Inverse Reinforcement Learning with Multiple Planning Horizons

Jiayu Yao
Gladstone Institutes
San Francisco, CA 94158
jiayu.yao@gladstone.ucsf.edu

Finale Doshi-Velez
SEAS, Harvard University
Allston, MA, 02134
finale@seas.harvard.edu

Barbara E Engelhardt
Gladstone Institutes & Stanford University
Stanford, CA, 94305
bengelhardt@stanford.edu

Abstract

In this work, we study an inverse reinforcement learning (IRL) problem where the experts are planning *under a shared reward function but with different, unknown planning horizons*. Without the knowledge of discount factors, the reward function has a larger feasible solution set, which makes it harder to identify a reward function. To overcome this challenge, we develop an algorithm that in practice, can learn a reward function similar to the true reward function. We give an empirical characterization of the identifiability and generalizability of the feasible set of the reward function.

1 Introduction

Designing reward functions in reinforcement learning (RL) can be difficult in domains such as healthcare [Riachi et al., 2021] and finance [Charpentier et al., 2021]. Inverse reinforcement learning (IRL) addresses this challenge by learning a reward function from expert demonstrations. The learned reward is a succinct description of the task and can be transferred to similar tasks with different environments. Recent research in IRL has focused on the additional challenge of learning the reward function from heterogeneous expert behaviors, for example, where expert demonstrations vary in quality [Shiarlis et al., 2016, Brown et al., 2019], or where each expert's policy is optimal under a distinct reward function [Mendez et al., 2018, Gleave and Habryka, 2018, Yu et al., 2019].

In this work, we focus on the setting where the expert behaviors vary because they are optimizing for *different planning horizons but a shared reward function*. In RL, the planning horizon is encoded in the discount factor, which discounts future (expected) rewards received for a given policy. A small discount factor corresponds to a short planning horizon, which implies that the expert focuses on short-term goals, whereas a large discount factor corresponds to a long planning horizon. For example, in an intensive care unit, the ventilation weaning practice is influenced by specific unit protocols and team or individual physician preferences [Kapnadak et al., 2015]. A more aggressive weaning practice implies a small discount factor. Another example is a mobile health application that helps with wellness self-management, where users set up short-term (< 1 year), long-term (1-2 years), or maintenance (> 2 years) health goals [Dicianno et al., 2017].

In this paper, unlike previous IRL work that relies on known discount factors [Cao et al., 2021, Rolland et al., 2022] , we assume that both the discount factors (one per expert) and the reward function are unknown. We show that existing IRL approaches cannot be directly adapted to our

NeurIPS 2023 Workshop on Generalization in Planning (GenPlan 2023).

IRL problem, e.g., linear programming IRL (LP-IRL [Ng et al., 2000]) admits undesirable feasible solutions such as identical discount factors for multiple experts. We develop a novel algorithm to learn a global multi-agent reward function with agent-specific discount factors based on LP-IRL. On a discrete MDP and grid domains, we show that our algorithm successfully identifies a reward function and a set of discount factors that can be used to reconstruct expert policies.

We also provide a better understanding of the identifiability and generalizability of IRL in settings where the discount factors associated with each expert, as well as the reward function, are unknown. Recent work shows that when there are two experts acting under a shared reward function and different but *known and correctly specified* discount factors, under certain conditions, IRL can identify the true reward function up to a constant [Cao et al., 2021, Rolland et al., 2022]. However, it is unclear how unknown or misspecified discount factors affect the identifiability and generalizability of the reward function. We empirically show that when the discount factors are unknown, the reward function is not identifiable. However, given a set of (misspecified) discount factors, the reward function has a unique feasible solution that generalizes well across most new RL tasks.

2 Related Work

IRL with homogeneous demonstrations. Previous IRL work focuses on identifying a reward function that explains expert behavior when the demonstrations are generated by a single expert. For example, max-margin IRL methods [Ng et al., 2000, Abbeel and Ng, 2004] seek a reward function that maximally separates the optimal policy and the second-most optimal policy. Max entropy IRL methods [Ziebart et al., 2008, Ziebart, 2010] estimate a reward function that maximizes the likelihood of the expert demonstrations. Bayesian IRL methods [Ramachandran and Amir, 2007, Jin et al., 2010] use prior knowledge to generate a posterior distribution over a set of possible reward functions. However, when given trajectories from multiple experts, each of these IRL approaches will learn a *separate* reward function for each expert by default, which is undesirable. In our work, we focus on a common scenario where the *global* reward function is shared by all experts with discount factors specific to each expert; these different discount factors lead to different optimal policies.

IRL with heterogeneous demonstrations. Recent IRL work explores heterogeneous expert demonstrations. Some methods study the scenario where expert demonstrations vary in quality. For example, Shiarlis et al. learns from demonstrations of both optimal policies and policies with undesirable behaviors (e.g., violating safety constraints). Similarly, Brown et al. assumes that one has access to a set of demonstrations ranked by their expected return. Other work studies the setting where each expert optimizes for a different task. For example, Babes et al. first identifies the tasks by clustering the expert demonstrations and then identifies a reward function for each task. In Yu et al. [2019], the authors use deep latent generative models to capture the shared reward structure of expert demonstrations. Finally, Mendez et al. consider a lifelong learning setting where the agent faces a sequence of similar tasks and optimizes overall performance. In contrast, we consider the scenario where experts share the same reward function but have different planning horizons. To the best of our knowledge, we are the first to study this type of IRL problem.

Identifiability and generalizability in IRL with respect to the reward function and discount factors. Recent work [Cao et al., 2021, Rolland et al., 2022] proves that, for entropy-regularized Markov decision processes (MDPs), if given optimal policies under two distinct discount factors, one can identify the true reward function up to a constant. However, the theory assumes that the discount factors are correctly specified, which is not realistic. In this work, we conduct identifiability and generalizability analyses on a toy domain without prior knowledge of the discount factors. Additionally, we develop algorithms to recover the reward function and the unknown discount factors simultaneously while previous work does not.

3 Background (LP-IRL)

An MDP is defined as a tuple $\mathcal{M}^* = (\mathcal{S}, \mathcal{A}, R^*, T, \gamma)$ where $\mathcal{S}$ is a finite state space, $\mathcal{A}$ is a set of discrete actions, the transition dynamics $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ describes the probability of reaching the next state s' by performing action a in the current state s, $R^*(s)$ is a deterministic reward function that we assume is action independent, and $\gamma \in [0, 1]$ is the discount factor. A Q-function RL

approach learns an optimal policy, $\pi^* : \mathcal{S} \to \mathcal{A}$, that maximizes the Q-function defined as

$$\pi^*(a|s) = \arg\max_a Q^\gamma_\pi(s,a) = \arg\max_a \Big(\sum_{s' \in \mathcal{S}} T(s'|s,a) R^*(s') + \mathbb{E}_\pi \Big[\sum_{t=1}^{\infty} \gamma^t R^*(S_t) \Big] \Big),$$

where the expectation is over the distribution of the state space induced by the policy π. The discount factor γ determines the bias towards short- or long-term outcomes. When $\gamma = 0$, the agent is myopic and ignores future rewards. When $\gamma = 1$, future rewards are valued as much as immediate ones.

In IRL problems, we are given an MDP, $\mathcal{M}^* \backslash R^*$, with an unknown reward function, and we consider a simple case where the expert policy can be fully observed. We wish to find a reward function R such that the expert policy is still optimal under the reconstructed MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, R, T, \gamma)$.

In Ng et al. [2000], the authors identify a reward function by solving,

$$\max_R \sum_{s \in S} \min_{a \in \mathcal{A} \backslash \pi^*(s)} Q^\gamma_{\pi^*}(s, \pi^*(s)) - Q^\gamma_{\pi^*}(s,a) - \lambda \|R\|_1 \tag{1a}$$

$$\text{s.t. } Q^\gamma_{\pi^*}(s, \pi^*(s)) - Q^\gamma_{\pi^*}(s,a) \geq 0 \quad \text{for } s \in \mathcal{S}, a \in \mathcal{A} \setminus a^* \tag{1b}$$

$$|R| \preceq R_{\max} \tag{1c}$$

$$\tag{1d}$$

where the l_1 norm penalty, $\lambda \|R\|_1$, regularizes the sparsity of the reward function. Constraints (1b) ensure that π^* is optimal under the inferred reward function R. The reward function solution to Equation 1a maximizes the sum of the differences of the Q-functions between the best and the next-best action over all states. Note that the difference of the Q-function is linear with respect to the reward function R,

$$Q^\gamma_{\pi^*}(s, \pi^*(s)) - Q^\gamma_{\pi^*}(s,a) = (T(\cdot|s,a^*) - T(\cdot|s,a))(I - \gamma T^{\pi^*})^{-1} R,$$

where T^{π^*} denotes the transition dynamics following the optimal policy π^*. Thus, the optimization problem in Equation 1 can be solved with linear programming (LP) [Ng et al., 2000].

4 Methods

In IRL problems with multiple planning horizons, we assume that we observe a set of expert policies $\Pi^* = \{\pi^*_k\}_{k=1}^K$, each optimized under the global reward function R^*, and transition dynamics T. But we allow a different discount factor γ^*_k. That is,

$$\pi^*_k(a|s) = \arg\max_a Q^{\gamma^*_k}_\pi(s,a) = \arg\max_a \Big(\sum_{s'} T(s'|s,a) R^*(s') + \mathbb{E}_\pi [\sum_{t=1}^{\infty} \gamma^{*t}_k R^*(S_t)] \Big).$$

We assume that both the reward function R^* and the set of discount factors $\Gamma^* = \{\gamma^*_k\}_{k=1}^K$ are unknown to us. We further make the following assumption:

Assumption 1. *For any two distinct expert policies $\pi^*_i, \pi^*_j \in \Pi^*,\ i \neq j$, optimized under γ^*_i, γ^*_j respectively, there is at least one state $s \in \mathcal{S}$ such that $Q^{\gamma^*_i}_{\pi^*_i}(s, \pi^*_i(s)) > Q^{\gamma^*_j}_{\pi^*_i}(s, \pi^*_j(s))$.*

The assumption ensures that each expert policy is uniquely optimal in at least one of the states, i.e., no two expert policies can be equally optimal in all states. The assumption implies that the discount factors lead to distinct policies, and thus are distinguishable from the data (i.e., $\gamma^*_i \neq \gamma^*_j$ for $i \neq j$).

Naive extension fails. To extend the formulation in Section 3 to our target IRL setting, one naive idea is to maximize the sum of the differences of the Q-functions over all experts as well as states, i.e.,

$$\max_{\Gamma \in [0,1]^K} \max_R \sum_{k \in [K]} \sum_{s \in \mathcal{S}} \min_{a \in A \backslash \pi^*_k(s)} Q^{\gamma_k}_{\pi^*_k}(s, \pi^*_k(s)) - Q^{\gamma_k}_{\pi^*_k}(s,a) - \lambda \|R\|_1 \tag{2a}$$

$$\text{s.t. } Q^{\gamma_k}_{\pi^*_k}(s, \pi^*_k(s)) - Q^{\gamma_k}_{\pi^*_k}(s,a) \geq 0 \quad \forall s \in \mathcal{S},\ a \in \mathcal{A} \backslash \pi^*_k(s), k \in [K] \tag{2b}$$

$$|R| \preceq R_{\max} \tag{2c}$$

There are two issues with the optimization problem in Equation 2. First, the objective function is not convex with respect to discount factors Γ. We will use Bayesian optimization (BO) techniques to solve the nonconvex optimization problem (Section 4.2). Second, and more problematic, this formulation allows for a solution that maximizes the Q-function difference of one particular policy with its discount factor. However, this allows all policies to be equally optimal under other discount factors, which violates Assumption 1 that requires each policy to be distinguishable with its discount factor. Ideally, the expert policies should be sufficiently different in terms of Q-functions under the learned reward function and discount factors.

4.1 LP-IRL with multiple discount factors

To avoid undesirable solutions, we propose a new optimization problem as follows:

$$\max_{\Gamma\in[0,1]^K} \max_{R} \min_{k\in[K],(s,a)\in\Omega_k} Q^{\gamma_k}_{\pi^*_k}(s,\pi^*_k(s)) - Q^{\gamma_k}_{\pi^*_k}(s,a) - \lambda\|R\|_1 \tag{3a}$$

$$\text{s.t. } Q^{\gamma_k}_{\pi^*_k}(s,\pi^*_k(s)) - Q^{\gamma_k}_{\pi^*_k}(s,a) \geq 0 \quad \forall s\in\mathcal{S},\ a\in\mathcal{A}\backslash\pi^*_k(s), k\in[K] \tag{3b}$$

$$|R| \preceq R_{\max}, \tag{3c}$$

where Ω_k is a pre-computed set of state-action tuples that ensures there exists a feasible reward solution R such that, for any $(s,a)\in\Omega_k,\ Q^{\gamma_k}_{\pi^*_k}(s,\pi^*_k(s)) - Q^{\gamma_k}_{\pi^*_k}(s,a) > 0$. Details of the algorithm that constructs Ω_k are described in Appendix A. In situations where there is a pair of expert policies π^*_i, π^*_j that cannot be distinguished under any reward function (i.e., $(s,\pi^*_i(s)) \notin \Omega_j$ for any $s\in\mathcal{S}$), we cannot find a solution that does not violate Assumption 1 and claim the optimization problem in Equation 3 to be infeasible. Intuitively, our proposed optimization problem seeks a reward function that maximizes the minimal non-zero difference of Q-functions of all experts over states where policies are distinguishable, thus encouraging expert policies to be sufficiently different and ensuring the satisfaction of Assumption 1.

4.2 Inference for LP-IRL

Unfortunately, the objective function in 3a is not convex with respect to the discount factors Γ. To solve for the global optima, we perform a bi-level optimization. Denote the upper-level objective function as

$$f(\Gamma,R) = \min_{k\in[K],(s,a)\in\Omega_k} Q^{\gamma_k}_{\pi^*_k}(s,\pi^*_k(s)) - Q^{\gamma_k}_{\pi^*_k}(s,a) - \lambda\|R\|_1.$$

We rewrite the optimization problem in Equation 3 as,

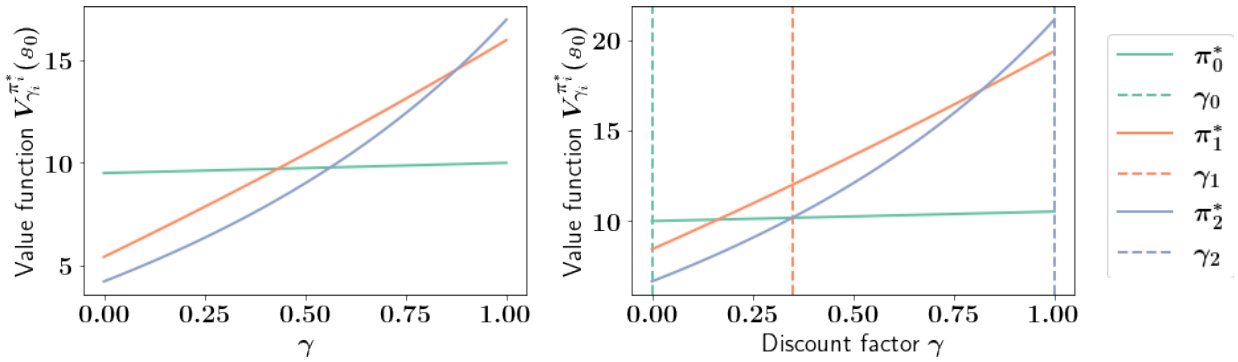
$$\max_{\Gamma\in[0,1]^k} g, \tag{4}$$

where $g = \max_R f(\Gamma,R)$ under constraints 3b , 3c. Given any $\Gamma\in[0,1]^K$, the lower-level objective function g can be solved analytically with LP. The upper-level optimization problem can be solved by performing a grid search over the space $[0,1]^K$. However, the computation complexity of grid search increases exponentially with respect to the total number of expert policies. To improve computation efficiency, we use Bayesian optimization (BO) techniques, which are suitable for nonconvex objective functions that are expensive to evaluate. The full algorithm is described in Appendix B.

5 Experiments and Results

5.1 Domains

We test Algorithm 1 in two domains: (1) a toy domain, a discrete MDP with 3 distinct expert policies (Figure 3a); (2) a grid domain [Ankile et al., 2023] with 4 distinct expert policies (Figure 5a). In the toy domain, experts trade-off between the probability of getting the reward and the reward magnitude. In the grid domain, experts choose between a small reward close by or a large one that is far away. The details are described in Appendix C. We first study the properties of the global optima of the reward function and the set of discount factors without considering the inference efficiency. That is, we solve the upper-level optimization problem in Equation 4 by performing a grid search. We then study the convergence performance of Algorithm 1, which utilizes BO for efficient inference.



(a) The value function $V_{\pi_k^*}^{\gamma}(s_0))$ of expert policies $\pi_0^*,\ \pi_1^*,\ \pi_2^*$ under the true reward R^*

(b) The value function $V_{\pi_k^*}^{\gamma}(s_0)$ of reconstructed optimal policies $\pi_0,\ \pi_1,\ \pi_2$ under the global maximum, $\tilde{R}$

Figure 1: Plots of the value function of the initial state, $V_{\pi_k^*}^{\gamma}(s_0)$, under (a) the true reward function R^* (b) and the global optimum of the reward function of Equation 3, $\tilde{R}$: $x,\ y$-axes represent the discount factor $\gamma \in [0, 1]$ and the value function $V_{\pi_k^*}^{\gamma}(s_0)$ of expert policy π_k^* respectively. Each color represents a different expert policy. With the true reward function R^*, each expert policy, π_k^*, is optimal under a continuous set of γ. The dashed lines in (b) represent the global optima of the discount factors, $\tilde{\Gamma}$.

5.2 Results

The discount factors optimizing our objective recover the true order of discount factors. From Figure 1, we see that although the learned discount factor ($\tilde{\Gamma} \approx \{0,\ 0.35,\ 1\}$) does not exactly align with the ground truth, they follow the same order of the true discount factors ($\gamma_0^* \leq \gamma_1^* \leq \gamma_2^*$). This is also true for the grid domain where the global maximum is achieved at $\tilde{\Gamma} \approx \{0.38,\ 0.72,\ 0.94,\ 1\}$. This property allows us to interpret the bias of each expert's goal—a small discount factor implies that the policy cares more about short-term outcomes and vice versa.

The reward function optimizing our object has a similar structure to the true reward function. Comparing Appendix Figure 3a to 3b, we see that the global optimum of the reward function has a larger reward at state s_2 than at state s_1. When the discount factor is large, this reward structure encourages the agent to collect the large reward even in the face of more stochasticity. Comparing Appendix Figure 5a to 5b, we see that the learned reward function learns a small reward for the left bottom grid and a large reward for the right bottom grid, which is similar to the true reward function.

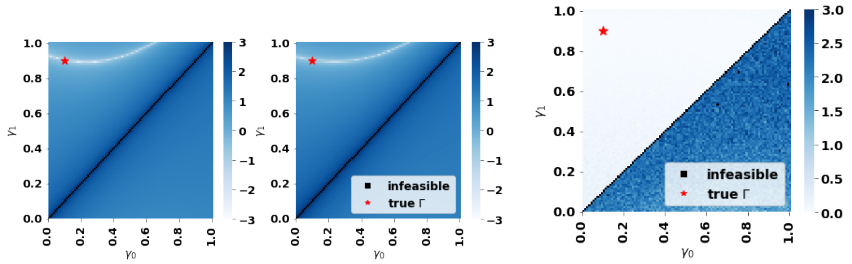
Bayesian Optimization converges quickly. From Appendix Figure 7, we see that on the toy domain, BO converges to the global maximum within 50 iterations while for the grid domain, within 200 iterations. If one were to perform a grid search over the space of $\Gamma \in [0, 1]^K$ with an interval of 0.1, it would require solving the linear programming 10^K times. Our algorithm is more computationally efficient and converges reliably to the global maximum.

5.3 Identifiability and Generalizability Analyses

In this section, we give an empirical characterization of the feasible set of the reward function and the set of discount factors under our IRL setting described in Section 3.

We assume that the observed expert policies are entropy-regularized optimal policies [Liu et al., 2019]. We conduct empirical identifiability and generalizability analyses on the described domain in Appendix Figure 4, whose feasible solution set is smaller than that of the toy domain in Appendix Figure 3a and easier to visualize. We show that when the discount factors are assumed to be unknown, the reward function is not identifiable anymore. However, if given a set of possibly misspecified discount factors, the reward function has a unique feasible solution and it generalizes well across a wide range of different RL tasks. See proofs in Corollary 1.

From Figure 2a, we can see that given two *misspecified* discount factors, the unique feasible reward function can be very different from the true reward function (the heatmap is dark in most of the plot



(a) Heatmap of the absolute error of the reward of states $s_1,\ s_2$ on a $\log_{10}$ scale ($\log_{10}|R - R^*|$)

(b) Heatmap of the absolute error of the value functions of the true optimal policy and the reconstructed optimal policy under randomly generated transition dynamics and discount factors (Equation 19).

Figure 2: Identifiability and generalizability of the reward function: x, y-axis represents (mis)specified γ_0, γ_1 respectively. The red star represents the true discount factors γ_0^*, γ_1^*. There are no feasible solutions along the diagonal. Lighter blue indicates a smaller error and vice versa.

area except the white curve on the top). From Figure 2a, we see that although the reward function is not identifiable, it generalizes well when $\gamma_0 < \gamma_1$ (the upper diagonal is lightly colored).

6 Conclusion and Future Work

In this work, we focus on an IRL setting with multiple experts optimizing with a global reward function but different discount factors. We assume that both the reward function and the discount factors are unknown. We show that in our targeted IRL setting, the feasible set of the reward function is larger and the reward function is not identifiable, which increases the difficulty of the IRL problem. To address the challenge, we make non-trivial modifications to the existing LP-IRL algorithm. We show that our algorithm can recover a reward structure that is similar to the true reward function, and also a set of discount factors whose magnitudes follow the same order as the true discount factors.

In the future, we want to extend the Max Causal Entropy IRL algorithm to our IRL problem and provide theoretical guarantees on the identifiability and generalizability of the reward function as well as the discount factors.

7 Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. IIS-2007076. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. JY and BEE were funded in part by Helmsley Trust grant AWD1006624, NIH NCI 5U2CCA233195, and CZI. BEE is a CIFAR Fellow in the Multiscale Human Program. JY acknowledges support from WP at Harvard University.

References

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.

Lars L Ankile, Brian S Ham, Kevin Mao, Eura Shin, Siddharth Swaroop, Finale Doshi-Velez, and Weiwei Pan. Discovering user types: Mapping user traits by task-specific behaviors in reinforcement learning. *arXiv preprint arXiv:2307.08169*, 2023.

Monica Babes, Vukosi Marivate, Kaushik Subramanian, and Michael L Littman. Apprenticeship learning about multiple intentions. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 897–904, 2011.

Daniel Brown, Wonjoon Goo, Prabhat Nagarajan, and Scott Niekum. Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations. In *International conference on machine learning*, pages 783–792. PMLR, 2019.

Haoyang Cao, Samuel Cohen, and Lukasz Szpruch. Identifiability in inverse reinforcement learning. *Advances in Neural Information Processing Systems*, 34:12362–12373, 2021.

Arthur Charpentier, Romuald Elie, and Carl Remlinger. Reinforcement learning in economics and finance. *Computational Economics*, pages 1–38, 2021.

Brad Edward Dicianno, Geoffrey Henderson, and Bambang Parmanto. Design of mobile health tools to promote goal achievement in self-management tasks. *JMIR mHealth and uHealth*, 5(7):e7335, 2017.

Adam Gleave and Oliver Habryka. Multi-task maximum entropy inverse reinforcement learning. *arXiv preprint arXiv:1805.08882*, 2018.

Zhuo-Jun Jin, Hui Qian, and Miao-Liang Zhu. Gaussian processes in inverse reinforcement learning. In *2010 International Conference on Machine Learning and Cybernetics*, volume 1, pages 225–230. IEEE, 2010.

Siddhartha G Kapnadak, Steve E Herndon, Suzanne M Burns, Y Michael Shim, Kyle Enfield, Cynthia Brown, Jonathon D Truwit, and Ajeet G Vinayak. Clinical outcomes associated with high, intermediate, and low rates of failed extubation in an intensive care unit. *Journal of Critical Care*, 30(3):449–454, 2015.

Jingbin Liu, Xinyang Gu, and Shuai Liu. Policy optimization reinforcement learning with entropy regularization. *arXiv preprint arXiv:1912.01557*, 2019.

Jorge Mendez, Shashank Shivkumar, and Eric Eaton. Lifelong inverse reinforcement learning. *Advances in neural information processing systems*, 31, 2018.

Andrew Y Ng, Stuart Russell, et al. Algorithms for inverse reinforcement learning. In *Icml*, volume 1, page 2, 2000.

Deepak Ramachandran and Eyal Amir. Bayesian inverse reinforcement learning. In *IJCAI*, volume 7, pages 2586–2591, 2007.

Elsa Riachi, Muhammad Mamdani, Michael Fralick, and Frank Rudzicz. Challenges for reinforcement learning in healthcare. *arXiv preprint arXiv:2103.05612*, 2021.

Paul Rolland, Luca Viano, Norman Schürhoff, Boris Nikolov, and Volkan Cevher. Identifiability and generalizability from multiple experts in inverse reinforcement learning. *Advances in Neural Information Processing Systems*, 35:550–564, 2022.

Kyriacos Shiarlis, Joao Messias, and Shimon Whiteson. Inverse reinforcement learning from failure. 2016.

Lantao Yu, Tianhe Yu, Chelsea Finn, and Stefano Ermon. Meta-inverse reinforcement learning with probabilistic context variables. *Advances in neural information processing systems*, 32, 2019.

Brian D Ziebart. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. Carnegie Mellon University, 2010.

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

A Construct Ω_k

In the following, we assume that we are given an arbitrary set of discount factors Γ. We show that for any π_k^*, how to construct a set of state-action tuples that ensures there exists a feasible reward solution R such that for any $(s,a) \in \Omega_k,\ Q_{\pi_k^*}^{\gamma_k}(s, \pi_k^*(s)) - Q_{\pi_k^*}^{\gamma_k}(s,a) > 0$. Recall that in an IRL setting with multiple experts (multiple discount factors), we solve the following optimization problem:

$$\max_{\Gamma \in [0,1]^K} \max_{R} \min_{k \in [K], (s,a) \in \Omega_k} Q_{\pi_k^*}^{\gamma_k}(s, \pi_k^*(s)) - Q_{\pi_k^*}^{\gamma_k}(s,a) - \lambda \|R\|_1 \tag{5a}$$

$$\text{s.t. } Q_{\pi_k^*}^{\gamma_k}(s, \pi_k^*(s)) - Q_{\pi_k^*}^{\gamma_k}(s,a) \geq 0 \quad \forall s \in \mathcal{S},\ a \in \mathcal{A} \backslash \pi_k^*(s), k \in [K] \tag{5b}$$

$$|R| \preceq R_{\max}, \tag{5c}$$

With Bellman Equations, the difference of the Q-functions can be written as,

$$Q_{\pi_k^*}^{\gamma_k}(s, \pi_k^*(s)) - Q_{\pi_k^*}^{\gamma_k}(s,a) = (T(\cdot|s, \pi_k^*(s)) - T(\cdot|s,a))(I - \gamma_k T^{\pi_k^*})^{-1} R.$$

Further, let $W \in \mathbb{R}^{(|\mathcal{S}| \times (|\mathcal{A}|-1) \times K) \times |\mathcal{S}|}$ be a matrix where each row

$$w_{s,a,k}^\intercal = (T(\cdot|s, \pi_k^*(s)) - T(\cdot|s,a))(I - \gamma_k T^{\pi_k^*})^{-1},$$

for $s \in \mathcal{S}, a \in \mathcal{A} \setminus \pi_k^*(s), k \in [K]$. The optimization problem in Equation 5 then becomes

$$\max_{R} \min_{k \in [K], (s,a) \in \Omega_k} WR \tag{6a}$$

$$\text{s.t. } WR \geq 0 \tag{6b}$$

$$|R| \preceq R_{\max} \tag{6c}$$

To construct Ω_k, we search for a reward function R that distinguishes the expert policy from other actions on as many states as possible. I.e., we solve the following optimization problem,

$$\max_{R} |WR|_0 \tag{7a}$$

$$\text{s.t. } WR \geq 0 \tag{7b}$$

Let R^* be the optimal solution of Optimization Problem 7. Then,

$$\Omega_k = \{(s,a) \in (\mathcal{S} \times \mathcal{A} \setminus \pi^*) | w_{s,a,k}^\intercal R^* \neq 0\}.$$

In the following, we show that Optimization Problem 7 can be solved with Linear Programming.

Claim 1. *The optimization problem in Equation 7 is equivalent to the following*

$$\min_{R,z} \mathbb{1}^\intercal z \tag{8a}$$

$$\textit{s.t. } WR + z \geq 1 \tag{8b}$$

$$WR \geq 0 \tag{8c}$$

$$z \geq 0 \tag{8d}$$

Specifically, an optimal solution R^* of Optimization Problem 8 is also an optimal solution of Optimization Problem 7. Furthermore, $\mathbb{1}^\intercal z^* = |\mathcal{S}| \times (|\mathcal{A}| - 1) \times K - |WR^*|_0$

Proof. Let R be a reward function that satisfies Constraint 8c. Then for any constant $c > 0$, cR also satisfies Constraint 8c. Now, let c be a constant such that any positive element in cWR is larger than 1. Denote the i-th element of any vector x by x_i. The vector z that minimizes $\mathbb{1}^\intercal z$ while satisfies Constraints 8b, 8d has the following form:

$$z_i = 1 \text{ if } (cWR)_i = 0,\ z_i = 0 \text{ if } (cWR^*)_i \geq 1. \tag{9}$$

We show that an optimal solution of Optimization Problem 8 is also an optimal solution of Optimization Problem 7 with proof by contradiction. Let (R^*, z^*) be an optimal solution of Optimization Problem 8 with z^* defined in Equation 9. We assume that R^* is optimal under Optimization Problem 7 Then there exists another $\tilde{R}$ such that $W\tilde{R} \geq 0$ and $|W\tilde{R}|_0 > |WR^*|_0$. Now, let c be a constant such that any positive element in $W(c\tilde{R})$ is larger than or equal 1 and construct a vector $\tilde{z}$ according to Equation 9. Because $|cW\tilde{R}|_0 = |W\tilde{R}|_0 > |WR^*|_0$, $W\tilde{R} \geq 0$ and $WR^* \geq 0$, $\tilde{z}$ will have more zero elements than z^*. Thus, $\mathbb{1}^\intercal \tilde{z} < \mathbb{1}^\intercal z^*$, which contradicts the assumption that z^* is optimal.

With the definition of z^* in Equation 9, it is easy to see that $\mathbb{1}^\intercal z^* = |z|_0 = |\mathcal{S}| \times (|\mathcal{A}| - 1) \times K - |WR^*|_0$. □

B Algorithm

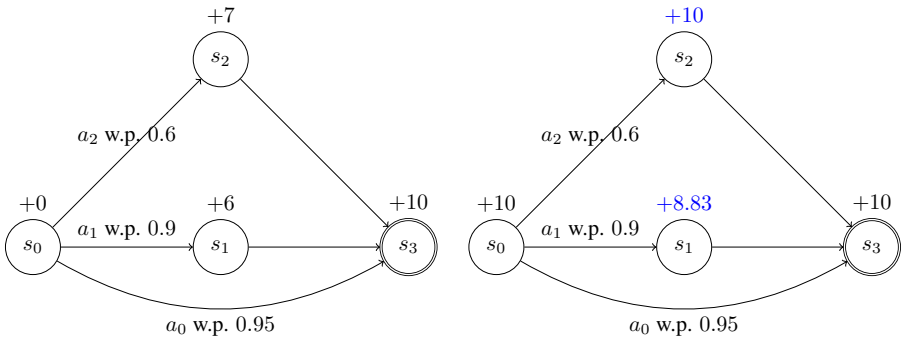
Algorithm 1 LP-IRL with multiple planning horizons

Given a set of K observed policies $\Pi^* = \{\pi_k^*\}_{k=1}^K$ and the transition dynamics T.
Place a Gaussian process prior on g.
Observe g at a set of m points, $\{\Gamma^{(i)}\}_{i=1}^m$, with $\Gamma^{(i)} \sim \texttt{Unif}(0,1)^K$
while $m \leq N$ **do**
    Update the posterior distribution of g given all observed points.
    Query a new point $\Gamma^{(m)}$ based on the acquisition function.
    Observe $g_m = \max_R f(\Gamma^{(m),R})$ with constraints in Equation 3b, 3c.
    increment m
end while
Return the reward function R and the set of discount factors Γ with the largest observed g.

C Domains

C.1 Toy Domain with a Discrete MDP

The toy domain is designed such that the optimal policy is a 3-step piecewise function with respect to the discount factor $\gamma \in [0, 1]$. We assume the transition dynamics T is given.



(a) Discrete MDP with the true reward function. (b) Discrete MDP with the learned function.

Figure 3: Toy Domain

Specifically, s_0 is the initial state, s_3 is the absorbing state. The agent gets a large positive reward when getting to the absorbing state ($r(s_3) = 10$). The agent gets some small rewards when getting to s_1, s_2 ($r(s_1) = 6,\ r(s_2) = 7$). With action a_0, the agent moves to the absorbing state w.p. 0.95. With action a_1, the agent moves to s_1 w.p. 0.9. With action a_2, the agent moves to s_2 w.p. 0.6. The agent stays otherwise. Note that although s_2 has a larger reward, the agent faces more stochasticity. Thus, there is a trade-off when the discount factor γ varies. See MDP details in Figure 3a.

Let $\pi_0^*, \pi_1^*, \pi_2^*$ denote the optimal action is a_0, a_1, a_2 ($\pi_i = \mathbb{1}_{a_i}$). We note that for this toy example, in state s_1, s_2, s_3, $\pi_0^*, \pi_1^*, \pi_2^*$ are equally optimal. Thus, we focus on s_0 in the later analysis. The value functions are

$$\begin{cases} V_\gamma^{\pi_0}(s_0) & = \frac{0.95 \cdot 10}{1-0.05\gamma} \\ V_\gamma^{\pi_1}(s_0) & = \frac{0.9(6+10\gamma)}{1-0.1\gamma} \\ V_\gamma^{\pi_2}(s_0) & = \frac{0.6(7+10\gamma)}{1-0.4\gamma} \end{cases}$$

Solve pairwise difference between value functions and let $\gamma_0 \approx 0.432,\ \gamma_1 \approx 0.876$. When $\gamma < \gamma_0$, $\pi^* = \pi_0$. When $\gamma_0 < \gamma < \gamma_1$, $\pi^* = \pi_1$. When $\gamma > \gamma_1$, $\pi^* = \pi_2$. See Figure 1a.

C.2 Discrete MDP for Identifiability Analyses

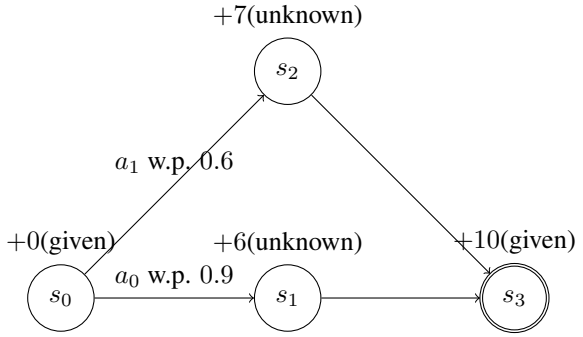


Figure 4: Discrete MDP for identifiability Analyses. We assume that $R(s_0)$, $R(s_3)$ are given and we need to infer $R(s_1),\ R(s_2)$

We modify the discrete MDP for easier analyses of the feasible solution set of the reward function R and the discount factor set Γ. Specifically, we only consider two actions. We assume that we are given the rewards at state $s_0,\ s_3$ ($R^*(s_0) = 0,\ R^*(s_3) = 10$) and we only need to infer the rewards at $s_1,\ s_2$, ($R^*(s_1), R^*(s_2)$ are unknown). With action a_0, the agent moves to s_1 w.p. 0.9. With action a_1, the agent moves to s_2 w.p. 0.6. The agent stays in state s_0 otherwise. See details in Figure 4.

We are given two entropy-regularized optimal policies π_0^*, π_1^* with $\gamma_0^* = 0.1,\ \gamma_1^* = 0.9$, respectively. We note that in state s_1, s_2, s_3, π_0^*, π_1^* are equally optimal. Thus for $s \in \{s_1, s_2, s_3\}$, $\pi_0^*(\cdot|s) = \pi_1^*(\cdot|s) = 0.5$.

C.3 Grid Domain

In the grid domain, there are two absorbing states: one at the left bottom cell with a small reward ($+10$) and one at the right bottom cell with a small reward ($+100$). Each step costs -10 if not reaching the rewarding state. The true reward function is plotted in 5a. The agent can start from anywhere in the grid world except the absorbing states. The agent can choose to move {`right, down, left, up`} at each state. If the agent comes across the wall by taking an action, the agent stays where it is and moves to the corresponding state otherwise. The transition dynamics of the grid domain are deterministic.

We observe 4 distinct expert policies whose optimal actions are visualized in Figure 6. We see that when the discount factor is small, the agent will go for the closest reward regardless of its magnitude (Figure 6a). When the discount factor is large, the agent prefers the large reward regardless of how far the reward is (Figure 6d).

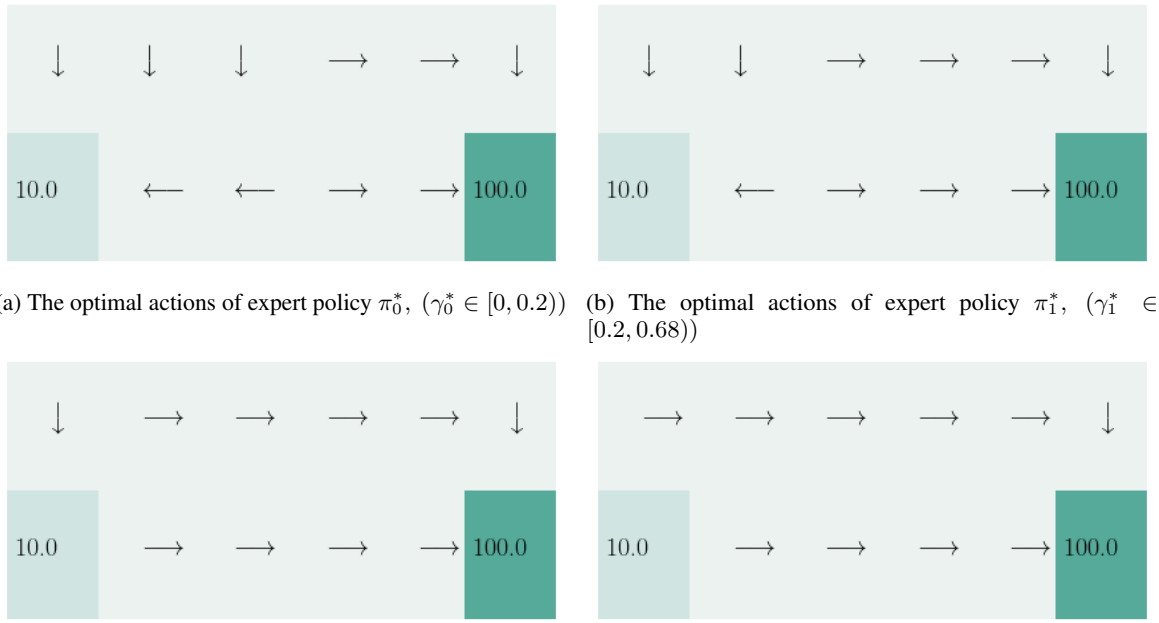
D Additional Results

We plot the trace plot of the best observed objective function of BO 7. We see that BO converges fast (within 50 iterations for the toy domain and 200 iterations for the grid domain).

(a) The true reward function R of the grid domain.

(b) The learned reward function $\tilde{R}$ of the grid domain.

Figure 5: The reward function of the grid domain.



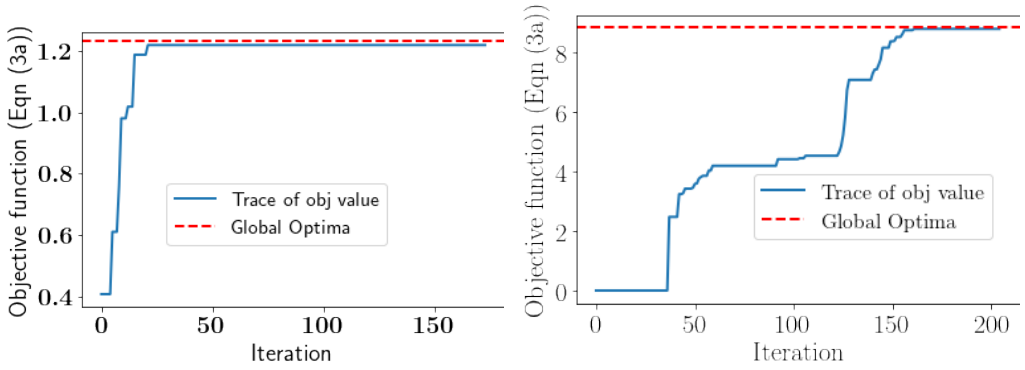
(a) The optimal actions of expert policy π_0^*, $(\gamma_0^* \in [0, 0.2))$

(b) The optimal actions of expert policy π_1^*, $(\gamma_1^* \in [0.2, 0.68))$

(c) The optimal actions of expert policy π_2^*, $(\gamma_2^* \in [0.68, 0.87))$

(d) The optimal actions of expert policy π_3^*, $(\gamma_3^* \in [0.87, 1])$

Figure 6: Visualization of the optimal actions at each state.



(a) The toy domain

(b) The grid domain

Figure 7: Trace plots of the best observed objective value of BO: x, y-axis represents the iteration and the best observed objective value, respectively. The red dashed line represents the global maximum from performing a grid search over $[0, 1]^K$.

E Identifiability and Generalizability Analyses

E.1 Identifiability Analyses

In entropy-regularized RL with MDP ,$\mathcal{M} = \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma$, the agent additionally maximizes its entropy at each visited state,

$$\pi^* = \arg\max_{\pi \in \mathcal{P}} V_\gamma^\pi = \arg\max_{\pi \in \mathcal{P}} \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t \Big(R(s_t) + \alpha \mathcal{H}(\cdot|s_t)\Big)\Big],$$

where $\mathcal{P}$ denotes the probability space, $\mathcal{H} = -\pi(a_t|s_t)\log\pi(a_t|s_t)$ and α is the temperature parameter that controls the trade-off between the value function and the entropy regularization. When α is large, the entropy penalization is large and the optimal policy is encouraged to be more stochastic.

In an IRL problem $\mathcal{M}\backslash R$ where we observe a single optimal policy, we have the following theorem

Theorem 1. *(Restate of Theorem 1 in [Cao et al., 2021]) Let $T^a(s, s')$ denote the probability of reaching state s' from the state s by taking action a. For a fixed policy $\pi(a|s)$, a discount factor $\gamma \in [0, 1)$, and an arbitrary choice of vector $v \in \mathbb{R}^{|S|}$, there is a unique corresponding action-independent reward function R*

$$T^a R = \alpha \log \pi^*(a|s) - \gamma T^a v + v \quad \textit{for } a \in \mathcal{A} \tag{10}$$

such that the MDP with reward function R yields an entropy-regularized optimal policy $\pi_\gamma^* = \pi$ with value function $V_\gamma^{\pi^*} = v$.

Corollary 1. *Consider the IRL problem described in Section C.2 with two observed policies $\pi_0^*,\ \pi_1^*$ acting optimally under the global reward function R^* but different discount factors γ_0^*, γ_1^*. Assume that the true discount factors Γ^* and the reward function are unknown to us. Then, for any given $\gamma_0,\ \gamma_1 \in [0, 1]$,*

1. *When $\gamma_0 = \gamma_1$, there is no feasible reward function.*
2. *When $\gamma_0 \neq \gamma_1$, there is a unique reward function such that $\pi_0^*,\ \pi_1^*$ are optimal under the discount factors $\gamma_1,\ \gamma_2$.*

Proof. Let $R = \begin{bmatrix} 0 \\ R_1 \\ R_2 \\ 10 \end{bmatrix}$, $\mathbf{v}_0 = \begin{bmatrix} v_{00} \\ v_{01} \\ v_{02} \\ v_{03} \end{bmatrix}$, $\mathbf{v}_1 = \begin{bmatrix} v_{10} \\ v_{11} \\ v_{12} \\ v_{13} \end{bmatrix}$. Plugging $R,\ \mathbf{v}_0,\ \mathbf{v}_1$ into Equation 10 results in,

$$T^a R = \alpha \log \pi_0^*(a|s) - \gamma_0 T^a v_0 + v_0 \quad \text{for } a \in \{a_0, a_1\} \tag{11}$$

$$T^a R = \alpha \log \pi_1^*(a|s) - \gamma_1 T^a v_1 + v_1 \quad \text{for } a \in \{a_0, a_1\}. \tag{12}$$

By re-writing Equations 11, 12 into a system of linear equations and solving it, we obtain the following equalities

$$\begin{cases} v_{01} = v_{02} = -\alpha(\gamma_0^2 + \gamma_0 + 1)\log 0.5 + 10 & (13) \\ v_{11} = v_{12} = -\alpha(\gamma_1^2 + \gamma_1 + 1)\log 0.5 + 10 & (14) \\ R_1 = \dfrac{1}{0.9}[\alpha \log \pi_0^*(a_0|s_0) - \gamma_0(0.1 v_{00} + 0.9 v_{01}) + v_{00}] & (15) \\ R_2 = \dfrac{1}{0.6}[\alpha \log \pi_0^*(a_1|s_0) - \gamma_0(0.4 v_{00} + 0.6 v_{02}) + v_{00}] & (16) \\ R_1 = \dfrac{1}{0.9}[\alpha \log \pi_1^*(a_0|s_0) - \gamma_1(0.1 v_{10} + 0.9 v_{11}) + v_{10}] & (17) \\ R_2 = \dfrac{1}{0.6}[\alpha \log \pi_1^*(a_1|s_0) - \gamma_1(0.4 v_{10} + 0.6 v_{12}) + v_{10}] & (18) \end{cases}$$

Subtract Equation 15 from Equation 17, Equation 16 from Equation 18 and we will have

$$\begin{bmatrix} 1 - 0.1\gamma_0 & -1 + 0.1\gamma_1 \\ 1 - 0.4\gamma_0 & -1 + 0.4\gamma_1 \end{bmatrix} \begin{bmatrix} v_{00} \\ v_{10} \end{bmatrix} = \begin{bmatrix} \alpha(\log \pi_1^*(a_0|s_0) - \log \pi_0^*(a_0|s_0)) - 0.9(\gamma_0 v_{01} - \gamma_1 v_{11}) \\ \alpha(\log \pi_1^*(a_1|s_0) - \log \pi_0^*(a_1|s_0)) - 0.6(\gamma_0 v_{02} - \gamma_1 v_{12}) \end{bmatrix}$$

The matrix $\begin{bmatrix} 1 - 0.1\gamma_0 & -1 + 0.1\gamma_1 \\ 1 - 0.4\gamma_0 & -1 + 0.4\gamma_1 \end{bmatrix}$ is invertible if and only if $\gamma_0 \neq \gamma_1$. Thus, when $\gamma_0 \neq \gamma_1$, $[v_{00}, v_{10}]$ have a unique solution, which corresponds to a unique $[R_1, R_2]$. □

E.2 Generalizability Analyses

We test the generalizability of the reward function under misspecified discount factors. For each pair of $\gamma_0,\ \gamma_1 \in [0, 1]$, we generate 100 random transition dynamics $\{T^{(i)}\}_{i=1}^{100}$ and discount factors $\{\gamma^{(i)}\}_{i=1}^{100}$. We then solve the optimal policies under the true reward function R^* and the unique feasible reward function R, denoted as $\pi^*, \pi^{(i)}$ respectively. We evaluate the policy quality by computing the difference of the value function under the true R^* and γ',

$$\triangle V = \frac{1}{100} \sum_{i=1}^{100} V^{\pi^*}_{R^*,T^{(i)},\gamma^{(i)}} - V^{\pi^{(i)}}_{R^*,T^{(i)},\gamma^{(i)}} \tag{19}$$