Comparing AI planning algorithms with humans on the Tower of London task

Anonymous submission

Abstract

Understanding problem solving or planning has been a shared challenge for both AI and cognitive science since the birth of both fields. We explore the extent to which modern planners from the field of AI can account for human performance on the Tower of London (TOL) task, a close relative of the Tower of Hanoi problem that has been extensively studied by psychologists. We characterize the task using the Planning Domain Definition Language (PDDL) and evaluate an adaptive online planner and a family of well-known planners, including online planners, optimal planners and satisficing planners. Each planner is evaluated based on its ability to predict the actions and planning times of participants in a new behavioral experiment. Our results suggest that participants use a range of strategies but that an adaptive lookahead planner provides the best overall account of both human actions and human planning times. This finding is consistent with the view that humans differ from standard AI planners by integrating a mechanism for evidence accumulation.

Introduction

When preparing a three course meal, fixing a leaky pipe or building a garden box, people must string together a sequence of actions in order to achieve a goal. Tasks like these are typically known as problem solving tasks by cognitive scientists and planning tasks by AI researchers. Problem solving or planning is a hallmark of intelligent behavior, and has been extensively studied by both AI researchers and cognitive scientists since the development of the Logic Theorist in 1956, a theorem prover sometimes described as the first AI program (Gugerty 2006; Newell and Simon 1956). In subsequent decades, psychologists have studied human performance on a wide range of problem solving tasks, including water jug problems (Atwood and Polson 1976) and the tower of Hanoi (Kotovsky, Hayes, and Simon 1985).

Planned behavior can be distinguished from reflex behavior, similar to the distinctions regarding goal-directed versus habitual behavior, model-based versus model-free decision making, and type II versus type I reasoning (Mattar and Lengyel 2022). Reflex-based approaches do not consider the outcome of each action or evaluate the utility of these outcomes, which limits their ability to perform well in dynamic situations. Both approaches are used by humans and animals, and in order to study human planning behavior, it is essential to choose tasks that cannot be solved through reflexive behavior, forcing participants to rely on planned behavior to solve the problem.

In this study, we build on an approach to planning that was initially developed by researchers including Newell, Simon et al. (1972). We have selected a simple task that is suitable for laboratory study and is unlikely to be solved through reflexive behavior (Mattar and Lengyel 2022). For us the task is the Tower of London (TOL) problem, a variant of the well-known Tower of Hanoi problem. Our goal is to identify a planning algorithm that matches human performance on the TOL task, and towards that end we evaluate a set of planning algorithms including several inspired by state-of-the art approaches in AI. Our approach therefore falls squarely in the tradition established by researchers like Newell and Simon who used computational models such as the General Problem Solver (GPS) to account for human performance on tasks like the Tower of Hanoi (Newell, Simon et al. 1972; Kotovsky, Hayes, and Simon 1985).

The Newell-Simon approach to problem solving arguably reached its pinnacle in the 1970s, and has been pursued less actively from the mid 1990s onwards (Ohlsson 2012). There are at least two reasons, however, why this approach may be worth revisiting. First, AI researchers have developed new approaches to planning that may help to capture aspects of human problem solving. For example, from the mid 1990s modern planning algorithms have relied on domain-general heuristics that can be derived automatically from a problem representation via *relaxations* (Geffner 2013). This approach to deriving heuristics could potentially lead to new models of human problem solving that can be applied to broad families of problems without requiring problem-specific strategies.

Second, psychologists have continued to construct new models to account for several aspects of human decision making (Solway and Botvinick 2015; Mormann et al. 2010). A key issue explored in recent modeling work is the trade-off between time cost and decision quality. Models exploring this idea build on the idea of bounded rationality (Simon 1990) and the framework of rational analysis (Anderson 1989). An agent that makes optimal use of bounded cognitive resources must decide when to stop the search process and act, and recent work on metareasoning has explored this *stopping problem* (Anderson and Oates 2007; Tajima et al. 2019). Solway and Botvinick (2015) use an evidence accu-

mulation mechanism to model performance in a two-step decision problem, but applying a similar approach to more complex sequential decision making problems (e.g. TOL) is a challenge that has not yet been addressed.

The next section summarizes some of the previous computational work on problem solving that forms the backdrop for the work described here. We then describe the Tower of London task and the behavioral experiment. The following sections introduce the specific planners that we evaluate and discusses the extent to which they account for the behavioral data. To preview our results, we find that people tend to use different strategies under different conditions and that the adaptive lookahead planner provides the best overall account of human performance.

Models of human problem solving

Perhaps the most influential cognitive model of problem solving is the General Problem Solver (Newell, Simon et al. 1972) and this model can be regarded as a variant of breadth first search. Subsequent work in this tradition used production systems such as ACT-R (Lebiere and Anderson 1993), 4CAPS (Varma and Just 2006) and SOAR (Laird, Newell, and Rosenbloom 1987) to develop models of problem solving on tasks including the Tower of Hanoi (Ruiz and Newell 1989) and the Tower of London (Varma and Just 2006).

In recent years researchers have departed from the earlier emphasis on production systems by considering a range of alternative approaches. Kuperwajs, Van Opheusden, and Ma (2019) used a tree search model with a domain-specific heuristic to predict human performance on a two-player game. Working within the framework of bounded rationality, Callaway et al. (2018) derived a meta-level Markov decision process model to simulate human behavior on a navigation task known as Mouselab. Donnarumma, Maisto, and Pezzulo (2016) developed an approach that combines probabilistic inference with subgoaling to account for human performance on the Tower of Hanoi task.

Across the recent literature there is evidence that the extent to which people look ahead while planning varies across individuals and across tasks (Callaway et al. 2021; Kryven et al. 2022). Meder et al. (2019) found that an approach that looks ahead only one step provided the best account of human performance in the 20-questions game, while Krusche et al. (2018) found that people have a planning horizon of at least 3 steps in the farming game that they considered. Several studies demonstrate that time pressure can lead to a shallower search tree (Keramati et al. 2016; Van Opheusden et al. 2017).

Most recent studies use non-deterministic or partially observable environments so that humans cannot easily derive optimal solutions (Kryven et al. 2022; Krusche et al. 2018), and there has been relatively little work on fully observable deterministic environments (e.g. TOL) in recent years. Our work, however, belongs to the Newell and Simon tradition that explores what can be learned from human performance on deterministic, fully observable-tasks.

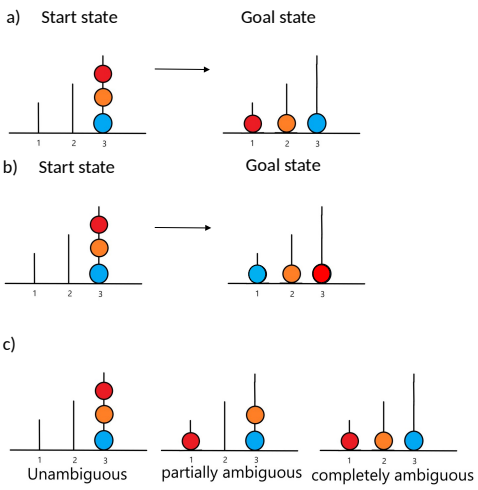

Figure 1: Tower of London problem. (a) A problem instance requiring two moves to transition from the start state to the goal state. (b) A problem instance requiring five moves. (c) *Start Hierarchy* is a structural parameter that classifies each instance as unambiguous (all balls on one peg), partially ambiguous or completely ambiguous (all balls on different pegs). The "ambiguity" refers to the initial action: unambiguous actions allow only one action, but completely ambiguous instances allow 4 possible actions.

The Tower of London task

Figure 1a shows an instance of the problem. The board illustrated in the figure has pegs that can hold up to 1, 2 and 3 balls respectively from left to right. Participants are given the board in some initial state, then asked to move balls from peg to peg until the board matches some specified goal state. The instance in Figure 1a can be solved in just two moves, but the shortest solution of the instance in Figure 1b involves 5 moves.

Previous work on the TOL has focused on identifying structural parameters that appear to influence the difficulty of a problem instance, and one such parameter is shown in Figure 1c (Kaller et al. 2004, 2011; Berg et al. 2010). Berg et al. (2010) carried out an experiment in which participants solved a set of TOL problems with optimal solutions of length between 4 and 7, and used their data to evaluate how 5 structural parameters relate to measures of human performance. A small amount of work has attempted to model the actions people choose when solving TOL problems (Varma and Just 2006; Donnarumma, Maisto, and Pezzulo 2016), but to our knowledge no previous work on the TOL task attempts to model both action selection and planning time as we do here.

AI planning and planners

Planning is the model-based approach to reasoning about the action(s) needed to achieve a goal given an initial scenario. In order to apply AI planners to the TOL problem, we translate the task into the propositional subset of the Planning

Domain Definition Language (PDDL), which is a standard language for modelling planning problems that extends the expressivity of the well known STRIPS language (Haslum et al. 2019). To encode the height constraints in the task, we simply enumerate all possible ball locations. In our setting, since there are just three pegs with heights of 1, 2, and 3 respectively, we have 6 different locations in total. In each state, there is a fluent (proposition) for each ball recording its current location. In addition, we also mark whether each ball is free to move and whether each location is available. For example, in the start state of Figure 1a, the red ball is in LOC3-3 (the third position on peg 3). There is no other ball on the red ball, so it is free to move to other locations. LOC1-1 is available, so we can execute the action that moves the red ball from LOC3-3 to LOC1-1 and the successor state is the middle state in Figure 1c.[1].

All of the AI planners evaluated here use the representation just described, but there is another way to model the problem within the PDDL framework. Namely, we can decompose each move action into two steps: first pick up a ball from one peg and then put it down on a peg. The major advantage of this approach is that it allows a player to pick up a ball then return it to the same peg, which occurs occasionally in our behavioral data. However, most previous work on the TOL treats each move as a single action and we follow the same approach for consistency.

Our model evaluation aimed to consider a set of planning algorithms (i.e. planners) that is broadly representative of prior work on planning in the fields of AI and psychology. The following sections describe the 6 different planners that we settled on.

Cognitive Architecture

4caps We chose 4caps to represent the broader family of cognitive architectures because an existing 4caps model of the TOL task is publicly available, and has previously been used to account for both behavioral and brain imaging data (Varma and Just 2006; Newman et al. 2003). This model includes some productions that are specific to the TOL task, and therefore does not qualify as a fully general model of problem solving.

Classical planners

Classical planners search until a complete path to the goal has been found.

BFS The three-peg TOL problem is sufficiently small that Breadth First Search (BrFS) is a viable algorithm. BrFS first tries all possible actions from the start state, and adds all states reached in this way to a queue. It then repeatedly takes a state from the front of the queue, tries all actions from that state, and adds all resulting states to the end of the queue, effectively always expanding the state closest to the initial state that has not been expanded yet. Proceeding in this way guarantees that BrFS will find an optimal solution, but the algorithm is blind because it does not consider the goal when choosing the state to expand next.

ASTAR ASTAR search algorithm (Hart, Nilsson, and Raphael 1968) is commonly used as a baseline heuristic search planner in AI planning research. A heuristic is a function that takes a state as input and returns an estimate of the distance between the state and the goal. A heuristic-based algorithm can therefore potentially capture the idea that people are most likely to focus on intermediate states that promise to bring them closer to their ultimate goal. If equipped with an admissible heuristic, then ASTAR is guaranteed to find an optimal solution.[2] When choosing which state to expand next, ASTAR picks the state that minimizes the cost to reach that state plus the heuristic estimate of the distance to the goal. Here we use the *goal-counting* heuristic, a domain-independent heuristic that can be automatically derived from the PDDL description of the problem, which evaluates a state based on how many goals are yet to be achieved (in our case, how many balls are not yet in their final positions).[3] This heuristic is equivalent to the "perceptual distance" heuristic in the psychological literature (Donnarumma, Maisto, and Pezzulo 2016), and has been explored by researchers including Simon (1963).

GBFS The heuristic search algorithm used in most state-of-the-art satisficing planners is greedy best-first search (GBFS) (Heusner, Keller, and Helmert 2017). In contrast to ASTAR, GBFS expands states using only the heuristic function, and chooses the state that lies closest to the goal according to this function. GBFS is not guaranteed to find an optimal solution and hence produces satisficing planners that trade off solution quality and solution speed. When combined with the goal-counting heuristic, GBFS yields a search strategy that captures some of the core ideas of means-ends analysis (Newell, Simon et al. 1972).

Online planners

Online planners are able to choose an action before a complete path has been found, and have been previously explored as models of human problem solving (Kuperwajs, Van Opheusden, and Ma 2019; Krusche et al. 2018). One prominent approach is Monte-Carlo Tree Search, but we did not consider this approach because it is best-suited for stochastic environments and the TOL is a deterministic task. Instead, we evaluate two lookahead planners that both rely on the goal-counting heuristic.

Lookahead The basic lookahead planners we consider have a fixed horizon that was set to values from 1 to 7 (maximum solution length). The planner evaluates the value of a state recursively using the minimal state value of its successors, and the state values of all leaf nodes are based on the heuristic function (goal-counting in this work). After computing these state values, the planner chooses the path with

[1]The PDDL representation of the start state of Figure 1a is {(in RED LOC3-3), (in ORANGE LOC3-2), (in BLUE LOC3-1), (free LOC1-1), (free LOC2-1), (free LOC2-2), (clear RED)}

[2]A heuristic function is *admissible* if it never overestimates the real distance between a state and the goal state

[3]The goal-counting heuristic is admissible. The ASTAR planner in this paper is therefore guaranteed to find an optimal solution.

Algorithm 1: Adaptive Lookahead

Parameters: Decision Threshold θ, Exploration constant C
Input: Search space P, Search goal g, Current state s_0
Output: Action selected a, Number of expanded nodes n (used as a proxy for the planning time for this step)

```
Let tree = Tree(s_0), n = 0 {Construct a tree rooted on state s_0}
Let node = tree.root, v' = -∞, v'' = -∞ {v' and v'' denote the best child node value and second best child node value of the root node respectively}

while tree.root is expanded and |v' - v''| ≤ θ do
    while node is expanded do
        node = UCBSelect(node, C)
    end while

    if node.state is g then
        return a ← softmaxSelect(tree.root), n
    end if

    n ← n + 1
    for succ in P.successors(node.state) do
        node.children.add(Node(succ, gc(succ)))
        {Initialize the new generated node using goal counting heuristic}
        tree.update(gc(succ)) {Backpropogate the new evidence so values of all ancestor nodes are updated by averaging the obtained values of all visits passing through the node}
    end for

    Update v', v''
end while

return a ← softmaxSelect(tree.root), n
```

minimal estimated cost. If multiple paths have the same minimal value, the planner randomly chooses one of these paths.

Adaptive Lookahead (A-LH) Although many online planners (e.g. Monte-Carlo Tree search) use a fixed planning horizon or a pre-defined timing budget, a small amount of work in AI has explored methods for optimizing lookahead depth (Bulitko, Levner, and Greiner 2002). For example, Kryven et al. (2022) develop a model with an adaptive planning horizon for a task that involves navigating through a maze.

Here we propose and evaluate an adaptive lookahead planner (see Algorithm 1) that draws on prior work on evidence integration and human meta-reasoning (Solway and Botvinick 2015; Anderson and Oates 2007; Chenyuan Zhang and Lipovetzky 2023). To achieve a balance between exploration and exploitation, this planner uses the upper confidence bound (UCB) algorithm as an action selection strategy (Kocsis and Szepesvári 2006), and keeps searching (evidence integration) until enough nodes have been expanded to suggest that the difference in value between the best action and the second best action exceeds some decision threshold. Our implementation in this work sets the threshold θ to 1 because the goal-counting heuristic is integer-valued. The exploration constant in UCB algorithm is set to 1.

After the adaptive lookahead planner either identifies a goal or surpasses the decision threshold, it will employ the softmax probability distribution of node values to generate an action, which is a common choice to model the uncertainty of human action selection (Sutton and Barto 2018).

Implementation

All classical planners, as well as the heuristics were implemented using the LAPKT framework (Ramirez, Lipovetzky, and Muise 2015). BrFS, and the online planners were implemented in Python. For 4caps, we used v1.2 of the TOL model.

Behavioral experiment

To allow us to compare the planners just described, we ran a behavioral experiment to collect fine-grained behavioral data (including response times) as participants solve instances of the TOL. Berg et al. (2010) previously ran a comprehensive experiment on the TOL, but their data are not publicly available. We therefore ran our own experiment using the same problem instances that they considered.

Our experiment included two between-participant conditions: a *full* condition and a *no-constraint* condition. In the full condition participants were asked to form a full plan to the target configuration before making their first move, and given feedback after each instance indicating whether they had found an optimal solution. In the no-constraint condition participants were simply asked to solve the task without any further instruction. The *full* condition matches the procedure used by Berg et al. (2010), and explicitly instructs participants to act as a classical offline planner. In the absence of this instruction, we anticipated that participants would behave more like an online planner.

We pre-registered the behavioral experiment on AsPredicted (see https://aspredicted.org/STK_41D). The experiment was programmed in javascript using the jspsych toolbox (De Leeuw 2015).

Instances. Following Berg et al. (2010), we considered all 117 problem instances with optimal solutions between 4 and 7 in length. For each instance, we generate a corresponding PDDL file automatically using the Python package Tarski (Francés, Ramirez, and Collaborators 2018).

Participants. 239 participants from standard sample completed the experiment on Prolific. Participants were randomly assigned to one of the two conditions, and completed 39 TOL instances randomly picked from 117 instances. Our final data set included 130 participants in the full condition and 109 in the no-constraint condition.

Outliers. Observations with abnormal response times were excluded according to a preregistered criterion. For each instance, responses more than 3 standard deviations away from the mean initial planning time for that instance

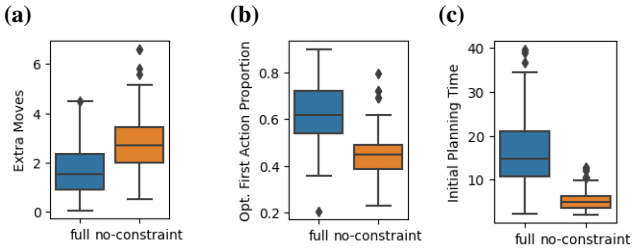

Figure 2: Comparison between the full and no-constraint conditions. (a) Extra moves (b) Optimal first action proportions (c) Initial planning times. Each data point shows mean performance per participant.

were considered abnormal. As a result, 239 out of 9321 (2.5%) responses are classified as outliers and excluded from our analysis.

Results

We consider two behavioral measures: the initial action selected for an instance and the initial planning time, or the time taken to select the initial action. Focusing on the first action only simplifies our analyses and facilitates comparisons across a relatively large set of planners.

Human performance in two conditions

We first compare human performance across the two conditions (full vs no-constraint) as shown in Figure 2. We focus on three performance measures. *Extra moves* (Figure 2a) is defined as the difference between the length of the plan provided by a participant and the length of the optimal plan. We also computed the proportion of participants who select an optimal first move (Figure 2b), and considered the time required to select this move (Figure 2c).

Figure 2 shows that participants in the full condition tend to generate plans that are 1.16 steps shorter than plans in the no-constraint condition, and that the first move in the full condition is more likely to be optimal (61% vs 44%). On average, however, participants in the full condition take an extra 11.23 seconds to produce this first move. Student's t-tests suggest that all three differences are statistically significant: extra moves ($t(238) = -8.12, p < 0.0001$), optimal first action proportion ($t(238) = 10.85, p < 0.0001$) and initial planning time ($t(238) = 15.07, p < 0.0001$).

Each data point in Figure 2 shows a participant rather than an instance, but an analysis at the level of problem instances produced converging results. For a given instance, plans generated in the full condition tend to have fewer steps ($t(116) = -7.99, p < 0.0001$), are more likely to include an optimal first move ($t(116) = 12.51, p < 0.0001$), and have a longer planning time for the the first move ($t(116) = 20.29, p < 0.0001$).

All of these results suggest that our condition manipulation had the expected effect, and that participants rely on different problem-solving strategies across the two conditions. We can now ask which planners provide the best account of responses in the two conditions.

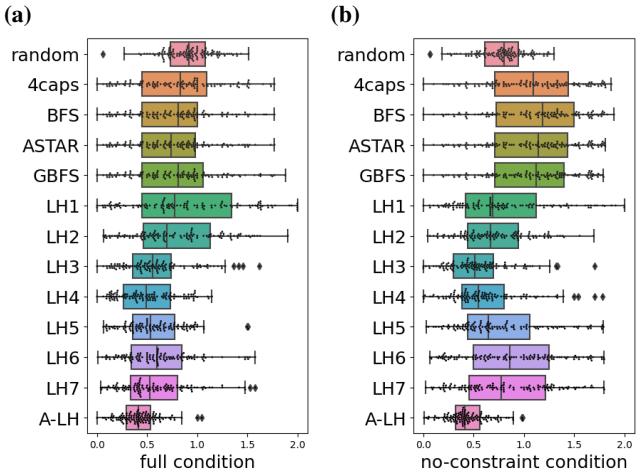

Figure 3: Evaluation of planner predictions about initial action selection. (a) Probability distance between planner distributions and the human distribution for the full condition. (b) Probability distance for the no-constraint condition. Each data point shows the probability distance for one instance, and smaller values of probability distance indicate better fits.

Predicting action selection

We first evaluate the extent to which planners can accurately predict the first action selected by participants. For each instance, we use the behavioral data to compute a distribution over initial actions chosen for that instance. We compare these distributions with distributions derived from the planners. All classical planners we used are non-stochastic, and most of the classical planner distributions therefore assign a probability of 1 to one action and a probability of zero to all remaining actions.

We compare human and planner distributions using *probability distance*, which is defined as the sum of the absolute differences in probability for each possible action. The results are summarized in Figure 3. Across both conditions, the online planners outperform the classical planners, and the adaptive lookahead planner achieves the best overall performance (smallest probability distance). The classical planners perform especially poorly in the no-constraint condition, where they are worse than the random baseline. It is not surprising that the classical planners perform poorly in the no-constraint condition, but it is striking that the best planners for the full condition are online planners, even though participants in this condition were instructed to behave like classical planners. This result suggests that the classical planners may have limited psychological validity even under conditions that are maximally favourable to these planners.

Predicting initial planning time

We now turn to initial planning times, and use mixed linear models to evaluate our family of planners. We first considered a regression model that is unrelated to all of our plan-

ners and has model string

$$\text{IPT} \sim 1 + \text{condition} + \text{order} + (1|\text{instance}) + (1|\text{participant})$$

The model takes initial planning time (IPT, measured in milliseconds) as the dependent variable, and includes fixed effects for condition (i.e. full or no constraint) and order (an integer from 1 to 39 that indicates the order in which a participant encountered a given instance). The model also includes random effects for problem instance and participant, and we obtained similar results regardless of whether instance is treated as a fixed or a random effect.

As expected, the base model performed better than the three simpler alternatives that omit either or both of the fixed effects. The Bayesian Information Criterion (BIC) was smaller for the base model than for the three alternatives by a factor of at least 81.

For the base model, the estimate for *condition* is 11192.72 (95%CI [9734.34, 12651.33]), which suggests that responses were around 11 seconds slower in the full condition compared to the no-constraint condition. The estimate for *order* was -102 (95%CI [-123.36, -81.19]), suggesting that participants became around 0.1 second faster with each additional instance that they solved. This order effect is consistent with the work of Berg et al. (2010), who report that solution times decrease with experience.

For each planner, we then asked whether the base model could be improved by replacing the random effect of instance with a fixed effect for planner response time, which is operationalized as the number of states expanded by a planner. For example, if the adaptive lookahead model predicted human planning times perfectly, then including response times for this model as a predictor in Equation should allow the resulting regression model to perfectly account for the human data. BIC values for each of these regression models are shown in Table 1. Among the fixed lookahead models, LH4 and LH6 achieved the best performance in the no-constraint and full conditions respectively. Table 1 also includes baselines that result from replacing the random effect in Equation with fixed effects for optimal cost (OC, or the length of the shortest solution) and start hierarchy (SH, see Figure 2ac). We consider both optimal cost and start hierarchy because these structural parameters predicted human performance best among the full set considered by Berg et al. (2010).

As expected, the online planners perform better than the classical planners in the no-constraint condition. In the full condition, one of the classical planners (ASTAR) performs relatively well but the best planner for this condition is the adaptive lookahead model. Our results for planning time are therefore broadly compatible with the finding in Figure 3 that the adaptive lookahead planner performs well across both conditions.

Table 1 reveals, however, that the single best predictor for the no-constraint condition is not a planner but rather the Start Hierarchy parameter shown in Figure 1c. It makes sense that participants should respond quickly when there is only one possible initial action (i.e. the instance is completely unambiguous), but common sense and previous work (Berg et al. 2010) suggest that people's responses are influenced by factors that go beyond Start Hierarchy alone. The strong performance of Start Hierarchy for the no-constraint condition therefore suggests that all of the planners that we evaluated are relatively far from a comprehensive account of human performance.

Table 1: BIC scores for regression models that take initial planning time as the dependent variable and incorporate planner predictions or structural parameters (OC and SH). For readability, scores are shown as offsets relative to 110066 (full condition) and 82053 (no-constraint condition).

Category	Planner	full	no-constraint
Baseline	random	625	141
	OC	47	114
	SH	584	**0**
Cognitive Architecture	4caps	120	89
Classical Planner	BFS	38	101
	ASTAR	4	83
	GBFS	162	104
Online Planner	LH1	597	78
	LH2	598	80
	LH3	597	79
	LH4	342	70
	LH5	110	88
	LH6	52	95
	LH7	87	110
	A-LH	**0**	87

Individual differences

The analysis summarized by Table 1 used individual-level data but did not focus on individual differences. A similar regression approach, however, can be applied to the subset of the data provided by a single participant, which yields regression scores indicating the extent to which each planner or structural parameter predicts the responses of that participant. Distributions of these regression scores across individuals are shown in Figure 4. Consistent with Table 1, the individual level analysis suggests that the adaptive lookahead and ASTAR planners provide the best account of the full condition, and that Start Hierarchy provides the best account of the no-constraint condition. In the full condition, ASTAR and the adaptive lookahead planner account for the responses of some individuals relatively well (regression scores around 0.6), but in the no-constraint condition no regression score for any individual exceeds 0.5. The results therefore suggest that none of the models provides a good account of individual performance in the no-constraint condition.

Discussion and Conclusion

We applied a set of planners to the TOL task and evaluated their ability to predict actions and response times collected in a new behavioral experiment. Prior work on the TOL task often asks participants to form a complete plan before acting (Berg et al. 2010), and in this condition we found that an adaptive lookahead planner provides the best account of

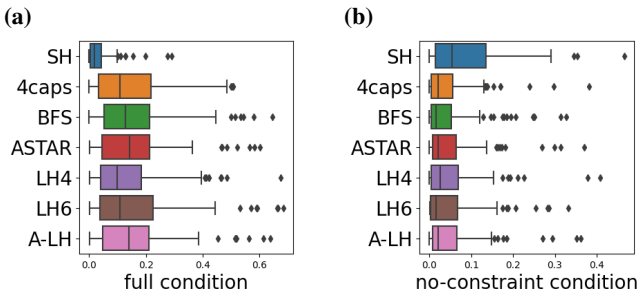

Figure 4: (a) Individual-level analysis of initial planning times. Panels (a) and (b) show regression scores for the full and no-constraint conditions, and each datapoint represents an individual participant.

both actions and response times. This planner allows the size of the search tree to depend on the difficulty of the current instance, and the good performance of this planner suggests that people flexibly navigate a speed-accuracy tradeoff when approaching sequential decision-making tasks.

The differences we observed between the full and no-constraint conditions confirm that people's problem solving strategies depend on task requirements, but our planner evaluation did not provide a consistent picture about performance in the no-constraint condition. The adaptive lookahead planner provided the best account of action selection in this condition, but our analysis of response times found that none of the planners was more predictive than a simple structural parameter (Start Hierarchy). It may not be surprising that removing task constraints increases variability and makes experimental data more difficult to model, but our results suggest that more work is needed to develop a satisfying account of human performance in this condition.

We presented a simple initial analysis of individual participants that revealed substantial variability, and future work can model individual differences more directly by introducing individual-level parameters to the models. For example, the success of the current adaptive lookahead model motivates future versions of the model that allow the decision threshold to vary across individuals.

When using the planning-based approach to model human behavior, it is important to consider not only the planner but also the alignment between human mental representations and the problem representation used by the planner. At least two representations of the TOL task can be considered. Our analysis treated pick-and-put as a single action, but an alternative modeling approach treats pick and put as two separate actions. We evaluated both representations and found that both of them led to similar conclusions, but more targeted experiments may be able to reveal which of the two is closer to the representation used by people. Similarly, our models used the goal-counting heuristic, but we also evaluated other general heuristics derived from widely used relaxations such as the delete-relaxation (Bonet and Geffner 2001), and found that these alternative heuristics produced similar results in our setting. Future studies, however, can consider experiments that aim to distinguish which of these heuristics provide the best account of human behavior.

Perhaps the most general message from our work is that the planning-based approach to human problem solving deserves to be revisited. Our results suggest that even relatively simple tasks such as the Tower of London continue to present challenges for cognitive models, and combining ideas from both cognitive psychology and AI planning continues to be a promising way to address these challenges.

References

Anderson, J. R. 1989. A Rational Analysis of Human. *Varieties of memory and consciousness: Essays in honour of Endel Tulving*, 195.

Anderson, M. L.; and Oates, T. 2007. A review of recent research in metareasoning and metalearning. *AI Magazine*, 28(1): 12–12.

Atwood, M. E.; and Polson, P. G. 1976. A process model for water jug problems. *Cognitive Psychology*, 8(2): 191–216.

Berg, W. K.; Byrd, D. L.; McNamara, J. P.; and Case, K. 2010. Deconstructing the tower: Parameters and predictors of problem difficulty on the Tower of London task. *Brain and Cognition*, 72(3): 472–482.

Bonet, B.; and Geffner, H. 2001. Planning as heuristic search. *Artificial Intelligence*, 129(1-2): 5–33.

Bulitko, V.; Levner, I.; and Greiner, R. 2002. Real-time lookahead control policies. In *Joint Workshop on Real-Time Decision Support and Diagnosis Systems, aaai*.

Callaway, F.; Lieder, F.; Das, P.; Gul, S.; Krueger, P. M.; and Griffiths, T. 2018. A resource-rational analysis of human planning. In *CogSci*.

Callaway, F.; van Opheusden, B.; Gul, S.; Das, P.; Krueger, P.; Lieder, F.; and Griffiths, T. 2021. Human planning as optimal information seeking. *Manuscript in preparation*.

Chenyuan Zhang, C. K.; and Lipovetzky, N. 2023. Goal recognition with timing information. In *Proceedings of the 33rd International Conference on Automated Planning and Scheduling (ICAPS)*. AAAI Press.

De Leeuw, J. R. 2015. jsPsych: A JavaScript library for creating behavioral experiments in a Web browser. *Behavior research methods*, 47(1): 1–12.

Donnarumma, F.; Maisto, D.; and Pezzulo, G. 2016. Problem solving as probabilistic inference with subgoaling: explaining human successes and pitfalls in the tower of Hanoi. *PLoS computational biology*, 12(4): e1004864.

Francés, G.; Ramirez, M.; and Collaborators. 2018. Tarski: An AI Planning Modeling Framework. https://github.com/aig-upf/tarski.

Geffner, H. 2013. Computational models of planning. *Wiley Interdisciplinary Reviews: Cognitive Science*, 4(4): 341–356.

Gugerty, L. 2006. Newell and Simon's logic theorist: Historical background and impact on cognitive modeling. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 50, 880–884. SAGE Publications Sage CA: Los Angeles, CA.

Hart, P. E.; Nilsson, N. J.; and 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.

Haslum, P.; Lipovetzky, N.; Magazzeni, D.; and Muise, C. 2019. An introduction to the planning domain definition language. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 13(2): 1–187.

Heusner, M.; Keller, T.; and Helmert, M. 2017. Understanding the search behaviour of greedy best-first search.

Kaller, C. P.; Rahm, B.; Köstering, L.; and Unterrainer, J. M. 2011. Reviewing the impact of problem structure on planning: A software tool for analyzing tower tasks. *Behavioural brain research*, 216(1): 1–8.

Kaller, C. P.; Unterrainer, J. M.; Rahm, B.; and Halsband, U. 2004. The impact of problem structure on planning: Insights from the Tower of London task. *Cognitive Brain Research*, 20(3): 462–472.

Keramati, M.; Smittenaar, P.; Dolan, R. J.; and 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.

Kocsis, L.; and Szepesvári, C. 2006. Bandit based monte-carlo planning. In *European conference on machine learning*, 282–293. Springer.

Kotovsky, K.; Hayes, J. R.; and Simon, H. A. 1985. Why are some problems hard? Evidence from Tower of Hanoi. *Cognitive psychology*, 17(2): 248–294.

Krusche, M. J.; Schulz, E.; Guez, A.; and Speekenbrink, M. 2018. Adaptive planning in human search. *BioRxiv*, 268938.

Kryven, M.; Kleiman-Weiner, M.; Tenenbaum, J.; and Yu, S. 2022. Planning ahead in spatial search. *PsyArXiv*.

Kuperwajs, I.; Van Opheusden, B.; and Ma, W. J. 2019. Prospective planning and retrospective learning in a large-scale combinatorial game. In *2019 Conference on Cognitive Computational Neuroscience*, 13–16.

Laird, J. E.; Newell, A.; and Rosenbloom, P. S. 1987. Soar: An architecture for general intelligence. *Artificial intelligence*, 33(1): 1–64.

Lebiere, C.; and Anderson, J. R. 1993. A connectionist implementation of the ACT-R production system. In *Proceedings of the fifteenth annual conference of the Cognitive Science Society*, 635–640.

Mattar, M. G.; and Lengyel, M. 2022. Planning in the brain. *Neuron*.

Meder, B.; Nelson, J. D.; Jones, M.; and Ruggeri, A. 2019. Stepwise versus globally optimal search in children and adults. *Cognition*, 191: 103965.

Mormann, M.; Malmaud, J.; Huth, A.; Koch, C.; and Rangel, A. 2010. The drift diffusion model can account for the accuracy and reaction time of value-based choices under high and low time pressure. *Judgment and Decision Making*, 5(6): 437–449.

Newell, A.; and Simon, H. 1956. The logic theory machine–A complex information processing system. *IRE Transactions on information theory*, 2(3): 61–79.

Newell, A.; Simon, H. A.; et al. 1972. *Human problem solving*, volume 104. Prentice-hall Englewood Cliffs, NJ.

Newman, S. D.; Carpenter, P. A.; Varma, S.; and Just, M. A. 2003. Frontal and parietal participation in problem solving in the Tower of London: fMRI and computational modeling of planning and high-level perception. *Neuropsychologia*, 41(12): 1668–1682.

Ohlsson, S. 2012. The problems with problem solving: Reflections on the rise, current status, and possible future of a cognitive research paradigm. *The Journal of Problem Solving*, 5(1): 101–128.

Ramirez, M.; Lipovetzky, N.; and Muise, C. 2015. Lightweight Automated Planning ToolKiT. http://lapkt.org/. Accessed: 2020.

Ruiz, D.; and Newell, A. 1989. Tower-noticing triggers strategy-change in the Tower of Hanoi: A Soar model. Technical report, Carnegie Mellon University Dept of Psychology.

Simon, H. A. 1963. Experiments with a heuristic compiler. *Journal of the ACM (JACM)*, 10(4): 493–506.

Simon, H. A. 1990. Bounded rationality. In *Utility and probability*, 15–18. Springer.

Solway, A.; and Botvinick, M. M. 2015. Evidence integration in model-based tree search. *Proceedings of the National Academy of Sciences*, 112(37): 11708–11713.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement learning: An introduction*. MIT press.

Tajima, S.; Drugowitsch, J.; Patel, N.; and Pouget, A. 2019. Optimal policy for multi-alternative decisions. *Nature neuroscience*, 22(9): 1503–1511.

Van Opheusden, B.; Galbiati, G.; Bnaya, Z.; Li, Y.; and Ma, W. J. 2017. A computational model for decision tree search. In *CogSci*.

Varma, S.; and Just, M. A. 2006. 4CAPS: An Adaptive Architecture for Human Information Processing. In *AAAI Spring Symposium: Between a Rock and a Hard Place: Cognitive Science Principles Meet AI-Hard Problems*, 91–96.