Maximum Likelihood Constraint Inference from Stochastic Demonstrations

David L. McPherson, Kaylene C. Stocking, S. Shankar Sastry

***Abstract*— When an expert operates a safety-critical dynamic system, constraint information is tacitly contained in their demonstrated trajectories and controls. These constraints can be inferred by modeling the system and operator as a constrained Markov Decision Process and finding which constraint is most likely to generate the demonstrated controls. Prior constraint inference work has focused mainly on deterministic dynamics. Stochastic dynamics, however, can capture the uncertainty inherent to real applications and the risk tolerance that requires.**

This paper extends maximum likelihood constraint inference to stochastic applications by using maximum causal entropy likelihoods. Furthermore, this extension does not come at increased computational cost, as we derive an algorithm that computes constraint likelihood and risk tolerance in a unified Bellman backup, thereby keeping the same computational complexity.

1. Introduction

Optimization-based control (notably, model predictive control) promises autonomous behavior [3] even in nonlinear [13] or stochastic dynamics [20][4]. It has already impacted industrial practice [15] as "model-predictive control", and its recent incarnation as "reinforcement learning" [13][18] pushes the paradigm further by leveraging large datasets and computing clusters.

Yet these optimizations only work if the clients' goals can be encoded as reward functions and their concerns encoded as safety constraint sets. One approach to this translation is to first solve the inverse of optimal control: given near-optimal demonstrations from the client, recover the reward function whose optimum would match the demonstrator's performance [8]. After fitting the task specification in this way, the objective can then be optimized to imitate the expert behavior [1] or used to predict human motion [23].

Often, inverse optimal control focuses on inferring the magnitude of the reward function. But as optimal control increasingly emphasizes working within constraints, inverse optimal control is interested in identifying those constraints [2][7][10][14][17]. Chou et al. [6] inferred constraints along the paths that would be low cost but were never observed. This intuition was grounded into a probabilistic (Bayesian) framework by Scobee and Sastry [17] by translating maximum entropy inverse reinforcement learning [22] to work for hard constraints. Unfortunately, the maximum entropy used in [17] only works for deterministic systems.

Non-deterministic models capture the uncertain dynamics inherent in applications. That uncertainty is especially important to consider when designing for robust safety constraint satisfaction. Stochasticity can model a variety of unpredictable dynamics in applications: from unpredictable power sources in renewable power systems [9] to hard-to-model turbulence in road conditions [20], from tumor cell growth in cancer treatment [16] to unforeseen changes in stormwater reservoirs [5].

The maximum entropy likelihoods can be extended to uncertain transition dynamics by conditioning the entropy at each time step only on the previously revealed state transitions [21]. This maximum causal entropy has been extended from running state-based rewards to learn signal temporal logic specifications [19]. Focusing this to just inclusion-for-all-time specifications that make up safety constraints allows for simpler algorithms as Scobee and Sastry [17] did for deterministic systems. This paper similarly focuses on constraints, paralleling [17], but can model stochasticity by factoring in the causality of dynamics as in [21].

A. Contributions and Guide

This work advances prior art [17] in inferring state-action constraints:

- by respecting causality using the principle of maximum **causal** entropy for likelihood generative models
- by extending the hypothesis family to include risk-tolerating **chance** constraints
- and by streamlining the algorithm into one backwards pass, thereby maintaining the same computational complexity as the non-stochastic version [17]

2. Background

Fitting models entails choosing the model out of some hypothesis class that is "best" along some metric. A natural metric is how likely the model would be to generate the true, observed demonstration data $\hat{x}_i$ for $i \in [0, 1, \cdots, T]$. Formally, assuming the space of possible models (called the hypothesis family) is indexed by some vector of parameters θ, the best model is the one with the highest probability of observing the dataset:

$$\theta = \arg\max_{\theta \in \Theta} P_\theta(X_0 = \hat{x}_0, X_1 = \hat{x}_1, \cdots X_N = \hat{x}_T) \tag{1}$$

Which is the maximum likelihood estimate of the parameter θ of the probability distribution P_θ.

This probability distribution $P_\theta(X_0, X_1, \cdots X_T)$ can factor into simpler terms when the X_i are states sampled over time from a causal dynamical system. If the state X_i contains all the evolving information, then the Markov property means that datapoints only depend on the past through the most recent preceding state. In particular:

$$P_\theta(X_i|X_{i-1}, X_{i-2}, \cdots, X_0) = P_\theta(X_i|X_{i-1}) \tag{2}$$

When these probabilistic dynamics over state are controlled by some exogenous input a to optimize some cost $R(x, a)$, these Markov dynamics become a Markov Decision Problem (MDP). In this work, we focus on discrete time and discrete state and action spaces, so the MDP can be written as a 4-tuple:

- state space $\mathcal{X} = \{x^0, x^1, \cdots, x^{N_X}\}$,
- set of actions $\mathcal{A} = \{a^0, a^1, \cdots, a^{N_A}\}$,
- transition probability function

  $$P(X_{t+1} = x_{t+1}|X_t = x_t, a_t) = S(x_t, a_t, x_{t+1})$$

  where $S : \mathcal{X} \times \mathcal{A} \times \mathcal{X} \to [0, 1]$,
- and objective metric $R(x_{[0:T]}, a_{[0:T-1]}) : \Xi \to \mathbb{R}$ where $\Xi = \mathcal{X}^T \times \mathcal{A}^{T-1}$ is the space of trajectories $\xi = (a_{[0:T-1]}, x_{[0:T]})$. This work assumes the reward to have a form that is decomposable over timesteps:

  $$R(x_{[0:T]}, a_{[0:T-1]}) = w(x_T) + \sum_{t=0}^{T-1} r(x_t, a_t) \tag{3}$$

  where $w(x_T)$ is the final reward and $r(x, a)$ is the running reward.

This probabilistic model can statistically fit to trajectory datasets; either tuning $S_\theta(x_t, a_t)$ to model fit to the dynamics or tuning $R_\theta(\xi)$ to model the expert demonstrator themselves. This latter modeling is the inverse optimal control approach to the imitation learning problem: to replicate expert performance given a set of their demonstrated trajectories $\mathcal{D} = \{\xi^1, \xi^2, \cdots, \xi^M\}$.

Continuing with using the maximum likelihood framework to estimate parameters θ, all that is needed is a likelihood that a particular $R_\theta(x, a)$ will generate $a_{[0:T-1]}$ and thereby $x_{[1:T]}$. Ziebart *et al* [22] introduced a random distribution on a designed to be robust to possible reward phenomena outside the necessarily limited hypothesis class. Specifically, they assume that the parameter to be estimated θ will multiply candidate feature functions $\phi(x, a)$ that will form the spanning basis functions of the hypothesis class. Mathematically:

$$r_\theta(x, a) = \theta^T \phi(x, a) \tag{4}$$

The estimation, then, is able to choose the best distribution along the $\phi(x, a)$ basis set, but is incapable of describing distributions outside of this linear subspace of function space. With the goal of remaining maximally agnostic to (and therefore robust to) this non-capturable space, Ziebart [21] deploys the distribution that maximizes entropy outside the candidate $\phi(x, a)$:

$$P_\theta(x_{[0:T]}) = \frac{e^{w(x_T)+\sum_{t=0}^{T-1} \theta^T \phi(x_t, a_t)}}{Z_\theta} \tag{5}$$

where Z_θ is a normalizing constant.

This exponential family distribution makes trajectories that are equally rewarding equally likely and more optimal trajectories exponentially more likely. This likelihood can be used to estimate what rewards $r(x, a)$ would generate the demonstrated behavior [23] which can in turn be used to replicate that expert performance [22]. This works well for modeling preferences and ideas of optimality, but is insufficient to learn the expert's safety rules that must be satisfied to avoid dangers (e.g. how bicyclists avoid potholes or drivers avoid black ice). This question of finding the feasible *domain* of the reward function $R(x, a)$ is addressed by Scobee and Sastry [17] as an optimization with the pre-identified reward magnitudes as an input. The task then becomes to find the constraint set C that will form the support of the maximum entropy distribution as:

$$P_{C,\theta}(x_{[0:T]}) = \begin{cases} \frac{e^{w(x_T)+\sum_{t=0}^{T-1} \theta^T \phi(x_t, a_t)}}{Z_C} & , \text{ if } \xi \in C \\ 0 & , \text{ if } \xi \notin C \end{cases} \tag{6}$$

where $\xi \in C$ means that all of trajectory ξ's states $x_{0:T}$ and actions $a_{0:T-1}$ are safe.

Similarly to how we decomposed the reward function over time into a sum of $r(x_t, a_t)$, we focus on the class of constraints that rule on individual timepoints' states $x_t \in C_X$ or actions $a_t \in C_A$ for all time. To model some risk-tolerance, we allow some small probability $\psi(x)$ of transitioning to an $x \in C_X$:

$$P(X_{t+1} = \bar{x}|X_t = x_t, a_t) \leq \psi(\bar{x}), \quad \forall \bar{x} \notin C_X \tag{7}$$

To deterministically constrain out a state x set $\psi(x) = 0$. On the other hand, setting $\psi(x) = 1$ means the constraint is inactive and transitioning to x is freely allowed. Therefore the set of state constraints C_X can be encoded as a $\psi(x)$ over all states $x \in \mathcal{X}$. Then the indicator that taking an action a from state x is safe is:

$$\begin{aligned} &\Phi_C(a, x) = \\ &= \mathbb{I}[a \in C_A \\ &\quad \cap (P(X_{t+1} = \bar{x}|X_t = x, a) \leq \psi(\bar{x}) \forall \bar{x} \notin C_X)] \end{aligned} \tag{8}$$

Let the set of all such safety constraints be $\mathcal{C}$.

Scobee and Sastry's [17] central insight to identify these constraints is that narrowing the support C after fixing θ will uniformly scale the distribution for all likelihoods still within the support C. As long as the demonstrations stay inside this C, tightening the constrained safe regions will increase the likelihood of observing those demonstrations. To avoid overfitting to ruling out all non-visited states and unused actions, states and actions would be inferred as unsafe one-by-one until improvement rate tapered off. The best such x^i or a^j to cut out would be the one that produces the best scaling factor corresponding to the new normalizing constant Z_{C^+}

where C^+ denotes the constraint set C after the candidate x^i or a^j is ruled out.

$$Z_{C^+} = \sum_{\xi \in C^+} e^{w(x_T) + \sum_{t=0}^{T-1} \theta^T \phi(x_t, a_t)} \tag{9}$$

Which can be tractably computed for all candidate constraint sets $C^+ \in \mathcal{C}$ by forward simulation of the dynamics (similar to Ziebart's backward-forward algorithm in [22]), since this quantity Z_{C^+} is directly proportional to the summed probability of all trajectories satisfying C^+:

$$P_{C^0,\theta}(\xi \in C^+) = \\ = \frac{\sum_{\xi \in C^+} e^{w(x_T) + \sum_{t=0}^{T-1} \theta^T \phi(x_t, a_t)}}{Z_{C^0,\theta}} \tag{10}$$

where $C^0 \in \mathcal{C}$ is the baseline constraint set with no additional x^i or a^j cut out, and so $Z_{C^0,\theta}$ is a constant across all candidates' forward simulations. This means that the $P_{C^0,\theta}(\xi \in C^+)$ will form an equivalent ranking on C^+ as Z_{C^+} would, and can be used interchangeably to identify the likelihood maximizing C^+.

Unfortunately the distribution in Equation 5 (and thereby the constrained distribution in Equation 6 derived from it) only work for non-stochastic dynamics as shown in [21]: the distribution does not factor in how the probabilistic dynamics reveal transition information over time and thereby makes the agents' selection of x non-causal. That paper solved the problem (for θ inference) by incorporating the dynamics' information structure via a recursive definition between the actions and succeeding states:

$$P_\theta(a_t|x_t) = \frac{e^{Q^{soft}_{\theta,t}(a_t,x_t)}}{e^{V^{soft}_{\theta,t}(x_t)}} \tag{11}$$

$$Q^{soft}_{\theta,t}(a_t, x_t) = r(x_t, a_t) + \mathbb{E}_{X_{t+1}} V^{soft}_{\theta,t+1}(x_{t+1}) \tag{12}$$

$$V^{soft}_{\theta,t}(x_t) = \log \sum_{a_t} e^{Q^{soft}_{\theta,t}(a_t,x_t)} \\ = \underset{a_t}{\text{softmax}}\, Q^{soft}_{\theta,t}(a_t, x_t) \tag{13}$$

where Q^{soft} can be interpreted as a state-action soft-optimal value-to-go and V^{soft} the state's soft-optimal value-to-go.

The present work will apply this improved causal distribution from [21] to the constraint inference approach designed in [17] in order to apply constraint inference to stochastic demonstrations. Towards this end, constraints can be added to Equations 11, 12, 13 as:

$$P_C(a_t|x_t) = \frac{e^{Q^{soft}_{C,t}(a_t,x_t)}}{e^{V^{soft}_{C,t}(x_t)}} \Phi_C(a_t, x_t) \tag{14}$$

$$Q^{soft}_{C,t}(a_t, x_t) = r(x_t, a_t) + \mathbb{E}_{X_{t+1}} V^{soft}_{C,t+1}(x_{t+1}) \tag{15}$$

$$V^{soft}_{C,t}(x_t) = \log \sum_{a_t} \Phi_C(a_t, x_t) e^{Q^{soft}_{C,t}(a_t,x_t)} \\ = \underset{a_t}{\text{softmax}}\, \Phi_C(a_t, x_t) Q^{soft}_{C,t}(a_t, x_t) \tag{16}$$

3. Derivation of Constraint Statistics for Causal Stochastic Demonstrations

The first step to applying the Scobee and Sastry's [17] key insight to the causal maximum entropy distribution defined in Equations 14-16 is writing the distribution joint over all timesteps in a horizon $[t : T]$ (with any starting $t \in [0, 1, 2, \cdots, T]$) as:

$$P_C(A_{[t:T]} = a_{[t:T]} | X_t = x_t) \tag{17}$$

$$= \begin{cases} \frac{e^{\mathbb{E}\left[R(X_{[t:T]}, a_{[t:T]})\right]}}{e^{V^{soft}_{C,t}(x_t)}} & , \text{ if } a_{[t:T]} \in W_C^{[t:T]} \\ 0 & , \text{ if } a_{[t:T]} \notin W_C^{[t:T]} \end{cases} \tag{18}$$

where $W_C^{[t:T]}$ is the set of feedback-controller sequences that satisfy the condition in Equation 8 for all times $\tau \in [t : T]$. This distribution over *controls*, rather than *states* as in the non-causal Equation 6, means that the dynamics' probability distribution $S(x_t, a_t, x_{t+1})$ is truly incorporated.

With this causal correction in place, the insight from Scobee and Sastry [17] applied to Equation 6 can be applied to Equation 18. Changing the constraint set C only changes the normalizing constant $Z_{C,t}$:

$$Z_{C,t} = e^{V^{soft}_{C,t}(x_t)} \tag{19}$$

Therefore incrementing from a constraint set C^0 to any tighter constraint set C^+, as long as this C^+ still includes the demonstrations, will strictly increase the likelihood of the observed demonstrations. To clarify the connection to the quantity in Equation 10 that was tracked in [17], note that:

$$P_{C^0,\theta}(\xi \in C^+) = \frac{Z_{C^+,t}}{Z_{C^0,t}} = \frac{e^{V^{soft}_{C^+,t}(x_t)}}{e^{V^{soft}_{C^0,t}(x_t)}} \tag{20}$$

which is the ratio for converting between C^0's normalizing constant and C^+'s normalizing constant. For brevity, we will denote this probability by $F_{C^+,t}(x_t)$. Most crucially, the probability of a trajectory starting at x_t and staying inside C^+ and C^0 scales by $1/F_{C^+,t}(x_t)$. Therefore this F forms a ranking on possible tightened constraint sets C^+; whichever has the smaller $F_{C^+,t}(x_t)$ will have larger likelihoods.

This fact will be used to infer C_X and C_A, but first note how we can infer $\psi(x)$. Inspired by the chance level specifications used in Vazquez-Chanlatte et al. [19], we observe the following:

Lemma 3.1. *When considering constraining out a single state x into C_X, it will maximize the likelihood to choose the lowest possible $\psi(x)$ that doesn't rule out any demonstrations.*

Proof: Consider two candidate constraints C^+ and C^+_- that differ only by C^+_- having exactly one $\psi(x)$ lower than C^+ has. C^+_- will always have $F_{C^+_-,0}(x_0) \leq F_{C^+,0}(x_0)$. Since a smaller $F_{C^+_-,0}(x_0)$ means that C^+_- will have a larger likelihood of observing the demonstrated trajectories if and

only if it doesn't rule out those trajectories as infeasible, the smallest possible $\psi(x)$ will be the maximum likelihood estimator.

The ratio $F_{C^+,t}(x_t)$ can be computed by modifying the soft Bellman backup defined in Equations (14) - (16). This modified backup procedure is described in the theorem below:

Theorem 3.2. *Let C^0 be a set of constraints and C^+ be an augmented version of C^0 with more states constrained. Then $F_{C^+,t}(x_t)$ can be computed as:*

$$F_{C^+,t}(x_t) = \mathbb{E}_{a_t \sim P_{C^0}}\left[\Phi_{C^+}(a_t, x_t) e^{\mathbb{E}_{x_{t+1}} \log\left(F_{C^+,t+1}(x_{t+1})\right)}\right]$$

Proof: It will be helpful to notate the set of legal actions from x under constraint set C as $A_C(x) = \{a \mid \Phi_C(a, x) = 1\}$

$$\begin{aligned} F_{C^+,t}(x_t) &= \frac{e^{V^{soft}_{C^+,t}(x_t)}}{e^{V^{soft}_{C^0,t}(x_t)}} \\ &= \frac{\sum_{a_t \in A_{C^+}(x_t)} e^{Q^{soft}_{C^+,t}(a_t,x_t)}}{e^{V^{soft}_{C^0,t}(x_t)}} \\ &= \sum_{a_t \in A_{C^+}(x_t)} \frac{e^{r(x_t,a_t)+\mathbb{E}_{x_{t+1}} V^{soft}_{C^+,t+1}(x_{t+1})}}{e^{V^{soft}_{C^0,t}(x_t)}} \end{aligned}$$

It will be convenient to define the logarithm of our $F_{C,t}$. Let it be Δ^t_C:

$$\begin{aligned} \Delta^{t+1}_{C^+}(x_{t+1}) &= \log\left(F_{C^+,t+1}(x_{t+1})\right) \\ &= \log\left(\frac{e^{V^{soft}_{C^+,t+1}(x_{t+1})}}{e^{V^{soft}_{C^0,t+1}(x_{t+1})}}\right) \\ &= V^{soft}_{C^+,t+1}(x_{t+1}) - V^{soft}_{C^0,t+1}(x_{t+1}) \end{aligned}$$

Then the ratio can be redefined in terms of previously calculated terms on C^0 and our iterating $F_{C,t}$

$$\begin{aligned} F_{C^+,t}(x_t) &= \sum_{a_t \in A_{C^+}(x_t)} \frac{e^{r(x_t,a_t)+\mathbb{E}_{x_{t+1}} V^{soft}_{C^0,t+1}(x_{t+1})}}{e^{V^{soft}_{C^0,t}(x_t)}} \\ &\qquad \cdot e^{\mathbb{E}_{x_{t+1}} \Delta^{t+1}_{C^+}(x_{t+1})} \\ &= \frac{\sum_{a_t \in A_{C^+}(x_t)} e^{Q_{C^0}(x_t,a_t)+\mathbb{E}_{x_{t+1}} \Delta^{t+1}_{C^+}(x_{t+1})}}{e^{V^{soft}_{C^0,t}(x_t)}} \\ &= \sum_{a_t \in A_{C^+}(x_t)} \frac{e^{Q_{C^0}(x_t,a_t)}}{e^{V^{soft}_{C^0,t}(x_t)}} e^{\mathbb{E}_{x_{t+1}} \Delta^{t+1}_{C^+}(x_{t+1})} \\ &= \sum_{a_t \in A_{C^+}(x_t)} P_{C^0}(a_t|x_t) e^{\mathbb{E}_{x_{t+1}} \Delta^{t+1}_{C^+}(x_{t+1})} \\ &= \mathbb{E}_{a_t \sim P_{C^0}} \Phi_{C^+}(a_t, x_t) e^{\mathbb{E}_{x_{t+1}} \Delta^{t+1}_{C^+}(x_{t+1})} \end{aligned} \tag{21}$$

□

4. Algorithm

Theorem 3.2 implies that an algorithm can compute the conversion ratios $F_C(x)$ (which will correspond to how much the distribution shrunk by) for all candidate constraints at the same time as the Bellman backup for the baseline set of constraints C^0. The Greedy Iterative Constraint Inference procedure pioneered in [17] suggests this selection can be performed iteratively adding just one constraint at a time. This iterative approach can be shown to be bounded sub-optimal compared to selecting all the constraints simultaneously [17]. In this iterative approach, the F_{C^+} optimizing C^+ will become the baseline set of constraints for the next iteration C^i.

A. Determining Chance Constraint Risk Levels from Demonstrated Transitions

Lemma 3.1 shows that this set of candidates can be further reduced to only those whose newly added $\psi(x)$ are as exclusive as possible without excluding any of the demonstrations $(\tilde{x}_{0:T}, \tilde{a}_{0:T}) \in \mathcal{D}$. That is, when adding state constraints, the newly added exclusion threshold $\psi(x)$ must be as low as possible while still being greater than all transition probabilities to x that were chosen by the expert in their demonstrations. For simplicity, we will lowerbound $\psi(x)$ to prevent any precursor states of x from having all its available actions ruled out thereby dooming any trajectory entering that precursor state to necessarily violate the chance constraint on x. Therefore this lowerbound $\underline{\psi}(x)$ must be defined:

$$\underline{\psi}(x') = \max_{\{x \mid \exists \hat{a} \ni P(x'|x,\hat{a})>0\}} \min_a P(x'|x,a) \tag{22}$$

This implies that the new $\psi(x)$ should be:

$$\begin{aligned} \psi(x) = \max\{&\underline{\psi}(x), \\ &\max_{\xi \in \mathcal{D}} \max_{t \in [0:T-1]} S(x_t, a_t, x)\} \end{aligned} \tag{23}$$

B. Comparing States and Actions Satisfaction Frequencies

Let $\mathcal{C}^+_i \subset \mathcal{C}$ be the subset of constraint sets that restrict only one more action or state than the nominal constraint set C_i. The most likely constraint $C \in \mathcal{C}^+_i$ is whichever still allows the observed demonstrations while having the smallest satisfaction frequency $F_{C^+,0}(x_0)$ from the starting state. This quantity can be computed via our proposed algorithm as described in Algorithm 1's pseudocode.

Note that Algorithm 1 has computations on the order of $O(|\mathcal{X}|^2(|\mathcal{X}|+|\mathcal{A}|))$, identical to the computational complexity of prior art in maximum likelihood constraint inference [17].

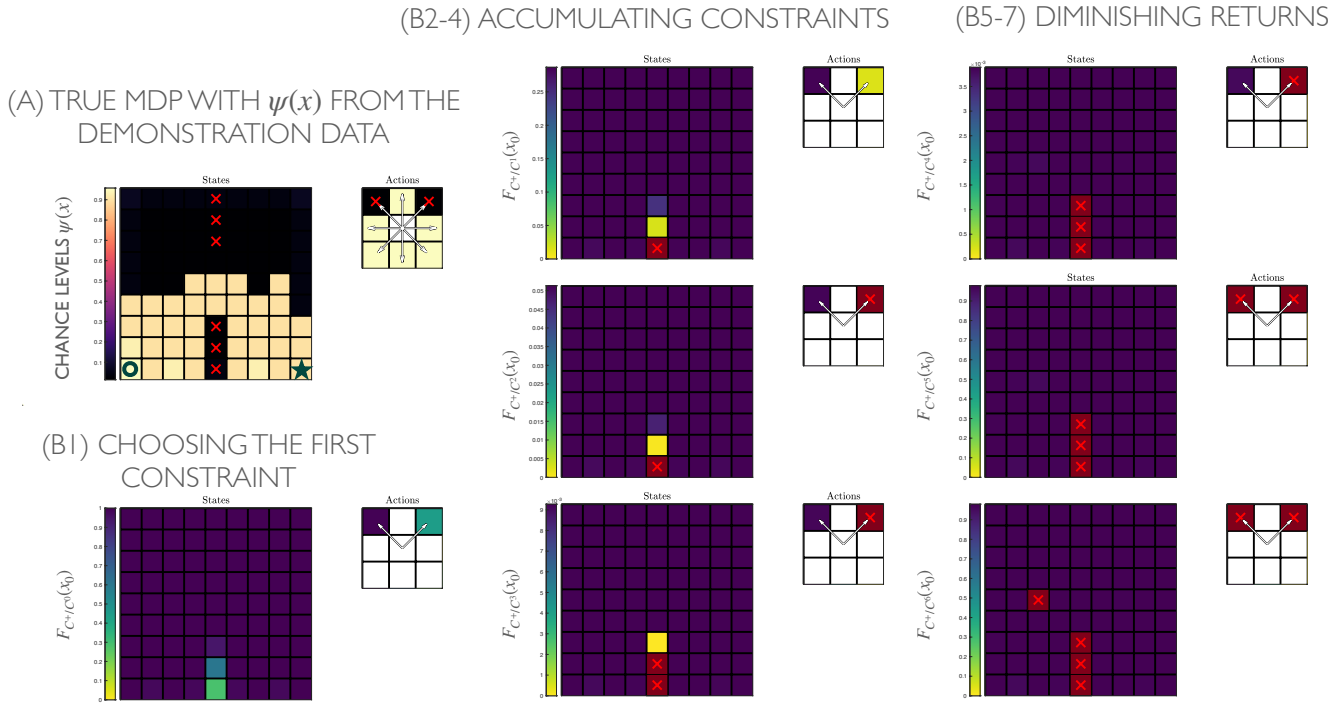

Fig. 1. The constraint inference algorithm was evaluated on a gridworld synthetic dataset with stochastic dynamics. The synthetic demonstrator optimized the task of finding the shortest path from the origin (O) in the bottom left to the starred goal in the bottom right. In the panels, the states and actions that are constrained out are marked with a red X. (A) The demonstration dataset is generated with the constraints shown in Panel A. The dark squares are states and actions the demonstrator never took. Conversely, the light state and actions squares were those chosen by the demonstrator. The relative shading on the states corresponds to the largest chance of transition to that state that was demonstrated in $\mathcal{D}$. That transition chance is as low as the risk threshold $\psi(x)$ on that state can be set for a constraint without rejecting the demonstrations as infeasible and making their likelihood 0. Therefore, inferred constraints will have $\psi(x)$ *equal* to that largest demonstrated transition chance. (B) The sequence of inferred constraints alongside the value of adding other candidate constraints in addition. Cell shading corresponds to how much further partition mass would be eliminated by introducing a constraint on that action or chance constraint on that state. This F_{C^+/C^i} value equals the probability that the demonstrator would happen to avoid that state even without a constraint (following C^i which doesn't count that state as dangerous like C^+ would). Only states that would be unlikely to be completely avoided the way they were in the demonstrations are likely to have constraints on them. (B1) The F values of choosing the first constraint over the empty constraint set C^0. Note especially the bottom middle square that only has a 29% chance of being avoided like the demonstrator did. In other words, it has a probability of $1 - 0.29 = 0.71$ of being transitioned to with a transition chance $S(x_-, a, x)$ higher than those demonstrated chances $\psi(x)$. The likely explanation for why the demonstrator avoided this straightshot state is that there was a constraint there. (B2-4) Adding the next three constraints continues to scale up the likelihood of observing $\mathcal{D}$ and infers true constraints of the groundtruth demonstrator (B5-7) After the fourth constraint gets inferred, the continued scaling shrinks by an order of magnitude and then effectively halts as F retracts to $F = 0.96$. This corresponds to inferring an untrue constraint. The final three true constraints can not be inferred since the demonstrator never traversed that unrewarding half of the grid.

5. Results

Algorithm 1 was implemented in MATLAB and tested on a synthetic dataset of $M = 100$ demonstrations. This dataset was synthesized from simulated trajectories of a stochastically optimal agent minimizing distance traveled on a two-dimensional "Gridworld" MDP with movement in all eight compass directions. These eight directions made up the action space $\mathcal{A}$ along with a loitering terminal action for once the goal was reached. Each directional action was given a fixed "slippage" chance of 0.1 where a random direction out of the other seven was followed instead. All ground-truth and candidate state constraints were fixed at a constant chance threshold of $\psi = 0.25$ for all states.

The simulated demonstrator only noisily optimized the task, following a Boltzmann choice distribution as described in Equation (18). The constraint inference algorithm was evaluated on this dataset as shown in Figure 1. By the fifth iteration (shown in Figure 1.B6), the algorithm suceeded in recovering the groundtruth constraints (see Figure 1.A).

6. Limitations and Future Work

The algorithms set forth in this paper focused on discretized state and action spaces. For controlling many systems on practical timescales, the state must be handled as a continuous parameter. Future work should investigate how gridded state spaces like in Figure 1 could be refined to approximate continuous state spaces. Reducing the algorithm to a variant Bellman backup, as we did in Theorem 3.2, suggests that the continuous variant may just be solving a Hamilton-Jacobi-Bellman equation. These partial differential equations have a rich literature investigating their solution, including toolsets like [12].

Extending constraint inference to stochastic systems raises questions of whether human experts might be better modeled using a prospect-theoretic or risk-sensitive measure as in [11]. Future work should investigate how human heuristics for statistical prediction might impact the way demonstra-

Algorithm 1: Modified Bellman Backup with Value Ratio

Data: Final reward $w(x)$ and running reward $r(x,a)$, Dynamics $S(x,a,x')$, Vector of indicators of constraint satisfaction Φ_C for nominal constraint set C^0 and all candidate constraints $C^+ \in \mathcal{C}_i^+$.

Result: $V_{C^i,t}$ and a column vector F where each entry corresponds to the $F_{C^+,0}$ for $C^+ \in \mathcal{C}_i^+$

for $x \in \mathcal{X}$ **do**
  $Z(T,x) \leftarrow \exp(w(x))$
  $F(T,x) \leftarrow 1$
end
for $t \in [T-1, 0]$ **do**
  for $x \in \mathcal{X}$ **do**
    $Z(t,x) \leftarrow 0$
    $F(t,x) \leftarrow 0$
    for $a \in \mathcal{A}$ **do**
      $Q(t,x,a) \leftarrow r(x,a)$
      $D(t,x,a) \leftarrow 0$
      for $x' \in \mathcal{X}$ **do**
        $Q(t,x,a) += S(x,a,x') \log(Z(t+1,x'))$
        $D(t,x,a) += S(x,a,x') \log(F(t+1,x'))$
      end
      $Z(t,x) += \Phi_{C^i}(x,a) \exp(Q(t,x,a))$
      $F(t,x) += \Phi_{C \in \mathcal{C}_\gamma^+}(x,a)\ \exp(Q(t,x,a))\ \exp(D(t,x,a))$
    end
    $F(t,x) = F(t,x)/Z(t,x)$
  end
end

tions are generated. The algorithm should be designed to be robust to these biases or even leverage their structure.

7. Discussion and Conclusion

By designing the likelihoods to maximize the causal entropy (that respects the information flow of state transition outcome revelation) this work makes maximum likelihood estimation possible for stochastic dynamics that reflect the uncertainties inherent in perilous situations. Moreover, by broadening the hypothesis class to include chance constraints our algorithm not only learns the constraints from expert operators, but also their risk tolerances. This opens the door to studying how expert operators plan risk-sensitively and what prospect-theoretic risk measures they may be employing.

Although increasing the complexity of systems that can be handled in constraint inference, this algorithm maintains the same computational complexity of $O(|\mathcal{X}|^2(|\mathcal{X}| + |\mathcal{A}|))$ as prior art. That is, control engineers can extract safety specifications from expert demonstration data for the same cost in both stochastic and deterministic dynamics.

References

[1] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 1, 2004.

[2] Leopoldo Armesto, Jorren Bosga, Vladimir Ivan, and Sethu Vijayakumar. Efficient learning of constraints and generic null space policies. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1520–1526. IEEE, 2017.

[3] Anil Aswani, Humberto Gonzalez, S Shankar Sastry, and Claire Tomlin. Provably safe and robust learning-based model predictive control. *Automatica*, 49(5):1216–1226, 2013.

[4] Bart van den Broek, Wim Wiegerinck, and Hilbert Kappen. Risk sensitive path integral control. *arXiv preprint arXiv:1203.3523*, 2012.

[5] Margaret P Chapman et al. A risk-sensitive finite-time reachability approach for safety of stochastic dynamic systems. In *2019 American Control Conference (ACC)*, pages 2958–2963. IEEE, 2019.

[6] Glen Chou, Dmitry Berenson, and Necmiye Ozay. Learning constraints from demonstrations. In *International Workshop on the Algorithmic Foundations of Robotics*, pages 228–245. Springer, 2018.

[7] Glen Chou, Necmiye Ozay, and Dmitry Berenson. Learning parametric constraints in high dimensions from demonstrations. In *Conference on Robot Learning*, pages 1211–1230. PMLR, 2020.

[8] Rudolf Emil Kalman. When is a linear control system optimal? *Journal of Basic Engineering*, 86(1):51–60, 1964.

[9] Mohammad E Khodayar, Mohammad Shahidehpour, and Lei Wu. Enhancing the dispatchability of variable wind generation by coordination with pumped-storage hydro units in stochastic power systems. *IEEE Transactions on Power Systems*, 28(3):2808–2818, 2013.

[10] Changshuo Li and Dmitry Berenson. Learning object orientation constraints and guiding constraints for narrow passages from one demonstration. In *International symposium on experimental robotics*, pages 197–210. Springer, 2016.

[11] Eric Mazumdar, Lillian J Ratliff, Tanner Fiez, and S Shankar Sastry. Gradient-based inverse risk-sensitive reinforcement learning. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 5796–5801. IEEE, 2017.

[12] Ian M Mitchell. A toolbox of level set methods. *UBC Department of Computer Science Technical Report TR-2007-11*, 2007.

[13] Andrew Y Ng, H Jin Kim, Michael I Jordan, Shankar Sastry, and Shiv Ballianda. Autonomous helicopter flight via reinforcement learning. In *NIPS*, volume 16. Citeseer, 2003.

[14] Claudia Pérez-D'Arpino and Julie A Shah. C-learn: Learning geometric constraints from demonstrations for multi-step manipulation in shared autonomy. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4058–4065. IEEE, 2017.

[15] S Joe Qin and Thomas A Badgwell. A survey of industrial model predictive control technology. *Control engineering practice*, 11(7):733–764, 2003.

[16] Tyler Rispom et al. Differentiation-state plasticity is a targetable resistance mechanism in basal-like breast cancer. *Nature communications*, 9(1):1–17, 2018.

[17] Dexter RR Scobee and S Shankar Sastry. Maximum likelihood constraint inference for inverse reinforcement learning. *arXiv preprint arXiv:1909.05477*, 2019.

[18] Niko Sünderhauf et al. The limits and potentials of deep learning for robotics. *The International Journal of Robotics Research*, 37(4-5):405–420, 2018.

[19] Marcell Vazquez-Chanlatte, Susmit Jha, Ashish Tiwari, Mark K Ho, and Sanjit Seshia. Learning task specifications from demonstrations. In *Advances in Neural Information Processing Systems*, pages 5367–5377, 2018.

[20] Grady Williams, Paul Drews, Brian Goldfain, James M Rehg, and Evangelos A Theodorou. Aggressive driving with model predictive path integral control. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1433–1440. IEEE, 2016.

[21] Brian D Ziebart. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. CMU, 2010.

[22] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

[23] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Human behavior modeling with maximum entropy inverse optimal control. In *AAAI Spring Symposium: Human Behavior Modeling*, volume 92, 2009.