Article

People construct simplified mental representations to plan

https://doi.org/10.1038/s41586-022-04743-9

Received: 25 January 2021

Accepted: 7 April 2022

Published online: 19 May 2022

Check for updates

Mark K. Ho[1,2]✉, David Abel[3,5], Carlos G. Correa[4], Michael L. Littman[3], Jonathan D. Cohen[1,4] & Thomas L. Griffiths[1,2]

One of the most striking features of human cognition is the ability to plan. Two aspects of human planning stand out—its efficiency and flexibility. Efficiency is especially impressive because plans must often be made in complex environments, and yet people successfully plan solutions to many everyday problems despite having limited cognitive resources[1–3]. Standard accounts in psychology, economics and artificial intelligence have suggested that human planning succeeds because people have a complete representation of a task and then use heuristics to plan future actions in that representation[4–11]. However, this approach generally assumes that task representations are fixed. Here we propose that task representations can be controlled and that such control provides opportunities to quickly simplify problems and more easily reason about them. We propose a computational account of this simplification process and, in a series of preregistered behavioural experiments, show that it is subject to online cognitive control[12–14] and that people optimally balance the complexity of a task representation and its utility for planning and acting. These results demonstrate how strategically perceiving and conceiving problems facilitates the effective use of limited cognitive resources.

In the short story *On Exactitude in Science*, Jorge Luis Borges describes cartographers who seek to create the perfect map—one that includes every possible detail of the country it represents. However, this innocent premise leads to an absurd conclusion: the fully detailed map of the country must be the size of the country itself, which makes it impractical for anyone to use. Borges' allegory illustrates an important computational principle. Namely, useful representations do not simply mirror every aspect of the world, but rather pick out a manageable subset of details that are relevant to some purpose (Fig. 1a). Here we examine the consequences of this principle for how humans flexibly construct simplified task representations to plan.

Classic theories of problem solving distinguish between representing a task and computing a plan[4,15,16]. For example, Newell and Simon[17] introduced heuristic search, in which a decision-maker has a full representation of a task (such as a chess board, chess pieces and the rules of chess), and then computes a plan by simulating and evaluating possible action sequences (that is, sequences of chess moves) to find one that is likely to achieve a goal (for example, checkmate). In artificial intelligence, the main approach to making heuristic search tractable involves limiting the computation of action sequences (such as thinking only a few moves into the future, or examining only moves that seem promising)[5]. Similarly, psychological research on planning largely focuses on how limiting, prioritizing, pruning or chunking action sequences can reduce computation[6–11,18–20].

However, people are not necessarily restricted to a single, full or fixed representation for a task. This matters as simpler representations can make better use of limited cognitive resources when they are tailored to specific parts or versions of a task. For example, in chess, considering the interaction of a few pieces, or focusing on part of the board, is easier than reasoning about every piece and part of the board. Furthermore, it affords the opportunity to adapt the representation, tailoring it to the specific needs of the circumstance—a process that we refer to as controlling a task construal. Although studies show that people can flexibly form representations to guide action (such as forming the ad hoc category of 'things to buy for a party' when organizing a social gathering[21]), a long-standing challenge for cognitive science and artificial intelligence is explaining, predicting and deriving such representations from general computational principles[22,23].

Our approach to studying how people control task construals starts with the premise that effective decision-making depends on making rational use of limited cognitive resources[1–3]. Specifically, we derive how an ideal, cognitively limited decision-maker should form value-guided construals that balance the complexity of a representation and its use for planning and acting. We then show that preregistered predictions of this account explain how people attend to task elements in several planning experiments (see Data availability). Our analysis and findings suggest that controlled, moment-to-moment task construals have a key role in efficient and flexible planning.

Task construals from first principles

We build on models of sequential decision-making expressed as Markov decision processes[24]. Formally, a task $\mathcal{T}$ consists of a state space $\mathcal{S}$; an initial state $s_0 \in \mathcal{S}$; an action space $\mathcal{A}$; a transition function

[1]Department of Psychology, Princeton University, Princeton, NJ, USA. [2]Department of Computer Science, Princeton University, Princeton, NJ, USA. [3]Department of Computer Science, Brown University, Providence, RI, USA. [4]Princeton Neuroscience Institute, Princeton University, Princeton, NJ, USA. [5]Present address: DeepMind, London, UK. ✉e-mail: mho@princeton.edu

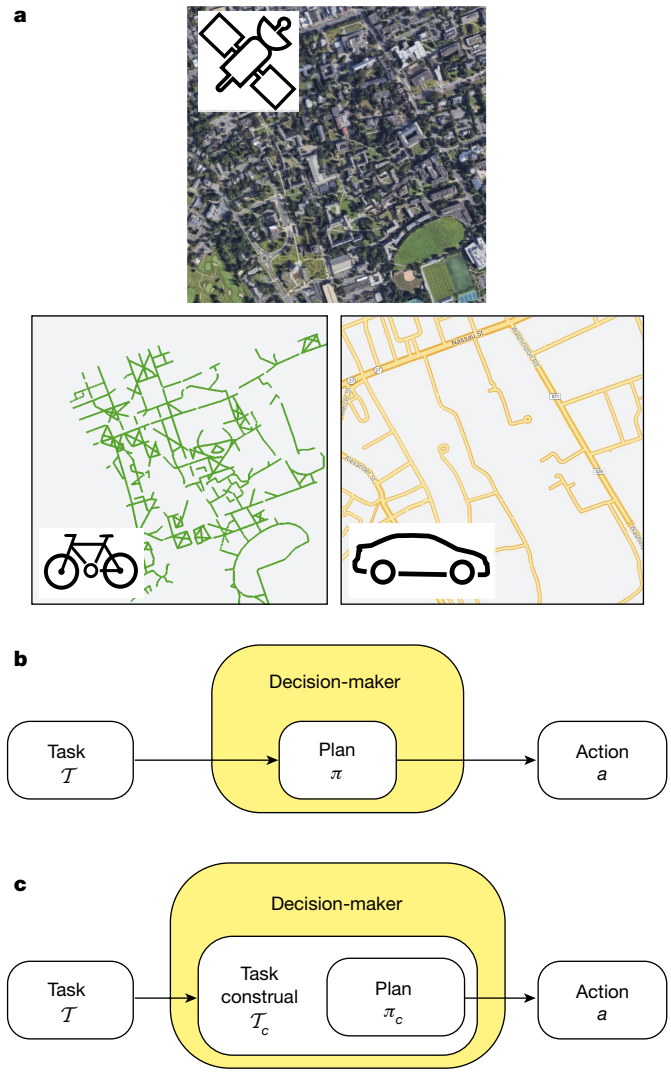


Fig. 1 | Construal and planning. a, A satellite photo of Princeton, New Jersey, USA (top) and maps of Princeton for bicycling versus automotive use cases (bottom). Like maps and unlike photographs, a decision-maker's construal picks out a manageable subset of details from the world relevant to their current goals. Imagery © 2022 Google, Map data 2022. **b**, Standard models assume that a decision-maker computes a plan, π, with respect to a fixed task representation, $\mathcal{T}$, and then uses it to guide their actions, a. **c**, According to our model of value-guided construal, the decision-maker forms a simplified task construal, $\mathcal{T}_c$, that is used to compute a plan, π_c. This process can be understood as two nested optimizations: an 'outer loop' of construal and an 'inner loop' of planning.

$P: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$; and a utility function $U: \mathcal{S} \to \mathbb{R}$. In standard formulations of planning, the value of a plan $\pi: \mathcal{S} \times \mathcal{A} \to [0, 1]$ from a state s is determined by the expected, cumulative utility of using that plan[25]: $V_\pi(s) = U(s) + \sum_a \pi(a|s) \sum_{s'} P(s'|s, a) V_\pi(s')$. Standard planning algorithms[5] (such as heuristic search methods) attempt to efficiently compute plans that optimize value by directly planning over a fixed task representation, $\mathcal{T}$, that is not subject to the decision-maker's control (Fig. 1b). Our aim is to relax this constraint and consider the process of adaptively selecting simplified task representations for planning, which we call the construal process (Fig. 1c).

Intuitively, a construal 'picks out' details in a task to consider. Here we examine construals that pick out cause–effect relationships in a task. This focus is motivated by the intuition that a key source of task complexity is the interaction of different causes and their effects with one another. For example, consider interacting with various objects in someone's living room. Walking towards the couch and hitting it is a cause–effect relationship, while pulling on the coffee table and moving it might be another such relationship. These individual effects can interact and may or may not be integrated into a single representation of moving around the living room. For example, imagine pulling on the coffee table and causing it to move, but in doing so, backing into the couch and hitting it. Whether or not a decision-maker anticipates and represents the interaction of multiple effects depends on what causes and effects are incorporated into their construal; this, in turn, can affect the outcome of behaviour.

Related work has studied how attention guides learning about how different state features predict rewards[26]. By contrast, to model construals, we require a way to express how attention flexibly combines different causes and their effects into an integrated model to use for planning. For this, we use a product of experts[27], a technique from the machine learning literature for combining distributions that is similar to factored approximations used in models of perception[28]. Specifically, we assume that the agent has N primitive cause–effect relationships that each assign probabilities to state, action and next-state transitions, $\phi_i: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1], i = 1, \ldots, N$. Each $\phi_i(s'|s, a)$ is a potential function representing, for example, the local effect of colliding with the couch or pulling on the coffee table. Then a construal is a subset of these primitive cause–effect relationships, $c \subseteq \{\phi_1, \ldots, \phi_N\}$, that produces a task construal, $\mathcal{T}_c$, with the following construed transition function:

$$P_c(s'|s, a) \propto \prod_{\phi_i \in c} \phi_i(s'|s, a). \quad (1)$$

Here, we assume that task construals ($\mathcal{T}_c$) and the original task ($\mathcal{T}$) share the same state space, action space and utility function. But, crucially, the construed transition function can be simpler than that of the actual task.

Ideally, a decision-maker would select a task construal that includes only those elements (cause–effect relationships) that lead to successful planning, excluding any others so as to make the planning problem as simple as possible. To make this intuition precise, it is essential to first distinguish between computing a plan with a construal and using the plan induced by a construal. In our example, suppose the decision-maker forms a construal of their living room that includes the effect of pulling on the coffee table but ignores the effect of colliding with the couch. They might then compute a plan in which they pull on the coffee table without any complications, but when they use that plan in the actual living room, they inadvertently stumble over their couch. This particular construal is less than optimal.

Thus, we formalize the distinction between the computed plan associated with a construal and its resulting behavioural utility: if the decision-maker has a task construal $\mathcal{T}_c$, denote the plan that optimizes it as π_c. Then, the utility of the computed plan when starting at state s_0 is given by its performance when interacting with the actual transition dynamics, P:

$$U(\pi_c) = U(s_0) + \sum_a \pi_c(a|s_0) \sum_{s'} P(s'|s_0, a) V_{\pi_c}(s'). \quad (2)$$

Put simply, the behavioural utility of a construal is determined by the consequences of using it to compute a plan and then act according to that plan in the actual task.

Having established the relationship between a construal and its utility, we can define the value of representation (VOR) associated with a construal. Our formulation resembles previous models of resource rationality[2] and the expected value of control[13] by discounting utilities with a cognitive cost, C. This cost could be further enriched by specifying algorithm-specific costs[29] or hard constraints[30]. However, our aim is to understand value-guided construal with respect to the complexity of the construal itself and with minimal algorithmic assumptions. To

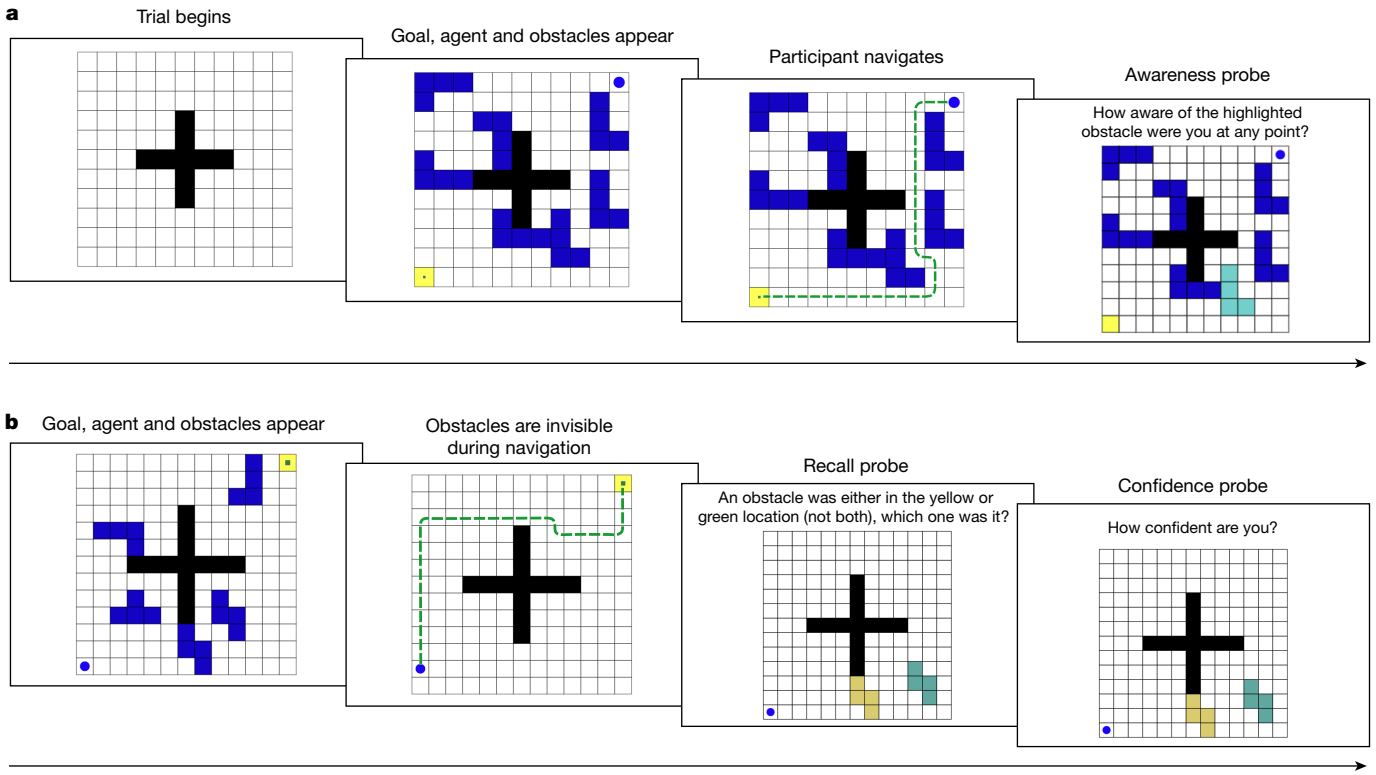


Fig. 2 | Value-guided construal predicts how people will form representations that are simple but useful for planning and acting. These predictions were tested in a new paradigm in which participants controlled a blue circle and navigated mazes composed of centre black walls in the shape of a cross, blue tetromimo-shaped obstacles, and a yellow goal state with a shrinking green square. We assume that attention to obstacles as a result of construal is reflected in memory of obstacles and used two types of probes to assess memory. **a**, In our initial experiment, the participants were shown the maze and navigated to the goal. The dashed line indicates an example path. After navigating, the participants were given awareness probes in which they were asked to report their awareness of each obstacle on an eight-point scale (for analyses, responses were scaled to range from 0 to 1). **b**, In a subsequent experiment, obstacles were visible only before moving to encourage planning up front, and participants were given recall probes in which they were shown a pair of obstacles in green and yellow, only one of which had been present in the maze that they had just completed. The participants were then asked which one had been in the maze as well as their confidence.

this end, we use a cost that penalizes the number of effects considered: $C(c) = |c|$, where $|c|$ is the cardinality of c. Intuitively, this cost reflects the description length of a program that expresses the construed transition function in terms of primitive effects[31]. It also generalizes recent economic models of sparsity-based behavioural inattention[32]. The VOR for construal c is then its behavioural utility minus its cognitive cost:

$$\mathrm{VOR}(c) = U(\pi_c) - C(c). \quad (3)$$

In brief, we introduce the notion of a task construal (equation (1)) that relaxes the assumption of planning over a fixed task representation. We then define an optimality criterion for a construal based on its complexity and its utility for planning and acting (equations (2) and (3)). This optimality criterion provides a normative standard that we can use to ask whether people form optimal value-guided construals[33,34]. Note that the question of precisely how people identify or learn optimal construals is beyond the scope of our current aims. Rather, here our goal is to simply determine whether their planning is consistent with optimal construal. If so, then understanding how people achieve (or approximate) this ability will be a key direction for future research (see the Supplementary Discussion for details about construal optimization algorithms).

A paradigm for examining construals

To examine whether people form construals that optimally balance complexity and utility, we designed a paradigm analogous to the example in Fig. 1a, in which participants were shown a two-dimensional map of a maze and had to move a blue dot to reach a goal location. On each trial, the participants were shown a new maze composed of a starting location, a goal location, centre black walls in the shape of a plus symbol (+) and an arrangement of blue obstacles. The goal, starting state and the blue obstacles (but not the centre black walls) changed on every trial, which required participants to examine the layout of the maze and plan an efficient route to the goal (Fig. 2a). In our framework, each obstacle corresponds to a cause–effect relationship, ϕ_i—that is, attempting to move into the space occupied by the obstacle and then being blocked. This is analogous to the effect of being blocked by a piece of furniture in our earlier example.

Two key features make our maze-navigation paradigm useful for isolating and studying the construal process. First, the mazes are fully observable: complete information about the task is immediately accessible from the visual stimulus. Second, each instance of a maze emerges from a particular composition of individual elements (for example, the obstacles). This means that, although all of the components of a particular maze are immediately accessible, participants need to choose which ones to integrate into an effective representation for planning (that is, select a construal). Fully observable but compositionally structured problems occur routinely in everyday life—for example, using a map to navigate through exhibits in a museum—as well as in popular games, such as in chess, figuring out how to move one's knight across a board occupied by an opponent's pieces. By providing people with immediate access to all of the components of a task while planning, we can examine which ones they attend to versus ignore and whether these patterns of awareness reflect a process of value-guided construal

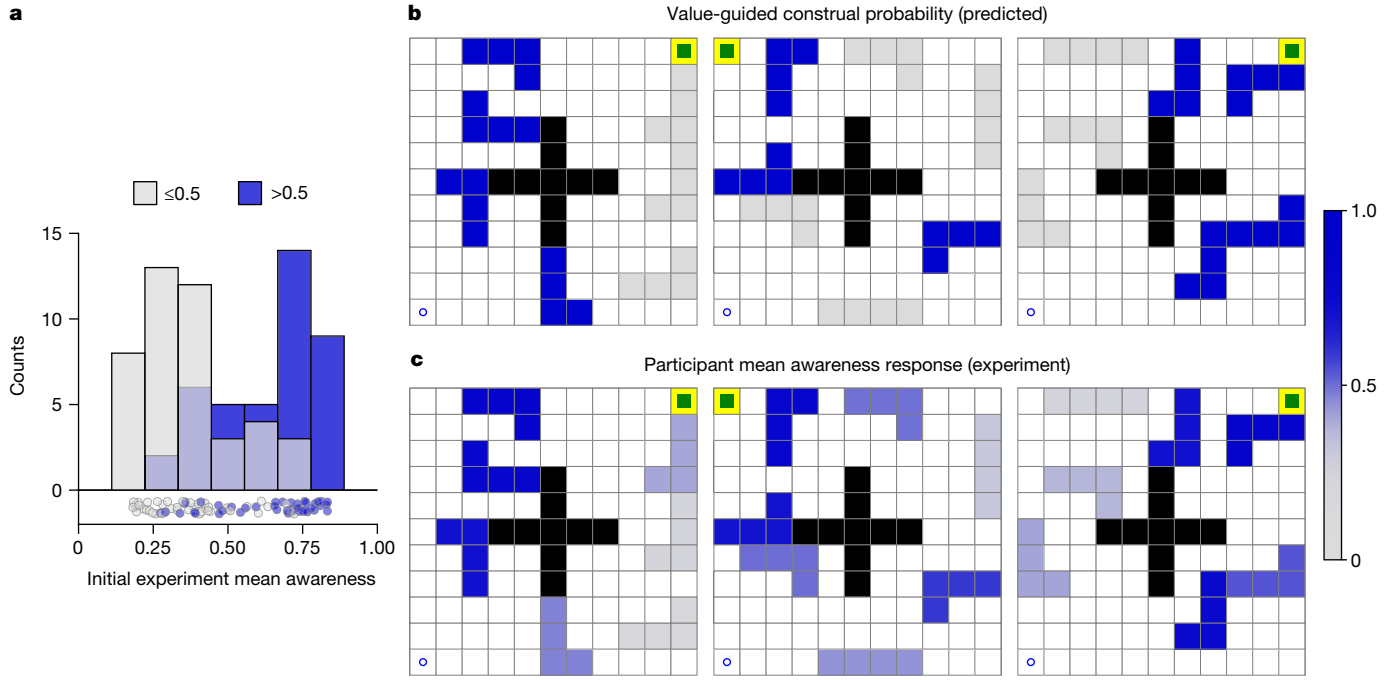


Fig. 3 | Initial experiment results. In our initial planning experiment (out of four), each person (n = 161 independent participants) navigated 12 2D mazes, each of which had 7 blue tetronimo-shaped obstacles. To assess whether attention to obstacles reflects a process of value-guided construal, the participants were given an awareness probe (Fig. 2a) for each obstacle in each maze. **a**, For our first analysis, we split the set of 84 obstacles across mazes on the basis of whether value-guided construal assigned a probability of less than or equal to 0.5 or greater than 0.5. The participants' mean awareness responses corresponding to the two sets of obstacles is shown (≤0.5 in grey, >0.5 in blue; individual by-obstacle mean awareness underlying the histograms are represented underneath). We then similarly split the obstacles on the basis of whether mean awareness responses were less than or equal to 0.5 or greater than 0.5 and, using a χ^2 test for independence, found that this split was predicted by value-guided construal $X_1^2 = 23.03$, $P = 1.6 \times 10^{-6}$, effect size $w = 0.52$, $n = 84$. **b**, Value-guided construal predictions for 3 out of the 12 mazes used in the experiment. The blue circles indicate the starting location; the green and yellow squares indicate the goal; the obstacle colours represent model probabilities according to the colour scale. **c**, Participants' mean awareness judgements for the same three mazes. Obstacle colours represent mean judgements according to the colour scale. Responses in this initial experiment generally reflect value-guided construal of mazes. The participants were recruited through the Prolific online experiment platform.

(see the 'Value-guided construal' section in the Methods and the Code availability). Furthermore, this general paradigm can be used in concert with several different experimental measures to assess attention (Extended Data Figs. 1–3, Data availability and Supplementary Methods).

Traces of construals in people's memory

We assume that the obstacles included in a construal will be associated with greater awareness and therefore memory; accordingly, we began by probing memory for obstacles after participants completed each maze to test whether they formed value-guided construals of the mazes. In our initial experiment, the participants received awareness probes in which, after navigation, they were shown a picture of the maze that they had just completed with one of the obstacles highlighted. They were then asked, "How aware of the highlighted obstacle were you at any point?" and responded on an eight-point scale that was later scaled to range from 0 to 1 for analyses (Fig. 2a). If the participants formed representations of the mazes that balance utility and complexity, their responses should be positively predicted by value-guided construal. This is precisely what we found: value-guided construal predicted awareness judgements (likelihood ratio test comparing hierarchical linear models with and without z-score normalized value-guided construal probabilities: $X_1^2 = 2{,}297.21$, $P < 1.0 \times 10^{-16}$; $\beta = 0.133$, s.e. = 0.003; see the 'Experiment analyses' section of the Methods; Fig. 3). Furthermore, we also observed the same results when the participants could not see the obstacles while moving and so needed to plan their route entirely up front ($X_1^2 = 726.95$, $P < 1.0 \times 10^{-16}$; $\beta = 0.115$, s.e. = 0.004). This was also the case when we probed awareness judgements immediately after planning but before execution ($X_1^2 = 679.20$, $P < 1.0 \times 10^{-16}$; $\beta = 0.106$, s.e. = 0.004; see the 'Up-front planning experiment' section of the Methods; Supplementary Analyses (memory experiment)).

Although the awareness probes provide useful insights into people's task construals, it is a step removed from their memory (which is already a step removed from the construal process itself) as it requires participants to reflect on their earlier awareness during planning. To address this limitation, we developed a second set of critical mazes with two properties. First, the mazes were designed to test the distinctive predictions of value-guided construal (Fig. 4a). Second, these new mazes enabled us to use a more stringent measure of memory for task elements. Specifically, we used obstacle recall probes, in which, after navigation, the participants were shown a grid with the black centre walls, a green obstacle, a yellow obstacle and no other obstacles. Either the green or yellow obstacle had actually been present in the maze, whereas the other obstacle did not overlap with any of those that had been present. The participants were then asked, "An obstacle was either in the yellow or green location (not both), which one was it?" and could select either option, followed by a confidence judgement on an eight-point scale that was scaled to range from 0 to 1 for analyses (Fig. 2b and Extended Data Fig. 4a). The recall probes therefore provided two measures, accuracy and confidence, and using hierarchical generalized linear models (HGLMs) we found that value-guided construal predicted both types of responses (likelihood ratio tests comparing models on accuracy: $X_1^2 = 249.34$, $P < 1.0 \times 10^{-16}$; $\beta = 0.648$, s.e. = 0.042; and confidence: $X_1^2 = 432.76$, $P < 1.0 \times 10^{-16}$; $\beta = 0.104$,

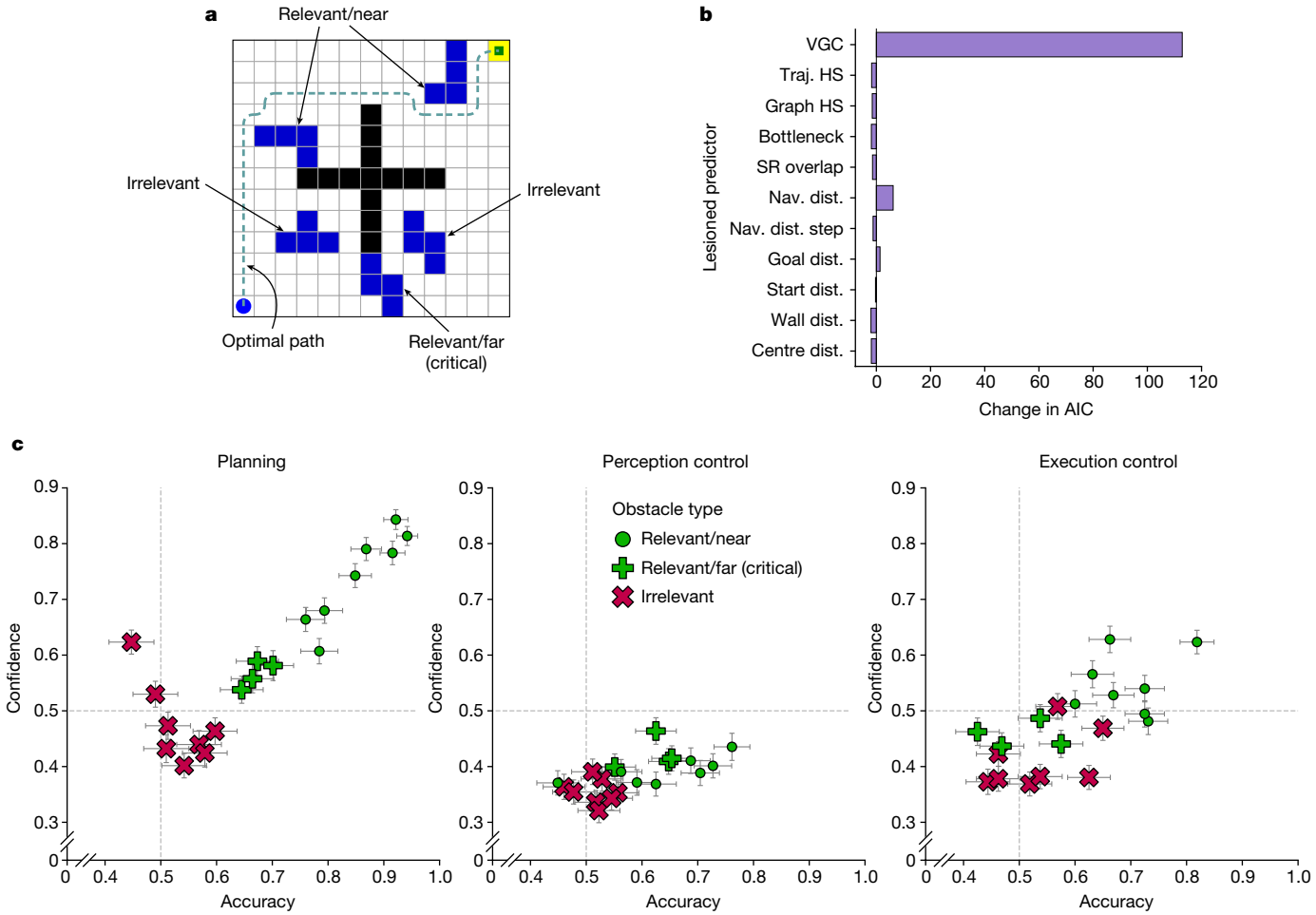


Fig. 4 | Critical mazes recall experiment, model comparisons and control studies. a, The critical mazes recall experiment ($n = 78$ independent participants; one version of one of the four planning experiments) used critical mazes that included critical obstacles, defined as obstacles that are highly relevant to planning but far from an optimal path (dashed line). Value-guided construal predicts critical obstacles will be included in a construal whereas irrelevant obstacles will not, independent of the distance to the optimal path. **b**, We fit a global model to recall responses that included the fixed parameter value-guided construal modification model (VGC) along with ten alternative predictors: trajectory-based heuristic search score (Traj. HS), graph-based heuristic search score (Graph HS), bottleneck state distance (Bottleneck), successor representation overlap (SR overlap), minimum navigation distance (Nav. dist.), timestep of minimum navigation distance (Nav. dist. step), distance to goal (Goal dist.), distance to start (Start dist.), distance to centre walls (Wall dist.) and distance to the centre of the maze (Centre dist.) (see the 'Experiment analyses' section of the Methods). Each predictor was then removed from this global model, and we calculated the resulting change in fit (in AIC). Removing value-guided construal led to the largest degradation of fit (greatest increase in AIC), underscoring its unique explanatory value. **c**, In a pair of non-planning control experiments, new participants either viewed patterns that looked exactly like the mazes (perception control; $n = 88$ independent participants) or followed 'breadcrumbs' through the maze along a path taken by a participant from the original experiment (execution control; $n = 80$ independent participants). The participants then answered the exact same recall questions. Value-guided construal remains a significant factor when explaining recall in the original critical mazes experiment (planning) while including mean recall from the perception and execution controls as covariates (likelihood ratio test for accuracy: $X_1^2 = 106.36$, $P = 6.2 \times 10^{-25}$; confidence: $X_1^2 = 18.56$, $P = 1.6 \times 10^{-5}$; P values are unmodified). This confirms that responses consistent with value-guided construal are not a simple function of perception and execution. The participants were recruited through the Prolific online experiment platform. For **c**, data are mean ± s.e.m. values for each obstacle, with relevant/near, relevant/far (critical) and irrelevant obstacle types distinguished.

s.e. = 0.005; see the 'Experiment analyses' section of the Methods). Moreover, when we gave a separate group of participants the awareness probes on these mazes, value-guided construal was again predictive (awareness: $X_1^2 = 837.47$, $P < 1.0 \times 10^{-16}$; $\beta = 0.175$, s.e. = 0.006). Thus, using three different measures of memory (recall accuracy, recall confidence and awareness judgements), we found further evidence that, when planning, people form task representations that optimally balance complexity and utility.

Controlling for perception and execution

The memory studies provide preliminary confirmation of our hypothesis, but they have several limitations. One is that, although the participants were engaged in planning, they were also necessarily engaged in other forms of cognitive processing, and these unrelated processes may have influenced memory of the obstacles. In particular, participants' perception of a maze or their execution of a particular plan through a maze may have influenced their responses to the memory probes. This potentially confounds the interpretation of our results, as a key part of our hypothesis is that task construals arise from planning, rather than simply perceiving or executing.

Thus, to test that responses to the memory probes cannot be fully explained by perception and/or execution, we administered two sets of yoked controls that did not require planning (see the 'Control experiments' section of the Methods). In the perception controls, new participants were shown patterns that looked exactly like the

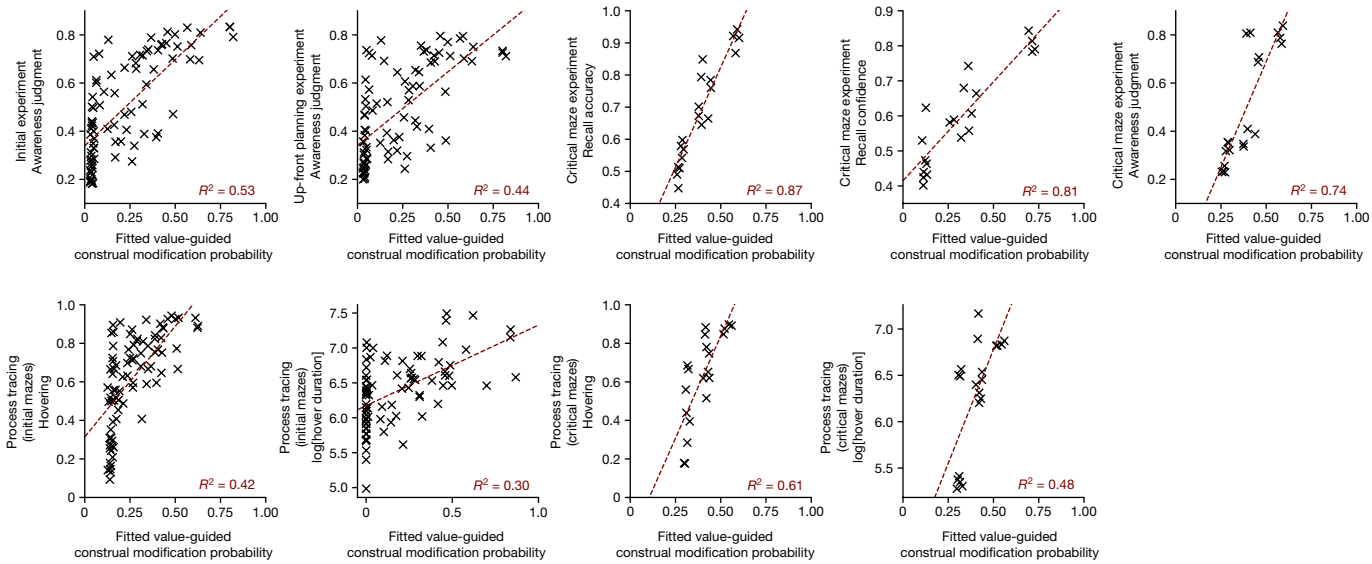


Fig. 5 | Fitted value-guided construal modification. Our initial model of value-guided construal focuses on whether an obstacle should or should not be included in a construal. We developed a generalization that additionally accounts for how much an obstacle influences a plan if a decision-maker is optimally modifying their construal during planning (see the 'Value-guided construal' section in the Methods). We used an ε-softmax noise model[39] for computed action plans and construal modification policies and, for each planning experiment and measure, searched for parameters that maximize the R^2 between model predictions and mean by-obstacle responses. Plots comparing the scores that the fitted construal modification model assigns to each obstacle with the participants' mean by-obstacle responses for the nine measures are shown (data are based on $n = 84{,}215$ observations taken from 825 independent participants).

mazes, but they performed an unrelated, non-planning task. Each pattern was presented to a new participant for the same amount of time that a participant in the original experiments had examined the corresponding maze before moving—that is, the amount of time that the original participant spent examining the maze to plan. The new participant then responded to the same probes, in the same order as the original participant. For the execution controls, we recruited another group of participants and gave them instructions similar to those in the planning experiments. However, in contrast to the original experiments, the task did not require planning. Rather, these mazes included 'breadcrumbs' that needed to be collected and that appeared every two steps. Breadcrumbs appeared along the exact path taken by one of the original participants, meaning that the new participant executed the same actions but without having planned. After completing each maze, the participant then received the same probes in the same order as the original participant.

We assessed whether responses in the planning experiments can be explained by a simple combination of perception and/or execution by testing whether value-guided construal remained a significant factor after accounting for control responses. Specifically, we used the mean by-obstacle responses from the perception and execution controls as predictors in HGLMs fit to the corresponding planning responses. We then tested whether adding value-guided construal as a predictor improved fits. For the awareness, accuracy and confidence responses in the recall experiment, we found that including value-guided construal significantly improved fits (likelihood ratio tests comparing models on accuracy: $X_1^2 = 106.36$, $P = 6.2 \times 10^{-25}$; confidence: $X_1^2 = 18.56$, $P = 1.6 \times 10^{-5}$; and awareness: $X_1^2 = 55.34$, $P = 1.0 \times 10^{-13}$) and that value-guided construal predictions were positively associated with responses (coefficients for accuracy: $\beta = 0.58$, s.e. = 0.058; confidence: $\beta = 0.039$, s.e. = 0.009; and awareness: $\beta = 0.054$, s.e. = 0.007). Thus, responses after planning are not reducible to a simple combination of perception and execution, and they can be further explained by the formation of value-guided construals (Fig. 4c and Supplementary Analyses (control experiment)).

Externalizing the planning process

Another limitation of the previous planning experiments is that they assess construal after planning is complete (that is, by probing memory). To obtain a measure of the planning process as it unfolds, we developed a process-tracing paradigm. In this version of the task, the participants never saw all of the obstacles at once. Instead, at the beginning of the trial, after being shown the start and goal locations, they could use their mouse to reveal individual obstacles by hovering over them (see the 'Process-tracing experiments' section of the Methods; Extended Data Fig. 4b). This led participants to externalize the planning process, and their behaviour on this task therefore provides insights into how planning computations unfolded internally. We tested whether value-guided construal accounted for behaviour by analysing two measures: whether an obstacle was hovered over and, if it was hovered over, the duration of hovering. Value-guided construal was a significant predictor for both these measures on both the initial mazes (likelihood ratio tests comparing HGLMs for hovering: $X_1^2 = 1{,}221.76$, $P < 1.0 \times 10^{-16}$; $\beta = 0.704$, s.e. = 0.021; and hover duration (log-transformed time in ms): $X_1^2 = 169.90$, $P < 1.0 \times 10^{-16}$; $\beta = 0.161$, s.e. = 0.012) and on the critical mazes (hovering: $X_1^2 = 1{,}361.92$, $P < 1.0 \times 10^{-16}$; $\beta = 0.802$, s.e. = 0.023; hover duration (log-transformed time in ms): $X_1^2 = 540.63$, $P < 1.0 \times 10^{-16}$; $\beta = 0.369$, s.e. = 0.016). Thus, these results complement our original memory-based measurements of people's task representations and strengthen the interpretation of them in terms of value-guided construal during planning.

Characterizing value-guided construal modification

Thus far, our account of value-guided construal has assumed that an obstacle is either always or never included in a construal. This simplification is useful as it enables us to derive clear qualitative predictions based on whether a plan is influenced by an obstacle, but it overlooks graded factors such as how much of a plan is influenced by an obstacle. For example, an obstacle may be relevant only for planning a few movements around a participant's initial location in a

maze and, as a result, could receive less total attention than one that is relevant for deciding how to act across a larger area of the maze. To characterize these more fine-grained attentional processes, we first generalized the original construal selection problem to a one in which the decision-maker revisits and potentially modifies their construal during planning. We then derived obstacle awareness predictions based on a theoretically optimal construal modification policy that balances complexity and utility (see the 'Value-guided construal' section in the Methods).

To assess value-guided construal modification, we reanalysed our data using three versions of the model with increasing ability to capture variability in responses. First, we used an idealized fixed-parameter model to derive a single set of obstacle attention predictions and confirmed that they also predict participant responses on the planning tasks (Supplementary Analyses (construal modification)). Second, for each planning measure and experiment, we calculated fitted-parameter models in which noise parameters for the computed plan and construal modification policy were fit (see the 'Value-guided construal' section in the Methods). Scatter plots comparing mean by-obstacle responses and model outputs for parameters with the highest R^2 are shown in Fig. 5. Finally, we fit a set of models that allowed for biases in computed plans (for example, a bias to stay along the edge of a maze or an explicit penalty for bumping into walls) and found that this additional expressiveness led to obstacle attention predictions with an improved correspondence to participant responses (Supplementary Analyses (construal modification)). Together, these analyses provide additional insights into the fine-grained dynamic structure of value-guided construal modification.

Accounting for alternative mechanisms

Although the analyses so far confirm the predictive power of value-guided construal, it is also important to consider alternative planning processes. For example, differential awareness could have been a passive side-effect of planning computations, rather than an active facilitator of planning computations as posited by value-guided construal. In particular, participants could have been planning by performing heuristic search over action sequences without actively construing the task, which would have led to differential awareness of obstacles as a byproduct of planning. Differential awareness could also have arisen from alternative representational processes, such as those based on the successor representation[35] or related subgoaling mechanisms[36]. Similarly, perceptual factors, such as the distance to the start, goal, walls, centre, optimal path or path taken, could have influenced responses.

On the basis of these considerations, we identified ten alternative predictors (see the 'Model Implementations' section in the Methods and the Code availability; Extended Data Figs. 5–7). All ten predictors plus the fixed value-guided construal modification predictions were included in global models that were fit to each of the nine planning experiment measures and, in all cases, value-guided construal was a significant predictor (Extended Data Table 1; see Supplementary Analyses (alternative mechanisms) for the same analyses with the single-construal model).

Furthermore, to assess the relative importance of each predictor, we calculated the change in fit (in terms of Akaike information criterion (AIC)) that resulted from removing each predictor from a global model (see the 'Experiment analyses' section of the Methods). Across all planning experiment measures, removing value-guided construal led to the first or second largest reduction in fit (Fig. 4b; Extended Data Table 1). These 'knock-out' analyses demonstrate the explanatory necessity of value-guided construal. To assess explanatory sufficiency, we fit a new set of single-predictor and two-predictor models using all predictors and then calculated their ΔAICs (see the 'Experiment analyses' section of the Methods; Extended Data Fig. 8). For all nine experimental measures, value-guided construal was one of the top two single-predictor models and was one of the two factors included in the best two-predictor model. Together, these analyses confirm the explanatory necessity and sufficiency of value-guided construal.

Discussion

We tested the idea that, when people plan, they do so by constructing a simplified mental representation of a problem that is sufficient to solve it—a process that we refer to as value-guided construal. We began by formally articulating how an ideal, cognitively limited decision-maker should construe a task so as to balance complexity and utility. We then showed that preregistered predictions of this model explain people's awareness, ability to recall problem elements (obstacles in a maze), confidence in recall ability and behaviour in a process-tracing paradigm, even after controlling for the baseline influence of perception and execution as well as ten alternative mechanisms. These findings support the hypothesis that people make use of a controlled process of value-guided construal, and that it can help to explain the efficiency of human planning. More generally, our account provides a framework for further investigating the cognitive mechanisms that are involved in construal. For example, future work can examine how construal strategies are acquired or how construal selection is shaped by computation costs, time or constraints. From a broader perspective, our analysis suggests a deep connection between the control of construals and the acquisition of structured representations like objects and their parts that can be cognitively manipulated[37,38], which can inform the development of intelligent machines. Future investigation into these and other mechanisms that interface with the control of representations will be crucial for developing a comprehensive theory of flexible and efficient intelligence.

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41586-022-04743-9.

1. Lewis, R. L., Howes, A. & Singh, S. Computational rationality: linking mechanism and behavior through bounded utility maximization. *Top. Cogn. Sci.* **6**, 279–311 (2014).
2. Griffiths, T. L., Lieder, F. & Goodman, N. D. Rational use of cognitive resources: levels of analysis between the computational and the algorithmic. *Top. Cogn. Sci.* **7**, 217–229 (2015).
3. Gershman, S. J., Horvitz, E. J. & Tenenbaum, J. B. Computational rationality: a converging paradigm for intelligence in brains, minds, and machines. *Science* **349**, 273–278 (2015).
4. Newell, A. & Simon, H. A. *Human Problem Solving* (Prentice Hall, 1972).
5. Russell, S. & Norvig, P. *Artificial Intelligence: A Modern Approach* 3rd edn (Prentice Hall, 2009).
6. Keramati, M., Smittenaar, P., Dolan, R. J. & Dayan, P. Adaptive integration of habits into depth-limited planning defines a habitual-goal–directed spectrum. *Proc. Natl Acad. Sci. USA* **113**, 12868–12873 (2016).
7. Huys, Q. J. M. et al. Bonsai trees in your head: how the Pavlovian system sculpts goal-directed choices by pruning decision trees. *PLoS Comput. Biol.* **8**, e1002410 (2012).
8. Huys, Q. J. M. et al. Interplay of approximate planning strategies. *Proc. Natl Acad. Sci. USA* **112**, 3098–3103 (2015).
9. Callaway, F. et al. Rational use of cognitive resources in human planning. *Nat. Hum. Behav.* https://doi.org/10.1038/s41562-022-01332-8 (2022).
10. Sezener, C. E., Dezfouli, A. & Keramati, M. Optimizing the depth and the direction of prospective planning using information values. *PLoS Comput. Biol.* **15**, e1006827 (2019).
11. Pezzulo, G., Donnarumma, F., Maisto, D. & Stoianov, I. Planning at decision time and in the background during spatial navigation. *Curr. Opin. Behav. Sci.* **29**, 69–76 (2019).
12. Miller, E. K. & Cohen, J. D. An integrative theory of prefrontal cortex function. *Ann. Rev. Neurosci.* **24**, 167–202 (2001).
13. Shenhav, A., Botvinick, M. M. & Cohen, J. D. The expected value of control: an integrative theory of anterior cingulate cortex function. *Neuron* **79**, 217–240 (2013).
14. Shenhav, A. et al. Toward a rational and mechanistic account of mental effort. *Ann. Rev. Neurosci.* **40**, 99–124 (2017).
15. Norman, D. A. & Shallice, T. in *Consciousness and Self-Regulation* (eds Davidson, R. J. et al.) 1–18 (Plenum Press, 1986).

16. Holland, J. H., Holyoak, K. J., Nisbett, R. E. & Thagard, P. R. *Induction: Processes of Inference, Learning, and Discovery* (MIT Press, 1989).
17. Newell, A. & Simon, H. A. Computer science as empirical inquiry: symbols and search. *Commun. ACM* **19**, 113–126 (1976).
18. Daw, N. D., Niv, Y. & Dayan, P. Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nat. Neurosci.* **8**, 1704–1711 (2005).
19. Gläscher, J., Daw, N., Dayan, P. & O'Doherty, J. P. States versus rewards: dissociable neural prediction error signals underlying model-based and model-free reinforcement learning. *Neuron* **66**, 585–595 (2010).
20. Ramkumar, P. et al. Chunking as the result of an efficiency computation trade-off. *Nat. Commun.* **7**, 12176 (2016).
21. Barsalou, L. W. Ad hoc categories. *Mem. Cogn.* **11**, 211–227 (1983).
22. Simon, H. A. The functional equivalence of problem solving skills. *Cogn. Psychol.* **7**, 268–288 (1975).
23. Brooks, R. A. Intelligence without representation. *Artif. Intell.* **47**, 139–159 (1991).
24. Puterman, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming* (John Wiley & Sons, 1994).
25. Bellman, R. *Dynamic Programming* (Princeton Univ. Press, 1957).
26. Leong, Y. C., Radulescu, A., Daniel, R., DeWoskin, V. & Niv, Y. Dynamic interaction between reinforcement learning and attention in multidimensional environments. *Neuron* **93**, 451–463 (2017).
27. Hinton, G. E. Training products of experts by minimizing contrastive divergence. *Neural Comput.* **14**, 1771–1800 (2002).
28. Whiteley, L. & Sahani, M. Attention in a Bayesian framework. *Front. Hum. Neurosci.* **6**, 100 (2012).
29. Lieder, F. & Griffiths, T. L. Resource-rational analysis: understanding human cognition as the optimal use of limited computational resources. *Behav. Brain Sci.* **43**, e1 (2020).
30. Yoo, A. H., Klyszejko, Z., Curtis, C. E. & Ma, W. J. Strategic allocation of working memory resource. *Sci. Rep.* **8**, 16162 (2018).
31. Grünwald, P. Model selection based on minimum description length. *J. Math. Psychol.* **44**, 133–152 (2000).
32. Gabaix, X. A sparsity-based model of bounded rationality. *Q. J. Econ.* **129**, 1661–1710 (2014).
33. Marr, D. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information* (W. H. Freeman, 1982).
34. Anderson, J. R. *The Adaptive Character of Thought* (Lawrence Erlbaum Associates, 1990).
35. Gershman, S. J. The successor representation: its computational logic and neural substrates. *J. Neurosci.* **38**, 7193–7200 (2018).
36. Stachenfeld, K. L., Botvinick, M. M. & Gershman, S. J. The hippocampus as a predictive map. *Nat. Neurosci.* **20**, 1643–1653 (2017).
37. Tversky, B. & Hemenway, K. Objects, parts, and categories. *J. Exp. Psychol.* **113**, 169–193 (1984).
38. Tenenbaum, J. B., Kemp, C., Griffiths, T. L. & Goodman, N. D. How to grow a mind: statistics, structure, and abstraction. *Science* **331**, 1279–1285 (2011).
39. Nassar, M. R. & Frank, M. J. Taming the beast: extracting generalizable knowledge from computational models of cognition. *Curr. Opin. Behav. Sci.* **11**, 49–54 (2016).

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

© The Author(s), under exclusive licence to Springer Nature Limited 2022

Methods

Model implementations

Value-guided construal. Our model assumes that the decision-maker has a set of cause–effect relationships that can be combined into a task construal that is then used for planning. To derive empirical predictions for the maze tasks, we assume a set of primitive cause–effect relationships, each of which is analogous to the example of interacting with furniture in a living room (see 'A paradigm for examining construals'). For each maze, we modelled the following: the default effect of movement (that is, pressing an arrow key causes the circle to move in that direction with probability $1-\varepsilon$ and stay in place with probability ε, $\varepsilon = 10^{-5}$), ϕ_{Move}; the effect of being blocked by the centre, plus-shaped (+) walls (that is, the wall causes the circle to not move when the arrow key is pressed), ϕ_{Walls}; and effects of being blocked by each of the N obstacles, ϕ_{Obstacle_i}, $i = 1,..,.N$. As every maze includes the same movements and walls, the model selected only which obstacle effects to include. The utility function for all mazes was given by a step cost of −1 until the goal state was reached.

Value-guided construal posits a bilevel optimization procedure involving an 'outer loop' of construal and an 'inner loop' of planning. Here we exhaustively calculate potential solutions to this nested optimization problem by enumerating and planning with all possible construals (that is, subsets of obstacle effects). We exactly solved the inner loop of planning for each construal using dynamic programming[40] and then evaluated the optimal stochastic computed plan under the actual task dynamics (that is, equation (2)). For planning and evaluation, transition probabilities were multiplied by a discount rate of 0.99 to ensure that values were finite. The general procedure for calculating the value of construals is outlined in the algorithm in Extended Data Table 2. To be clear, our current research strategy is to derive theoretically optimal predictions for the inner loop of planning and outer loop of construal in the spirit of resource-rational analysis[2]. Thus, this specific procedure should not be interpreted as a process model of human construal. In the Supplementary Discussion (algorithms for construal optimization), we discuss the feasibility of optimizing construals and how an important direction for future research will involve investigating tractable algorithms for finding good construals.

Given a VOR function that assigns a value to each construal, we model participants as selecting a construal according to a softmax decision rule:

$$P(c) \propto \exp\{\alpha^{-1}\text{VOR}(c)\}, \tag{4}$$

where $\alpha > 0$ is a temperature parameter (for our preregistered predictions $\alpha = 0.1$). We then calculated a marginalized probability for each obstacle being included in the construal, from the initial state, s_0, corresponding to the expected awareness of that obstacle:

$$P(\text{Obstacle}_i) = \sum_c \mathbb{1}\left[\phi_{\text{Obstacle}_i} \in c\right] P(c), \tag{5}$$

where, for a statement X, $\mathbb{1}[X]$ evaluates to 1 if X is true and 0 if X is false. We implemented this model in Python v.3.7.4 using the msdm library (see Code availability).

The basic value-guided construal model makes the simplifying assumption that the decision-maker plans with a single static construal. We can extend this idea to consider a decision-maker who revisits and potentially modifies their construal at each stage of planning. In particular, we can conceptualize this process in terms of a sequential decision-making problem induced by the interaction between task dynamics (such as those of a maze) and the internal state of an agent (for example, a construal)[41]. The solution to this problem is then a sequence of modified construals associated with planning over different parts of the task (for example, planning movements for different areas of the maze).

Formally, we denote the set of possible construals as $\mathcal{C} = \mathcal{P}(\{\phi_1, ..., \phi_N\})$, the powerset of cause–effect relationships, and define a construal modification Markov decision process, which has a state space corresponding to the Cartesian product of task states and construals, $(s, c) \in \mathcal{S} \times \mathcal{C}$, and an action space corresponding to possible next construals, $c' \in \mathcal{C}$. Having chosen a new construal c', the probability of transitioning from task state s to s' comes from first calculating a joint distribution using the actual transition function $P(s'|s, a)$ and plan $\pi_{c'}(a|s)$ and then marginalizing over task actions a:

$$P(s'|s, c') = \sum_a \pi_{c'}(a|s) P(s'|s, a). \tag{6}$$

In this construal modification setting, the analogue to the VOR (equation (3)) is the optimal construal modification value function, defined over all s, c:

$$V(s, c) = U(s) + \max_{c'} \left[\sum_{s'} P(s'|s, c') V(s', c') - C(c', c) \right], \tag{7}$$

where $C(c', c) = |c' - c|$ is the number of additional cause–effect relationships in the new construal c′ compared to c (for sets A and B, the set difference $A - B = \{a : a \in A \text{ and } a \notin B\}$). Importantly, this cost on modifying the construal encourages consistency—that is, without $C(c', c)$, a decision-maker would have no disincentive to completely change their construal for each state. Note that, in the special case where $c = \emptyset$, we recover the original static construal cost for a single step. Finally, using the construal modification value function, we define a softmax policy over the task/construal state space, $\pi(c'|s, c) \propto \exp\{\alpha_c^{-1}[\sum_{s'} P(s'|s, c')V(s', c') - C(c', c)]\}$. For the fixed parameter model, we set $\alpha_c = 0.1$ (as with the single-construal model).

The construal modification formulation enables us to consider not only whether an obstacle appears in a construal, but also how long it appears in a construal. In particular, we would like to compute a quantity that is analogous to equation (5) that assigns model scores for each obstacle. To do this, we use the normalized task/construal state occupancy induced by a construal policy π from the initial task/construal state, $\rho_\pi(s, c|s_0, c_0) \propto M_\pi(s_0, c_0, s, c)$, where $c_0 = \emptyset$ and M_π is the successor representation under π (for a self-contained review of M_π, see the 'Successor representation-based predictors' section below). Given a policy π and starting task state s_0, for each obstacle, we calculate the probability of having a construal that includes that obstacle:

$$P(\text{Obstacle}_i) = \sum_{s,c} \mathbb{1}\left[\phi_{\text{Obstacle}_i} \in c\right] \rho_\pi(s, c|s_0, c_0). \tag{8}$$

To calculate the optimal construal modification value function, $V(s, c)$, for each maze, we constructed construal modification Markov decision processes in Python (v.3.7.4) using scipy (v.1.5.2) sparse matrices[42]. We then exactly solved for $V(s, c)$ using a custom implementation of policy iteration[43] designed to take advantage of the sparse matrix data structure (see Code availability). For the fitted parameter models, we used separate ε-softmax noise models[39] for the computed plans, $\pi_c(a|s)$, and construal modification policy, $\pi(c'|s,c)$, and performed a grid search over the four parameters for each of the nine planning measures ($\alpha_a^{-1} \in \{1, 3, 5, 7\}$; $\varepsilon_a \in \{0.0, 0.1, 0.2\}$; $\alpha_c^{-1} \in \{1, 3, 5, 7, 9\}$; $\varepsilon_c \in \{0, 0.05, 0.1, 0.2, 0.3\}$). Moreover, for parameter fitting, we limited the construals $c' \in \mathcal{C}$ to be of size three. This improves the speed of parameter evaluation and yields results comparable to the fixed parameter model, which uses the full construal set. Finally, to obtain obstacle value-guided construal probabilities, we simulate 1,000 rollouts of the construal modification policy to estimate $\rho_\pi(\cdot|s_0, c_0)$. As with the initial model, we emphasize that these procedures are not intended as an algorithmic account of construal modification but, rather, enable us

to derive theoretically optimal predictions of the fine-grained dynamics of value-guided construals during planning.

Heuristic search over action sequences. Value-guided construal posits that people control their task representations to actively facilitate planning, which, in the maze navigation paradigm, leads to differential attention to obstacles. However, differential attention could also occur as a passive side-effect of planning, even in the absence of active construal. In particular, heuristic search over action sequences is another mechanism for reducing the cost of planning, but it accomplishes this in a different way—by examining possible action sequences in order of how promising they seem, not by simplifying the task representation. If people are simulating candidate action sequences through heuristic search (and not engaged in an active construal process), differential attention to task elements could have simply been a side effect of how those simulations unfolded.

Thus, we wanted to derive predictions of differential awareness as a byproduct of search over action sequences. To do so, we considered two general classes of heuristic search algorithms. The first, a variant of real-time dynamic programming (RTDP)[44,45], is a trajectory-based search algorithm that simulates physically realizable trajectories (that is, sequences of states and actions that could be generated by repeatedly calling a fixed transition function). The algorithm works by first initializing a heuristic value function (for example, based on domain knowledge). It then simulates trajectories that greedily maximize the heuristic value function while also performing Bellman updates at simulated states[44]. This scheme then leads RTDP to simulate states in order of how promising they are (according to the continuously updated heuristic value function) until the value function converges. Importantly, RTDP can end up visiting a fraction of the total state space, depending on the heuristic. Our implementation was based on the labelled RTDP algorithm of Bonet & Geffner[45], which additionally includes a labelling scheme that marks states where the estimate of the value function has converged, leading to faster overall convergence.

To derive obstacle awareness predictions, we ran RTDP (implemented in msdm; see Code availability) on each maze and initialized it with a heuristic corresponding to the optimal value function assuming that there are plus-shaped walls but no obstacles. This models the background knowledge that participants have about distances, while also providing a fair comparison to the initial information provided to the value-guided construal implementation. Moreover, if at any point the algorithm had to choose actions on the basis of an estimated value, ties were resolved randomly, making the algorithm stochastic. For each maze, we ran 200 simulations of the algorithm to convergence and examined which states were visited by the algorithm over all simulations. We calculated the mean number of times that each obstacle was 'hit' by the algorithm, where a hit was defined as a visit to a state adjacent to an obstacle such that the obstacle was in between the state and the goal. As the distribution of hit counts has a long tail, we used the natural log of hit counts +1 as the obstacle hit scores. The reason why the raw hit counts have a long tail is due to the particular way in which RTDP calculates the value of regions where the heuristic value is much higher than the actual value (for example, dead ends in a maze). Specifically, RTDP explores such regions until it has confirmed that it is no better than an alternative path, which can take many steps. More generally, trajectory-based algorithms are limited in that they can only update states by simulating physically realizable trajectories starting from the initial state.

The limitations of trajectory-based planning algorithms motivated our use of a second class of graph-based planning algorithms. We used LAO*[46], a version of the classic A* algorithm[47] generalized to be used on Markov decision processes (implemented in msdm; see Code availability). In contrast to trajectory-based algorithms, graph-based algorithms such as LAO* maintain a graph of previously simulated states. LAO* in particular builds a graph of the task rooted at the initial state and then continuously plans over the graph. If it computes a plan that leads it to a state at the edge of the graph, the graph is expanded according to the transition model to include that state and then the planning cycle is restarted. Otherwise, if it computes an optimal plan that only visits states in the simulated graph, the algorithm terminates. By continuously expanding the task graph and performing planning updates, the algorithm can intelligently explore the most promising (according to the heuristic) regions of the state space being constrained to physically realizable sequences. In particular, graph-based algorithms can quickly 'backtrack' when they encounter dead ends.

Obstacle awareness predictions based on LAO* were derived by using the same initial heuristic as was used for RTDP and a similar scheme for handling ties. We then calculated the total number of times an obstacle was hit during graph expansion phases only, using the same definition of a hit as above. For each maze, we generated 200 planning simulations and used the raw hit counts as the hit score.

Algorithms like RTDP and LAO* plan by simulating realizable action sequences that begin at the start state. As a result, these models tend to predict greater awareness to obstacles that are near the start state and are consistent with the initial heuristic, regardless of whether those obstacles strongly affect or lie along the final optimal path. For example, obstacles down initially promising dead ends have a high hit score. This contrasts with value-guided construal, which predicts greater attention to relevant obstacles, even if they are distant, and lower attention to irrelevant ones, even if they are nearby. For an example of these distinct model predictions, see maze 14 in Extended Data Fig. 6.

To be clear, our goal was to obtain predictions for search over action sequences in the absence of an active construal process for comparison with value-guided construal. However, in general, heuristic search and value-guided construal are complementary mechanisms, as the former is a way to plan given a representation and the latter is a way to choose a representation for planning. For example, one could perform heuristic search over a construed planning model, or a construal could help with selecting a heuristic to guide search over actions. These types of interaction between action-sequence search and construal are important directions for future research that can be built on the ideas developed here.

Successor representation-based predictors. We also considered two measures based on the successor representation, which has been proposed as a component in several computational theories of efficient sequential decision-making[35,48]. Importantly, the successor representation is not a specific model; rather, it is a predictive coding of a task in which states are represented in terms of the future states likely to be visited from that state, given the decision-maker follows a certain policy. Formally, the value function of a policy $\pi(a|s)$ can be expressed in the following two equivalent ways:

$$V_\pi(s) = U(s) + \sum_a \pi(a|s) \sum_{s'} P(s'|s,a) V_\pi(s') \tag{9}$$

$$= \sum_{s^+} M_\pi(s, s^+) U(s^+), \tag{10}$$

where $M_\pi(s, s^+)$ is expected occupancy of s^+ starting from s, when acting according to π. The successor representation of a state s under π is then the vector $M_\pi(s, \cdot)$. Algorithmically, M_π can be calculated by solving a set of recursive equations (implemented in Python with numpy[49]; see Code availability):

$$M_\pi(s, s^+) = \mathbb{1}[s = s^+] + \sum_{a,s'} \pi(a|s) P(s'|s,a) M_\pi(s', s^+). \tag{11}$$

Again, the successor representation is not itself an algorithm, but rather a policy-conditioned recoding of states that can be a component of a larger computational process (for example, different kinds

of learning or planning). Here, we focus on its use in the context of transfer learning[48,50] and bottleneck states[36,51].

Research on transfer learning posits that the successor representation supports transfer that is more flexible than pure model-free mechanisms but less flexible than model-based planning. For example, previous work[50] modelled agents that learned a successor representation for the optimal policy in an initial maze and then examined transfer when the maze was changed (for example, adding in a new barrier). While their research focuses on learning, rather than planning, we can borrow the basic insight that the successor representation induced by the optimal policy for a source task can influence the encoding of a target task, which constitutes a form of construal. In our experiments, the participants were not trained on any particular source task, but we can use the maze with all of the obstacles removed as a proxy (that is, representing what all mazes had in common). Thus, we calculated the optimal policy π for the maze without any obstacles (but with the start and goal), computed the successor representation M_π and then calculated, for each obstacle i in the actual maze with the obstacles, a successor representation overlap (SR-Overlap) score:

$$\text{SR}-\text{Overlap}(i) = \sum_{s \in \text{Obs}_i} M_\pi(s_0, s), \tag{12}$$

where s_0 is the starting state and Obs_i is the set of states occupied by the obstacle i. This quantity can be interpreted as the amount of overlap between an obstacle and the successor representation of the starting state. If the successor representation shapes how people represent tasks, this quantity would be associated with greater awareness of certain obstacles.

The second predictor is related to the idea of bottleneck states. These emerge from how the successor representation encodes multiscale task structure[36], and they have been proposed as a basis for subgoal selection[51]. If bottlenecks guide subgoal selection, then distance to bottleneck states could give rise to differential awareness of obstacles through subgoaling processes. Thus, we wanted to test that responses consistent with value-guided construal were not entirely attributable to the effect of bottleneck states calculated in the absence of an active construal process. Importantly, we note that as with alternative planning mechanisms like heuristic search, the identification of bottleneck states for subgoaling is compatible with value-guided construal (for example, one could identify subgoals for a construed version of a task).

When viewing the transition function of a task (such as a maze) as a graph over states, bottleneck states lie on either side of a partitioning of the state space into two regions such that there is high intra-region connectivity and low inter-region connectivity. This can be computed for any transition function using the normalized min-cuts algorithm[52] or derived from the second eigenvector of the successor representation under a random policy[36]. Here we use a variant of the second approach as described in the appendix of ref. [36]. Formally, given a transition function, $P(s'|s, a)$, we define an adjacency matrix, $A(s, s') = \mathbb{1}[\exists a \,.\, P(s'|s, a) > 0]$, and a diagonal degree matrix, $D(s, s) = \sum_{s'} A(s, s')$. Then, the graph Laplacian, a representation often used to derive low-dimensional embeddings of graphs in spectral graph theory, is $L = D - A$. We take the eigenvector with the second largest eigenvalue, which assigns a positive or negative value to each state in the task. This vector can be interpreted as projecting the state space onto a single dimension in a way that best preserves connectivity information, with a zero point that represents the mid-point of the projected graph. Bottleneck states correspond to those states nearest to 0. For each maze, we used this method to identify bottleneck states and further reduced these to the optimal bottleneck states, defined as bottleneck states with a non-zero probability of being visited under the optimal stochastic policy for the maze. Finally, for each obstacle, we calculated a bottleneck distance score, the minimum Manhattan distance from an obstacle to any of these bottleneck states.

Notably, value-guided construal also predicts greater attention to obstacles that form bottlenecks because one often needs to carefully navigate through them to reach the goal. However, the predictions of our model differ for obstacles that are distant from the bottleneck. Specifically, value-guided construal predicts greater attention to relevant obstacles that affect the optimal plan, even if they are far from the bottleneck (see the model predictions for maze 2 in Extended Data Fig. 5).

Perceptual landmarks. Finally, we considered several predictors based on low-level perceptual landmarks and participants' behaviour. These included the minimum Manhattan distance from an obstacle to the start location, the goal location, the centre black walls, the centre of the grid and any of the locations visited by the participant in a trial (navigation distance). We also considered the timestep at which participants were closest to an object as a measure of how recently they were near an object. In cases in which navigation distance was not an appropriate measure (for example, if the participants never navigated to the goal), we used the minimum Manhattan distance to trajectories sampled from the optimal policy averaged over 100 samples.

Experimental design

All of the experiments were preregistered (see Data availability) and approved by the Princeton Institutional Review Board (IRB). All of the participants were recruited from the Prolific online platform and provided informed consent. Sample sizes were determined on the basis of pilot experiments (see Reporting Summary). At the end of each experiment, the participants provided free-response demographic information (age and gender, coded as male/female/neither). Experiments were implemented with psiTurk[53] and jsPsych[54] frameworks (see Code availability). Instructions and example trials are shown in the Supplementary Methods.

Initial experiment. Our initial experiment used a maze-navigation task in which the participants moved a circle from a starting location on a grid to a goal location using the arrow keys. The set of initial mazes consisted of twelve 11 × 11 mazes with seven blue tetronimo-shaped obstacles and centre walls arranged in a cross that blocked movement. On each trial, the participants were first shown a screen displaying only the centre walls. When they pressed the spacebar, the circle they controlled, the goal and the obstacles appeared, and they could begin moving immediately. Moreover, to ensure that the participants remained focused on moving, we placed a green square on the goal that shrank and would disappear after 1,000 ms but reset whenever an arrow key was pressed, except at the beginning of the trial when the green square took longer to shrink (5,000 ms). The participants received US$0.10 for reaching the goal without the green square disappearing (in addition to the base pay of US$0.98). The mazes were pseudorandomly rotated or flipped, so the start and end state was constantly changing, and the order of mazes was pseudorandomized. After completing each trial, the participants received awareness probes, which showed a static image of the maze they had just navigated, with one of the obstacles shown in light blue. The participants were asked "How aware of the highlighted obstacle were you at any point?" and could respond using an eight-point scale (which was rescaled to 0–1 for analyses). Probes were presented for the seven obstacles in a maze. None of the probes were associated with a bonus.

We requested 200 participants on Prolific and received 194 complete submissions. Following preregistered exclusion criteria, a trial was excluded if, during navigation, >5,000 ms was spent at the initial state, >2,000 ms was spent at any non-initial state, >20,000 ms was spent on the entire trial or >1,500 ms was spent in the last three steps in total. Participants with <80% of trials after exclusions or who failed 2

of 3 comprehension questions were excluded, which resulted in $n = 161$ participants' data being analysed (median age of 28; 81 male, 75 female, 5 neither).

Up-front planning experiment. The up-front planning version of the memory experiment was designed to dissociate planning and execution. The main change was that, after participants took their first step, all of the blue obstacles (but not the walls or goal) were removed from the display (although they still blocked movement). This strongly encouraged planning before execution. To provide sufficient time to plan, the green square took 60,000 ms to shrink on the first step. Furthermore, on a random half of the trials, after taking two steps, the participants were immediately presented with the awareness probes (early termination trials). The other half were full trials. We reasoned that responses after early termination trials would better reflect awareness after planning but before execution (see the Supplementary Analyses (memory experiment) for analyses comparing early versus full trials).

We requested 200 participants on Prolific and received 188 complete submissions. The exclusion criteria were the same as in the initial experiment, except that the initial state and total trial time criteria were raised to 30,000 ms and 60,000 ms, respectively. After exclusions, we analysed data from $n = 162$ participants (median age of 28; 85 male, 72 female, 5 neither).

Critical mazes experiment. In the critical mazes experiment, participants again could not see the obstacles while executing and therefore needed to plan up front, but no trials ended early. There were two main differences compared with the previous experiments. First, we used a set of four critical mazes that included critical obstacles chosen to test predictions specific to value-guided construal. These were obstacles relevant to decision-making, but distant from the optimal path (see Supplementary Analyses (memory experiment) for analyses focusing on these critical obstacles). Second, half of the participants received recall probes in which they were shown a static image of the grid with only the walls, a green obstacle and a yellow obstacle. They were then asked "An obstacle was either in the yellow or green location (not both), which one was it?" and could select either option, followed by a confidence judgement on an eight-point scale (rescaled to 0–1 for analyses). Pairs of obstacles and their contrasts in the critical mazes are shown in Extended Data Fig. 4a. Participants each received two blocks of the four critical mazes, pseudorandomly oriented and/or flipped.

We requested 200 participants on Prolific and received 199 complete submissions. The trial and participant exclusion criteria were the same as in the up-front planning experiment. After exclusions, we analysed data from $n = 156$ participants (median age of 26; 78 male, 75 female, 3 neither).

Control experiments. The aim of the control experiments was to obtain yoked baselines for perception and execution for comparison with probe responses in the memory studies. The perception control used a variant of the task in which the participants were shown patterns that were perceptually identical to the mazes. Instead of solving a maze, they were told to "catch the red dot". On each trial, a small red dot could appear anywhere on the grid, and the participants were rewarded on the basis of whether they pressed the spacebar after it appeared. Each participant was yoked to the responses of a participant from either the up-front planning or critical mazes experiments. On yoked trials, the participants were shown the exact same maze/pattern as their counterpart. They were also shown the pattern for the amount of time that their counterpart took before making their first move—as the obstacles were not visible during execution for the counterpart, this is approximately the time the counterpart spent looking at the maze to plan. A red dot never appeared on these trials, and they were followed by the exact same probes that the counterpart received. References to 'obstacles' were changed to 'tiles' (for example, "highlighted tiles" as opposed to "highlighted obstacle" for the awareness probes). We also included dummy trials, which showed mazes in orientations not appearing in the yoked trials, for durations sampled from the yoked durations. Half of the dummy trials had red dots. We recruited enough participants such that at least one participant was matched to each participant from the original experiments and excluded people who said that they had participated in a similar experiment. This resulted in data from $n = 164$ participants being analysed for the initial mazes perception control (median age of 30.5; 84 male, 79 female, 1 neither) and $n = 172$ for the critical mazes perception control (median age of 36.5; 86 male, 85 female, 1 neither).

The execution control used a variant of the task in which participants followed a series of 'breadcrumbs' through the maze to the goal and so did not need to plan a path to the goal. Each participant was yoked to a counterpart in either the initial experiment or the critical mazes experiment so that the breadcrumbs were generated based on the exact path taken by the counterpart. The ordering of the mazes and obstacle probes (that is, awareness or location recall) were also the same. We recruited participants until at least one participant was matched to each participant from the original experiments. Furthermore, we used the same exclusion criteria as in the initial experiment with the additional requirement that all black dots be collected on a trial. This resulted in data from $n = 163$ participants being analysed for the initial mazes execution control (median age of 29; 86 male, 77 female) and $n = 161$ for the critical mazes execution control (median age of 30; 94 male, 63 female; 4 neither).

Process-tracing experiments. We ran process-tracing experiments using the initial mazes and the critical mazes. These experiments were similar to the memory experiments, except they used a process-tracing paradigm designed to externalize the planning process. Specifically, the participants never saw all of the obstacles in the maze at once. Rather, at the beginning of a trial, after clicking on a red X in the centre of the maze, the goal and agent appeared, and the participants could use their mouse to hover over the maze and reveal individual obstacles. An obstacle would become completely visible if the mouse hovered over any tile that was part of it for at least 25 ms, until the mouse was moved to a tile that was not part of that obstacle. Once the participant started to move using the arrow keys, the cursor became temporarily invisible (to prevent using the cursor as a cue to guide execution), and the obstacles could no longer be revealed. We examined two dependent measures for each obstacle: whether participants hovered over an obstacle and, if so, the log-transformed duration of hovering in milliseconds.

For each experiment with each set of mazes, we requested 200 participants on Prolific. The participants who completed the task had their data excluded if they did not hover over any obstacles on more than half of the trials. For the experiment with the initial mazes set, we received completed submissions from 174 people and, after exclusions, analysed data from $n = 167$ participants (median age of 30; 82 male, 82 female, 3 neither). For the experiment with the critical mazes set, we received completed submissions from 188 people and, after exclusions, analysed data from $n = 179$ participants (median age of 32; 89 male, 86 female, 4 neither).

Experiment analyses

HGLMs were implemented in Python and R using the lme4[55] and rpy2[56] packages (see Code availability). For all models, we included by-participant and by-maze random intercepts, unless the resulting model was singular, in which case we removed by-maze random intercepts. For the memory experiment analyses testing whether value-guided construal predicted responses, we fit models with and without z-score normalized value-guided construal probabilities as a fixed effect and performed likelihood ratio tests to assess significance.

For the control experiment analyses reported in the main text, we calculated mean by-obstacle responses from the perception and execution controls, and then included these values as fixed effects in models fit to the responses in the planning experiments. We then contrasted models with and without value-guided construal and performed likelihood ratio tests (additional analyses are reported in the Supplementary Analyses (memory experiment and control experiment)).

For our comparison with alternative models, we considered 11 different predictors that assign scores to obstacles in each maze: fixed-parameter value-guided construal modification probability (VGC), trajectory-based heuristic search score (Traj. HS), graph-based heuristic search score (Graph HS), bottleneck state distance (Bottleneck), successor representation overlap (SR overlap), minimum navigation distance (Nav. dist.), timestep of minimum navigation distance (Nav. dist. step), minimum optimal policy distance (Opt. dist.), distance to goal (Goal dist.), distance to start (Start dist.), distance to centre walls (Wall dist.) and distance to the centre of the maze (Centre dist.). We included predictors in the analysis of each experiment's data where appropriate. For example, in the up-front planning experiment, the participants did not navigate on early termination trials, and we therefore used the optimal policy distance rather than navigation distance. All predictors were *z*-score normalized before being included as fixed effects in HGLMs to facilitate comparison of estimated coefficients.

We performed three types of analyses using the 11 predictors. First, we wanted determine whether value-guided construal captured variability in responses from the planning experiments even when accounting for the other predictors. For these analyses, we compared HGLMs that included all predictors to HGLMs with all predictors except value-guided construal and tested whether there was a significant difference in fit using likelihood ratio tests. Second, we wanted to evaluate the relative necessity of each mechanism for explaining attention to obstacles when planning. For these analyses, we compared global HGLMs to HGLMs with each of the predictors removed and calculated the resulting change in AIC (see Extended Data Table 1 for estimated coefficients and resulting AIC values). Finally, we wanted to assess the relative sufficiency of predictors in accounting for responses on the planning tasks. For these analyses, we fit HGLMs to each set of responses that included only individual predictors or pairs of predictors and, for each model, we calculated the ΔAIC relative to the best-fitting model (Extended Data Fig. 8). Note that, for all of these models, AIC values are summed over participants.

Reporting summary

Further information on research design is available in the Nature Research Reporting Summary linked to this paper.

Data availability

Data for the current study are available through the Open Science Foundation repository https://doi.org/10.17605/OSF.IO/ZPQ69.

Code availability

Code for this study is available through the Open Science Foundation repository https://doi.org/10.17605/OSF.IO/ZPQ69, which links to a GitHub repository and contains an archived version of the repository. The value-guided construal model and alternative models were implemented in Python (v.3.7.4) using the msdm (v.0.6) library, numpy (v.1.19.2) and scipy (v.1.5.2). Experiments were implemented using psiTurk (v.3.2.0) and jsPsych (v.6.0.1). Hierarchical generalized linear regressions were implemented using rpy2 (v.3.3.6), lme4 (v.1.1.21) and R (v.3.6.1).

40. Sutton, R. S. & Barto, A. G. *Reinforcement Learning: An Introduction* (MIT Press, 2018).
41. Parr, R. & Russell, S. in Proc. *Advances in Neural Information Processing Systems* (eds Jordan, M. I. et al.) 10 (MIT Press, 1997).
42. Virtanen, P. et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods* **17**, 261–272 (2020).
43. Howard, R. A. *Dynamic Programming and Markov Processes* (MIT Press, 1960).
44. Barto, A. G., Bradtke, S. J. & Singh, S. P. Learning to act using real-time dynamic programming. *Artif. Intell.* **72**, 81–138 (1995).
45. Bonet, B. & Geffner, H. Labeled RTDP: improving the convergence of real-time dynamic programming. In *Proc. International Conference on Planning and Automated Scheduling* Vol. 3 (ed. Giunchiglia, E.) 12–21 (AAAI Press, 2003).
46. Hansen, E. A. & Zilberstein, S. LAO*: a heuristic search algorithm that finds solutions with loops. *Artif. Intell.* **129**, 35–62 (2001).
47. Hart, P. E., Nilsson, N. J. & Raphael, B. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Syst. Sci. Cybern.* **4**, 100–107 (1968).
48. Momennejad, I. et al. The successor representation in human reinforcement learning. *Nat. Hum. Behav.* **1**, 680–692 (2017).
49. Harris, C. R. et al. Array programming with NumPy. *Nature* **585**, 357–362 (2020).
50. Russek, E. M., Momennejad, I., Botvinick, M. M., Gershman, S. J. & Daw, N. D. Predictive representations can link model-based reinforcement learning to model-free mechanisms. *PLoS Comput. Biol.* **13**, e1005768 (2017).
51. Solway, A. et al. Optimal behavioral hierarchy. *PLoS Comput. Biol.* **10**, e1003779 (2014).
52. Shi, J. & Malik, J. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**, 888–905 (2000).
53. Gureckis, T. M. et al. psiTurk: an open-source framework for conducting replicable behavioral experiments online. *Behav. Res. Methods* **48**, 829–842 (2016).
54. De Leeuw, J. R. jsPsych: a JavaScript library for creating behavioral experiments in a web browser. *Behav. Res. Methods* **47**, 1–12 (2015).
55. Bates, D., Mächler, M., Bolker, B. & Walker, S. Fitting linear mixed-effects models using lme4. *J. Stat. Softw.* **67**, 1–48 (2015).
56. The rpy2 Contributors. *rpy2* version 3.3.6. (2020); https://rpy2.github.io/

Acknowledgements We thank J. Hamrick, L. Gularte, C. Sayalı, Q. Zhang, R. Dubey and W. Thompson for feedback on this work. This work was funded by NSF grant 1545126, John Templeton Foundation grant 61454 and AFOSR grant FA 9550-18-1-0077.

Author contributions All of the authors contributed to conceptualizing the project and editing the manuscript. M.K.H., D.A., M.L.L. and T.L.G. developed the value-guided construal model. M.K.H. implemented the value-guided construal model. M.K.H. and C.G.C. implemented the heuristic search models and msdm library. M.K.H., J.D.C. and T.L.G. designed the experiments. M.K.H. implemented the experiments, analysed the results and drafted the manuscript.

Competing interests The authors declare no competing interests.

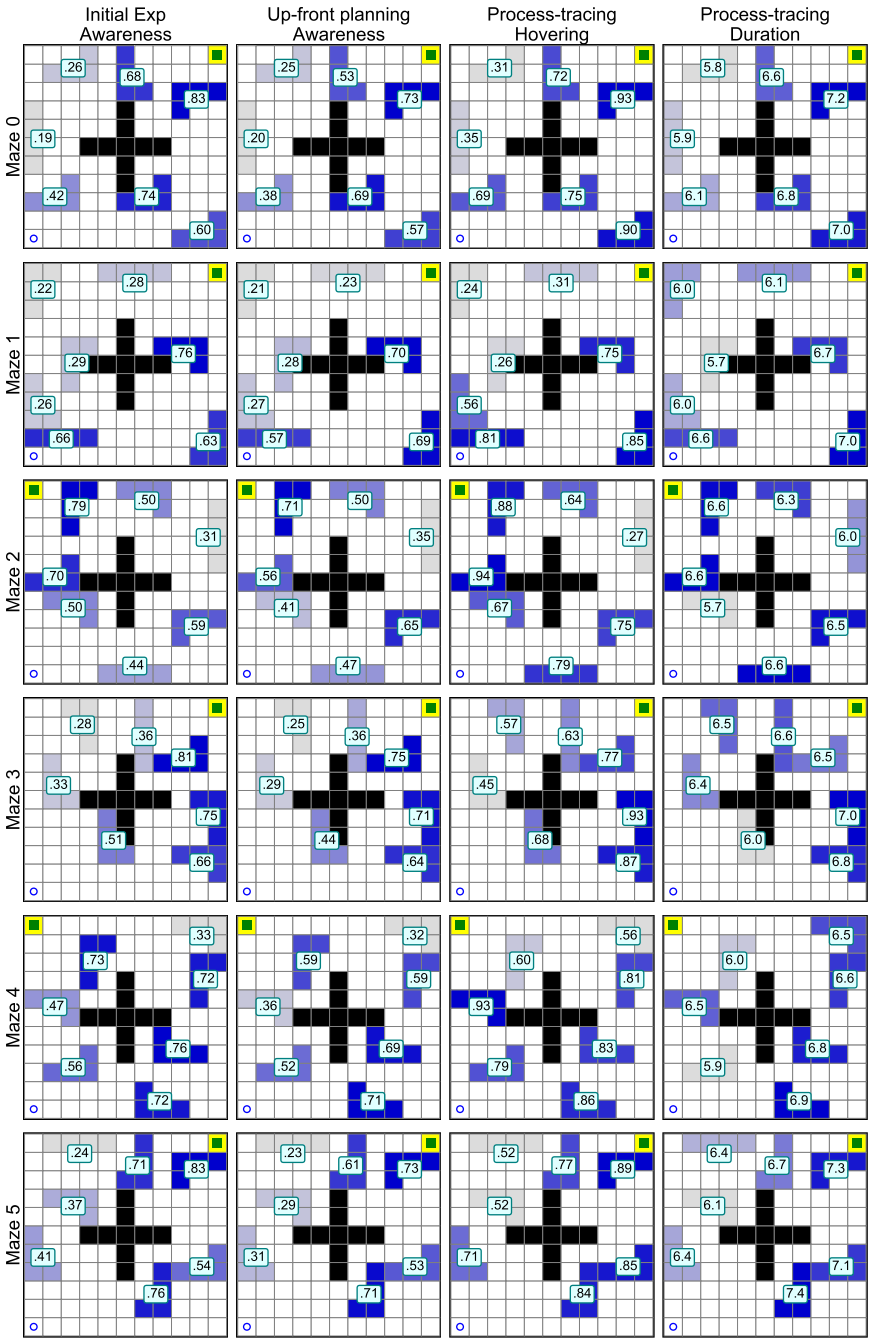
Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41586-022-04743-9.

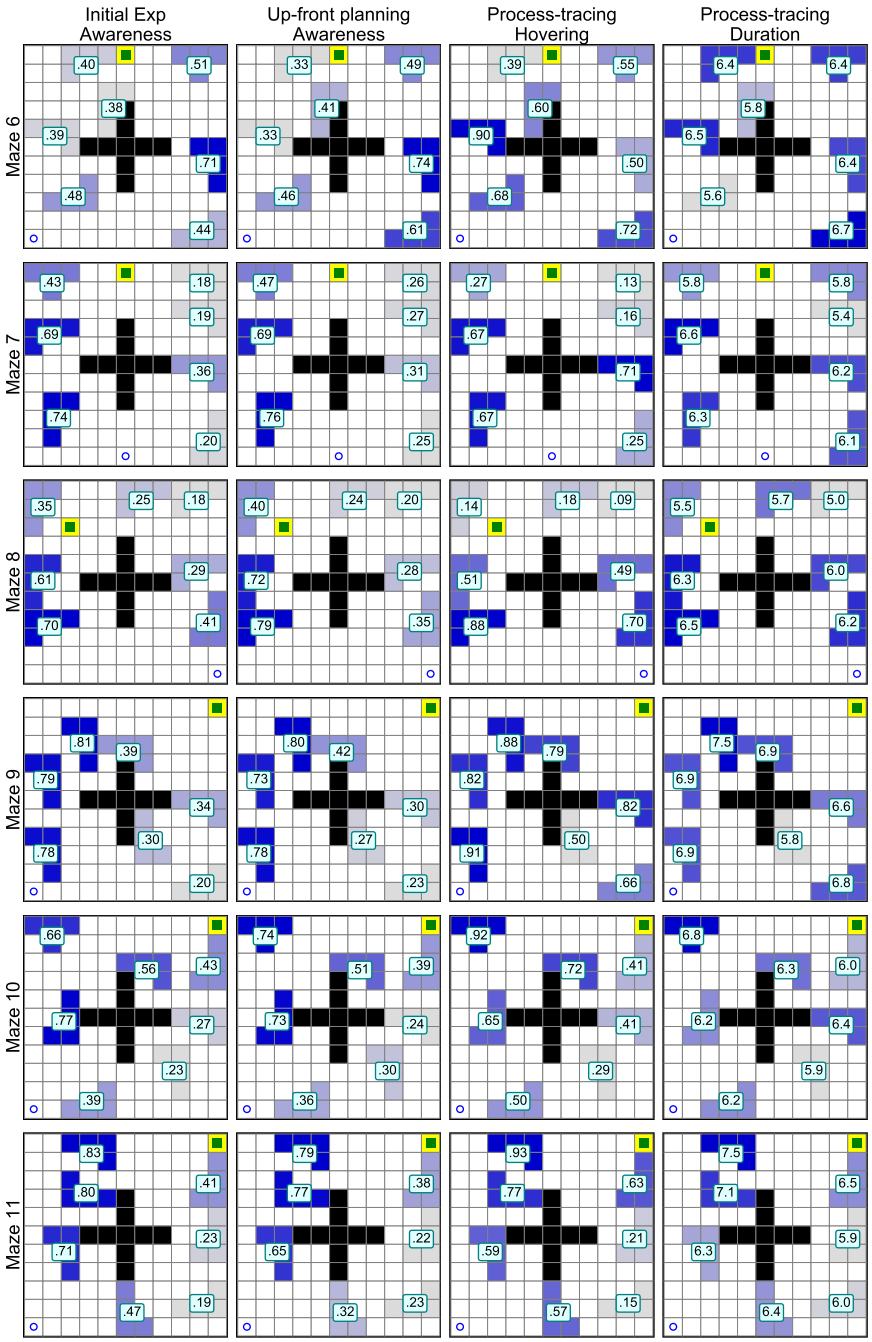
Correspondence and requests for materials should be addressed to Mark K. Ho.

Peer review information *Nature* thanks Wei Ji Ma, Redmond O'Connell and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

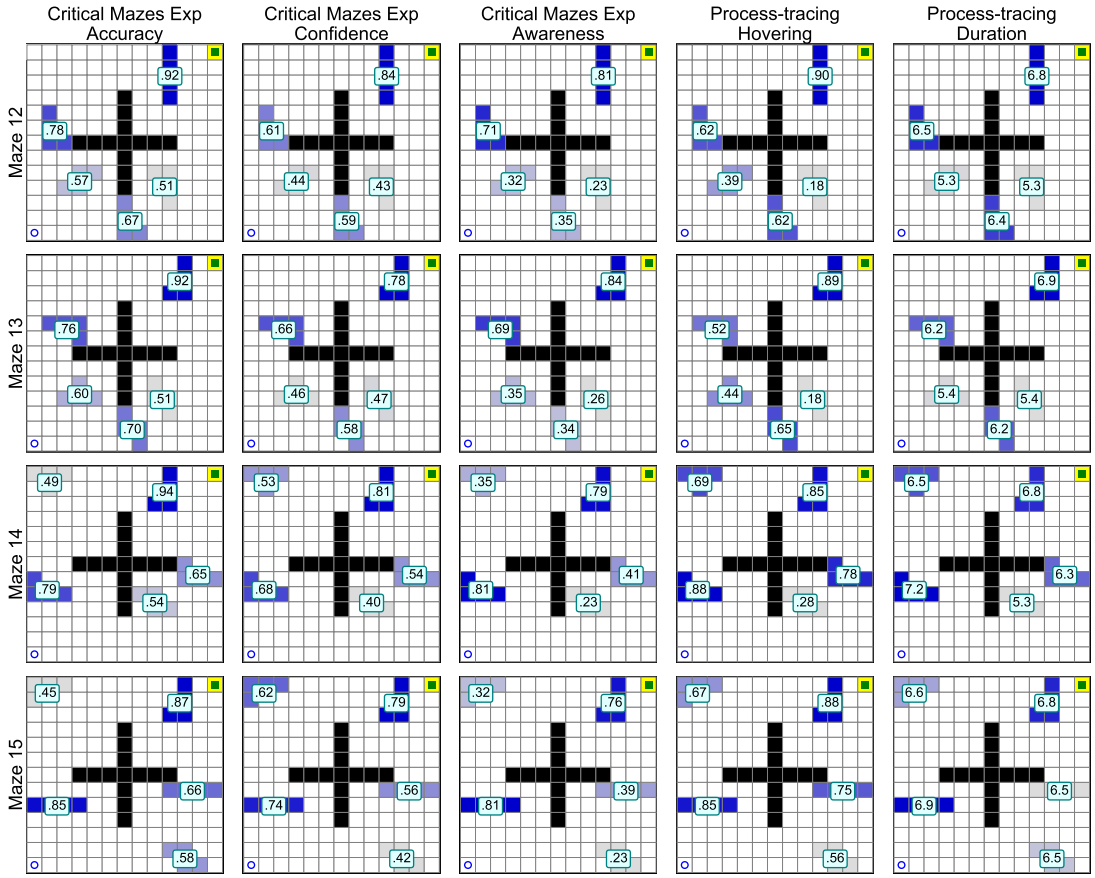
Reprints and permissions information is available at http://www.nature.com/reprints.



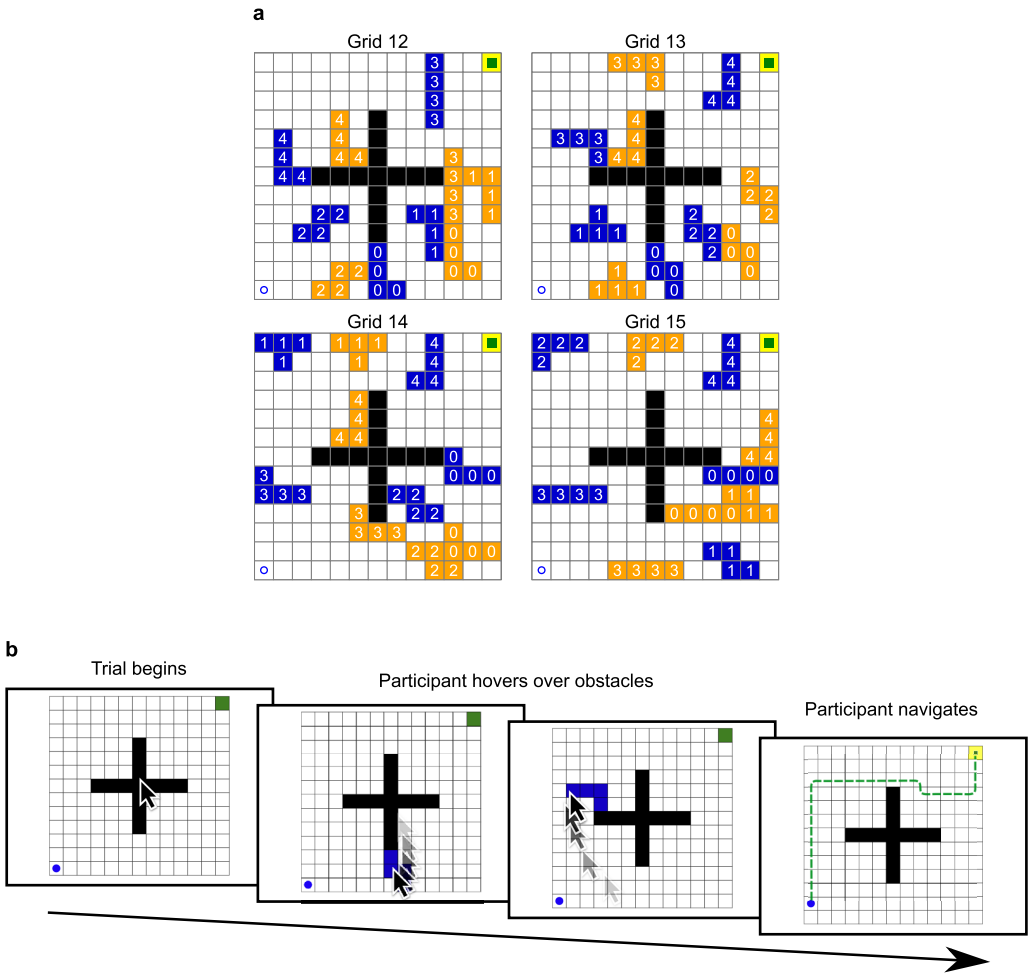
Extended Data Fig. 1 | Experimental measures on mazes 0 to 5. Average responses associated with each obstacle in mazes 0 to 5 in the initial experiment (awareness judgement), the up-front planning experiment (awareness judgement), and the process-tracing experiment (whether an obstacle was hovered over and, if so, the duration of hovering in log milliseconds). Obstacle colours are normalized by the minimum and maximum values for each measure/maze, except for awareness judgements, which are scaled from 0 to 1.



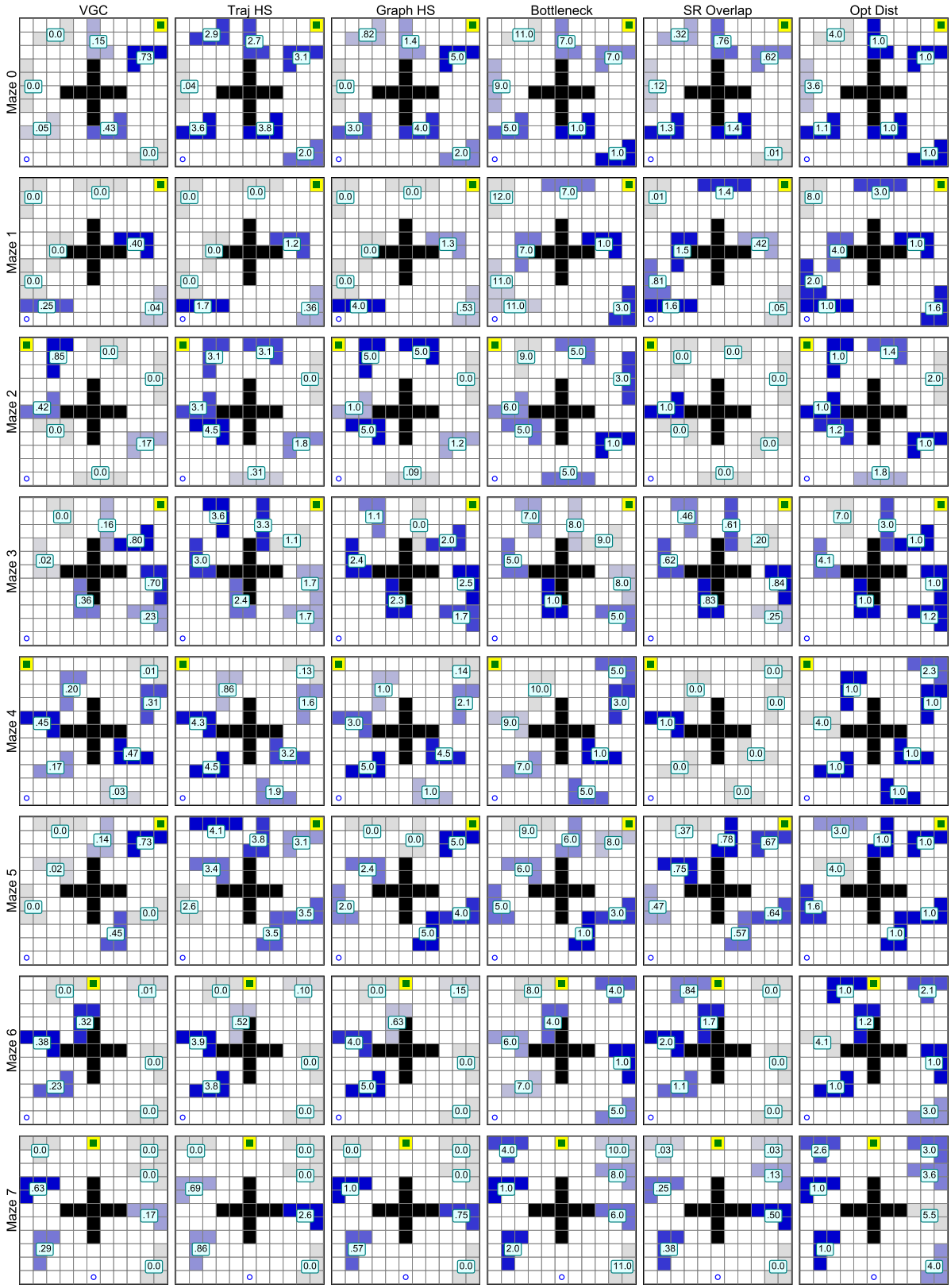
Extended Data Fig. 2 | Experimental measures on mazes 6 to 11. Average responses associated with each obstacle in mazes 6 to 11 in the initial experiment (awareness judgement), the up-front planning experiment (awareness judgement), and the process-tracing experiment (whether an obstacle was hovered over and, if so, the duration of hovering in log milliseconds). Obstacle colours are normalized by the minimum and maximum values for each measure/maze, except for awareness judgements, which are scaled from 0 to 1.



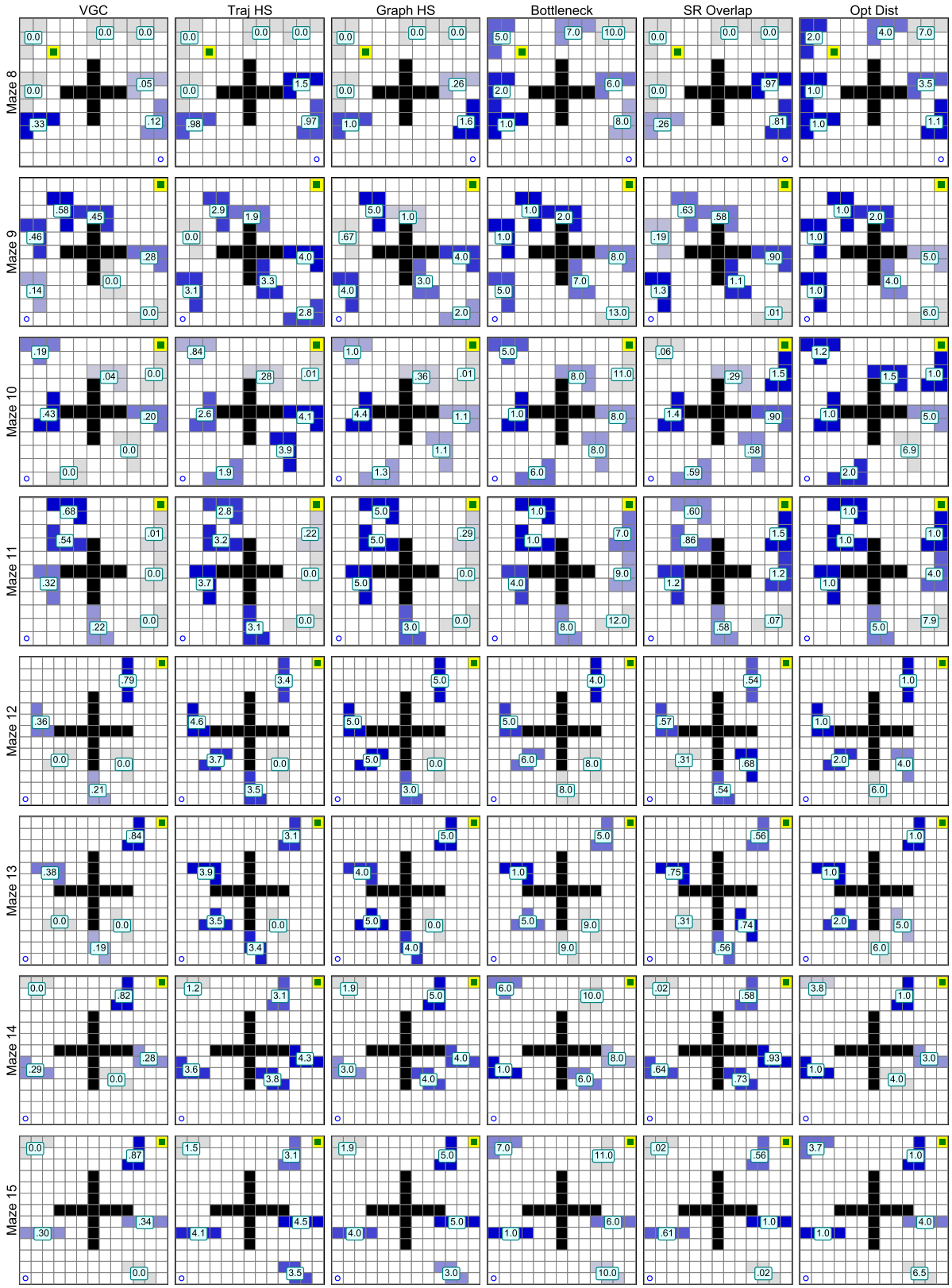
Extended Data Fig. 3 | Experimental measures on mazes 12 to 15. Average responses associated with each obstacle in mazes 12 to 15 in the critical mazes experiment (recall accuracy, recall confidence, and awareness judgement) and the process-tracing experiment (whether an obstacle was hovered over and, if so, the duration of hovering in log milliseconds). Obstacle colours are scaled to range from 0.5 to 1.0 for accuracy, 0 to 1 for hovering, confidence, and awareness judgements, and the minimum to maximum values across obstacles in a maze for hovering duration in log milliseconds.



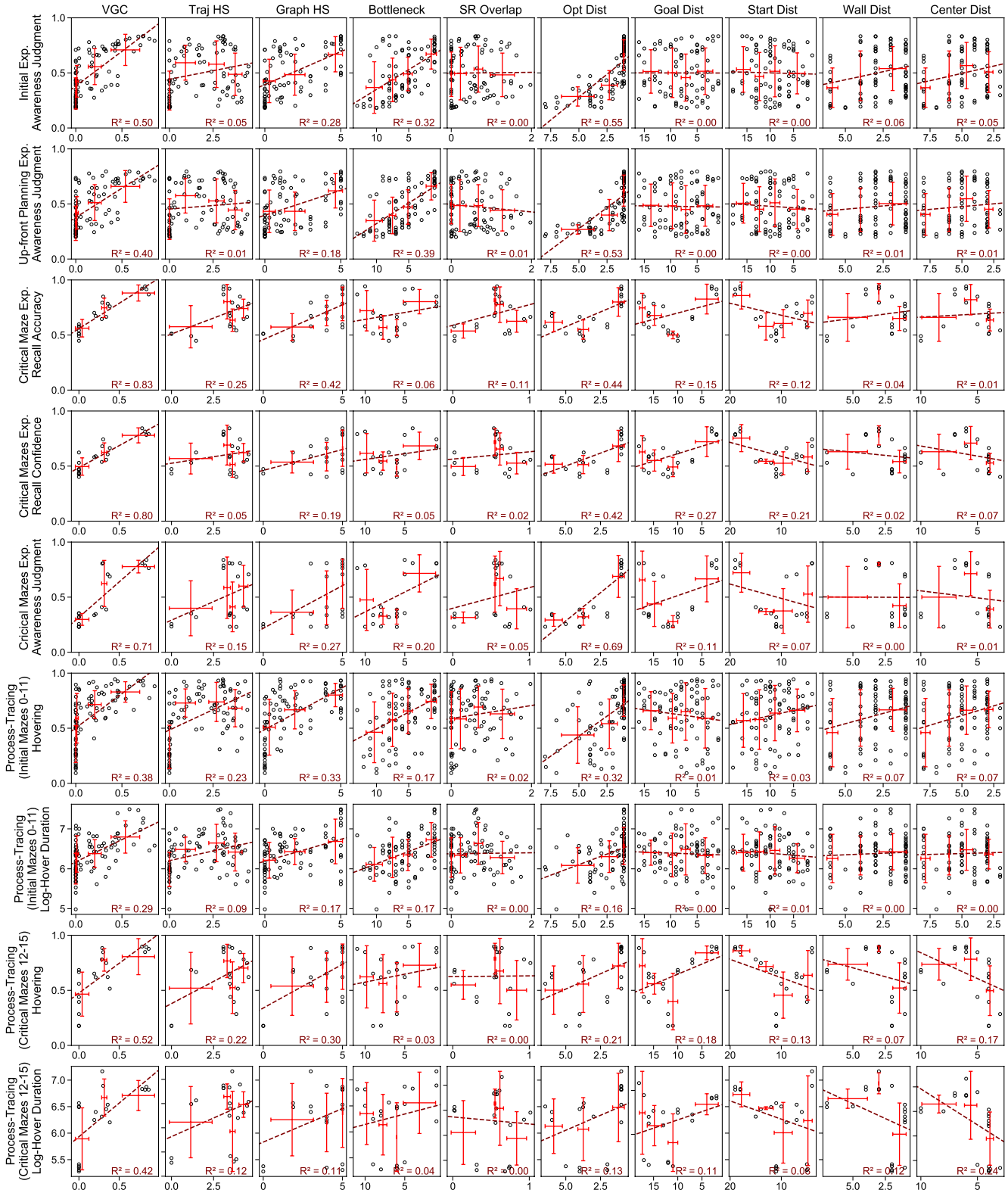
Extended Data Fig. 4 | Additional Experimental Details. a, Items from critical mazes experiment. Blue obstacles are the location of obstacles during the navigation part of the trial. Orange obstacles with corresponding number are copies that were shown during location recall probes. During recall probes, participants only saw an obstacle paired with its copy. **b**, Example trial from process-tracing experiment. Participants could never see all of the obstacles at once, but, before navigating, could use their mouse to reveal obstacles. We analyzed whether value-guided construal predicted which obstacles people tended to hover over and, if so, the duration of hovering.



Extended Data Fig. 5 | Model predictions on mazes 0 through 7. Shown are the predictions for six of the eleven predictors we tested: fixed parameter value-guided construal modification obstacle probability (VGC, our model); trajectory-based heuristic search obstacle hit score (Traj HS); graph-based heuristic search obstacle hit score (Graph HS); distance to optimal bottleneck (Bottleneck); successor representation overlap score (SR Overlap); and distance to optimal paths (Opt Dist) (see Methods, Model Implementations). Mazes 0 to 7 were all in the initial set of mazes. Darker obstacles correspond to greater predicted attention according to the model. Obstacle colours normalized by the minimum and maximum values for each model/maze.

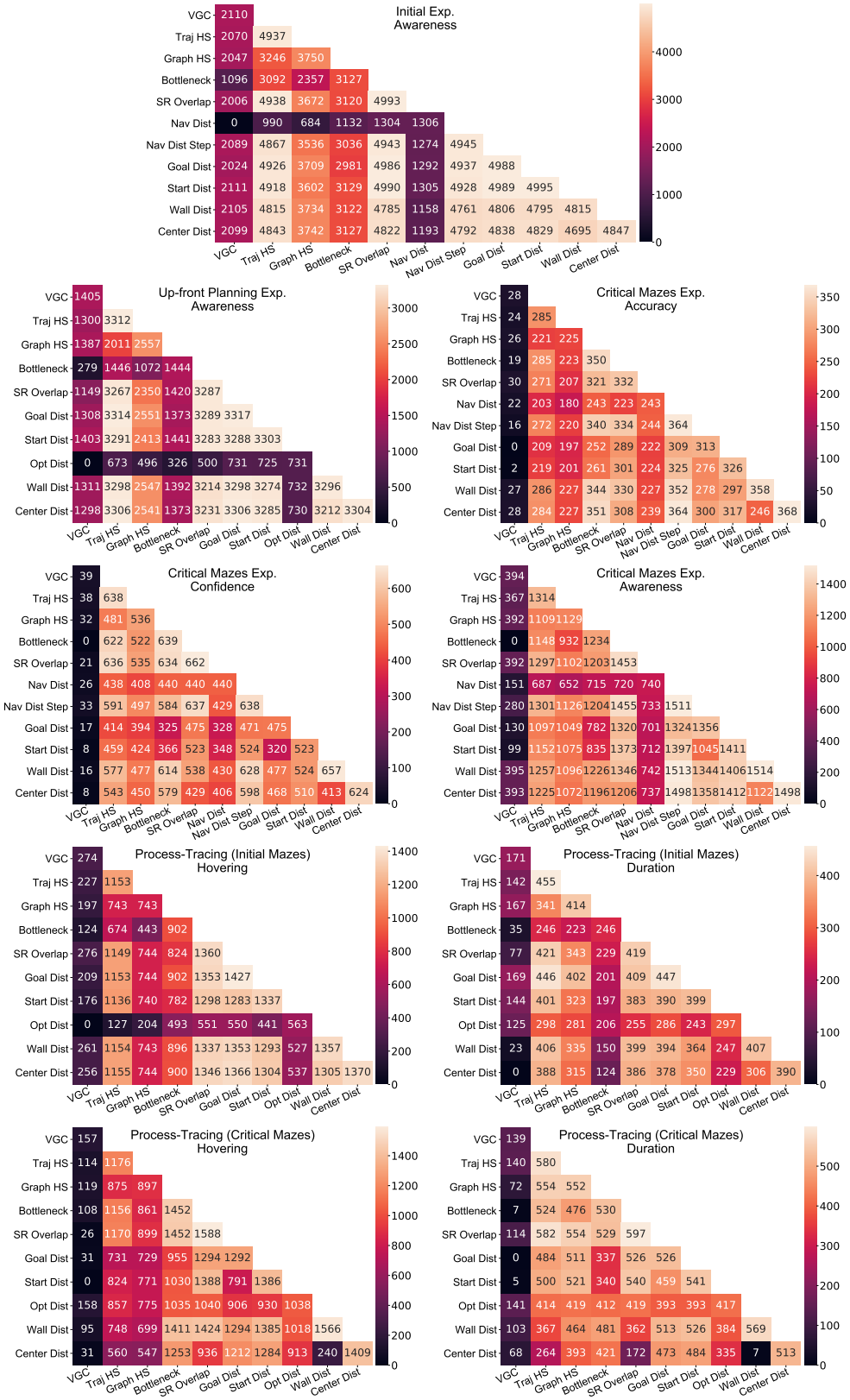


Extended Data Fig. 6 | Model predictions on mazes 8 through 15. Shown are the predictions for six of the eleven predictors we tested (see Methods, Model Implementations). Mazes 8 to 11 were part of the initial set of mazes, while mazes 12 to 15 constituted the set of critical mazes. Darker obstacles correspond to greater predicted attention according to the model. Obstacle colours normalized by the minimum and maximum values for each model/maze.



Extended Data Fig. 7 | See next page for caption.

Extended Data Fig. 7 | Summaries of candidate models and data from planning experiments. Each row corresponds to a measurement of attention to obstacles from a planning experiment: Awareness judgements from the initial memory experiment, the up-front planning experiment, and the critical mazes experiment; recall accuracy and confidence from the critical mazes experiment; and the binary hovering measure and hovering duration measure (in log milliseconds) from the two process-tracing experiments. Each column corresponds to candidate processes that could predict attention to obstacles: fixed parameter value-guided construal modification obstacle probability (VGC, our model), trajectory-based heuristic search hit score (Traj HS), graph-based heuristic search hit score (Graph HS), distance to bottleneck states (Bottleneck), successor-representation overlap (SR Overlap), expected distance to optimal paths (Opt Dist), distance to the goal location (Goal Dist), distance to the start location (Start Dist), distance to the invariant black walls (Wall Dist), and distance to the centre of the maze (Centre Dist). Note that for distance-based predictors, the x-axis is flipped. For each predictor, we quartile-binned the predictions across obstacles, and for each bin we plot (bright red lines) the mean and standard deviation of the predictor and mean by-obstacle response (overlapping bins were collapsed into a single bin). Black circles correspond to the mean response and prediction for each obstacle in each maze. Dashed dark red lines are simple linear regressions on the black circles, with R^2 values shown in the lower right of each plot. Across the nine measures, value-guided construal tracks attention to obstacles, while other candidate processes are less consistently associated with obstacle attention (data are based on $n = 84215$ observations taken from 825 independent participants).



Extended Data Fig. 8 | Sufficiency of individual and pairs of mechanisms for explaining attention to obstacles when planning. To assess the individual and pairwise sufficiency of each predictor for explaining responses in the planning experiments, we fit hierarchical generalized linear models (HGLMs) that included pairs of predictors as fixed effects. Each lower-triangle plot corresponds to one of the experimental measures, where pairs of predictors included in a HGLM as fixed-effects are indicated on the x- and y-axes. Values are the ΔAIC for each model relative to the best fitting model associated with an experimental measure (lower values indicate better fit). Values along the diagonals correspond to models fit with a single predictor. According to this criterion, across all experimental measures, value-guided construal is in the first or second best single-predictor HGLM, and is always in the best two-predictor HGLM.

Extended Data Table 1 | Necessity of different mechanisms for explaining attention to obstacles when planning

a

	Initial Exp		Up-front Exp				Critical Mazes Exp					
	Awareness		Awareness (F)		Awareness (E)		Accuracy		Confidence		Awareness	
	β	SE	β	SE	β	SE	β	SE	β	SE	β	SE
Intercept	0.530	0.017	0.325	0.019	0.494	0.016	0.560	0.066	0.644	0.041	0.429	0.044
VGC	0.085	0.004	0.095	0.006	0.081	0.006	0.150	0.015	0.118	0.009	0.212	0.009
Traj HS	0.000	0.005	0.003	0.008	0.015	0.008	0.026	0.030	-0.011	0.017	0.013	0.019
Graph HS	0.029	0.005	0.012	0.007	0.005	0.007	-0.034	0.034	-0.036	0.020	-0.042	0.021
Bottleneck	-0.028	0.003	-0.044	0.005	-0.052	0.005	0.004	0.014	-0.013	0.008	-0.034	0.009
SR Overlap	-0.031	0.003	-0.036	0.005	-0.037	0.005	-0.033	0.046	0.004	0.027	0.086	0.028
Nav Dist	-0.114	0.003	-0.277	0.012	—	—	-0.035	0.012	-0.009	0.007	-0.064	0.007
Nav Dist Step	0.014	0.004	-0.027	0.003	—	—	-0.009	0.010	-0.007	0.006	-0.043	0.007
Opt Dist	—	—	—	—	-0.084	0.005	—	—	—	—	—	—
Goal Dist	-0.007	0.005	0.005	0.007	-0.006	0.007	0.116	0.073	-0.095	0.043	0.083	0.046
Start Dist	-0.021	0.005	0.017	0.007	-0.023	0.007	0.069	0.069	-0.126	0.040	-0.014	0.043
Wall Dist	-0.237	0.024	-0.153	0.037	-0.191	0.037	-0.039	0.068	-0.095	0.040	0.066	0.043
Center Dist	0.238	0.025	0.182	0.038	0.214	0.038	0.019	0.058	0.114	0.034	0.007	0.036

b

	Process-Tracing Exp (Initial Mazes)				Process-Tracing Exp (Critical Mazes)			
	Hovering		Duration		Hovering		Duration	
	β	SE	β	SE	β	SE	β	SE
Intercept	0.717	0.128	6.515	0.071	0.544	0.320	5.887	0.181
VGC	0.588	0.035	0.234	0.017	0.702	0.054	0.481	0.030
Traj HS	0.452	0.046	0.102	0.028	0.197	0.134	0.352	0.080
Graph HS	-0.196	0.045	-0.089	0.024	0.395	0.141	-0.311	0.084
Bottleneck	-0.224	0.031	-0.175	0.016	0.058	0.059	-0.057	0.030
SR Overlap	-0.017	0.028	0.024	0.016	1.359	0.221	0.388	0.106
Opt Dist	-0.402	0.032	-0.022	0.020	0.329	0.054	0.147	0.032
Goal Dist	-0.146	0.045	0.025	0.024	-1.037	0.334	0.071	0.190
Start Dist	-0.331	0.042	0.048	0.022	-1.618	0.311	-0.190	0.182
Wall Dist	-0.987	0.207	-0.908	0.117	-0.145	0.252	-0.339	0.147
Center Dist	1.294	0.211	1.143	0.119	1.282	0.189	0.817	0.127

c

	Initial Exp	Up-front Exp		Critical Mazes Exp			Process-Tracing Exp (Initial Mazes)		Process-Tracing Exp (Critical Mazes)	
	Awareness	Awareness (F)	Awareness (E)	Accuracy	Confidence	Awareness	Hovering	Duration	Hovering	Duration
VGC	499†	280†	204†	112*	179*	484*	292*	175*	181*	249*
Traj HS	-1	-1	1	-1	-1	-1	95	10	13	13
Graph HS	30	0	-1	-1	1	1	17	11	4	9
Bottleneck	69	72	96	-1	0	13	49	113†	2	1
SR Overlap	93	55	57	-1	-1	7	-1	0	94†	10
Nav Dist	1080*	508*	—	6†	0	72†	—	—	—	—
Nav Dist Step	12	59	—	-1	0	40	—	—	—	—
Opt Dist	—	—	244*	—	—	—	163†	0	15	19
Goal Dist	0	-1	-1	1	2	1	8	0	9	-1
Start Dist	17	3	8	0	7	-1	59	2	37	-1
Wall Dist	94	14	24	-1	3	0	20	57	-1	3
Center Dist	90	20	30	-1	9†	-1	34	89	39	38†

*Largest increase in AIC after lesioning; †Second-largest increase

For each measure in each planning experiment, we fit hierarchical generalized linear models (HGLMs) that included the following predictors as fixed-effects: fixed parameter value-guided construal modification obstacle probability (VGC, our model); trajectory-based heuristic search obstacle hit score (Traj HS); graph-based heuristic search obstacle hit score (Graph HS); distance to optimal bottleneck (Bottleneck); successor representation overlap score (SR Overlap); distance to path taken (Nav Dist); timestep of point closest along path taken (Nav Dist Step); distance to optimal paths (Opt Dist); distance to the goal state (Goal Dist); distance to the start state (Start Dist); distance to any part of the centre walls (Wall Dist); and distance to the centre of the maze (Centre Dist) (Methods, Model Implementations). If the measure was taken before participants navigated, distance to the optimal paths was used, otherwise, distance to the path taken and its timestep were used. **a**, **b**, Estimated coefficients and standard errors for z-score normalized predictors in HGLMs fit to responses from the initial experiment, up-front planning experiment (F = full trials, E = early termination trials), the critical mazes experiment, and the process-tracing experiments. We found that value-guided construal was a significant predictor even when accounting for alternatives (likelihood ratio tests between full global models and models without value-guided construal: Initial Exp, Awareness: $\chi^2(1)=501.11$, $p<1.0\times10^{-16}$; Up-front Exp, Awareness (F): $\chi^2(1)=282.17$, $p<1.0\times10^{-16}$; Up-front Exp, Awareness (E): $\chi^2(1)=206.14$, $p<1.0\times10^{-16}$; Critical Mazes Exp, Accuracy: $\chi^2(1)=114.87$, $p<1.0\times10^{-16}$; Critical Mazes Exp, Confidence: $\chi^2(1)=181.28$, $p<1.0\times10^{-16}$; Critical Mazes Exp, Awareness: $\chi^2(1)=486.99$, $p<1.0\times10^{-16}$; Process-Tracing Exp (Initial Mazes), Hovering: $\chi^2(1)=294.40$, $p<1.0\times10^{-16}$; Process-Tracing Exp (Initial Mazes), Duration: $\chi^2(1)=177.58$, $p<1.0\times10^{-16}$; Process-Tracing Exp (Critical Mazes), Hovering: $\chi^2(1)=183.52$, $p<1.0\times10^{-16}$; Process-Tracing Exp (Critical Mazes), Duration: $\chi^2(1)=251.16$, $p<1.0\times10^{-16}$). **c**, To assess the relative necessity of each predictor for the fit of a HGLM, we conducted lesioning analyses in which, for each predictor in a given *global* HGLM, we fit a new *lesioned* HGLM with only that predictor removed. Each entry of the table shows the change in AIC when comparing global and lesioned HGLMs, where larger positive values indicate a greater reduction in fit as a result of removing a predictor. According to this criterion, across all experiments and measures, value-guided construal is either the first or second most important predictor. *Largest increase in AIC after lesioning; †Second-largest increase.

Extended Data Table 2 | Algorithm for Computing the VOR Function

Algorithm 1 Calculate value of representation function for construals, VOR, given states $\mathcal{S}$, initial state s_0, actions $\mathcal{A}$, transition function P, state utility function U, and primitive cause-effect relationships $\{\phi_1, ..., \phi_N\}$.

Require: ComputePlan and EvaluatePlan functions

function ConstrualValues($\mathcal{S}, s_0, \mathcal{A}, P, U, \{\phi_1, ..., \phi_N\}$)
  Initialize value of representation map VOR
  for $c \subseteq \{\phi_1, ...\phi_N\}$ **do**
    for $s \in \mathcal{S}, a \in \mathcal{A}, s' \in \mathcal{S}$ **do**
      $P_c(s' \mid s, a) = \prod_{\phi \in c} \phi(s' \mid s, a)$ ▷ Multiply primitive effects in construal
    end for
    for $s \in \mathcal{S}, a \in \mathcal{A}, s' \in \mathcal{S}$ **do**
      $P_c(s' \mid s, a) = P_c(s' \mid s, a) / \sum_z P_c(z \mid s, a)$ ▷ Normalize construed transitions
    end for
    $x_c = |c|$ ▷ Complexity of construal
    $\pi_c =$ ComputePlan$(\mathcal{S}, \mathcal{A}, P_c, U)$ ▷ Plan with construal
    $u_c =$ EvaluatePlan(P, U, s_0, π_c) ▷ Utility of construal
    VOR$[c] = u_c - x_c$ ▷ Value of representation
  end for
  return VOR
end function

To obtain predictions for our our ideal model of value-guided construal, we calculated the VOR of all construals in a maze. This was done by enumerating all construals (subsets of obstacle effects) and then, for each construal, calculating its behavioural utility and cognitive cost. This allows us to obtain theoretically optimal value-guided construals. For a discussion of alternative ways of calculating construals, see the Supplementary Discussion of Construal Optimization Algorithms.