Object Empowerment-Driven Tool Selection for Exploration in Reinforcement Learning

Faizan Rasheed, Kenzo Clauw, Daniel Polani, Nicola Catenacci Volpi
Adaptive Systems Research Group
University of Hertfordshire
United Kingdom
{f.rasheed,k.clauw,d.polani,n.catenacci-volpi}@herts.ac.uk

Abstract

Tool use enhances problem solving by enabling complex tasks, but remains challenging for RL agents due to long horizons and sparse, delayed rewards that hinder exploration and learning efficiency. While classic intrinsic motivation (IM) improves exploration, common methods lack focus on object-tool interactions, causing agents to discover many irrelevant details. In this paper, we show how RL agents can efficiently learn tool use by optimizing object empowerment, an IM measuring control over specific objects. We extend this to multi-tool, multi-object settings, enabling agents to identify key tool-object relations, learn when and how to use tools, and understand their lasting effects. Experiments in MiniHack environments show improved exploration, generalization, and efficiency over PPO under sparse reward conditions.

1 Introduction

Efficient exploration is essential for lifelong learning agents to continuously adapt, discover and efficiently solve new tasks. In cognitive science, affordances — action possibilities offered by objects—are linked to tool use in problem-solving [5, 19]. Inspired by this, we explore affordance - based object interactions to enhance exploration in reinforcement learning (RL).

Exploration in environments involving object-tool interactions is challenging due to sparse rewards, delayed feedback and long-horizon dependencies. Intrinsic motivation (IM) offers a solution by providing internal rewards that guide exploration beyond external signals. Common IM strategies include novelty [29], curiosity via prediction error [20], and information gain [8]. Among these, empowerment [12, 24] is particularly suited to lifelong learning, as it measures an agent's capacity to influence its environment, encouraging exploration of controllable and meaningful aspects. However, classical empowerment treats all controllable states equally—including irrelevant areas like empty rooms—leading to inefficient exploration. To address this, [23] proposed object empowerment, measuring control over specific objects instead of the entire environment. This improved learning in simple single-tool-object tasks. Yet, real-world environments often involve multiple tools and objects requiring the agent to evaluate and select optimal tool-object pairs.

In this work, we extend the object empowerment framework to multi-tool, multi-object environments for efficient exploration in RL. We propose multi-object empowerment, generalizing the measure across multiple objects, and developing an empowerment-based tool selection mechanism. In addition to determining which tools to use, another challenge for RL agents is deciding where to use them. For instance, some tools only become effective when the agent is near a task-relevant object (e.g., chimpanzees use stones to crack nuts only near specific nut trees [6]), while others can act on objects from a distance (e.g., a remote control). In more complex scenarios, one may need to reason about the downstream effects of tool use. The interaction with an object can depend on the satisfaction of

18th European Workshop on Reinforcement Learning (EWRL 2025).

specific preconditions or intermediate sub-goals—such as manipulating an object that lies behind a locked door. A tool may be essential for addressing such intermediary tasks (e.g., using a key to open the door), even when it is not directly involved in achieving the agent's primary objective, making the context of tool use harder to identify. Interestingly, this state-dependence of tool utility is mirrored in the nature of object empowerment, which is a function of the agent's state and can provide the necessary spatial context for meaningful tool-object interactions. As we will show, object empowerment landscapes can guide agents in discovering where a tool exerts maximum influence over its target object. Another crucial dimension of tool use is the temporal evolution of tool-object interactions. Some tools afford repeated or persistent transformations, while others have a one-time effect and then lose control over the object. For instance, unlocking a door with a key preserves long-term interaction possibilities (e.g., the door can be re-locked), whereas breaking the door eliminates future interactions. We will show that object empowerment enables the characterization tools in terms of the temporal extent to which they can continue interacting meaningfully with an object.

2 Related Work

Several studies have modeled tool use and learning. [10] propose a Bayesian framework capturing the triadic relationship between tools, actions, and effects, which relates to our use of tool-to-object empowerment for exploration. Similarly, [28] show how tool-use capabilities can emerge through behavior-grounded exploration based on effect representations. RL has also been used to acquire tool-use skills [30, 15], including via auxiliary rewards to optimize resource constraints. Closer to our approach, [27] propose a developmental robotics model where tool use emerges from intrinsic motivation and planning.

Intrinsic drives that support the discovery of functional affordances are central to guided self-organization, which emphasizes internal information gradients over external rewards [22]. Empowerment [12, 24], unlike novelty-based IMs [21, 2], quantifies how much control an agent has over its environment. As a biologically inspired, information-theoretic measure, empowerment fosters structured exploration and skill acquisition, especially in sparse-reward settings [17, 3]. Recent work by [14] demonstrates its role in open-ended skill discovery in tool-rich environments, reinforcing its relevance to tool use.

3 Methodology

3.1 Tool Learning Framework

We model tool-use within a RL framework [1], where an autonomous agent learns to handle tools by manipulating objects in its surrounding. The environment is represented as a Markov Decision Process (MDP), defined by a quadruple $(\mathcal{S}, \mathcal{A}, T, R)$. Here, $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, T is the transition function and R is the reward function. The agent aims to find policy that maximizes the expected return $\sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$, where $\gamma \in [0, 1)$ is a discount factor that prioritizes immediate rewards over distant ones.

We use object empowerment ($\mathfrak{E}_{\mathfrak{O}}$) as an intrinsic motivation signal to guide exploration. Assuming the extrinsic reward $R(s)$ depends only on the state $s \in \mathcal{S}$, we combine it with object empowerment $\mathfrak{E}_{\mathfrak{O}}(s)$ into a regularized reward function $\hat{R}(s)$ as follows:

$$\forall s \in \mathcal{S} \;\; \hat{R}(s) \coloneqq R(s) + \beta \mathfrak{E}_{\mathfrak{O}}(s) \quad , \tag{1}$$

where $\beta \in \mathbb{R}_{\geq 0}$ is a weighting factor that balances the contribution of extrinsic and intrinsic reward. The maximization of the regularized reward $\hat{R}$ encourages the agent to explore actions and states that increase object empowerment even in the absence of immediate extrinsic rewards. A small β places greater emphasis on the completion of the task encoded by R, while a larger β pushes the agent to maintain its control over objects of the environment, even at the cost of not addressing the task at all when β is very large. A suitable trade-off can guide the agent to interact with objects during early learning, thereby facilitating task completion in later stages.

3.1.1 State Space

Formally, the environment consists of an agent, a set of n tools $\mathfrak{T} = \{\mathfrak{T}_1, \mathfrak{T}_2, \ldots, \mathfrak{T}_n\}$, and a set of m objects $\mathfrak{O} = \{\mathfrak{O}_1, \mathfrak{O}_2, \ldots, \mathfrak{O}_m\}$. Each of these entities contributes to the overall state space, defined as:

$$\mathcal{S} \coloneqq \mathcal{S}^{\mathfrak{A}} \times \left(\prod_{j=1}^{n} \mathcal{S}^{\mathfrak{T}_j} \right) \times \left(\prod_{i=1}^{m} \mathcal{S}^{\mathfrak{O}_i} \right) \times \mathcal{S}^{\mathfrak{W}} \tag{2}$$

Here, $\mathcal{S}^{\mathfrak{A}}$ is the agent's state space (e.g., its location in the environment), $\mathcal{S}^{\mathfrak{T}_j}$ is the state space of the j-th tool (e.g., its position or whether it is equipped by the agent), $\mathcal{S}^{\mathfrak{O}_i}$ is the state space of the i-th object (e.g., its location or condition), $\mathcal{S}^{\mathfrak{W}}$ includes other static components of the environment, such as walls or goal positions.

3.1.2 Action Space

Among all the actions in $\mathcal{A}$ that an agent can perform, we define now the subsets of actions that are executed while using the tools in $\mathfrak{T}$. We distinguish between the following subsets of $\mathcal{A}$: (i) the actions of the agent $\mathcal{A}^{\mathfrak{A}} \subseteq \mathcal{A}$ that do not involve the use of a tool; (ii) the actions $\mathcal{A}^{\mathfrak{T}_j} \subseteq \mathcal{A}$ that allow the agent to use the tool $\mathfrak{T}_j$, for $j = 1, 2, \ldots n$; (iii) the set $\mathcal{A}^{\mathfrak{A}\mathfrak{T}_j} \coloneqq \mathcal{A}^{\mathfrak{A}} \cup \mathcal{A}^{\mathfrak{T}_j}$ for $j = 1, 2, \ldots n$, containing both the actions of the agent that are not relevant to tool use and its actions specifically relevant to tool $\mathfrak{T}_j$. For instance, in a navigation task, $\mathcal{A}^{\mathfrak{A}}$ could contain the action "north", which moves the agent towards the north direction, where if the agent equips an axe, the set $\mathcal{A}^{\mathfrak{T}_{\text{axe}}}$ could include the action "chop".[1]

3.2 Object Empowerment

Empowerment[12] is defined as the Shannon capacity of an agent's actuation channel between action sequences and resulting states. Object empowerment extends the classical empowerment formalism by measuring an agent's influence over the state subspace $\mathcal{S}^{\mathfrak{O}_i}$ of specific objects of the environment $\mathfrak{O}_i$, rather than over the entire state space $\mathcal{S}$ [23]. Furthermore, given a tool $\mathfrak{T}_j$, object empowerment is defined by using the tool actions subset $\mathcal{A}^{\mathfrak{T}_j}$ as source of the agent's actuation channel, instead of the full agent action set $\mathcal{A}$ as in classical empowerment. This formulation not only quantify the degree of influence that an agent has over specific objects of the environment $\mathfrak{O}_i$, but also allows one to measure the impact of those interactions that are exclusively mediated via tool $\mathfrak{T}_j$.

Given a tool $\mathfrak{T}_j \in \mathfrak{T}$, let $a^h_{\mathfrak{T}_j} \coloneqq (a_1^{\mathfrak{T}_j}, a_2^{\mathfrak{T}_j}, \ldots, a_h^{\mathfrak{T}_j}) \in \mathcal{A}^h_{\mathfrak{T}_j}$ be a tool action sequence of length h, where $\mathcal{A}^h_{\mathfrak{T}_j}$ denotes the set of all possible sequences of h tool $\mathfrak{T}_j$ actions. Let S_t be a random variable representing the agent's state at time t, and $A^h_{\mathfrak{T}_j}$ the random variable for the h-step tool $\mathfrak{T}_j$ action sequence starting at time t. [2] Let $S^{\mathfrak{O}_i}_{t+h}$ be the random variable representing the state of object $\mathfrak{O}_i$ at time $t + h$. The h-step *object empowerment* $\mathfrak{E}^h_{\mathfrak{T}_j \mathfrak{O}_i}(s)$ of state $s \in \mathcal{S}$ from tool $\mathfrak{T}_j$ to object $\mathfrak{O}_i$ is defined as the Shannon capacity of the channel between the tool $\mathfrak{T}_j$ action sequence and the resulting state of object $\mathfrak{O}_i$, conditioned on the current state s:

$$\mathfrak{E}^h_{\mathfrak{T}_j \mathfrak{O}_i}(s) \coloneqq \max_{P(a^h_{\mathfrak{T}_j}|s)} I(S^{\mathfrak{O}_i}_{t+h}; A^h_{\mathfrak{T}_j} \mid S_t = s) \tag{3}$$

where $I(X;Y)$ denotes the mutual information between the random variables X and Y. $\mathfrak{E}^h_{\mathfrak{T}_j \mathfrak{O}_i}$ measures how much the actions of the tool $\mathfrak{T}_j$ can reliably influence the state of the object $\mathfrak{O}_i$. To capture an agent's control over multiple objects jointly, we extend the above formulation to define *multi-object empowerment*. Let $\mathfrak{D} = \{\mathfrak{D}_1, \mathfrak{D}_2, \ldots, \mathfrak{D}_q\} \subseteq \mathfrak{O}$ be a subset of objects. The h-step multi-object empowerment from tool $\mathfrak{T}_j$ to objects $\mathfrak{D}$ is then defined as:

$$\mathfrak{E}^h_{\mathfrak{T}_j \mathfrak{D}}(s) \coloneqq \max_{P(a^h_{\mathfrak{T}_j}|s)} I(S^{\mathfrak{D}_1}_{t+h} \ldots S^{\mathfrak{D}_q}_{t+h}; A^h_{\mathfrak{T}_j} \mid S_t = s) \tag{4}$$

[1] In this paper, interactions between objects and tools $\mathfrak{T}_j$ are always performed using actions from the set $\mathcal{A}^{\mathfrak{A}\mathfrak{T}_j}$. For notational simplicity, we denote this set as $\mathcal{A}^{\mathfrak{T}_j}$ throughout the text.

[2] When writing $A^h_{\mathfrak{T}_j}$ we omit the time index t to have a more compact notation.

In deterministic settings, where transitions and observations are uniquely determined by actions and state, the mutual information in Equation 3 reduces to the log-cardinality of the set of distinct states of object $\mathfrak{O}_i$ that the agent can observe from s after executing all possible h-step tool action sequences $a^h_{\mathfrak{T}_j}$:

$$\mathfrak{E}^h_{\mathfrak{T}_j\mathfrak{O}_i}(s) = \log_2\left(\left|\mathcal{S}^h_{\mathfrak{T}_j\mathfrak{O}_i}(s)\right|\right) \tag{5}$$

where $\mathcal{S}^h_{\mathfrak{T}_j\mathfrak{O}_i}(s) \coloneqq \{s_{t+h} \mid a^h_{\mathfrak{T}_j} \in \mathcal{A}^h_{\mathfrak{T}_j}, s_{t+h} = T^h(s, a^h_{\mathfrak{T}_j})\}$ is the set of states reachable from s after applying all $a^h_{\mathfrak{T}_j}$ in $\mathcal{A}^h_{\mathfrak{T}_j}$, and $T^h(s, a^h)$ denotes the h-step transition function.

3.3 Tool Selection Mechanism

In scenarios with multiple tools and objects an agent may benefit by knowing which tools enables the largest control over each object of the environment. While more than one tool may exert some influence on a certain object, the level of influence may vary, with some tools that may be very effective while others may be useless. To represent all possible tool-object relationships, we define the *tool-object empowerment matrix*. It contains the state-averaged object empowerment $\hat{\mathfrak{E}}^h_{\mathfrak{T}_j\mathfrak{O}_i}$ of each tool to each object in the environment (see Table 1). We define the h-step tool-object empowerment matrix $\mathbb{T} \in \mathbb{R}^{n\times m}$ as

$$\mathbb{T}[j,i] = \hat{\mathfrak{E}}^h_{\mathfrak{T}_j\mathfrak{O}_i} \quad j = 1,\ldots,n,\ i = 1,\ldots,m.$$

	$\mathfrak{O}_1$	$\ldots$	$\mathfrak{O}_{i^*}$	$\cdots$	$\mathfrak{O}_m$
$\mathfrak{T}_1$	$\hat{\mathfrak{E}}^h_{\mathfrak{T}_1\mathfrak{O}_1}$	$\cdots$	$\hat{\mathfrak{E}}^h_{\mathfrak{T}_1\mathfrak{O}_{i^*}}$	$\cdots$	$\hat{\mathfrak{E}}^h_{\mathfrak{T}_1\mathfrak{O}_m}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathfrak{T}_j*$	$\hat{\mathfrak{E}}^h_{\mathfrak{T}_j*\mathfrak{O}_1}$	$\cdots$	$\hat{\mathfrak{E}}^h_{\mathfrak{T}_j*\mathfrak{O}_{i^*}}$	$\cdots$	$\hat{\mathfrak{E}}^h_{\mathfrak{T}_j*\mathfrak{O}_m}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathfrak{T}_n$	$\hat{\mathfrak{E}}^h_{\mathfrak{T}_n\mathfrak{O}_1}$	$\cdots$	$\hat{\mathfrak{E}}^h_{\mathfrak{T}_n\mathfrak{O}_{i^*}}$	$\cdots$	$\hat{\mathfrak{E}}^h_{\mathfrak{T}_n\mathfrak{O}_m}$

Table 1: Tool-object empowerment matrix $\mathbb{T}$ showing the state-averaged empowerment $\hat{\mathfrak{E}}^h_{\mathfrak{T}_j\mathfrak{O}_i}$ for each tool-object pair. Values indicate the degree of influence each tool has over each object and i^* indicates the object of interest.

Tools with non-zero average object empowerment with certain objects constitute candidates tools for interacting with those objects. On the contrary, if an item exhibits zero average object empowerment toward all objects, it can not be considered a tool for that environment. Finally, there is the tool with maximum average object empowerment for an object. Given an object of interest $\mathfrak{O}_{i^*}$, this is defined as follows:

$$\mathfrak{T}_{j^*} \coloneqq \arg\max_j \ \hat{\mathfrak{E}}^h_{\mathfrak{T}_j\mathfrak{O}_{i^*}}. \tag{6}$$

Equation 6 enables the design of a *tool selection mechanism* that can be used by artificial agents to automatically select tools when interacting with specific objects $\mathfrak{O}_{i^*}$. One can expect that, without prior knowledge about the tools, on average, the tool selected by Equation 6 has the largest chance of being useful when interacting with the task object $\mathfrak{O}_{i^*}$. Thus, in our RL experiments, we used Equation 6 to choose the object empowerment $\mathfrak{E}^h_{\mathfrak{T}_j*\mathfrak{O}_{i^*}}$ from the selected tool $\mathfrak{T}_{j^*}$ to the task objects $\mathfrak{O}_{i^*}$ as intrinsic reward in Equation 1. We will show that the object empowerment of the selected tool can guide exploration toward meaningful object interactions.

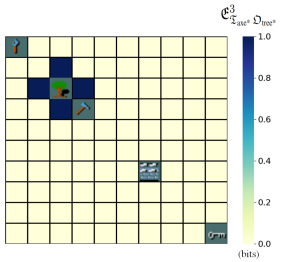
4 Experiments

We conduct our experiments in MiniHack environments [25], which support rich interactions between tools and objects in a grid-based world. A detailed description of the environment is provided in Appendix A.

5 Experiment 1: Empowerment-Guided Tool Selection in Single-Object Task

The environment reported in Figure 1a includes two manipulable objects, a tree and a wall, and four available tools: an axe, a pickaxe, a tin opener and a key. We considered the task of chopping the tree and the one of destroying the wall. We start with the first one, hence, here the tree is the task-relevant object $\mathfrak{O}_{\text{tree}*}$.

(a) Initial state of the environment. Black cells represent unobserved areas outside the agent's field of view.



(b) 3-step axe to tree empowerment $\mathfrak{E}^3_{\mathfrak{T}_{\text{axe}*}\mathfrak{O}_{\text{tree}*}}$ landscape for all possible agent's locations (in bits), when the agent is equipped with the axe.

Figure 1: (a) Experiment 1 environment setup. The agent must use the appropriate tool (e.g., axe for the tree) while other tools act as distractors. (b) Empowerment landscape shows non-zero values only when the agent is adjacent to the tree.

To support tool selection, we compute the tool-object empowerment matrix $\mathbb{T}$ for this environment and report it in Table 2. Among all the tools of this environment, only the axe has an influence over the state of the tree ($\hat{\mathfrak{E}}^h_{\mathfrak{T}_{\text{axe}}\mathfrak{O}_{\text{tree}*}} = 4.233 \times 10^{-8}$ bits). [3] The pickaxe has only an effect on the state of the wall ($\hat{\mathfrak{E}}^h_{\mathfrak{T}_{\text{pickaxe}}\mathfrak{O}_{\text{wall}}} = 4.233 \times 10^{-8}$ bits). The tin opener and the key have no impact on any object ($\hat{\mathfrak{E}}^h_{\mathfrak{T}_{\text{tinop}}\mathfrak{O}_{\text{tree}*}} = 0$ bits, $\hat{\mathfrak{E}}^h_{\mathfrak{T}_{\text{key}}\mathfrak{O}_{\text{wall}}} = 0$ bits)), so they should not be considered tools for this environment. Since the axe yields the highest object empowerment for the tree, $\mathfrak{T}_{\text{axe}*}$ is selected through the tool selection mechanism of Equation (6).

	$\mathfrak{O}_{\text{tree}*}$	$\mathfrak{O}_{\text{wall}*}$
$\mathfrak{T}_{\text{axe}*}$	4.233×10^{-8}	0
$\mathfrak{T}_{\text{pickaxe}*}$	0	4.233×10^{-8}
$\mathfrak{T}_{\text{tinop}}$	0	0
$\mathfrak{T}_{\text{key}}$	0	0

Table 2: State-averaged tool-to-object empowerment $\hat{\mathfrak{E}}^h_{\mathfrak{T}_j\mathfrak{O}_i}$ in bits for each tool-object combination of Experiment 1.

To illustrate the spatial distribution of object empowerment in this environment, we examine the empowerment landscape before and after the axe is equipped. When the axe is not equipped, it $\mathfrak{E}^8_{\mathfrak{T}_{\text{axe}*}\mathfrak{O}_{\text{tree}*}}$ is nonzero only in the cell where the axe is located, indicating that from there in 8 steps the agent can reach tree and chop it. There, the value of $\mathfrak{E}^8_{\mathfrak{T}_{\text{axe}*}\mathfrak{O}_{\text{tree}*}}(s_p^{\mathfrak{T}_{\text{axe}*}})$ is 1 bit, because the agent can either chop the tree or leave it intact. When $\mathfrak{E}^8_{\mathfrak{T}_{\text{axe}*}\mathfrak{O}_{\text{tree}*}}$ is used as intrinsic reward, this acts as a beacon towards the tool location, helping the agent to find the axe while exploring the environment. In Figure 1b we report the landscape of $\mathfrak{E}^3_{\mathfrak{T}_{\text{axe}*}\mathfrak{O}_{\text{tree}*}}$ for when the tool is equipped. The landscape shows non-zero values (i.e., 1 bit) of $\mathfrak{E}^3_{\mathfrak{T}_{\text{axe}*}\mathfrak{O}_{\text{tree}*}}$ in locations adjacent to the tree. This indicates that from those positions, the agent can chop the tree in the 3 steps necessary to execute the "apply"→"choose"→"direction" sequence of actions illustrated in the previous section. This landscape reflects the fact that the axe is a tool whose influence is highly localized and effective only

[3] The MDP representing this environment has more than 70000 states, due to the combinatorial contribution of the states of all the tools an objects in the environment, each one having three possible states. For this reason, and the sparsity of the landscape, in this experiment, and the following ones, the state averaged object empowerment is a very small number.

when the agent is next to its target object. When used as intrinsic reward, $\mathfrak{E}^{3}_{\mathfrak{T}_{\text{axe}*}\mathfrak{O}_{\text{tree}*}}$ acts as beacon that attracts the agent to the tree once the axe is equipped, helping it to fulfill the spatial conditions under which meaningful tool-object interactions become possible. Since the agent needs more steps to interact with an object when a tool is unequipped (i.e., additional steps are necessary to reach the tool and pick it up), in our experiments we have used a longer horizon h for states where the tool is unequipped and a shorter horizon for states where the tool is equipped (here, $h = 8$ and $h = 3$ respectively).

To evaluate learning performance under sparse reward conditions, we compare a standard PPO agent with an intrinsically motivated agent whose reward is regularized with object empowerment. Figure 2 shows the average cumulative reward across training episodes, computed over 10 independent runs. The agent using $\mathfrak{E}^{h}_{\mathfrak{T}_{\text{axe}*}\mathfrak{O}_{\text{tree}*}}$ ($\beta = 0.0009$) shows a faster convergence to optimal performance compared to the baseline PPO agent. This improvement highlights how object empowerment helps guide exploration in sparse reward environments where useful tool-object interactions must be discovered. Similar results were obtained when the objective was to destroy the wall $\mathfrak{O}_{\text{wall}*}$ and the

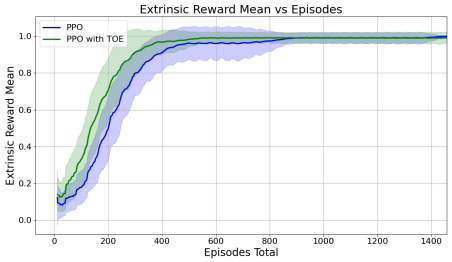


Figure 2: The agent using the axe-to-tree empowerment $\mathfrak{E}^{h}_{\mathfrak{T}_{\text{axe}*}\mathfrak{O}_{\text{tree}*}}$ as a regularizer (green) shows faster convergence compared to standard PPO (blue). Shaded regions represent standard deviation across 10 independent runs.

the pickaxe $\mathfrak{T}_{\text{pickaxe}*}$ was selected as a tool, confirming the generality of the proposed approach.

6 Experiment 2: Empowerment-Guided Tool Selection in Multi-Object Task

In this experiment, we explore a more challenging scenario where the agent is required to destroy two distinct objects: a boulder and a door (i.e., with multiple targets $\mathfrak{O}_{\text{bould}*}\mathfrak{O}_{\text{door}*}$). The agent receives a reward of 1 for each object successfully destroyed.

(a) Initial state of the environment of experiment 2.

(b) Empowerment landscape of experiment 2

Figure 3: (a) Experiment 2 environment setup. (b) 6-step wand to boulder-door empowerment $\mathfrak{E}^{6}_{\mathfrak{T}_{\text{wand}*}\mathfrak{O}_{\text{bould}*}\mathfrak{O}_{\text{door}*}}$ landscape for all possible agent's locations, when the agent is equipped with the wand.

The environment (see Figure 3a) contains four tools: a wand, an axe, a tin opener, and a katana. Here, the wand can destroy both the boulder and the door, the axe is capable of only destroying the door, while the tin opener and katana serve as distractors with no effect on the environment's objects. In addition, the environment includes walls that act as static barriers, preventing agent movement, which do not serve as manipulable objects.

We report the tool-object empowerment matrix $\mathbb{T}$ for this environment in Table 3. In addition to the average tool to object empowerment of the individual objects, this table also reports the average tool to object empowerment $\hat{\mathfrak{E}}^h_{\mathfrak{T}_j\,\mathfrak{O}_{\text{bould*}}\mathfrak{O}_{\text{door*}}}$ of the two objects considered together (see Equation 4). Being $\hat{\mathfrak{E}}^h_{\mathfrak{T}_{\text{wand}}\,\mathfrak{O}_{\text{bould*}}\mathfrak{O}_{\text{door*}}}$ the largest average object empowerment for both targets, our tool selection method chooses the wand $\mathfrak{T}_{\text{wand*}}$ and its boulder-door empowerment as intrinsic reward for RL.

	$\mathfrak{O}_{\text{bould}}$	$\mathfrak{O}_{\text{door}}$	$\mathfrak{O}_{\text{bould*}}\mathfrak{O}_{\text{door*}}$
$\mathfrak{T}_{\text{wand*}}$	5.292×10^{-7}	6.138×10^{-7}	9.564×10^{-7}
$\mathfrak{T}_{\text{axe}}$	0	3.281×10^{-7}	3.281×10^{-7}
$\mathfrak{T}_{\text{tinop}}$	0	0	0
$\mathfrak{T}_{\text{kata}}$	0	0	0

Table 3: State-averaged tool to object empowerment $\hat{\mathfrak{E}}^h_{\mathfrak{T}_j\mathfrak{O}_i}$ for each tool-object combination. The last column reflects the multi-object empowerment $\hat{\mathfrak{E}}^h_{\mathfrak{T}_j\,\mathfrak{O}_{\text{bould*}}\mathfrak{O}_{\text{door*}}}$.

In this experiment we use $h = 5$ when the wand is unequipped for reward regularization, because this horizon yields an object empowerment landscape peaked in the location of the wand, and $h = 6$ when the wand is equipped. We report the wand-equipped landscape of $\mathfrak{E}^6_{\mathfrak{T}_{\text{wand}}\,\mathfrak{O}_{\text{bould*}}\mathfrak{O}_{\text{door*}}}$ in Figure 3b. Unlike the tools in Experiment 1, which can affect objects only when adjacent to it, here the wand's area of influence spans a larger portion of the grid. This is because in MiniHack, the wand can strike objects at arbitrary distances along the orthogonal directions from the agent's position. This observation suggests that object empowerment formalism could be used to characterized tools-object range of interaction. The 2.0 bits peaks of $\mathfrak{E}^6_{\mathfrak{T}_{\text{wand}}\,\mathfrak{O}_{\text{bould*}}\mathfrak{O}_{\text{door*}}}$ are in the two cells where in 6 steps the agent can destroy both the boulder and the door (i.e., 3 steps to destroy one plus 3 steps to destroy the other). Intermediate values, such as 1.58 and 1.0 bits, appear in locations where the agent can affect either one of the two objects or only one of the two, respectively.

We compare learning performance using standard PPO and PPO regularized with $\mathfrak{E}^h_{\mathfrak{T}_{\text{wand*}}\,\mathfrak{O}_{\text{bould*}}\mathfrak{O}_{\text{door*}}}$. Figure 4 reports the the cumulated reward mean per episode averaged over 10 independent runs. The TOE-augmented agent converges rapidly and attains higher final performance compared to the baseline. In contrast, the standard PPO agent frequently plateaus at suboptimal values, indicating it gets trapped in local optima, for instance by learning to destroy only one object. TOE-based regularization helps overcome this limitation by encouraging policies that expand future influence towards both objects, driving the agent toward broader interaction strategies that ultimately solve the full task.

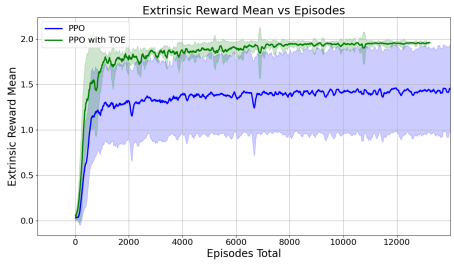


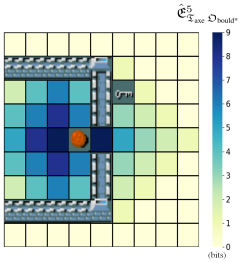
Figure 4: The agent using $\mathfrak{E}^h_{\mathfrak{T}_{\text{wand*}}\,\mathfrak{O}_{\text{bould*}}\mathfrak{O}_{\text{door*}}}$ as a regularizer with $\beta = 0.0009$ (green) learns faster and more reliably than the standard PPO agent (blue).

7 Experiment 3: Tool Use to Achieve Sub-goals

In this last experiment, we examine a more complex scenario, where tools enable task completion not through direct manipulation of the goal object, but via the interaction with another object whose manipulation is a pre-condition to reach the goal object. We depict the environment in Figure 5a. The environment contains an axe and a key as tools, and a door and a boulder as objects. The task for the agent is to move the boulder onto a designated goal location (highlighted as a blue square). To "push"

the boulder the agent must occupy a cell adjacent to a it and execute a movement action toward it, then both the agent and the boulder are displaced by one cell in the same direction. Hence, the boulder in this environment can be moved directly by the agent without requiring any tool. However, the boulder is initially inaccessible, positioned behind a locked door and surrounded by walls that restrict movement. To reach and push the boulder, the agent must first move through the door—either by opening it with the key or by destroying it using the axe. These tools therefore provide instrumental affordances: they do not act directly on the boulder but instead enable access to it by modifying the environment. Although the axe and the key do not impact the state of the boulder directly, the average axe to boulder empowerment $\hat{\mathfrak{E}}^{h}_{\mathfrak{T}_{\text{axe}}\,\mathfrak{O}_{\text{bould*}}}$ and key to boulder empowerment $\hat{\mathfrak{E}}^{h}_{\mathfrak{T}_{\text{key}}\,\mathfrak{O}_{\text{bould*}}}$ are non-zero for $h \geq 7$: they emerge indirectly, through a causal chain of actions where the agent alters the door state using a tool and subsequently moves the boulder using its own body.

(a) Initial state of the environment of experiment 3.



(b) 5-step boulder empowerment landscape after the room becomes accessible in experiment 3.

In Figure 5b we report the landscape of 5-step axe to boulder empowerment $\hat{\mathfrak{E}}^{5}_{\mathfrak{T}_{\text{axe}}\,\mathfrak{O}_{\text{bould*}}}$ when the axe is equipped and the room is accessible. Differently from the objects of the previous experiments, which had only two possible states, the boulder can be repeatedly pushed in multiple directions and transit in always more states as the number of interactions with it increases, creating a richer object empowerment landscape. As a result, $\hat{\mathfrak{E}}^{5}_{\mathfrak{T}_{\text{axe}}\,\mathfrak{O}_{\text{bould*}}}$ increases with h and with the proximity to the boulder, two features that, when used as intrinsic reward, not only make the boulder empowerment to act as beacon for the object, but also as a sort of gradient towards it, which in turn facilitates learning.

The tools-door interactions can themselves be characterized by their axe to door empowerment $\mathfrak{E}^{h}_{\mathfrak{T}_{\text{axe}}\,\mathfrak{O}_{\text{door}}}$ and key to door empowerment $\mathfrak{E}^{h}_{\mathfrak{T}_{\text{key}}\,\mathfrak{O}_{\text{door}}}$, which enables downstream influence to the boulder empowerment $\mathfrak{E}^{h}_{\mathfrak{T}_{\text{axe}}\,\mathfrak{O}_{\text{bould*}}}$ and $\mathfrak{E}^{5}_{\mathfrak{T}_{\text{key}}\,\mathfrak{O}_{\text{bould*}}}$, respectively. When the agent is located in the cell in front of the door, both tools yield a 3-step door empowerment of $\mathfrak{E}^{3}_{\mathfrak{T}_{\text{axe}}\mathfrak{O}_{\text{door}}} = \mathfrak{E}^{3}_{\mathfrak{T}_{\text{key}}\mathfrak{O}_{\text{Door}}} = 1.0$ bit (the door is either cleared, i.e. opened by the key or destroyed by the axe, or closed). Furthermore whether the door has been opened by the key or destroyed by the axe does not make any difference w.r.t. enabling the interaction of the agent with the boulder. But, although the axe and the door seems to act in a similar manner here, there is a fundamental difference between the two tools. The key operates reversibly: the door can be repeatedly opened and re-closed. In contrast, when the axe irreversibly destroys the door, it precludes any further interaction with it. From a purely object empowerment-based perspective, both interactions have the same door empowerment $\mathfrak{E}^{h}_{\mathfrak{T}_{j}\,\mathfrak{O}_{\text{door}}}$ of 1 bit for $h >= 3$. However, if we condition the object empowerment on the door not being closed, these two quantities dramatically differ. As shown in Figure 6, when conditioned, the $\mathfrak{E}^{h}_{\mathfrak{T}_{j}\,\mathfrak{O}_{\text{door}}}$ of 1 bit for $h >= 3$ evolves differently depending on whether the agent uses a key or an axe. In the Figure, the x-axis represents time steps t, and the y-axis encodes the possible state of the door (i.e., open, closed, or destroyed). Object empowerment values are represented by the color of the curves (red for 1 bit and blue for 0 bits). For the key, the door empowerment remains at a steady value of 1 bit even as the door changes state, highlighting the reversibility and temporal persistence of the key's influence on the door. By contrast, when using the axe, the agent can only destroy the door once and, after that, no further state transitions are possible, and empowerment drops and stays at 0 bits.

In Figure 7 we show that the agent learning with $\mathfrak{E}^{h}_{\mathfrak{T}_{\text{axe*}}\,\mathfrak{O}_{\text{door*}}\mathfrak{O}_{\text{bould*}}}$ as intrinsic rewards ($h = 7$ for states with the axe unequipped and $h = 5$ for equipped states) and ($\beta = 0.00006$) learns quicker and reaches higher asymptotic performance compared to the PPO baseline. This improvement demonstrates how object empowerment can encourage policies that account for multi-step dependencies, such as clearing intermediate obstacles to eventually reach the goal object.

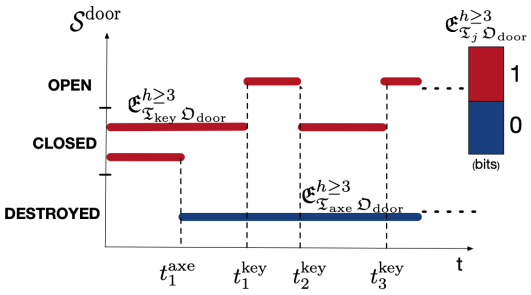


Figure 6: Temporal evolution of object empowerment $\mathfrak{E}^{h\geq 3}_{\mathfrak{T}_j,\mathfrak{O}_{\text{Door}}}$ as the agent interacts with the door using either the key or the axe.

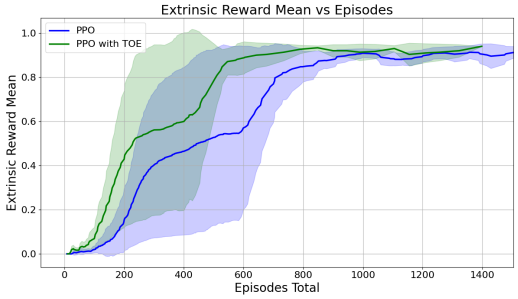


Figure 7: The agent using $\mathfrak{E}^{h}_{\mathfrak{T}_{\text{axe}^*}\,\mathfrak{O}_{\text{door}^*}\mathfrak{O}_{\text{bould}^*}}$ as a regularizer with $\beta = 0.0006$ (green) learns faster compared to the standard PPO agent (blue).

8 Conclusion

This work has explored how object empowerment, an object-centric intrinsic motivation, can guide learning of tool-use behaviors in environments with multiple tools and objects. Rooted in cognitive science-inspired principles, our approach reflects the idea that adaptive organisms seek to maximize their potential influence over the objects that surround them. Across increasingly complex experiments, we demonstrated that object empowerment enables to: (i) identify tools with the highest potential influence over task-relevant objects, (ii) characterize tools interactions that support interactions with multiple objects, that are long ranged, or that are persistent and reversible (iii) reason about downstream effects of tool use over sequence of objects. Our experiments showed that agents guided by object empowerment consistently outperform vanilla baseline RL agents, converging faster and more reliably despite sparse rewards.

9 Limitations and Future Work

A key limitation of this work is that object empowerment is assumed and computed from known dynamics rather than learned. Future work aims to learn object empowerment directly from interactions, allowing agents to infer which tools control which objects and under what conditions. Our broader goal is to scale this to lifelong learning environments like Craftax [16] and MineRL [7] where agents must transfer and reuse skills across diverse tasks. In this setting, generalization across tools, goals, and environments becomes crucial. Recent work suggests that exploration plays a key role in improving generalization [11], yet existing approaches often ignore the consistent structure of object-tool interactions that persist across tasks. We propose that learning object empowerment can guide exploration toward such transferable structure, acting as an inductive bias to support compositional generalization. To do so, we refer to the Appendix B for two future research directions.

References

[1] Andrew Barto and S. Richard Sutton. Reinforcement learning: an introduction. 2018.

[2] Nicolas Bougie and Ryutaro Ichise. Skill-based curiosity for intrinsically motivated reinforcement learning. *Machine Learning*, 109:493–512, 2020.

[3] Siyu Dai, Wei Xu, Andreas Hofmann, and Brian Williams. An empowerment-based solution to robotic manipulation tasks with sparse rewards. *Autonomous Robots*, 47(5):617–633, 2023.

[4] Dibya Ghosh, Jad Rahme, Aviral Kumar, Amy Zhang, Ryan P Adams, and Sergey Levine. Why Generalization in RL is Difficult: Epistemic POMDPs and Implicit Partial Observability. In *Advances in Neural Information Processing Systems*, volume 34, pages 25502–25515. Curran Associates, Inc., 2021.

[5] James J. Gibson. The theory of affordances. pages 127–143, 1979. Accessed: 2025-05-17.

[6] MM Günther and C Boesch. Energetic cost of nut-cracking behaviour in wild chimpanzees. *Hands of primates*, pages 109–129, 1993.

[7] William H Guss, Manuel Lopes, Richard Liao, Carlos Florensa, Joel Lehman, and Kenneth O Stanley. Minerl: A large-scale dataset of minecraft demonstrations. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pages 1430–1432, 2019.

[8] Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[9] Maximilian Igl, Kamil Ciosek, Yingzhen Li, Sebastian Tschiatschek, Cheng Zhang, Sam Devlin, and Katja Hofmann. Generalization in Reinforcement Learning with Selective Noise Injection and Information Bottleneck. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[10] Raghvendra Jain and Tetsunari Inamura. Learning of tool affordances for autonomous tool manipulation. In *2011 IEEE/SICE international symposium on system integration (SII)*, pages 814–819. IEEE, 2011.

[11] Yiding Jiang, J. Zico Kolter, and Roberta Raileanu. On the importance of exploration for generalization in reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[12] Alexander S Klyubin, Daniel Polani, and Chrystopher L Nehaniv. Empowerment: A universal agent-centric measure of control. In *2005 ieee congress on evolutionary computation*, volume 1, pages 128–135. IEEE, 2005.

[13] Eric Liang, Richard Liaw, Robert Nishihara, Philipp Moritz, Roy Fox, Ken Goldberg, Joseph Gonzalez, Michael Jordan, and Ion Stoica. Rllib: Abstractions for distributed reinforcement learning. In *International conference on machine learning*, pages 3053–3062. PMLR, 2018.

[14] Aly Lidayan, Yuqing Du, Eliza Kosoy, Maria Rufova, Pieter Abbeel, and Alison Gopnik. Intrinsically-motivated humans and agents in open-world exploration. *arXiv preprint arXiv:2503.23631*, 2025.

[15] Ziang Liu, Stephen Tian, Michelle Guo, C Karen Liu, and Jiajun Wu. Learning to design and use tools for robotic manipulation. *arXiv preprint arXiv:2311.00754*, 2023.

[16] Michael Matthews, Michael Beukman, Benjamin Ellis, Mikayel Samvelyan, Matthew Jackson, Samuel Coward, and Jakob Foerster. Craftax: A Lightning-Fast Benchmark for Open-Ended Reinforcement Learning. 2024.

[17] Shakir Mohamed and Jimenez Danilo Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. *Advances in neural information processing systems*, 28, 2015.

[18] Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[19] François Osiurak, Christophe Jarry, and Didier Le Gall. Grasping the affordances, understanding the reasoning: Toward a dialectical theory of human tool use. 117(2):517–540, 2010.

[20] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017.

[21] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pages 2778–2787. PMLR, 2017.

[22] Mikhail Prokopenko. *Guided self-organization: Inception*, volume 9. Springer Science & Business Media, 2013.

[23] Faizan Rasheed, Daniel Polani, and Nicola Catenacci Volpi. Leveraging empowerment to model tool use in reinforcement learning. In *2023 IEEE International Conference on Development and Learning (ICDL)*, pages 28–36. IEEE, 2023.

[24] Christoph Salge, Cornelius Glackin, and Daniel Polani. Empowerment–an introduction. *Guided Self-Organization: Inception*, pages 67–114, 2014.

[25] Mikayel Samvelyan, Robert Kirk, Vitaly Kurin, Jack Parker-Holder, Minqi Jiang, Eric Hambro, Fabio Petroni, Heinrich Küttler, Edward Grefenstette, and Tim Rocktäschel. Minihack the planet: A sandbox for open-ended reinforcement learning research. *arXiv preprint arXiv:2109.13202*, 2021.

[26] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[27] Kristsana Seepanomwan, Daniele Caligiore, Kevin J O'Regan, and Gianluca Baldassarre. Intrinsic motivations and planning to explain tool-use development: A study with a simulated robot model. *IEEE Transactions on Cognitive and Developmental Systems*, 14(1):75–89, 2020.

[28] Alexander Stoytchev. Behavior-grounded representation of tool affordances. In *Proceedings of the 2005 ieee international conference on robotics and automation*, pages 3060–3065. IEEE, 2005.

[29] Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. #Exploration: A Study of Count-Based Exploration for Deep Reinforcement Learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[30] Sam Wenke, Dan Saunders, Mike Qiu, and Jim Fleming. Reasoning and generalization in rl: A tool use perspective. *arXiv preprint arXiv:1907.02050*, 2019.

A Environment Details

Let represent with $\mathcal{W}$ all possible cells of the grid-word. The employed state space $\mathcal{S}$ includes the grid location of the agent $s^{\mathfrak{A}} \in \mathcal{W}$, the positions of all the tools, $s_p^{\mathfrak{T}_1}, \ldots, s_p^{\mathfrak{T}_n} \in \mathcal{W}^n$, and the locations of all the objects, $s_p^{\mathfrak{O}_1}, \ldots, s_p^{\mathfrak{O}_m} \in \mathcal{W}^m$. In addition, the tools states $\mathcal{S}^{\mathfrak{T}_j}$ include an equipped status, indicating whether the tool has been picked up by the agent and added to its inventory, and a hidden status that says whether a tool is visible from the agent's point of view. Similarly, the object states $\mathcal{S}^{\mathfrak{O}_i}$ include a hidden status and a flag that indicates whether an object has been destroyed by the agent. The employed action space $\mathcal{A}$ includes the agent movements in the grid $\mathcal{A}^{\mathfrak{A}}$ (i.e., north, south, east, west) and the tools actions $\mathcal{A}^{\mathfrak{T}_j}$, which only take effect when a tool is equipped. Tools are equipped automatically when the agent moves onto the the cell where the tool is located (i.e., $s^{\mathfrak{A}} = s_p^{\mathfrak{T}_j}$). Tool actions are based on the MiniHack game mechanics, where the use of a tool involves the following three transitions: first, the agent needs to decide that it wants use a tool by executing

the "apply" action; then, it chooses which tool from its inventory to use via tool identifiers actions; finally, the agent specifies one of the four cardinal directions to which apply the tool. For example, applying an axe to the north may destroy a tree located in that direction. The grid-world dynamics T is deterministic, so we have used Equation 5 to compute object empowerment $\mathfrak{E}^h_{\mathfrak{T}_j\mathfrak{D}}$. Each experiment is formulated as an episodic MDP. The employed reward structure is sparse: agent is rewarded only for achieving the task objective, such as destroying designated objects, when the agent receives a reward of $+1$ and transitions to a terminal state. Otherwise, each other transition incurs a reward of 0. To solve the MDPs considered in our experiments, we used the Proximal Policy Optimization (PPO) algorithm [26], as implemented in the Ray's RLlib library [13].

B Future research directions on generalization and exploration

1. Epistemic Uncertainty for Learning Object Empowerment. Generalization in RL can be framed as a POMDP, where limited samples lead to uncertainty over latent structure [4]. Current object empowerment methods ignore epistemic uncertainty over which object-tool interactions are effective, limiting adaptation in novel settings. Incorporating epistemic uncertainty—e.g., via Bayesian methods, ensembles, or memory—could help agents actively resolve affordance ambiguities, improving sample efficiency and generalization to new tools, objects, and contexts [18].

2. Filtering Distractors via the Information Bottleneck. Naïvely maximizing object empowerment can lead agents to overfit to spurious or irrelevant object-tool interactions. To mitigate this, we propose regularizing empowerment using an information bottleneck (IB), focusing learning on interactions that are causally useful for control. IB-based methods are known to discard task-irrelevant information and promote compact, transferable representations [9].