Resource-rational Task Decomposition to Minimize Planning Costs

Carlos G. Correa[*] **(cgcorrea@princeton.edu)**
Princeton Neuroscience Institute, Princeton University, Princeton, NJ 08542 USA

Mark K. Ho[*] **(mho@princeton.edu)**
Fred Callaway (fredcallaway@princeton.edu)
Thomas L. Griffiths (tomg@princeton.edu)
Department of Psychology, Princeton University, Princeton, NJ 08542 USA
[*]Contributed equally

Abstract

People often plan hierarchically. That is, rather than planning over a monolithic representation of a task, they decompose the task into simpler subtasks and then plan to accomplish those. Although much work explores *how* people decompose tasks, there is less analysis of *why* people decompose tasks in the way they do. Here, we address this question by formalizing task decomposition as a *resource-rational representation problem*. Specifically, we propose that people decompose tasks in a manner that facilitates efficient use of limited cognitive resources given the structure of the environment and their own planning algorithms. Using this model, we replicate several existing findings. Our account provides a normative explanation for how people identify subtasks as well as a framework for studying how people reason, plan, and act using resource-rational representations.

Keywords: planning; task decomposition; option discovery; hierarchical reinforcement learning; subgoals

Introduction

Your brother's birthday is coming up, so you decide to leave work early to drop his gift off at the post office. Although you know your way around town, you haven't been to the post office in several months, so you need to think. In particular, you start to *plan*: "How do I get to the post office from here?" you ask yourself. "Well, there's that café where I sometimes get my morning coffee. If I can first get there, then I should be able to get to the post office easily. Now, how should I get to the café? I know its east of where I am, so I can walk that way until I hit the main road..." Continuing this line of thought for a few moments, you come up with a plan before setting off with determination.

The seemingly mundane choice to navigate to the café before navigating from the café to the post office is an example of *task decomposition*. That is, rather than reason about a task in its totality (e.g., going from work to the post office), people decompose a task into manageable subtasks (e.g., going from work to the café; going from the café to the post office) and then reason in terms of those subtasks. Planning at multiple levels of abstraction has been extensively documented in psychology and neuroscience (Botvinick et al., 2009) and plays an important role in developing systems that can solve complex, high-dimensional problems (Sacerdoti, 1974). In short, hierarchical planning and decision-making is a key element of intelligent behavior in both humans and machines.

Although much research has explored how people leverage hierarchical representations (Ribas-Fernandes et al., 2011;

Cushman & Morris, 2015; Balaguer et al., 2016), there has been less systematic investigation into the principles that determine task decompositions in the first place. There are a few notable exceptions, including accounts that emphasize the value of inferring the hidden structure to guide behavior across tasks (Collins & Frank, 2013; Tomov et al., 2020) as well as accounts based on compressing a representation of optimal behavior (Solway et al., 2014; Maisto et al., 2015). But, whereas these existing models emphasize decomposition in relation to statistical inference about the environment or behavior, our account focuses on a separate role that task decomposition plays: It makes *reasoning* easier.

Here, we approach task decomposition as a *resource-rational representation problem*. That is, we model people as solving the problem of how to break down a task in a manner that makes efficient use of planning resources. In the following sections, we provide background on related work before discussing the mathematical details of our normative account of task decomposition. We then report several simulations and show how our model can explain human data from four experiments reported by Solway et al. (2014). Finally, we conclude by discussing future directions for resource-rational approaches to problem solving representations.

Background

Planning is hard because of the *curse of dimensionality* (Bellman, 1957): As one attempts to plan into an increasingly distant future, over a larger state space, or under conditions of greater uncertainty, computation quickly becomes intractable. Nonetheless, humans have numerous strategies that allow us to plan in complex domains. Some of these strategies involve modifying the *search process*. For example, during search, people have been shown to limit their depth of planning (MacGregor et al., 2001; Keramati et al., 2016), prune away unpromising paths (Huys et al., 2012), and direct their search using model-free value estimates (Anderson, 1990; Newell & Simon, 1972; van Opheusden et al., 2017). Another strategy is to modify the *problem representation* itself. Various forms of hierarchical planning (Botvinick, 2012) and task decomposition (Solway et al., 2014; Huys et al., 2015) are characteristic of this approach. But while these two types of strategies are distinct, they are also clearly intertwined: How one represents a problem can make search anywhere from impossible to trivial (Kaplan & Simon, 1990).

The present work takes inspiration from the deep relationship between search—a type of computation—and task decomposition—a type of representation—in the cognitively demanding setting of planning. Task representations can play a key role in making problem-solving computations more efficient (Ho et al., 2019), and identifying general principles for automatically learning such representations is an active area of research in artificial intelligence. For instance, Jinnai et al. (2019) examine how penalizing dynamic programming iterations can guide decomposition, while Harb et al. (2018) introduce a deliberation cost for switching subtasks to help shape a decomposition. Here, we extend these ideas by analyzing human task decomposition in terms of *search costs*. Broadly, our approach is in the spirit of *resource-rational analysis* (Griffiths et al., 2015; Lieder & Griffiths, 2020), a formal framework for deriving cognitive models under the assumption that people make rational use of their limited cognitive resources. Previous resource-rational analyses of planning have focused primarily on the search process itself (e.g., Callaway et al., 2018). However, the interdependence of computations and the representations over which they operate means that this general framework can be readily applied to the latter.

Resource-Rational Task Decomposition

By treating task decomposition as a resource-rational problem, we assume people acquire task representations that enable them to plan efficiently and perform tasks successfully. Our account distinguishes between three nested levels of optimization (Figure 1A). The lowest level is *action-level planning*, where concrete actions are chosen that solve a subtask (e.g., which direction should I walk to get to the café). The next level is *subtask-level planning*, where a sequence of subtasks is chosen (e.g., navigating to the café and then to the post office). Finally, the highest level is *task decomposition*, where a set of subtasks that constitute the decomposed task is chosen (e.g., setting the café as a possible subgoal across multiple tasks).

Importantly, solutions to the higher levels of optimization depend on what happens at lower levels: A good task decomposition depends on how the subtask-level planner will compose the subtasks, and the selection of subtasks depends on how the action-level planner will accomplish each one. Furthermore, a resource-rational task decomposition is sensitive not only to how well the planners solve their subproblems (e.g., does action-level planning identify a good route to the café?), but also on the computational cost of identifying those solutions (e.g. how much thought did it take to find that route?). Next, we discuss each of the three levels of our model.

Action-level Planning

Action-level planning computes the optimal actions that one should take to reach a subgoal. Here, we focus on deterministic, shortest path problems (e.g., finding a route to the café). Formally, action-level planning occurs over a task defined by

a set of states, $\mathcal{S}$; a transition graph, $T \subseteq \mathcal{S} \times \mathcal{S}$; and a subgoal state, $z \in \mathcal{S}$. In our running example, states are possible locations (e.g., at the office, at work, at the café, at the post office); the transition graph represents which locations in town are accessible to one another; and a subgoal could be the café.

Given an initial state, s, and a subgoal, z, action-level planning seeks to find a minimum-length sequence of states that begins at s and ends at z. We denote the length of this minimum-length sequence to be $D(s,z)$. For computing the optimal sequence of actions, we consider two broad classes of search algorithms: uninformed search and heuristic search (Newell & Simon, 1972; Russell & Norvig, 2009).

Uninformed Search   When faced with a domain that lacks features to guide exploration, the best that a planning agent can do is blindly but systematically explore their model of the problem starting from an initial state. This strategy describes a broad class of search algorithms known as *uninformed search*. For example, *breadth-first search* (BFS) explores states in the order of their distance from the starting state. As a result, for an initial node s and subgoal node z, BFS will explore all states that are less than the minimum path length $D(s,z)$, as shown in Figure 1B. The cost of BFS, $C_{\mathrm{BFS}}(s,z)$, is proportional to the number of these states.

Heuristic Search   Unlike uninformed search, *heuristic search* leverages domain knowledge in the form of a *heuristic function* that can provide an optimistic estimate of the distance to a goal. For instance, when navigating to the café, you might know that it is North-East of work, leading you to consider walking North or East before South or West. The canonical heuristic search algorithm is A* (Hart et al., 1968), which considers states in the order of an optimistic estimate of the total cost of a solution passing through that state (Figure 1C). This estimate is the cost of reaching that state plus a lower bound on the cost from that state to the goal, which is given by the heuristic function. For example, when navigating to the café, one might use Euclidean distance as a heuristic, which is optimistic because it assumes you can walk directly towards your destination (e.g., no obstacles will be in the way). By prioritizing states that are more promising (as estimated by the heuristic), A* can search far fewer states than BFS, resulting in a lower search cost, $C_{\mathrm{A}^*}(s,z)$.

Subtask-Level Planning

A number of formalisms have been used to model hierarchical decision-making (Sutton et al., 1999; Dieterich, 2000; Parr & Russell, 1998). Here, we assume a simple model of hierarchical planning that involves only a single level above action-level planning, which we call *subtask-level planning*. Formally, subtask-level planning occurs over a set of subgoals, $\mathcal{Z} \subset \mathcal{S}$.[1] Given a set of subgoals, subtask-level planning consists of choosing the best sequence of subgoals that accomplish a larger goal. Each subgoal is then provided to the action-level planner in turn, and the resulting action-level plans are combined into a complete plan to reach the goal
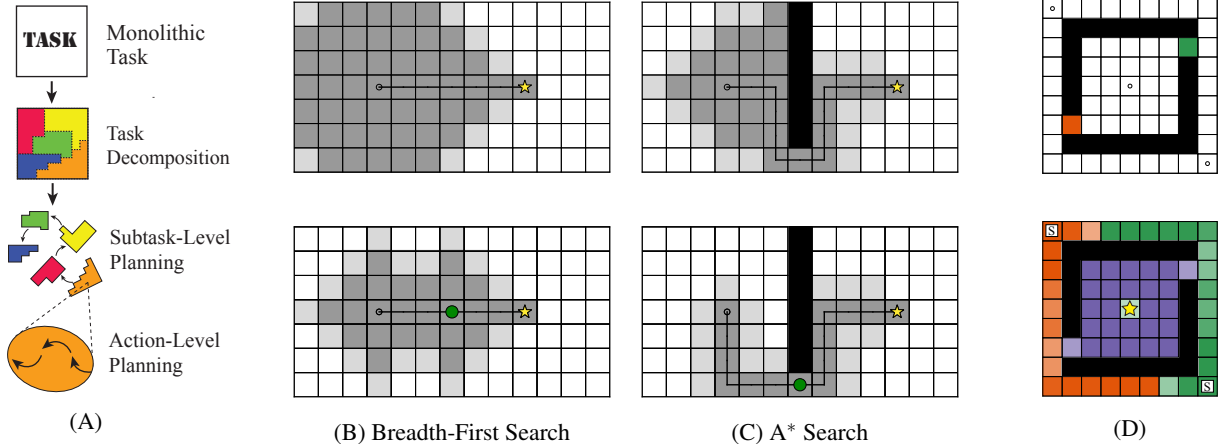
Figure 1: (A) Our account relates three levels of optimization during problem solving: task decomposition, subtask-level planning, and action-level planning. (B) *Top*: Planning with breadth-first search (BFS) from a start state (small circle) to the goal state (star) with no subgoals. Grey squares indicate nodes visited during search process (49 nodes visited). *Bottom*: BFS with the optimal size 1 task decomposition, where the green circle represents a subgoal (26 nodes visited). (C) *Top*: Planning with A* search with a Manhattan distance heuristic and no subgoals (28 nodes visited). *Bottom*: A* search with the optimal size 1 task decomposition (13 nodes visited). (D) Indoor/Outdoor domain with optimal size 2 task decomposition. *Top*: The three dots represent possible start/goal states. The green and red tiles are the non-trivial subgoals in the optimal task decomposition. *Bottom*: Subtask-level policy when the center tile is the goal (star) and the diagonal corners are possible start states ("S"). Colors correspond to the most likely subgoal chosen at each state. Purple is the trivial "go to goal" subtask.

state. For example, when navigating to the post office, the subtask-level planner might decide to first go to the café and then go to the post office from there, and the action-level planner would figure out the precise sequence of steps to get from work to the café and from the café to the post office.

The objective of the subtask-level planner is to identify the sequence of subgoals that brings the agent to the goal state while maximizing task rewards and minimizing computational costs. Here, we focus on tasks in which the task is simply to reach the goal state in as few steps as possible. Additionally, note that we only consider action-level planners that return the optimal shortest path. Thus, formally, the task reward associated with choosing a subgoal z from state s is the negative distance: $R(s,z) = -D(s,z)$. We can then compactly express the optimization problem faced by the subtask-level planner as a Bellman equation (Bellman, 1957). Given a task goal g, a set of subgoals $\mathcal{Z}$, and an algorithm with a search cost function $C_{\texttt{Alg}}$, the optimal subtask-level planning utility from any state $s \in \mathcal{S}$ is:

$$V_{\mathcal{Z}}^g(s) = \max_{z \in \mathcal{Z}} \left\{ R(s,z) - C_{\texttt{Alg}}(s,z) + V_{\mathcal{Z}}^g(z) \right\}. \quad (1)$$

The fixed point of Equation 1 can be used to identify the optimal subtask-level policy (Puterman, 1994). Additionally, we assume that the ultimate goal, g, is always included in $\mathcal{Z}$ to ensure that it is possible for the subtask-level planner

[1]For readers familiar with the *options framework* (Sutton et al., 1999), we note that what we call a subgoal is equivalent to a simple option where the set of initial states is the full state space, $\mathcal{S}$, and the termination function is $\beta(s) = \mathbf{1}(s = z)$. This means that subtask-level planning is a semi-Markov decision process.

to solve the task. Finally, although we do not explore this possibility here, note that this formulation allows us to easily express tradeoffs between task rewards, R, and algorithm-specific computation costs, $C_{\texttt{Alg}}$.

Task Decomposition

Having defined action-level planning and subtask-level planning over subgoals, we can now turn to our original motivating question: How should people decompose tasks? In this context, this reduces to the problem of selecting the best set of subgoals. Importantly, we assume that people rely on a common set of subgoals for all the different possible tasks that they might have to accomplish in a given environment. For example, *be at the downtown train station* is a good subgoal because it is often along relatively-optimal paths, whereas *be at a friend's place on the other side of town* is probably not a good subgoal because it is only relevant when visiting that friend.

We formalize subgoal selection as an optimization problem: Identify the set of subgoals, $\mathcal{Z}^*$, that maximize the value attained by the subtask-level planner on average. That is,

$$\mathcal{Z}^* = \operatorname*{argmax}_{\mathcal{Z}} \mathbb{E}_{s,g}[V_{\mathcal{Z}}^g(s)], \quad (2)$$

where the expectation is with respect to a *task distribution*, $p(s,g)$, over starting states s and goals g. Importantly, this objective takes into account both the expected task rewards and the costs of action-level planning mediated by subtask-level planning.
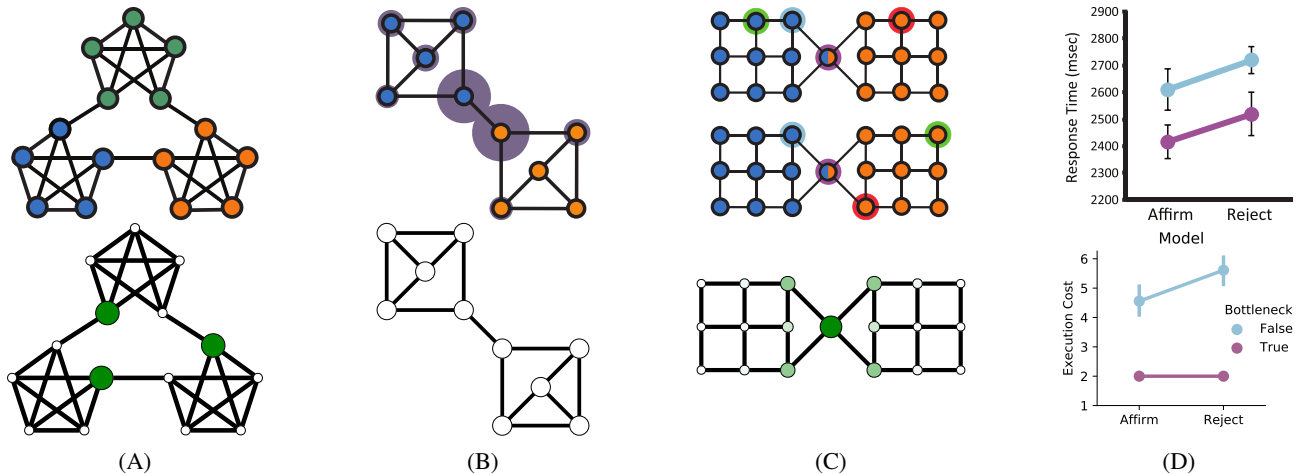
Figure 2: Analyzing Solway et al. (2014) Experiments 1-3 with uninformed search. (A) *Top*: Solway et al. (2014) decomposition of graph from **Schapiro et al. (2013)**. *Bottom*: Resource-rational task decomposition. (B) *Top*: Graph used in **Experiment 1** from Solway et al. with proportion of "bus stops" placed at states. *Bottom*: Subgoals do not facilitate more efficient breadth-first search (BFS) in this graph, so our model does not learn to use any states as subgoals. (C) *Top*: Graph for **Experiment 2** with bottlenecks depicted. Same graph was used in Experiment 3. *Bottom*: Distribution over subgoals z from softmax of value of subgoal with inverse temperature of 100. Task decomposition facilitates more efficient BFS in this graph. In particular, the "bottleneck state" connecting the two regions is the optimal subgoal, followed by the adjacent states. (D) *Top*: **Experiment 3** in Solway et al. probed whether participants' plans included states in the graph. Responses were significantly faster for bottleneck (purple) compared to non-bottleneck states (blue). *Bottom*: Simulations of probe response computations in experimental trials with the optimal task decomposition learned by our model. Our model takes fewer steps to respond when the probe is a bottleneck state, mirroring the experimental findings.

Implementation

The code for all the analyses we report is available at https://bit.ly/2T44Tun. Here, we briefly describe the implementation. For both BFS and A*, we calculated action-level computational costs $C_{\texttt{Alg}}(s,z)$ and minimum path lengths $D(s,z)$ for every state $s \in \mathcal{S}$ and subgoal $z \in \mathcal{S}$. With these quantities, a set of subgoals $\mathcal{Z}$, and distribution over goals and starting states $p(s,g)$, we can define the optimal expected subtask-level planning value function, $\mathbb{E}_{s,g}\left[V_{\mathcal{Z}}^{g}(s)\right]$ (see Equation 1). We compute this function using value iteration with a threshold of $\varepsilon = 10^{-5}$ (Bellman, 1957).

Finally, to solve for the optimal set of subgoals, $\mathcal{Z}^{*}$, we explored two methods. The first was an exact method—enumeration and evaluation of all subgoal sets. The second was a gradient-based method. This method used a differentiable version of value iteration at the subtask-planning stage (Haarnoja et al., 2017; Ho et al., 2020) and distributions over subgoals instead of discrete subgoals at the task decomposition level. While enumeration is intractable for large state spaces, we found that the methods produced similar results when both were feasible. Thus, we present results using the exact enumeration method when it was computationally tractable—for the small environments used in Solway et al. (2014)—and the gradient method otherwise.

Gridworld Simulations

To illustrate the properties of our model, we begin by analyzing optimal decompositions of simple gridworld tasks. The grids we tested include *Open Field*, *2-Room*, and *Indoor/Outdoor*. We tested our model with both BFS and A*. As shown in Figure 1, our model produces intuitive task decompositions as a function of a task and planning algorithm.

Analysis of Solway et al. (2014) Experiments

Solway et al. (2014) reported four studies that investigated how people decompose tasks and engage in hierarchical planning. Here, we ask if our resource-rational model can account for these findings. We first discuss Experiments 1-3 (Figure 2), which relied on tasks in which people could not leverage prior knowledge and then turn to Experiment 4 (Figure 3), which used the Tower of Hanoi (Nilsson, 1971), a task that allows for the use of prior knowledge.

Experiments 1-3: Uninformed Search

Summary of Findings Experiments 1-3 reported by Solway et al. (2014) experimentally tested the hierarchical structure used by participants when performing navigation tasks over abstract state spaces. Figures 2B and 2C show the connectivity structure of the domains people were given.

A key qualitative finding reported by Solway et al. (2014) was that people's responses reflected a decomposition of the
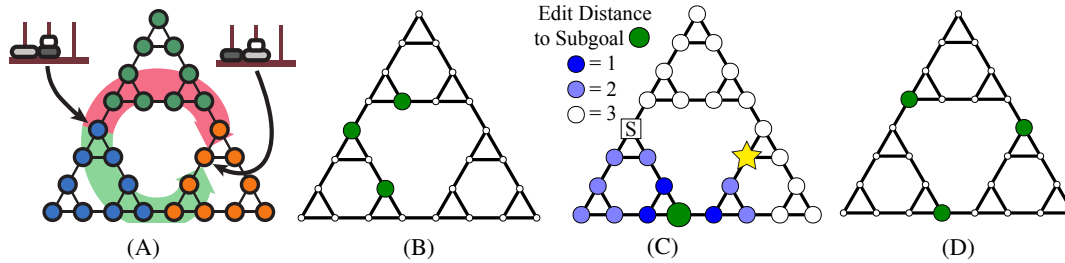
Figure 3: Analyzing Solway et al. (2014) Experiment 4. (A) Hierarchically optimal (green) vs. non-hierarchically optimal (red) paths in Tower of Hanoi. (B) Optimal task decomposition with BFS has one bottleneck subgoal and two neighboring non-bottleneck states as subgoals. (C) Edit distance to a subgoal (green circle). Edit distance provides useful local information for heuristic search. (D) Optimal task decomposition with A* and edit distance heuristic has three equidistant bottleneck states as subgoals.

state space based on *bottlenecks*, states or transitions that connect more densely-connected regions of a state space (Şimşek & Barto, 2009). A preliminary modeling result they reported recovered a task decomposition along community boundaries in a graph (Figure 2A) studied in Schapiro et al. (2013). Experiment 1 assessed this in the task with the transition structure in Figure 2B (top) by having participants choose a "bus stop" that would be most useful for making "deliveries" between locations represented by the icons. Participants overwhelmingly chose the bottleneck states, as displayed in the figure. Experiment 2 had participants actually navigate to and from random locations in the task with the structure in Figure 2C. However, on test trials, participants were asked to either identify locations along the path *in any order* or identify a single location on the path. Participants tended to report bottleneck states first, suggesting they were thinking about these states first in their planning process. Finally, Experiment 3 used the same domain as Experiment 2, but participants were probed about whether a state was on the optimal path between two states. Participants answered faster for bottleneck states compared to non-bottleneck states, providing additional evidence that these states are the first to come to mind (Figure 2D).

Model Implementation and Results For the Schapiro et al. graph (Figure 2A), we assumed a uniform distribution over all start and goal states, and set the number of subgoals to $|\mathcal{Z}| = 3$. The best subgoals separated the three communities at their boundaries, allowing the action-planner to first find the community containing the goal and then search for the goal within that community.

For the second graph (Figure 2B; Solway et al. Experiment 1), we again assumed a uniform distribution over all start and goal states, and set the number of subgoals to $|\mathcal{Z}| = 1$. Interestingly, no set of subgoals achieved greater value than planning without subgoals: $\mathcal{Z}^* = \emptyset$. Thus, the model did not reflect the empirical results. Although "bus stop" judgments recovered bottleneck states, they may reflect a process that is distinct from task decomposition for planning. Further experiments are needed to evaluate this.

We made the same assumptions for the third graph (Figure 2C; Experiments 2 & 3), and found that the bottleneck state was always in the optimal decomposition. The next-best decompositions all included one of the four states connected to the bottleneck state, indicating that navigating near the bottleneck state is a useful subgoal in this task.

To replicate the reaction time results reported in Experiment 3 (Figure 2D), we simulated hierarchical planning with the optimal decomposition using trials as described by Solway et al. Specifically, the model first constructed a subtask-level plan. If the queried state was a subgoal state, the model replied "Affirm" as soon as the subtask-level planner encountered the state and "Reject" if the state was not in the completed subtask-level plan. If the queried state was not a subgoal state, the model proceeded to construct each action-level plan in turn. As soon as the queried state was encountered, the model replied "Affirm". If the final action-level plan to the goal was completed without encountering the state, the model replied "Reject". In either case, we used the total number of subgoals and states that were simulated before the response was produced as a proxy for reaction time. These results are plotted for bottleneck vs. non-bottleneck probes and "Affirm" vs. "Reject" type probes in Figure 2D.

Experiment 4: Tower of Hanoi and Heuristic Planning

Summary of Findings In a final experiment, Solway et al. tested participants solving the Tower of Hanoi (Nilsson, 1971). The experiment focused on "problems of interest", trials where two paths of the same length led to a goal but one crossed more bottleneck states. They found that participants preferred to take paths that crossed fewer bottleneck states. Assuming that people prefer hierarchically shorter paths (i.e., ones that use fewer subgoals), this has been taken to reflect a decomposition of the task based on bottleneck states.

Model Implementation and Results The Tower of Hanoi is an important contrast to the tasks in the first three experiments because states have features that provide clues for search. For example, the *edit distance* between two states provides an optimistic estimate of their minimum path length: it

ignores that some transitions are forbidden and assumes you can rearrange blocks arbitrarily. Much like how spatial distance can guide planning in navigation tasks, heuristics like edit distance can guide problem solving in structured tasks.

To understand the relationships between heuristics, task decomposition, and Solway et al.'s results, we ran several versions of our model on the Tower of Hanoi. We set the number of subgoals to $|\mathcal{Z}| = 3$ and used BFS as the action-level planner. Notably, the optimal subgoals under this scheme were systematically "skewed", consisting of a bottleneck state and two nearby points (Figure 3B). For our second simulation, we used the same procedure and parameters, but rather than using BFS (uninformed search), we used A* with an edit-distance heuristic for action-level planning (Figure 3C). The top two decompositions both contained three bottleneck states in separate communities (Figure 3D). Unlike BFS, A* can efficiently navigate between these points, allowing for a task decomposition that spans the full extent of the problem space.

Discussion

We have proposed a resource-rational account of task decomposition based on the idea that subgoals are decomposed to make planning easier. Our model specifies three levels of nested optimization: *Task decomposition* identifies a set of subgoals for a given domain, *subtask-level planning* chooses sequences of subgoals to reach a goal, and *action-level planning* chooses sequences of concrete actions to reach a subgoal. Optimal task decomposition thus depends on both the structure of the environment and the computational resource usage specific to the planning algorithm. We find that our model produces interpretable task decompositions in grid-world tasks and decompositions consistent with three of the four findings reported by Solway et al. (2014).

The model presented here departs from and complements other normative proposals in the literature. Most existing approaches pose task decomposition as an *inference problem*: People are modeled as inferring a generative model of the environment (Collins & Frank, 2013; Tomov et al., 2020) or as compressing optimal behavior (Solway et al., 2014; Maisto et al., 2015). In contrast, we pose task decomposition as a *resource-rational representation problem*: People are modeled as having subgoals that reduce the computational overhead of action-level planning. This change in view has several consequences worth noting.

First, unlike inferential approaches that abstract away the underlying reasoning process, our framework requires specifying a planning algorithm. Different assumptions at this lowest level (e.g. using breadth first search or A*) can dramatically influence the task decomposition (e.g., Figure 3). On the one hand, this makes model identification more challenging since the space of planning algorithms and parameterizations is vast. On the other hand, because our model is both algorithmic and normative, it can characterize the interplay of planning computations and representations in a well-posed manner. Additionally, this approach allows us to an-

alyze how behavioral suboptimality can arise from rational tradeoffs between task rewards and computation costs associated with particular search algorithms. Future empirical work on resource-rational planning representations will need to examine these questions in greater depth.

A second difference is that inferential models generally emphasize learning from task interactions as data, while we have deliberately set aside how resource-rational decompositions are learned. Specifically, our formulation assumes the existence of an optimization process that can select a decomposition, whether it be through direct experience with a task or other means. Although this temporarily defers important and interesting questions about online problem solving, characterizing any learning process requires first identifying what is being learned (i.e., what is being optimized). It is in this sense that the model presented here is a resource-rational analysis of task decomposition (Griffiths et al., 2015).

More broadly, the work presented here is consistent with other recent efforts within cognitive science to understand how people engage in computationally efficient decision-making (Griffiths et al., 2015; Lewis et al., 2014; Gershman et al., 2015; Lieder & Griffiths, 2020). It is also complementary to recent work in artificial intelligence that explores the interaction between planning and task representations (Jinnai et al., 2019; Harb et al., 2018). Our hope is that future work on human planning and problem solving will continue to investigate the relationships between computation, representation, and resource-rational decision-making.

Acknowledgements

References

Anderson, J. R. (1990). *The Adaptive Character of Thought*. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Balaguer, J., Spiers, H., Hassabis, D., & Summerfield, C. (2016). Neural mechanisms of hierarchical planning in a virtual subway network. *Neuron*, *90*(4), 893 - 903.

Bellman, R. (1957). *Dynamic programming*. Princeton University Press.

Botvinick, M. M. (2012). Hierarchical reinforcement learning and decision making. *Current Opinion in Neurobiology*, *22*(6), 956–962.

Botvinick, M. M., Niv, Y., & Barto, A. C. (2009). Hierarchically organized behavior and its neural foundations: A reinforcement learning perspective. *Cognition*, *113*(3), 262–280.

Callaway, F., Lieder, F., Das, P., Gul, S., Krueger, P., & Griffiths, T. (2018). A resource-rational analysis of human planning. In *Proceedings of the Annual Conference of the Cognitive Science Society*.

Collins, A. G., & Frank, M. J. (2013). Cognitive control over learning: Creating, clustering, and generalizing task-set structure. *Psychological Review*, *120*(1), 190–229.

Cushman, F., & Morris, A. (2015). Habitual control of goal selection in humans. *Proceedings of the National Academy of Sciences*, *112*(45), 13817–13822.

Dietterich, T. G. (2000). Hierarchical Reinforcement Learning with the MAXQ Value Function Decomposition. *Journal of artificial intelligence research*, *13*, 227–303.

Gershman, S. J., Horvitz, E. J., & Tenenbaum, J. B. (2015). Computational rationality: A converging paradigm for intelligence in brains, minds, and machines. *Science*, *349*(6245), 273–278.

Griffiths, T. L., Lieder, F., & Goodman, N. D. (2015). Rational Use of Cognitive Resources: Levels of Analysis Between the Computational and the Algorithmic. *Topics in Cognitive Science*, *7*(2), 217–229.

Haarnoja, T., Tang, H., Abbeel, P., & Levine, S. (2017). Reinforcement learning with deep energy-based policies. In *ICML* (pp. 1352–1361).

Harb, J., Bacon, P.-L., Klissarov, M., & Precup, D. (2018). When waiting is not an option: Learning options with a deliberation cost. In *Thirty-Second AAAI Conference on Artificial Intelligence.*

Hart, P. E., Nilsson, N. J., & Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, *4*(2), 100-107.

Ho, M. K., Abel, D., Cohen, J., Littman, M. L., & Griffiths, T. L. (2020). The Efficiency of Human Cognition Reflects Planned Information Processing. In *Thirty-Fourth AAAI Conference on Artificial Intelligence.*

Ho, M. K., Abel, D., Griffiths, T. L., & Littman, M. L. (2019). The value of abstraction. *Current Opinion in Behavioral Sciences*, *29*, 111–116.

Huys, Q. J. M., Eshel, N., O'Nions, E., Sheridan, L., Dayan, P., & Roiser, J. P. (2012). Bonsai trees in your head: how the Pavlovian system sculpts goal-directed choices by pruning decision trees. *PLoS Computational Biology*, *8*(3), e1002410.

Huys, Q. J. M., Lally, N., Faulkner, P., Eshel, N., Seifritz, E., Gershman, S. J., . . . Roiser, J. P. (2015). Interplay of approximate planning strategies. *Proceedings of the National Academy of Sciences*, *112*(10), 3098–103.

Jinnai, Y., Abel, D., Hershkowitz, D., Littman, M., & Konidaris, G. (2019). Finding options that minimize planning time. In *ICML* (Vol. 97, pp. 3120–3129).

Kaplan, C. A., & Simon, H. A. (1990). In search of insight. *Cognitive Psychology*, *22*(3), 374 - 419.

Keramati, M., Smittenaar, P., Dolan, R. J., & Dayan, P. (2016). Adaptive integration of habits into depth-limited planning defines a habitual-goal–directed spectrum. *Proceedings of the National Academy of Sciences*, *113*(45), 12868–12873.

Lewis, R. L., Howes, A., & Singh, S. (2014). Computational rationality: Linking mechanism and behavior through bounded utility maximization. *Topics in Cognitive Science*, *6*(2), 279-311.

Lieder, F., & Griffiths, T. L. (2020). Resource-rational analysis: understanding human cognition as the optimal use of limited computational resources. *Behavioral and Brain Sciences*, 1–60.

MacGregor, J. N., Ormerod, T. C., & Chronicle, E. P. (2001). Information processing and insight: a process model of performance on the nine-dot and related problems. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *27*(1), 176.

Maisto, D., Donnarumma, F., & Pezzulo, G. (2015). Divide et impera: subgoaling reduces the complexity of probabilistic inference and problem solving. *Journal of The Royal Society Interface*, *12*(104), 20141335–20141335.

Newell, A., & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice-Hall.

Nilsson, N. J. (1971). *Problem-solving methods in artificial intelligence*. McGraw-Hill.

Parr, R., & Russell, S. J. (1998). Reinforcement learning with hierarchies of machines. In *Advances in Neural Information Processing Systems* (pp. 1043–1049).

Puterman, M. L. (1994). *Markov decision processes: Discrete stochastic dynamic programming*. John Wiley & Sons, Inc.

Ribas-Fernandes, J., Solway, A., Diuk, C., McGuire, J., Barto, A., Niv, Y., & Botvinick, M. (2011). A neural signature of hierarchical reinforcement learning. *Neuron*, *71*(2), 370 - 379.

Russell, S., & Norvig, P. (2009). *Artificial intelligence: A modern approach* (3rd ed.). USA: Prentice Hall Press.

Sacerdoti, E. D. (1974). Planning in a hierarchy of abstraction spaces. *Artificial Intelligence*, *5*(2), 115–135.

Schapiro, A. C., Rogers, T. T., Cordova, N. I., Turk-Browne, N. B., & Botvinick, M. M. (2013). Neural representations of events arise from temporal community structure. *Nature Neuroscience*, *16*(4), 486–492.

Şimşek, Ö., & Barto, A. G. (2009). Skill characterization based on betweenness. In *Advances in Neural Information Processing Systems* (pp. 1497–1504).

Solway, A., Diuk, C., Córdova, N., Yee, D., Barto, A. G., Niv, Y., & Botvinick, M. M. (2014). Optimal behavioral hierarchy. *PLoS Computational Biology*, *10*(8), e1003779.

Sutton, R. S., Precup, D., & Singh, S. (1999). Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, *112*(1-2), 181–211.

Tomov, M. S., Yagati, S., Kumar, A., Yang, W., & Gershman, S. J. (2020). Discovery of hierarchical representations for efficient planning. *PLOS Computational Biology*, *16*(4).

van Opheusden, B., Galbiati, G., Bnaya, Z., Li, Y., & Ma, W. J. (2017). A computational model for decision tree search. In *Proceedings of the Annual Conference of the Cognitive Science Society.*