Bayesian Inverse Motion Planning for Online Goal Inference in Continuous Domains

Tan Zhi-Xuan, Jovana Kondic, Stewart Slocum,
Joshua B. Tenenbaum, Vikash K. Mansinghka, Dylan Hadfield-Menell

***Abstract*— Humans and other agents navigate their environments by acting efficiently to achieve their goals. In order to infer agents' goals from their actions, it is thus necessary to model how agents achieve their goals efficiently. Here, we show how online goal inference and trajectory prediction in continuous domains can be performed via *Bayesian inverse motion planning*: By modeling an agent as an approximately Boltzmann-rational motion planner that produces low-cost trajectories while avoiding obstacles, and placing a prior over goals, we can infer the agent's goal and future trajectory from partial trajectory observations. We compute these inferences online using a sequential Monte Carlo algorithm, which accounts for the multimodal distribution of trajectories due to obstacles, and exhibits better calibration at early timesteps than a Laplace approximation and a greedy baseline.**

I. Introduction

For autonomous agents to collaborate fluidly with humans in physical spaces, they need to rapidly infer human goals from low-level observations of motion. This goal inference problem has been tackled through a variety of approaches, including deep learning [1], inverse reinforcement learning [2], [3], [4], plan recognition [5], [6], [7], and Bayesian inverse planning [8], [9]. Most of these approaches build upon the insight that humans can be modeled as goal-directed planners who act efficiently to achieve their goals. Under this assumption, a goal is more likely if observed behavior corresponds with an efficient plan to achieve that goal.

Here we demonstrate how this insight can be applied to continuous domains. Unlike most goal inference approaches, which are typically restricted to discretized [10] or symbolic domains [6], we model agents as goal-directed *motion* planners, allowing us to handle continuous environments and observations. Building upon work in legible motion planning [11] and probabilistic programming [12], [13], we incorporate motion planning algorithms into a Bayesian agent model, allowing us to condition on motion trajectories to infer the agent's underlying goal. In contrast to approaches that use RRT planners [7], [12], [13], we model agents as *Boltzmann-rational* motion planners, directly capturing the relationship that lower-cost plans are more probable, while enabling the use of gradient-based trajectory optimization to generate plans [14], [15], [16]. For inference, we develop an asymptotically-accurate sequential Monte Carlo (SMC) algorithm, which approximates the full posterior distribution over goals and future trajectories in an online manner.

The authors are with the Department of Brain and Cognitive Sciences (BCS) and Department of Electrical Engineering and Computer Science (EECS) of the Massachusetts Institute of Technology. {xuan, jkondic, sslocum3, jbt, vkm, dhm}@mit.edu

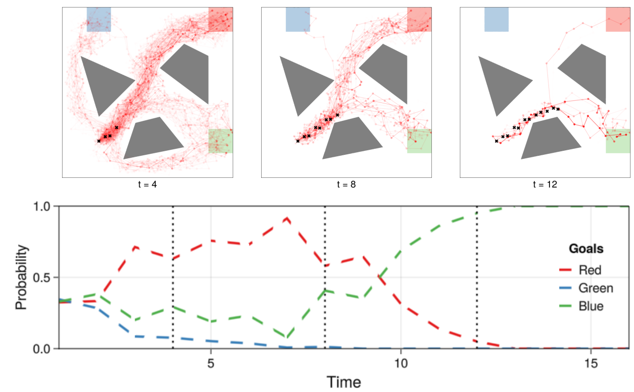

Fig. 1: Online goal inference (below, dashed) and trajectory prediction (above, red) from noisy observations (above, black markers) in a continuous obstacle-laden environment, via Bayesian inference over a motion planning agent.

II. Modeling Agents as Goal-Directed Motion Planners

We define a Bayesian model of agents as goal-directed motion planners, where motion plans are generated in accordance with the principle of rational action [17], [8]. Specifically, we place a uniform prior over possible goal regions $g \in G$, and over trajectory endpoints in each goal region $x_T \in g$. We then assume that agent trajectories (i.e. motion plans) $x_{1:T}$ are drawn from a distribution $P(x_{1:T}|x_1, x_T)$ over low-cost, obstacle-avoiding trajectories with (known) start and (sampled) end points (x_1, x_T). Finally, for each timestep $t \in [1, T]$, we model observations o_t as normally distributed around the agent's location at that timestep ξ_t with noise σ:

$$\textit{Goal Prior:} \quad g \sim \text{Uniform}(G) \tag{1}$$

$$\textit{Endpoint Prior:} \quad x_T \sim \text{Uniform}(g) \tag{2}$$

$$\textit{Motion Planning:} \quad x_{1:T} \sim P(x_{1:T}|x_1, x_T) \tag{3}$$

$$\textit{Observation Noise:} \quad o_t \sim \text{Normal}(x_t, \sigma) \tag{4}$$

To model approximately rational, low-cost trajectories, it is common to use a Boltzmann distribution with a cost function $C(x_{1:T})$ and rationality parameter α [10], [11]

$$\pi(x_{1:T}|x_1, x_T) = \frac{1}{Z(x_1, x_T)} \exp[-\alpha C(x_{1:T})] \tag{5}$$

$$C(x_{1:T}) = C_{\text{smooth}}(x_{1:T}) + \lambda C_{\text{obs}}(x_{1:T}) \tag{6}$$

where $Z(\cdot, \cdot)$ is an endpoint-dependent normalizing constant, and the cost function includes terms for smoothness and obstacle avoidance [14], [15].

However, evaluating the probability of a trajectory $x_{1:T}$ under the Boltzmann distribution π (a.k.a. the maximum entropy distribution [3]) requires the normalizing constant $Z(x_1, x_T)$, which is intractable to compute in continuous domains. We instead assume that agents sample motion plans $x_{1:T}$ according to a *Monte Carlo approximation* $\hat{\pi}$ of the true Boltzmann distribution π, drawing upon the insight that planning can formulated as inference [18], [19], [20]. In particular, we use a SMC approximation of π, which proposes initial trajectories then iteratively adapts them to the target distribution π via a series of Langevin Monte Carlo (LMC) [21] and Newtonian Monte Carlo (NMC) [22] kernels. These kernels are stochastic analogues to first and second-order trajectory optimization steps [14], [15], so our SMC approximation can be viewed as an algorithm for stochastic trajectory optimization. Our SMC approximation $\hat{\pi}$ also produces unbiased estimates $\hat{Z}$ of the normalizing constant Z. This allows us to compute unbiased estimates of the *normalized* probability $\hat{\pi}(x_{1:T}|x_1, x_T)$ of a trajectory, which is sufficient for sound Bayesian inference [12], [23].

III. Online Goal Inference via Bayesian Inverse Motion Planning

Having defined our agent model, our aim is to infer the agent's goal g and future trajectory $x_{\tau+1:T}$ given observations of its trajectory so far $o_{1:\tau}$ and a known initial location x_1:

$$P(g, x_{\tau+1:T}|x_1, o_{1:\tau}) \propto P(g)P(x_T|g)P(x_{1:T}|x_1, x_T)\prod_{t=1}^{\tau} P(o_t|x_t) \quad (7)$$

We sequentially approximate this posterior distribution using another SMC algorithm called Sequential Monte Carlo for Inverse Motion Planning (SMC-IMP). Like other particle filtering algorithms, SMC-IMP works by first sampling a set of N hypotheses from the model $(g, x_{1:T})^{1:N}$ and assigning them equal weights $w^{1:N}$. At each timestep t, a new observation o_t arrives, and the hypotheses are adjusted to better to explain o_t, then reweighted according to how well they explain o_t. After reweighting, the hypotheses are further adjusted by LMC and NMC rejuvenation kernels [24]. The weighted collection of hypotheses $((g, x_{1:T}), w)^{1:N}$ at timestep t represents a discrete approximation to the posterior over goals and full trajectories $P(g, x_{1:T}|x_1, o_{1:t})$. Dropping the first t steps of each hypothesized trajectory gives an approximation to our desired posterior $P(g, x_{t+1:T}|x_1, o_{1:t})$. Pseudocode for SMC-IMP is shown in Algorithm 1.

Algorithm 1 SMC for Inverse Motion Planning (SMC-IMP)

procedure SMC-IMP(x_1, $o_{1:\tau}$, N)
- $(g, x_{1:T})^i \sim P(g, x_{1:T}|x_1)$ **for** $i \in [1, N]$ ▷ Sample N hypotheses
- $w^i \leftarrow P(o_1|x_1)$ **for** $i \in [1, N]$ ▷ Initialize weights
- **for** $t \in [2, \tau],\ i \in [1, N]$ **do**
  - $\tilde{x}_t^i \sim K(x_t; o_t)$ ▷ Propose new $\tilde{x}_t$ close to o_t
  - $\tilde{x}_{1:T}^i \leftarrow (x_{1:t-1}^i, \tilde{x}_t, x_{t+1:T}^i)$ ▷ Replace x_t with new $\tilde{x}_t$
  - $w^i \leftarrow w^i \frac{K(\tilde{x}_t^i; o_t)}{L(x_t^i; \tilde{x}_{1:T}^i)} \frac{P(g^i, \tilde{x}_{1:T}^i|x_1)}{P(g^i, x_{1:T}^i|x_1)} P(o_t|\tilde{x}_t^i)$ ▷ Reweight hypotheses
  - $x_{1:T}^i \sim \mathrm{LMC}(\cdot, \mathrm{NMC}(\cdot; \tilde{x}_{1:T}^i))$ ▷ Rejuvenate via NMC & LMC
- **end for**
- **return** $((g, x_{1:T}), w)^{1:N}$ ▷ Return weighted hypotheses

end procedure

	$P(g_{\text{true}}\|o_{1:t})$				Brier Score			
Method	$t = T/5$	$T/4$	$T/3$	$T/2$	$t = T/5$	$T/4$	$T/3$	$T/2$
Greedy	0.42	0.45	0.51	0.62	0.67	0.65	0.59	0.45
Laplace	**0.60**	**0.61**	**0.73**	**0.85**	0.68	0.66	0.45	**0.25**
SMC-IMP	0.53	**0.61**	0.70	0.79	**0.48**	**0.43**	**0.40**	0.33

TABLE I: Probability assigned to the true goal $P(g_{\text{true}}|o_{1:t})$ and Brier scores at various timesteps t (as a fraction of trajectory length T), averaged across trajectories.

IV. Experiments & Discussion

We evaluated the goal inference capabilities of our method on 45 agent trajectories across 5 different 2D scenes, each of which contained 3 possible goal regions, and varying obstacle setups (e.g. smaller scattered obstacles, a maze, a tunnel). These scenes were designed to test our model's ability to generate diverse trajectory hypotheses, as well as our inference algorithm's ability to handle multimodal posterior distributions over trajectories to a goal. In each scene, we generated a set of three trajectories per goal (of length $T = 21$), corresponding to different paths through the obstacles whenever more than one path was plausible.

We configured our SMC algorithm to sample $N = 600$ particles, and use a rationality parameter of $\alpha = 20$. We compared against two baselines: A greedy distance-based heuristic, which assigns higher probability to goals that are closer to the most recent observation $P(g|o_{1:t}) \propto \exp(\text{min-dist}(o_t))$; and a Laplace approximation to the posterior over goals, which models the distribution over trajectories to each goal as a multivariate Gaussian around the lowest-cost trajectory, and ignores observation noise [11]. In contrast to both of these baselines, SMC-IMP accounts for the multimodal distribution over trajectories due to the presence of obstacles.

In Table I, we report the posterior probability each method assigns to the true goal at $\frac{1}{5}$, $\frac{1}{4}$, $\frac{1}{3}$, and $\frac{1}{2}$ the length of the trajectory (rounded to the nearest timestep), averaged across all trajectories in the dataset. We also report the Brier score at the same timesteps, which reflects how well-calibrated the predictions are (lower is better) [25]. As expected, SMC-IMP out-performs the greedy baseline. Compared to the Laplace approximation, SMC-IMP tends to assign lower probability to the true goal, but with a better Brier score at earlier timesteps. This reflects that SMC-IMP is better at maintaining uncertainty when the data is ambiguous, whereas the Laplace approximation is over-confident at earlier timesteps, assigning high probabilities to the wrong goals.

While our method performs reasonably on this dataset, many challenges remain. In the future, we aim to improve the runtime and accuracy of the algorithm through better Monte Carlo approximations of the Boltzmann distribution, which may involve the use of smarter proposal distributions or specialized solvers to generate candidate motion plans. We also aim to scale this approach to higher dimensions, and to explore the possibility of integrating SMC-IMP with existing algorithms for inverse task-level planning [9], with the eventual hope of performing goal inference and trajectory prediction over task-and-motion plans.

Potential applications for our method include human motion-prediction for collision avoidance and human-robot collaboration, especially for manipulation problems. Here, accurate inference of human goals and future trajectories remains a major obstacle towards safe and high autonomy collaborative systems [26]. In contrast to existing deep learning approaches to motion prediction [27], [28], [29], our method would not require hours of data collection and retraining for each new problem.

Concretely, adapting our method to a human-robot collaboration problem would involve scaling to 3D domains and estimating the human's state in such a way that a cost function and trajectory optimization could be applied (e.g. with pose estimation). We are optimistic about the potential of our method to scale to difficult 3D motion planning problems since our trajectory sampling algorithm can be viewed as a probabilistic modification to successful and widely-used SQP-based motion planners [16]. The inferred trajectories produced by our algorithm could then be used as inputs to the robot's cost function and problem definition, which could be solved using standard robotic planning methods, especially probabilistic algorithms like belief-space planning [30], [31].

ACKNOWLEDGEMENTS

The authors would like to thank Alex Lew for helpful discussion and comments on using SMC to compute an unbiased estimate of the normalizing constant over trajectories.

REFERENCES

[1] N. Rabinowitz, F. Perbet, F. Song, C. Zhang, S. A. Eslami, and M. Botvinick, "Machine theory of mind," in *International conference on machine learning*, pp. 4218–4227, PMLR, 2018.

[2] D. Ramachandran and E. Amir, "Bayesian inverse reinforcement learning.," in *IJCAI*, vol. 7, pp. 2586–2591, 2007.

[3] B. D. Ziebart, A. L. Maas, J. A. Bagnell, A. K. Dey, *et al.*, "Maximum entropy inverse reinforcement learning.," in *AAAI*, vol. 8, pp. 1433–1438, Chicago, IL, USA, 2008.

[4] D. Hadfield-Menell, S. J. Russell, P. Abbeel, and A. Dragan, "Cooperative inverse reinforcement learning," *Advances in neural information processing systems*, vol. 29, 2016.

[5] M. Ramırez and H. Geffner, "Plan recognition as planning," in *Proceedings of the 21st International Joint Conference on Artifical Intelligence.*, pp. 1778–1783, Citeseer, 2009.

[6] M. Ramırez and H. Geffner, "Probabilistic plan recognition using off-the-shelf classical planners," in *Proceedings of the Conference of the Association for the Advancement of Artificial Intelligence (AAAI 2010)*, pp. 1121–1126, Citeseer, 2010.

[7] G. Kaminka, M. Vered, and N. Agmon, "Plan recognition in continuous domains," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, Apr. 2018.

[8] C. L. Baker, R. Saxe, and J. B. Tenenbaum, "Action understanding as inverse planning," *Cognition*, vol. 113, no. 3, pp. 329–349, 2009.

[9] T. Zhi-Xuan, J. Mann, T. Silver, J. Tenenbaum, and V. Mansinghka, "Online Bayesian goal inference for boundedly rational planning agents," *Advances in Neural Information Processing Systems*, vol. 33, pp. 19238–19250, 2020.

[10] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. Srinivasa, "Planning-based prediction for pedestrians," in *ICRIS*, pp. 3931–3936, IEEE, 2009.

[11] A. D. Dragan, *Legible robot motion planning*. PhD thesis, Carnegie Mellon University, 2015.

[12] M. F. Cusumano-Towner, A. Radul, D. Wingate, and V. K. Mansinghka, "Probabilistic programs for inferring the goals of autonomous agents," *arXiv preprint arXiv:1704.04977*, 2017.

[13] I. R. Seaman, J.-W. van de Meent, and D. Wingate, "Nested reasoning about autonomous agents using probabilistic programs," *arXiv preprint arXiv:1812.01569*, 2018.

[14] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa, "Chomp: Gradient optimization techniques for efficient motion planning," in *2009 IEEE International Conference on Robotics and Automation*, pp. 489–494, IEEE, 2009.

[15] J. Schulman, J. Ho, A. X. Lee, I. Awwal, H. Bradlow, and P. Abbeel, "Finding locally optimal, collision-free trajectories with sequential convex optimization," *Robotics: Science and Systems IX*, 2013.

[16] D. Hadfield-Menell, C. Lin, R. Chitnis, S. Russell, and P. Abbeel, "Sequential quadratic programming for task plan optimization," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5040–5047, IEEE, 2016.

[17] G. Gergely and G. Csibra, "Teleological reasoning in infancy: The naıve theory of rational action," *Trends in cognitive sciences*, vol. 7, no. 7, pp. 287–292, 2003.

[18] M. Botvinick and M. Toussaint, "Planning as inference," *Trends in cognitive sciences*, vol. 16, no. 10, pp. 485–488, 2012.

[19] S. Levine, "Reinforcement learning and control as probabilistic inference: Tutorial and review," *arXiv preprint arXiv:1805.00909*, 2018.

[20] A. Piché, V. Thomas, C. Ibrahim, Y. Bengio, and C. Pal, "Probabilistic planning with sequential monte carlo methods," in *International Conference on Learning Representations*, 2019.

[21] P. J. Rossky, J. D. Doll, and H. L. Friedman, "Brownian dynamics as smart monte carlo simulation," *The Journal of Chemical Physics*, vol. 69, no. 10, pp. 4628–4633, 1978.

[22] N. S. Arora, N. K. Tehrani, K. D. Shah, M. Tingley, Y. L. Li, N. Torabi, D. Noursi, S. A. Masouleh, E. Lippert, and E. Meijer, "Newtonian Monte Carlo: A second-order gradient method for speeding up MCMC," in *AABI*, 2019.

[23] A. K. Lew, M. Cusumano-Towner, and V. K. Mansinghka, "Recursive monte carlo and variational inference with auxiliary variables," in *Uncertainty in Artificial Intelligence*, pp. 1096–1106, PMLR, 2022.

[24] N. Chopin, "A sequential particle filter method for static models," *Biometrika*, vol. 89, no. 3, pp. 539–552, 2002.

[25] G. W. Brier *et al.*, "Verification of forecasts expressed in terms of probability," *Monthly Weather Review*, vol. 78, no. 1, pp. 1–3, 1950.

[26] A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, and O. Khatib, "Progress and prospects of the human–robot collaboration," *Autonomous Robots*, vol. 42, pp. 957–975, 2018.

[27] P. Schydlo, M. Rakovic, L. Jamone, and J. Santos-Victor, "Anticipation in human-robot cooperation: A recurrent neural network approach for multiple action sequences prediction," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5909–5914, IEEE, 2018.

[28] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 961–971, 2016.

[29] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4674–4683, 2017.

[30] L. P. Kaelbling and T. Lozano-Pérez, "Integrated task and motion planning in belief space," *The International Journal of Robotics Research*, vol. 32, no. 9-10, pp. 1194–1227, 2013.

[31] D. Hadfield-Menell, E. Groshev, R. Chitnis, and P. Abbeel, "Modular task and motion planning in belief space," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4991–4998, IEEE, 2015.