Action Understanding as Inverse Planning
Appendix

Chris L. Baker, Rebecca Saxe & Joshua B. Tenenbaum
Department of Brain and Cognitive Sciences
Massachusetts Institute of Technology

A Markov decision problems

This section formalizes the encoding of an agent's environment and goal into a Markov decision problem (MDP), and describes how this MDP can be solved efficiently by algorithms for rational planning. Let π be an agent's plan, referred to here (and in the MDP literature) as a *policy*, such that $P_\pi(a_t|s_t, g, w)$ is a probability distribution over actions a_t at time t, given the agent's state s_t at time t, the agent's goal g and world state w. This distribution formalizes $P(\mathsf{Actions}|\mathsf{Goal}, \mathsf{Environment})$, the expression for probabilistic planning sketched in the main text. The policy π encodes all goal-dependent plans the agent could make in a given environment.

We assume that agents' policies follow the principle of rationality. Within a goal-based MDP, this means that agents choose action sequences that minimize the expected cost to achieve their goals, given their beliefs about the environment. Let $C_{g,w}(a, s)$ be the environment- and goal-dependent cost to an agent of taking action a in state s. The expected cost to an agent of executing policy π starting from state s is given by the agent's *value function*, which sums the costs the agent is expected to incur over an infinite horizon:

$$V^\pi_{g,w}(s) = E_\pi\left[\sum_{t=1}^{\infty}\sum_{a_t} P_\pi(a_t|s_t, g, w)C_{g,w}(a_t, s_t)\middle| s_1 = s\right]. \tag{1}$$

In general, cost functions may differ between agents and environments. For the environments we consider, action costs are assumed to be proportional to the negative length of the resulting movement, and the Stay action incurs a small cost as well. We assume that agents stop incurring costs once they reach their goals, implying that rational agents will try to reach their goals as quickly as possible.

The state-action value function, or Q, defines the expected cost of taking action a_t from state s_t and executing policy π afterwards by averaging possible outcomes s_{t+1} caused by a_t:

$$Q^\pi_{g,w}(s_t, a_t) = \sum_{s_{t+1}} P(s_{t+1}|s_t, a_t, w)V^\pi_{g,w}(s_{t+1}) + C_{g,w}(a_t, s_t). \tag{2}$$

Here, $P(s_{t+1}|s_t, a_t, w)$ is the state transition distribution, which specifies the probability of moving to state s_{t+1} from state s_t, as a result of action a_t, in world w. For simplicity, we assume state transitions are deterministic in our experiments.

The optimal action from any state is computed by greedily choosing the action that maximizes the Q-function. However, instead of this deterministic policy, we assume that agents have a probability distribution over actions associated with policy π:

$$P_\pi(a_t|s_t, g, w) \propto \exp\big(\beta Q^\pi_{g,w}(s_t, a_t)\big). \tag{3}$$

This type of policy is called a *Boltzmann policy*, which takes the "soft-max" of the Q-function, yielding an approximate principle of rationality, where β controls the amount of noise in the agent's actions.

The optimal Boltzmann policy π^* and the value function of π^* satisfy the Bellman equation [1] for all states:

$$V^{\pi^*}_{g,w}(s) = \sum_{a_t} P_{\pi^*}(a_t|s_t, g, w) Q^{\pi^*}_{g,w}(s_t, a_t). \tag{4}$$

These equations can be solved efficiently using the value iteration algorithm [2], which iteratively updates the left- and right-hand sides of Equation 4 for all states until convergence to a unique fixed point.

Given an agent's situation-specific policy, goal inferences and goal-based action predictions depend crucially on the prior over goals, corresponding to $P(\mathsf{Goal}|\mathsf{Environment})$ in the main text. This prior is instantiated by different models within the inverse planning framework. In the next section, we describe three inverse planning models based on different hypotheses about people's prior knowledge about goals, denoted M1(β), M2(β,γ), and M3(β,κ), which roughly correspond to the three kinds of explanations we offered for the woman's anomalous behavior in the introductory vignette in the main text. In addition to these models, we also consider a simple heuristic alternative, denoted H(β), that people might apply in action understanding. For each model, we will describe the computations involved in each of our three key tasks: online goal inference, retrospective goal inference and goal-based action prediction.

These three kinds of inferences all depend on applying Bayes' rule, in which we compare all possible goal hypotheses against each other, in terms of how well they explain an observed action sequence. Each goal hypothesis yields a different MDP that must be solved. In this paper, we exhaustively enumerate the full hypothesis space of goals under each model, solving the corresponding MDP for each one. This is computationally feasible for the simple environments and the restricted hypothesis spaces of goals that we consider here. Real world settings will likely require more complex goal priors which allow goals to be selected from a very large or infinite hypothesis space. In this case, full enumeration will not be possible, and efficient and accurate approximate inference schemes will be required [7, 9]. Likewise, in MDPs defined over complex environments with many state variables, the state space quickly grows too large to solve the MDP exactly. In these cases, approximate rational planning algorithms are necessary, and this is an active area of research [5, 4, 10, 8] within the field of artificial intelligence.

B Inverse planning models

B.1 Model 1: single underlying goal

Our first model, denoted M1(β), is our most basic instantiation of inverse planning. M1 assumes that agents have one underlying goal in each action sequence that must be inferred. A graphical model of M1 is shown in Fig. 1(a). Observations of agents are assumed to begin at time $t = 1$, with the agent occupying state s_1 in environment w. The agent is assumed to have an invariant goal g, which generates actions a_1 through a_{T-1} according to the policy from Equation 3, where the parameter β determines the agent's level of determinism. At high values of β, agents rarely deviate from the optimal path to their goals, but at low β values, agents' behavior is noisy, becoming a random walk at $\beta = 0$. Agents' actions generate state transitions, producing the state sequence $s_1, s_2, \ldots, s_T$. The objective of inverse planning is to invert this model, yielding inferences of agents' goals g, given observations of the state sequence $s_{1:T}$.

Given an observed state sequence and the environment, the distribution over the agent's goal in M1 is computed using Bayes' rule:

$$P(g|s_{1:T}, w) \propto P(s_{2:T}|s_1, g, w) P(g|w), \tag{5}$$

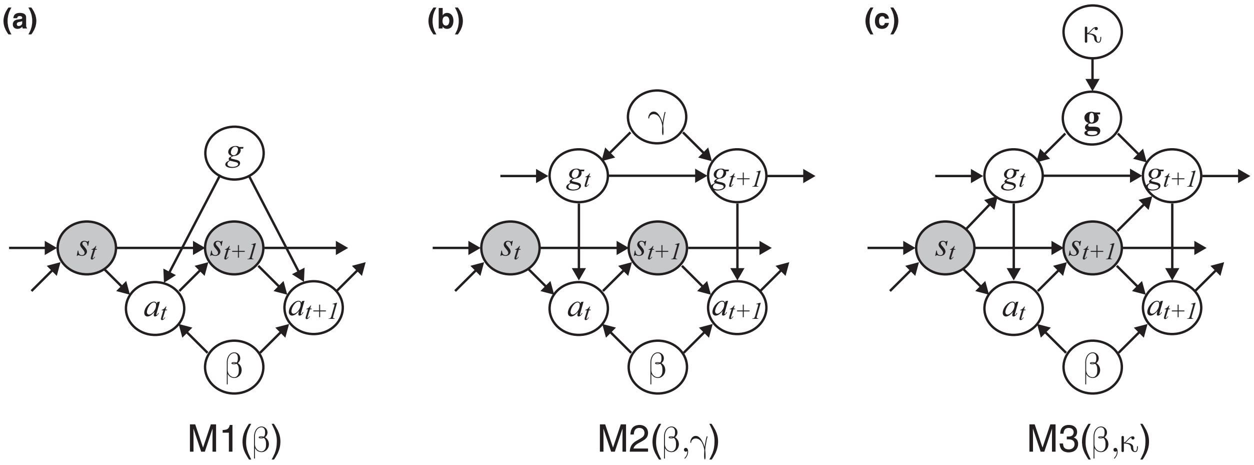

Figure 1: Graphical models of M1, M2 and M3. All variables except β, γ and κ are implicitly conditioned on the environment w.

where $P(g|w)$ is the prior over the agent's goal. There are many possible ways to treat $P(g|w)$, but in our experiments, for simplicity, we will assume it is uniform over the set of goals that are realizable in the environment w, *i.e.* each position in the grid not blocked by an obstacle. In Equation 5, the probability of a state sequence given a goal and the environment, $P(s_{2:T}|s_1, g, w)$, is computed as the product of the probabilities of the individual state transitions from s_t to s_{t+1}, given the goal and the environment:

$$P(s_{2:T}|s_1, g, w) = \prod_{t=1}^{T-1} P(s_{t+1}|s_t, g, w). \tag{6}$$

In Equation 6, we marginalize over actions to compute the probability of the next state s_{t+1}, given the current state s_t, the goal g, and the environment w:

$$P(s_{t+1}|s_t, g, w) = \sum_{a_t \in A_{s_t}} P(s_{t+1}|s_t, a_t, w) P_\pi(a_t|s_t, g, w). \tag{7}$$

Equation 5 corresponds directly to Equation 1 in the main text, and allows us to model people's online and retrospective goal inferences in Experiments 1 and 2.

Prediction of future actions, given observations of previous actions, is done by computing the probability of a novel state sequence $s'_{1:T}$ in the environment w, conditioned on a set of N observed state sequences of length T $s^{1:N}_{1:T}$ in w. Assuming the agent's goal is unchanged, this is just an average over the predicted actions for each possible goal g, weighted by the posterior probability of g given the observed sequences:

$$P(s'_{2:T}|s'_1, s^{1:N}_{1:T}, w) = \sum_g P(s'_{2:T}|s'_1, g, w) P(g|s^{1:N}_{1:T}, w). \tag{8}$$

Equation 8 corresponds directly to Equation 2 in the main text, and allows us to model people's action predictions in Experiment 3.

Unlike the more complex models presented below, M1 explains agents' actions in terms of a single underlying goal, and accounts for all unexplained deviations from the shortest path to the goal in terms of random departures from optimality. However, there is often not a single invariant goal that fully explains an agent's actions. Instead, real-world agents typically pursue goals with rich structure, such as subgoals, multiple objectives, and goal changes, as in the introductory example. Next, we describe models M2 and M3, which assume more complex goal structures. These models generalize M1: M1(β) is a special case of both M2(β,γ) and M3(β,κ), with γ or κ equal to 0.

B.2 Model 2: changing goals

The next model we consider extends M1 by assuming that agents' goals can change over time. We denote this model M2(β,γ), where γ is the probability that the agent will change its goal at each time. The graphical model of M2, shown in Fig. 1(b), expands on the graphical model of M1 by replacing the single node g with a Markov chain representing the probability distribution over the sequence of goals $g_1, g_2, \ldots, g_{T-1}$.

In M2, goals are indexed by time, and we modify Equation 5 for M1 to represent $P(g_t|s_{1:T}, w)$, the posterior probability of an agent's goals at each time, with $t < T$. In the online case, goal inferences are provided by the posterior distribution over goals at time t, given a state sequence $s_{1:t+1}$ and world w. To compute this, we recursively define the *forward distribution*:

$$P(g_t|s_{1:t+1}, w) \propto P(s_{t+1}|g_t, s_t, w) \sum_{g_{t-1}} P(g_t|g_{t-1}, w) P(g_{t-1}|s_{1:t}, w), \tag{9}$$

where the recursion is initialized with $P(g_1|w)$, the probability distribution over initial goals at time $t = 1$, before any state transitions have been observed. $P(g_t|g_{t-1}, w)$ is the conditional distribution over changing to goal g_t from g_{t-1}. We assume that $P(g_1 = i|w) = \theta_i$, and that $P(g_t = i|g_{t-1} = j, w) = (1-\gamma)\delta_{ij} + \gamma\theta_i$, where θ is typically assumed to be uniform over the set of goals[1].

Equation 9 is an online version of Equation 1 in the main text, which allows us to model people's online goal inferences in Experiment 1. When $\gamma = 0$, M2 reduces to M1, and goal changing is prohibited. When $\gamma = 1$, the model is equivalent to randomly choosing new goal i with probability θ_i at each time step. Intermediate values of γ between 0 and 1 interpolate between these extremes.

The retrospective version of Equation 5 for M2 is given by $P(g_t|s_{1:T}, w)$, the marginal probability of a goal at time t given the entire state sequence $s_{1:T}$, with $t < T - 1$. To compute this, we use a variant of the forward-backward algorithm [6]. The forward distribution is defined by Equation 9 above. The *backward distribution* is recursively defined by:

$$P(s_{t+2:T}|g_t, s_{1:t+1}, w) = \sum_{g_{t+1}} P(g_{t+1}|g_t, w) P(s_{t+2}|g_{t+1}, s_{t+1}, w) P(s_{t+3:T}|g_{t+1}, s_{t+2}, w). \tag{10}$$

The marginal probability of goal g_t given the state sequence $s_{1:T}$ is the product of the forward and backward distributions:

$$P(g_t|s_{1:T}, w) \propto P(g_t|s_{1:t+1}, w) P(s_{t+2:T}|g_t, s_{1:t+1}, w). \tag{11}$$

Equation 11 is a smoothed version of Equation 1 in the main text, allowing us to model subjects' retrospective goal inferences in Experiment 2. The parameter γ plays a key role in retrospective inferences, determining how information from past and future movements is integrated into the distribution over current goals. When $\gamma = 0$, M2 reduces to M1: changing goals is prohibited, and future information constrains the probability of all past goals to be equal to $P(g_{T-1}|s_{1:T}, w)$. When $\gamma = 1$, only the movement from

[1] Note that this implies that given a goal change, the agent sometimes chooses the same goal as before, and only chooses a *new* goal with probability $\gamma(K-1)/K$. We choose this parameterization for clarity and consistency, but M2 can easily be reparameterized by $\gamma' \leftarrow \gamma K/(K-1)$.

s_t to s_{t+1} carries information about g_t; all other past and future movements carry no information about the current goal.

Prediction of agents' future actions in M2 differs from prediction in M1 because agents' goals are no longer assumed constant across observation sequences. Rather than inferring an invariant goal, in M2, prediction of agents' future actions, given their past actions, is done by inferring θ. The posterior distribution over θ, given a set of N observed state sequences of length T $s_{1:T}^{1:N}$ in environment w is computed by Bayes' rule:

$$P(\theta|s_{1:T}^{1:N}, w) \propto P(s_{2:T}^{1:N}|s_1^{1:N}, \theta, w)P(\theta|w). \tag{12}$$

The probability of a novel state sequence $s'_{1:T}$, given environment w and previously observed state sequences $s_{1:T}^{1:N}$ in w is then:

$$P(s'_{1:T}|s_{1:T}^{1:N}, w) = \int_\theta P(s'_{2:T}|s'_1, w, \theta)P(\theta|s_{1:T}^{1:N}, w)d\theta. \tag{13}$$

Equation 13 corresponds to Equation 2 in the main text. Because the integral in Equation 13 is intractable, we use a grid approximation to integrate over θ to model people's judgments in Experiment 3[2].

B.3 Model 3: complex goals

Our final model extends M1 by assuming that agents' goals can be *complex*, and can include the constraint that agents must pass through a sequence of "subgoals" along the way to their end goals. We denote this model M3(β,κ). A graphical model of M3 is shown in Fig. 1(c).

In M3, complex goals $\mathbf{g}$ are a list of states, instead of a single state g, as in M1. Let $\mathbf{g}$ be a complex goal with 0 or more subgoals, represented as a list of length $\dim(\mathbf{g})$, where $\dim(\mathbf{g}) \geq 1$. $P(\mathbf{g}|w)$ is the prior over complex goals. The prior assumes that with probability κ, agents pick a subgoal uniformly from the state space, and with probability $1-\kappa$, agents do not choose a subgoal. Agents continue choosing additional subgoals with probability κ until the sampling terminates with probability $1-\kappa$, implying that number of subgoals chosen is geometrically distributed. Then, agents pick an end goal uniformly from the state space. M3 includes M1 as a special case with $\kappa = 0$, implying that the agent picks 0 subgoals.

The posterior probability of a complex goal under this model is computed identically to M1, except now $\mathbf{g}$ is a complex goal:

$$P(\mathbf{g}|s_{1:T}, w) \propto P(s_{2:T}|s_1, \mathbf{g}, w)P(\mathbf{g}|w). \tag{14}$$

The probability of an agent's state sequence, given a complex goal $\mathbf{g}$ and environment w, is computed by segmenting the full state sequence into sequences in which the agent pursues particular subgoals. Once a subgoal is reached, a new segment begins in which the agent pursues its next subgoal. Formally,

$$P(s_{2:T}|s_1, \mathbf{g}, w) = \prod_{i=1}^{\dim(\mathbf{g})} \prod_{t=k_{i-1}}^{k_i} P(s_{t+1}|s_t, g_t = i, w), \tag{15}$$

where $k_0 = 1$ and $k_i = \min(\{t|s_t = g_i \wedge t > k_{i-1}\})$, $i > 0$.

Inferences about an agent's end goal are obtained by marginalizing over goal types, and within the complex goal type, marginalizing over possible subgoals. Let $\mathbf{g} = [g_{1:\dim(\mathbf{g})-1}\ g_{\dim(\mathbf{g})}]$. Then

$$P(g_{\dim(\mathbf{g})}|s_{1:T}, w) \propto \sum_{g_{1:\dim(\mathbf{g})-1}} P(s_{2:T}|s_1, \mathbf{g}, w)P(\mathbf{g}|w) \tag{16}$$

[2] Another possible approximation is to compute the maximum likelihood (ML) estimate θ_{ML}, conditioned on state sequences $s_{1:T}^{1:N}$ and environment w using the EM algorithm, and use this ML estimate to predict hypothetical actions.

is the marginal probability of an end goal, given a state sequence and the environment. Equation 16 corresponds to Equation 1 in the main text, and allows us to model people's online and retrospective inferences in Experiments 1 and 2.

Prediction in M3 is similar to prediction in M1, but instead of averaging over simple goals, we average over complex goals:

$$P(s'_{1:T}|s^{1:N}_{1:T}, w) = \sum_{\mathbf{g}} P(s'_{2:T}|s'_1, \mathbf{g}, w)P(\mathbf{g}|s^{1:N}_{1:T}, w). \tag{17}$$

Equation 17 corresponds directly to Equation 2 in the main text, and allows us to model people's action predictions in Experiment 3.

B.4 Heuristic alternative H

The heuristic alternative we consider, denoted H(β), assumes that the probability of a goal is a function of only the agent's last observed movement. H is inspired by heuristics that categorize intentional movement based on temporally local motion cues, such as the rate of change of the relative distance between two objects. For the sake of comparison with our other models, we formulate H as a limiting case of inverse planning. H is equivalent to a special case of M2, with probability of changing goals at each timestep $\gamma = 1$, implying that the agent chooses a new goal at each timestep. This yields a version of Equation 11 that only depends on the agent's last movement, regardless of the length of the agent's movement history:

$$P(g_t|s_{1:t+1}, w) \propto P(s_{t+1}|s_t, g_t, w)P(g_t|w). \tag{18}$$

In contrast, Equation 5 for M1, Equation 11 for M2, and Equation 16 for M3 infer a goal based on how well it explains the agent's entire state sequence, rather than just the last movement.

C Experiment 1

C.1 Bootstrap cross-validated correlational analyses

We assessed the statistical significance of the differences between M1, M2, M3 and H with a bootstrap cross-validated [3] correlational analysis. Bootstrap cross-validation (BSCV) is a non-parametric technique for model selection, which measures the goodness-of-fit of models to data while preventing overfitting and controlling for model complexity. For a large number of iterations N, our analysis selected random training subsets of size k of participants' data (by sampling uniformly with replacement from all data points) and found the parameters of each model that yielded the highest correlations with these training datasets. For each iteration, we then computed the correlation of the model fit on that iteration with a testing dataset given by the complement of the training dataset on that iteration. Across all iterations, we computed the average correlation $\langle r \rangle$ of each model with the testing datasets, and we computed the proportion of iterations in which the correlation of one model was greater than another model. This allowed us to estimate the goodness-of-fit of each model class, and to compute p-values for significance tests of whether one model class predicted people's data better than another. Our BSCV analyses of Experiment 1 (overall and targeted) used parameters $N = 10000$ and $k = 50$.

C.2 Analysis of parameters

This analysis examined a grid of parameter values for each model. We tested each model using 10 evenly spaced β values from 0.5 to 5. For M2, for each value of β we tested 20 evenly spaced γ values between

0.05 and 1. For M3, for each value of β we tested 20 evenly spaced κ values between 0.05 and 1. For each model, a grid of correlation coefficients with people's judgments using these parameters is shown in Fig. 2.

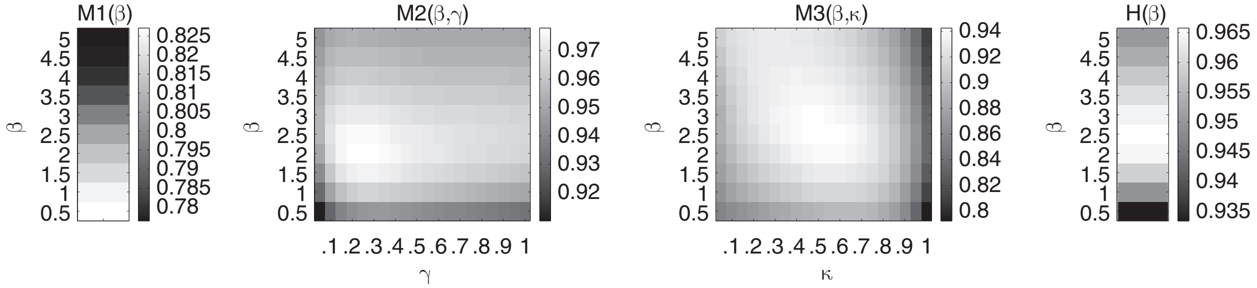

Figure 2: Correlation grids for M1, M2, M3 and H for Experiment 1. These plots show the correlation of each model with people's data for a grid of parameter values. M2, M3 and H all correlated highly with people's ratings, with maximum correlations with people's data of $r > 0.97$, $r > 0.94$ and $r > 0.96$, respectively, while M1 correlated less highly with people's judgments, with a maximum correlation of $r > 0.82$.

For each model, a robust range of parameters yielded correlations that were close to the optimum. For M1, all parameter values yielded correlations that were low relative to M2, M3 and H. Among the parameter settings we tested for M1, low β values yielded the highest correlations with people's data, because M1 could only explain the indirect paths in our stimuli in terms of noisy actions.

For M2, all parameter values that we tested yielded correlations above $r = 0.9$. The optimal range for β was between 1.5 and 2.5, and the optimal range for γ was between 0.10 and 0.25. The worst correlations were obtained at β values below 1.0, which yielded a uniform distribution over actions in the limit at $\beta = 0$, and at γ values below 0.5, which yielded M1 in the limit at $\gamma = 0$. The other limit, at $\gamma = 1$, yielded H, for which the same range of β values yielded good performance as for M2.

For M3, changing the parameter values resulted in a wider numerical range of correlations with subjects' data, from a minimum around $r = 0.8$, to a maximum near $r = 0.94$. M3 correlated relatively highly with people's judgments at values of β above 1.0 and values of κ below 0.8. The optimal range for β was between 1.0 and 3.0, and the optimal range for κ was between 0.2 and 0.7.

D Experiment 2

D.1 Bootstrap cross-validated correlational analysis

Our BSCV analysis of Experiment 2 used parameters $N = 10000$ and $k = 50$.

D.2 Analysis of parameters

This analysis examined a grid of parameter values for each model, using the same values as our analysis of the parameters for Experiment 1. A grid of the correlation coefficients of each model with people's judgments using these parameters is shown in Fig. 3.

In Experiment 2, models based on changing goals (M2 and H) correlated highly with people's data, while models based on static goals (M1 and M3) correlated relatively poorly with people's data. For all the models we tested, β values below 1.0 yielded the best correlations with people's judgments, which was

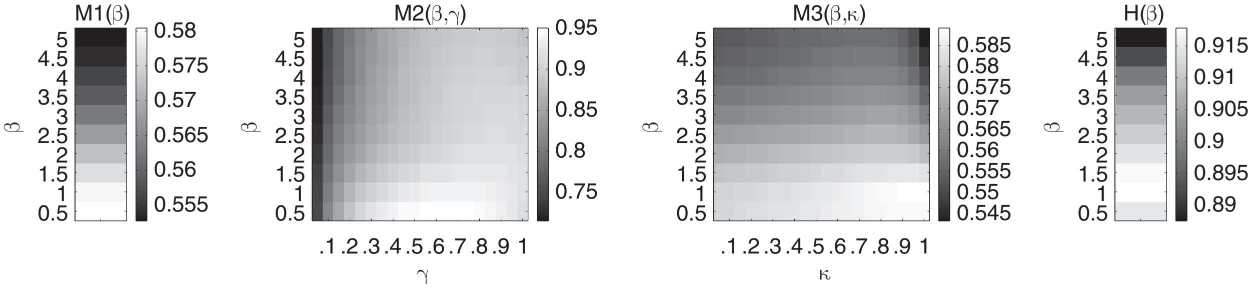

Figure 3: Correlation grids for M1, M2, M3 and H for Experiment 2. These plots show the correlation of each model with people's data for a grid of parameter values. Models that could infer changing goals (M2 and H) correlated highly with people's ratings, with maximum correlations of $r > 0.95$ and $r > 0.91$, respectively. Models that could represent only static goals (M1 and M3) correlated less highly with people's judgments, both with maximum correlations with people's data of $r > 0.58$.

lower than in Experiment 1. For M2, γ values above 0.2 yielded the highest correlations with people's data, and the optimal range was between 0.4 and 0.7. For M3, the optimal range for κ was from 0.7 to 1.0.

E Experiment 3

E.1 Data processing and model fitting

We modeled people's judgments in Experiment 3 using the log posterior odds ratios between the two hypothetical actions in the response phase. Because the log posterior odds ratios varied between $-\infty$ and ∞, and mean subject ratings varied between 1 and 9, we first mapped the log posterior odds ratios to the interval $(0, 1)$, then rescaled these values to the interval $(1, 9)$. Our mapping was based on the assumption that the log posterior odds ratios were normally distributed, and that subjects made their ratings by dividing the range of log posterior odds into 9 regions of equal probability, corresponding to the 9 points on the rating scale. To map the log posterior odds ratios to the interval $(0, 1)$, we first computed the z-scores of all log posterior odds ratios, then mapped these z-scores through the sigmoidal normal cumulative density function. We then computed correlation coefficients between these values and people's ratings.

For M1 and M3, we assumed that the agent's goal could be visible or invisible, and that the space of potential goals and subgoals was given by all grid squares in the environment. For M2, we restricted the space of goals to just the marked end goal and the potential subgoal of the particular condition. We modeled participants' learning over training trials as inferring the probability of changing to either goal in M2. The probability of the agent choosing the end goal, given a goal change, was represented by the parameter θ, and the probability of choosing the subgoal was given by $1 - \theta$. To compute the integral over θ for posterior prediction of future actions, we approximated the continuous range of θ values from 0 to 1 using a discretization of 21 intervals.

E.2 Bootstrap cross-validated correlational analysis

Our BSCV analysis of Experiment 3 used parameters $N = 10000$ and $k = 20$.

E.3 Analysis of parameters

This analysis examined a grid of parameter values for each model, using the same values as our analyses of the parameters for Experiment 1 and Experiment 2. A grid of the correlation coefficients of each model with people's judgments using these parameters is shown in Fig. 4.

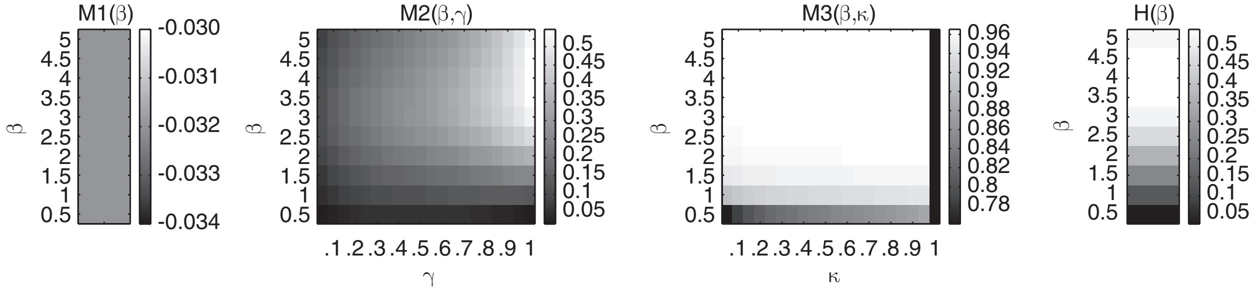

Figure 4: Correlation grids for M1, M2, M3 and H for Experiment 3. These plots show the correlation of each model with people's data for a grid of parameter values. Only M3 correlated highly with people's judgments ($r > 0.96$), while M2 and H correlated relatively poorly with people's data ($r > 0.5$), and the predictions of M1 were uncorrelated with people's data ($r > -0.03$).

For Experiment 3, the predictions of M1 were uncorrelated with people's judgments because M1 made the same prediction in each trial: that the agent would take the direct path to its goal. M1 made this prediction regardless of the value of β. For M2, M3 and H, higher β values yielded a better fit to people's data than in the previous experiments, probably because the agents took straighter, less noisy paths in these stimuli.

For M2 and H, all parameter values yielded a low correlation coefficient with people's judgments. Although our simulation of M2 (and H) used an approximation (described in Appendix B.2) this approximation faithfully captured the aspects of the stimuli that were most relevant for our subjects. Our approximation first restricted the hypothesis space of goals to include just the subgoal and end goal from the stimuli, which was warranted by the fact that these goals explained the full range of agents' actions in the stimuli. Second, our approximation of the integral from Equation 13 was accurate, introducing a minimal amount of error into the model predictions.

For M2, high values of γ, which included H as a special case, performed best. High γ values allowed M2 to infer goal changes at every timestep, and to absorb the probability of switching to the subgoal versus the end goal into the stimulus-dependent θ posterior. However, with this parameterization, M2 was still a poor predictor of people's judgments. M2 could potentially predict a sequence of goals that first pursued the subgoal, and then pursued the end goal, but M2 lacked a strong inductive bias over the sequence of goals. Even if M2 learned from the training stimuli that the agent pursued the subgoal half of the time, and the end goal half of the time, it had no way to predict in what order the agent would pursue these goals in the test stimuli.

For M3, all values of κ from .05 to .95 performed well for β above 1.5. Values of β below 1.5 performed worst because they tended to explain indirect paths as random "noise", rather than inferring a subgoal, and thus did not generalize evidence for subgoals from the training stimuli to the test stimuli. The correlation coefficient between M3 and people's ratings was relatively insensitive to the value of κ because κ had an approximately additive effect on the log posterior odds ratio for moderate values, which did not affect the correlation.

F General discussion

F.1 Hierarchical Bayesian analysis: model selection across experiments

Here we make precise the hierarchical Bayesian analysis laid out in the General Discussion of the main text for explaining how participants inferred which model or goal prior was appropriate to use in each experiment. Let D denote the ensemble of data that subjects saw during the course of a particular experiment, consisting of a pair of $\mathsf{Actions}$ and $\mathsf{Environment}$ for each of N trials, such that $D = \{\langle \mathsf{Actions}_i, \mathsf{Environment}_i \rangle | i \in 1, \ldots, N\}$. The probability of a particular model M given D can then be computed using Bayes' rule:

$$P(M|D) \propto P(D|M)P(M), \tag{19}$$

where $P(M)$ is the prior probability of the model. The likelihood of the model $P(D|M)$ is the product of the marginal probability of the actions in each trial, conditioned on the environment in that trial and the model. To compute the marginal probability for each trial, we sum over all possible goals because the agent's actions also depend on its goal:

$$P(D|M) = \prod_{i=1}^{N} \sum_{\mathsf{Goals}} P(\mathsf{Actions}_i | \mathsf{Goal}, \mathsf{Environment}_i, M) P(\mathsf{Goal} | \mathsf{Environment}_i, M).$$

The term inside the sum is just the probabilistic planning likelihood multiplied by the prior over goals, taken directly from Equation 1 in the main text. Thus, the likelihood of a model depends on how well it explains all the agents' actions across all trials.

To test whether people's inferences can be explained by rational inference in an HBM, we compared the log-marginal likelihood of M1, M2, M3 and H for each experiment, $\log P(D|M)$ from Equation 19 (this is denoted $\log P(\mathsf{Stimuli}|\mathsf{Model})$ in the main text). This involves integrating over all degrees of freedom in the HBM, including all possible goals and the parameters β, γ and κ for each model, assuming a uniform prior.

References

[1] Richard Bellman. *Dynamic Programming*. Princeton University Press, Princeton, N.J., 1957.

[2] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 2. Athena Scientific, Belmont, MA, 2nd edition, 2001.

[3] Paul R. Cohen. *Empirical methods in artificial intelligence*. MIT Press, Cambridge, MA, 1995.

[4] Carlos Guestrin, Daphne Koller, Ronald Parr, and Shobha Venkataraman. Efficient solution algorithms for factored MDPs. *Journal of Artificial Intelligence Research*, 19:399–468, 2003.

[5] Andrew Y. Ng and Michael I. Jordan. Pegasus: A policy search method for large MDPs and POMDPs. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, 2000.

[6] Lawrence R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, 77, 1989.

[7] Deepak Ramachandran and Eyal Amir. Bayesian inverse reinforcement learning. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence*, 2007.

[8] Emanuel Todorov. Linearly-solvable Markov decision problems. In *Advances in Neural Information Processing Systems 19*, pages 1369–1376. MIT Press, 2006.

[9] Emanuel Todorov. Efficient algorithms for inverse optimal control. In *International Conference on Machine Learning*, Under Review.

[10] David Wingate and Kevin D. Seppi. Prioritization methods for accelerating MDP solvers. *Journal of Machine Learning Research*, 6:851–881, 2005.