses to terminate at any position (with reward 2 if it is a task goal). We train an agent on the task where it must learn to navigate to the middle of the top-left or bottom-right rooms. Figure 1 (left) shows the learned WVF. The figure is generated by plotting the value functions for every goal position (since $\mathcal{G} = \mathcal{S}$), and displaying it at their respective xy position. Note how the values with respect the "top-left" and "bottom-right" goals are high (red), reflecting the high rewards the agent received for reaching those goals when it was trying to reach them. Figure 1 (middle) shows a close up view of the learned WVF around the "top-left" goal. We can observe from the value gradient of the plots that the WVF does indeed learn how to reach all positions in the gridworld. We can then maximise over goals to obtain the regular value function and policy (Figure 1 (right)).*

Finally, we demonstrate that the transition probabilities can be inferred from the learned WVF. Figures 2 (left) and (middle) respectively show the transitions inferred by solving the Bellman equations with $s', g \in \mathcal{S} \times \mathcal{S}$ and $s', g \in \mathcal{N}(s) \times \mathcal{N}(s)$. For each, we inferred the next state probabilities for taking each cardinal action at the center of each room, and placed the corresponding arrow in the state with highest probability. The red arrows in Figure 2 (left) correspond to incorrectly inferred next states, which is a consequence of the learned WVF not being near optimal at all states for all goals. Figure 2 (middle) shows that in practice, if the WVF is not near-optimal, we can still infer dynamics by using $s', g \in \mathcal{N}(s) \times \mathcal{N}(s)$. Figure 2 (right) shows sample trajectories for following the optimal policy using the inferred transition probabilities. The gray-scale color of each arrow corresponds to the normalised value prediction for that state.

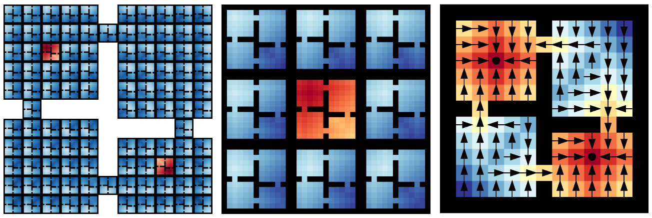


Figure 1: Learned WVF (left), close up of WVF for "top-left" goal (middle), and inferred values and policy (right).

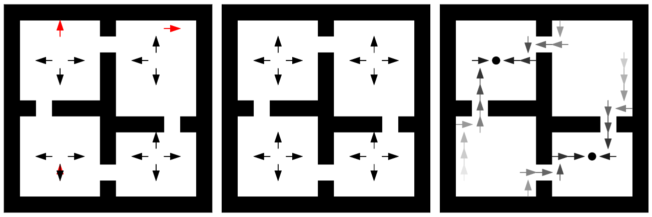


Figure 2: Inferred transitions (left-middle) and imagined rollouts using the learned WVF (right).

3 Benefits of WVFs across tasks in the same world

We now highlight some benefits of WVFs that are a natural result of assuming tasks come from the same world—that is we assume that all tasks share the same state space, action space and dynamics, but differ in their reward functions.

More specifically, we define the world as a background MDP $M_0 = \langle \mathcal{S}_0, \mathcal{A}_0, P_0, R_0 \rangle$ with its own state space, action space, transition dynamics and background reward function. Any individual task M is specified by a task-specific reward function $R^\tau_M(s,a)$ that is non-zero only for transitions entering terminal states. The reward function for the resulting MDP is then simply $R_M(s,a,s') := R_0(s,a,s') + R^\tau_M(s,a)$. We denote the set of all these tasks as $\mathcal{M}$ and the set of their corresponding optimal WVFs as $\bar{\mathcal{Q}}^*$.

Interestingly, this definition means that if tasks come from the same world, then their WVFs share the same world policy—that is, the agent has the same notion of goals and how to reach them no matter the task.

Theorem 3. *Let $\bar{\mathcal{Q}}^*$ be the set of optimal world $\bar{Q}$-value functions with mastery of tasks in $\mathcal{M}$. Then for all $s \neq g \in \mathcal{S} \times \mathcal{G}$,*

$$\bar{\pi}^*(s,g) \in \arg\max_{a\in\mathcal{A}} \bar{Q}^*_{M_1}(s,g,a) \iff \bar{\pi}^*(s,g) \in \arg\max_{a\in\mathcal{A}} \bar{Q}^*_{M_2}(s,g,a)\ \forall M_1, M_2 \in \mathcal{M}.$$

Similarly, if we require that the world policies need to be the same across tasks, then we have that the tasks must come from the same world as defined above. In a sense, this gives us a notion of task invariance (the world does not change with changes in task) implying policy conservation (the world policy is constant across tasks), and vice-versa.

3.1 Zero-shot values and policies from rewards

Since each task $M \in \mathcal{M}$ share the same dynamics (and consequently the same world policy), their corresponding WVFs can be written as $\bar{Q}^*_M(s,g,a) = G^*_{s,g,a} + \bar{R}^\tau_M(s',a')$ for some $s', a' \in \mathcal{S} \times \mathcal{A}$, where $G^*_{s,g,a}$ is a constant across tasks that represents the sum of rewards starting from s and taking action a up until g, but not including the terminal reward. Using this fact, we can show that the optimal value function and policy for any task can be obtained zero-shot from an arbitrary WVF given the task-specific rewards:

Theorem 4. *Let R^τ_M be the given task-specific reward function for a task $M \in \mathcal{M}$, and let $\bar{Q}^* \in \bar{\mathcal{Q}}^*$ be an arbitrary WVF. Define*

$$\tilde{\bar{V}}_M(s,g) := \max_{a\in\mathcal{A}} \bar{Q}^*(s,g,a) + \left(\max_{a\in\mathcal{A}} R^\tau_M(g,a) - \max_{a\in\mathcal{A}} \bar{Q}^*(g,g,a) \right), \text{ the estimated WVF of } M.$$

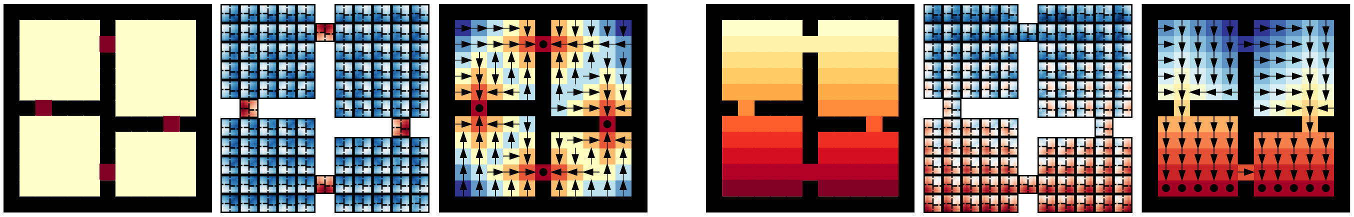
Then,

(i) for all $g \in \mathcal{G}$ reachable from $s \in \mathcal{S}$, $\bar{V}^*_M(s,g) = \tilde{\bar{V}}_M(s,g)$.

(ii) $V^*_M(s) = \max_{g\in\mathcal{G}} \tilde{\bar{V}}(s,g)$, and $\pi^*_M(s) \in \arg\max_{a\in\mathcal{A}} \bar{Q}^*(s, \arg\max_{g\in\mathcal{G}} \tilde{\bar{V}}_M(s,g), a)$.

This has several important implications for transfer learning. For example, one can learn an arbitrary WVF with unsupervised pretraining, then solve any new task by only learning the reward function (from experience or demonstrations).

Experiment 3.1. *Consider the 4-rooms domain introduced in Experiment 2.1 where the agent has learned the WVF for navigating to the top-left or bottom-right rooms. Figure 3 shows the estimated WVFs, values and policy for each task, computed as per Theorem 4 using the given task rewards and the learned WVF for the bottom-left task. Notice how the WVFs look similar to the rewards. This highlights the structural similarity between the task space and value function space which enables the zero-shot inference of task policies from task rewards alone.*



(a) Navigating to the hallways.

(b) Navigating to the bottom of the grid.

Figure 3: Inferring values and policies zero-shot from goal rewards. From left to right on each figure: The given task specific rewards, the inferred WVF using Theorem 4, and the inferred values and policy from maximising over goals.

3.2 Zero-shot logical composition

We recently proposed a framework for agents to apply logical operations—conjunction ($\wedge$), disjunction ($\vee$) and negation ($\neg$)—over the space of tasks and WVFs [1]. We first define these logical operations over tasks and WVFs as follows (we omit the value functions' parameters for readability):

$$R_{M_1 \vee M_2} := \max\{R_{M_1}, R_{M_2}\}; \quad R_{M_1 \wedge M_2} := \min\{R_{M_1}, R_{M_2}\}; \quad R_{\neg M_1} := (R_{\mathcal{M}_{SUP}} + R_{\mathcal{M}_{INF}}) - R_{M_1},$$
$$\bar{Q}^*_{M_1} \vee \bar{Q}^*_{M_2} := \max\{\bar{Q}^*_{M_1}, \bar{Q}^*_{M_2}\}; \quad \bar{Q}^*_{M_1} \wedge \bar{Q}^*_{M_2} := \min\{\bar{Q}^*_{M_1}, \bar{Q}^*_{M_2}\}; \quad \neg\bar{Q}^*_{M_1} := \left(\bar{Q}^*_{SUP} + \bar{Q}^*_{INF}\right) - \bar{Q}^*_{M_1},$$

where $\bar{Q}^*_{SUP}$ is the goal-oriented value function for the maximum task $R_{\mathcal{M}_{SUP}}$ where $r = R_{\text{MAX}}$ for all $\mathcal{G}$. Similarly $\bar{Q}^*_{INF}$ is goal-oriented value function for the minimum task $R_{\mathcal{M}_{INF}}$ where $r = R_{\text{MIN}}$ for all $\mathcal{G}$.

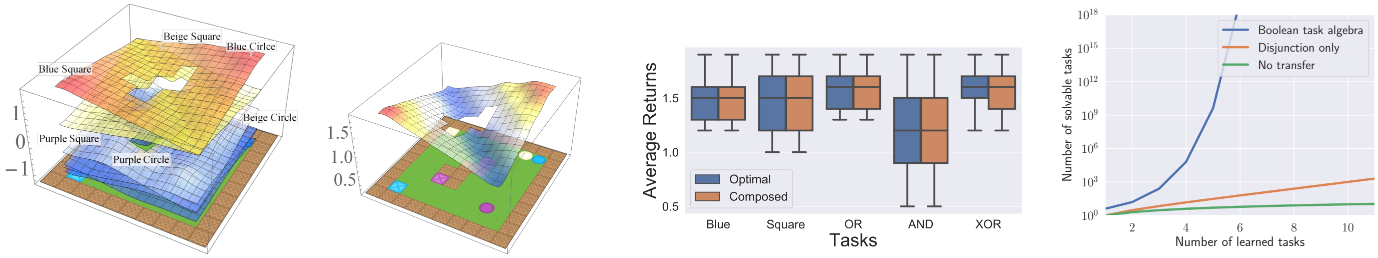
We then showed that the logical composition of WVFs produces exactly the WVF for the corresponding task composition (Theorem 5 (i)). We further showed that when the task specific rewards are binary, $\mathcal{M}$ and $\bar{\mathcal{Q}}^*$ from homomorphic Boolean algebras (Theorem 5 (ii)). This formally defines classical logic over the space of tasks and value functions.

Theorem 5. *Let $\bar{\mathcal{Q}}^*$ be the set of optimal world $\bar{Q}$-value functions for tasks in $\mathcal{M}$. Then,*

(i) for all tasks $M_1, M_2 \in \mathcal{M}$, we have $\bar{Q}^*_{\neg M_1} = \neg\bar{Q}^*_{M_1}$, $\bar{Q}^*_{M_1 \vee M_2} = \bar{Q}^*_{M_1} \vee \bar{Q}^*_{M_2}$, and $\bar{Q}^*_{M_1 \wedge M_2} = \bar{Q}^*_{M_1} \wedge \bar{Q}^*_{M_2}$.

(ii) If the task specific rewards are binary, then $(\mathcal{M}, \vee, \wedge, \neg)$ and $(\bar{\mathcal{Q}}^*, \vee, \wedge, \neg)$ are homomorphic Boolean algebras.

Experiment 3.2. *Consider video game world where an agent must navigate a 2D world and collect objects of different shapes and colours [4]. The state space is an 84×84 RGB image, and the agent is able to move in any of the four cardinal directions. The agent also possesses a `pick-up` action, which allows it to collect an object when standing on top of it. The positions of the agent and objects are randomised at the start of each episode. We first use deep Q-learning (similarly to Algorithm 1) to learn two base tasks: collecting blue objects and collecting squares. Figure 4a shows the values with respect to each goal of the learned WVF for collecting blue objects, and maximising over goals gives us the regular value function (Figure 4b). We can then compose them to solve their disjunction (OR), conjunction (AND) and exlusive-or (XOR), as shown in Figure 4c. In general, after learning n base tasks we can solve all their 2^{2^n} logical compositions, giving agents a super-exponential explosion of skills (Figure 4d).*



(a) Learned WVF for collecting blue objects.

(b) Inferred regular value function for collecting *Blue* objects.

(c) Average returns over 1000 episodes of optimal and composed WVFs.

(d) Super-exponential number of solvable tasks from learned ones.

Figure 4: Experiments in a 2D domain where the agent needs to collect objects of various shapes and colors.

4 Conclusion

For an agent to learn how to solve multiple tasks in a given world from a single stream of experience, it should not only encode knowledge about how to optimally solve the current task, but also how achieve agent-centric goals in the world. We introduced WVFs as a principled knowledge representation that captures this idea. We then showed that WVFs learned on any task have mastery of the world—they encode how to solve all possible goal-reaching tasks. This results in agents that can infer the dynamics of the environment (which can be used for model based approaches like planning), enables them to solve new tasks zero-shot given just their terminal rewards, and finally enables zero-shot logical composition of learned skills (an important ability for the combinatorial explosion of skills in long-lived agents).

References

[1] G. Nangue Tasse, S. James, and B. Rosman. A Boolean task algebra for reinforcement learning. *Advances in Neural Information Processing Systems*, 33, 2020.

[2] R. Sutton, A. Barto, et al. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.

[3] R. Sutton, J. Modayil, M. Delp, T. Degris, P. Pilarski, A. White, and D. Precup. Horde: A scalable real-time architecture for learning knowledge from unsupervised sensorimotor interaction. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 761–768, 2011.

[4] B. van Niekerk, S. James, A. Earle, and B. Rosman. Composing value functions in reinforcement learning. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 6401–6409. PMLR, 09–15 Jun 2019.

[5] V. Veeriah, J. Oh, and S. Singh. Many-goals reinforcement learning. *arXiv preprint arXiv:1806.09605*, 2018.