Analysis of Inverse Reinforcement Learning with Perturbed Demonstrations

Francisco S. Melo[1] and **Manuel Lopes**[2] and **Ricardo Ferreira**[3]

Abstract.

Inverse reinforcement learning (IRL) addresses the problem of recovering the unknown reward function for a given Markov decision problem (MDP) given the corresponding optimal policy or a perturbed version thereof. This paper studies the space of possible solutions to the general IRL problem, when the agent is provided with incomplete/imperfect information regarding the optimal policy for the MDP whose reward must be estimated. We focus on scenarios with finite state-action spaces and discuss the constraints imposed on the set of possible solutions when the agent is provided with (i) perturbed policies; (ii) optimal policies; and (iii) incomplete policies. We discuss previous works on IRL in light of our analysis and show that, with our characterization of the solution space, it is possible to determine non-trivial closed-form solutions for the IRL problem. We also discuss several other interesting aspects of the IRL problem that stem from our analysis.

1 Introduction

Inverse reinforcement learning (IRL) addresses the problem of recovering the unknown reward function for a given Markov decision problem (MDP) given the corresponding optimal policy. Originally formulated in [8], the first formal treatment of IRL is due to Ng and Russel [6]. In their work, the authors provide a formal characterization of the solution space for the IRL problem and several algorithms designed to tackle different variations thereof. However, most results in [6] rest on the underlying assumption that the learner is provided access to the optimal policy for the target reward.

Several posterior works proposed modifications to the original IRL formulation. For example, in [9] the authors address *IRL with evaluation*, in which the expert is unable to describe the optimal policy but can only evaluate two policies comparatively. In [4, 7] the authors propose a different view of IRL, in which the reward function is seen as providing a *parameterization* of the target policy. IRL then reduces to a supervised learning problem, where the goal is to approximate, within a parameterized family of policies, one specific target policy from (noisy) samples thereof. The particular approach in [7] relies on *Bayesian inference* and proposes an algorithm to estimate the posterior distribution over the possible reward functions given the demonstration. The same work also shows that the original algorithms in [6] can be recovered by an adequate choice of prior and likelihood function. In [4], on the other hand, a gradient-descent approach is proposed to minimize a quadratic loss function.

In a somewhat different line of work, several recent works have adopted IRL-based approaches to *apprenticeship learning* [2, 10, 11]. In apprenticeship learning, the learner is less concerned with recovering a reward function than to recover a policy that closely matches the performance of the demonstrator in some precise sense. IRL-based approaches to apprenticeship learning assume that the demonstrator is following a policy (not necessarily optimal) for a known underlying Markov decision problem. Then, by recovering an intermediate reward function in an IRL-like fashion, the desired policy can be computed as the optimal policy associated with this reward function. We refer to [1] for additional details and references.

In this paper we contribute to the existing literature in two aspects. On one hand, while currently there is a rich body of work on algorithmic approaches to IRL and apprenticeship learning, the only theoretical analysis of the IRL problem and corresponding solution space is provided by the pioneer work of Ng and Russel [6]. Unfortunately, as already mentioned, the analysis in [6] assumes that the optimal policy is available to the learner, complete and error-free. In this paper we complement that analysis and characterize the IRL solution space when (a) the optimal policy may not be completely specified; and/or (b) the learner can only access a *perturbed* version of the optimal policy. Our results also differ from those in [6] in that we consider a different reward model.

Our analysis of the solution space for the IRL problem in turn leads to our second contribution in this paper: we show that, by considering a more restrictive notion of optimal policy, we are able to derive analytically *non-trivial, closed-form solutions* to the IRL problem. This is in contrast with current methods in the literature that typically resort to some optimization routine to tackle IRL.

Incidentally, our analysis also highlights several smaller results/facts that are of independent interest per se. Specifically,

- We briefly discuss relations between IRL and reward shaping [5];
- We show that it is possible to compute a non-trivial reward function for any optimal policy;
- We show that the previous fact does not hold if a parameterized reward model is used.
- We show that the two approaches in [4, 7] share the same solution space, and analytically describe this solution space;

Although some of the above results may seem obvious at first glance, our analysis shows that they yield subtle consequences that have not been properly explored in the literature and shed a new light on the structure of the policy space in MDPs.

2 Background

The "classical" IRL problem is formulated within the framework of Markov decision problems (MDPs). We thus start by reviewing MDPs and some related concepts, before formalizing the IRL problem in the next section.

[1] INESC-ID/IST, Portugal, email: fmelo@inesc-id.pt
[2] University of Plymouth, UK, email: manuel.lopes@plymouth.ac.uk
[3] Institute for Systems and Robotics, Portugal, email: ricardo@isr.ist.utl.pt

2.1 Markov Decision Problems

A *Markov decision problem* (MDP) describes a sequential decision problem in which an agent must choose the sequence of actions that maximizes some reward-based optimization criterion. Formally, an MDP is a tuple $\mathcal{M} = (\mathcal{X}, \mathcal{A}, \mathsf{P}, r, \gamma)$, where $\mathcal{X}$ represents the (finite) state-space, $\mathcal{A}$ represents the (finite) action-space, $\mathsf{P}(x, a, y)$ represents the transition probability from state x to state y when action a is taken and $r(x, a)$ represents the expected reward for taking action a in state x. The scalar γ is a discount factor.

We consider a *policy* as a mapping $\pi : \mathcal{X} \times \mathcal{A} \longrightarrow [0, 1]$ that assigns to each $x \in \mathcal{X}$ a distribution $\pi(x, \cdot)$ over $\mathcal{A}$. The purpose of the agent is to determine a policy π so as to maximize, for all $x \in \mathcal{X}$,

$$V^{\pi}(x) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t r\big(\mathsf{X}_t, \mathsf{A}_t\big) \mid \mathsf{X}_0 = x\right],$$

where X_t is the random variable (r.v.) representing the state at time t, A_t is the r.v. corresponding to the action taken at that time instant and is such that $\mathbb{P}\left[\mathsf{A}_t = a \mid \mathsf{X}_t = x\right] = \pi(x, a)$. We define the Q-function associated with a policy π as

$$Q^{\pi}(x, a) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t r\big(\mathsf{X}_t, \mathsf{A}_t\big) \mid \mathsf{X}_0 = x, \mathsf{A}_0 = a\right].$$

where, again, A_t is distributed according to $\pi(\mathsf{X}_t, \cdot)$ for all $t > 0$. Finally, we defined the *advantage function* associated with π as $A^{\pi}(x, a) = Q^{\pi}(x, a) - V^{\pi}(x)$.

2.2 Optimal Policies

For any finite MDP, there is at least one *optimal policy* π^* such that $V^{\pi^*}(x) \geq V^{\pi}(x)$ for any π and $x \in \mathcal{X}$. The corresponding value function, V^*, verifies the Bellman optimality equation,

$$V^*(x) = \max_{a \in \mathcal{A}} \left[r(x, a) + \gamma \sum_{y \in \mathcal{X}} \mathsf{P}(x, a, y) V^*(y)\right]. \tag{1}$$

The associated Q-function in turn verifies

$$\begin{aligned} Q^*(x, a) &= r(x, a) + \gamma \sum_{y \in \mathcal{X}} \mathsf{P}(x, a, y) V^*(y) \\ &= r(x, a) + \gamma \sum_{y \in \mathcal{X}} \mathsf{P}(x, a, y) \max_{u \in \mathcal{A}} Q^*(y, u). \end{aligned} \tag{2}$$

For any given MDP, the Bellman equation provides a two-way relation between optimal policies and optimal value functions, summarized in the following expressions:

$$V^*(x) = \mathbb{E}_{\pi^*}\left[r(x, A) + \gamma V^*(Y)\right] \tag{3a}$$

$$\operatorname{supp}(\pi^*_x) \subset \arg\max_{a \in \mathcal{A}} \mathbb{E}\left[r(x, a) + \gamma V^*(y)\right], \tag{3b}$$

where $\mathbb{E}_{\pi^*}[\cdot]$ denotes the expectation with respect to (w.r.t.) the joint distribution over the action A and next state Y induced by π^*, and $\operatorname{supp}(\pi^*_x)$ denotes the support of the distribution $\pi^*(x, \cdot)$. Given a function $Q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$, we define the *greedy action set associated with* Q at state x as

$$\mathcal{A}^Q(x) = \big\{a^* \in \mathcal{A} \mid a^* \in \arg\max_{a \in \mathcal{A}} Q(x, a)\big\}. \tag{4}$$

Using this definition, the relations in (3) become

$$V^*(x) = \sum_{a \in \mathcal{A}} \pi^*(x, a) Q^*(x, a) \qquad \operatorname{supp}(\pi^*_x) \subset \mathcal{A}^{Q^*}(x).$$

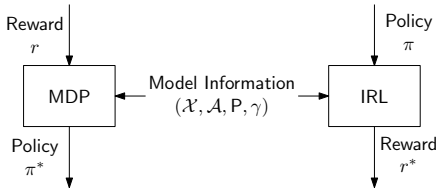

Figure 1. Block diagram representing an MDP and an IRL problem.

Finally, we define *greedy policy* associated with a function $Q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ as the policy verifying

$$\pi^Q(x, a) = \begin{cases} 1/\left|\mathcal{A}^Q(x)\right| & \text{if } a \in \mathcal{A}^Q(x) \\ 0 & \text{otherwise,} \end{cases}$$

for all $x \in \mathcal{X}$, where $\left|\mathcal{A}^Q(x)\right|$ denotes the cardinality of $\mathcal{A}^Q(x)$.

3 Inverse Reinforcement Learning

In this section we formalize the inverse reinforcement learning problem and review the results in [6]. We also provide a brief overview of the main ideas behind in [4, 7].

3.1 Inverse Bellman Equation

As seen above, an MDP represents a *decision problem* in which the task to be completed is represented by the reward function r. The optimal solution to such a task consists in a policy π^* for which both relations in (3) hold. Solving an MDP $\mathcal{M} = (\mathcal{X}, \mathcal{A}, \mathsf{P}, r, \gamma)$ thus amounts to computing one such π^* given the model of $\mathcal{M}$.

Inverse reinforcement learning (IRL) deals with the inverse problem to that of an MDP (see Fig. 1). Solving an IRL problem consists in recovering the *reward function* r given the corresponding optimal policy π^*. In other words, given a policy π^* and the model $(\mathcal{X}, \mathcal{A}, \mathsf{P}, \gamma)$, we want to compute a reward function r such that π^* is optimal for the MDP $(\mathcal{X}, \mathcal{A}, \mathsf{P}, r, \gamma)$.

From (2) and the fact that $V^*(x) = \sum_a \pi^*(x, a) Q^*(x, a)$, we get

$$Q^*(x, a) = r(x, a) + \gamma \sum_{y \in \mathcal{X}} \mathsf{P}(x, a, y) \sum_{b \in \mathcal{A}} \pi^*(y, b) Q^*(y, b).$$

Then, given a general function $Q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$, it is possible to invert the above relation for each pair (x, a), to yield

$$r(x, a) = Q^*(x, a) - \gamma \sum_{y \in \mathcal{X}} \mathsf{P}(x, a, y) \sum_{b \in \mathcal{A}} \pi^*(y, b) Q^*(y, b). \tag{5}$$

If the Bellman equation defines the optimal value-function/Q-function from the corresponding reward function, the expression above defines the reward function from its corresponding Q-function. As such, we henceforth refer to (5) as the *inverse Bellman equation*. Together, the Bellman equation and (5) define a *one-to-one* relation between reward-functions and Q-functions. In other words, given *any* Q-function Q there is a corresponding reward function r such that Q is the optimal Q-function associated with r.

3.2 Solution Characterization of Ng and Russel

We now review the main result in [6] that describes the solution space for the IRL problem, when the learner is provided with complete and error-free access to the optimal policy π^* for the MDP whose reward must be estimated. To this purpose, it is convenient to write the Bellman equation (1) in vector notation as

$$\mathbf{v}^* = \boldsymbol{\pi}^*[\mathbf{R}] + \gamma \boldsymbol{\pi}^*[\mathsf{P}]\mathbf{v}^*,$$

where $\mathbf{v}^*$ is a column vector representing V^*, $\mathbf{R}$ is a matrix representing the unknown reward r, $\boldsymbol{\pi}^*[\cdot]$ represents the expectation w.r.t. the optimal policy, π^*, and P is the transition matrix for the MDP. Concretely, $\boldsymbol{\pi}[\mathbf{R}]$ is a column vector with xth component given by $\sum_a \pi^*(x,a) r(x,a)$ and $\boldsymbol{\pi}[\mathsf{P}]$ is a matrix with (x,y) component given by $\sum_a \pi^*(x,a)\mathsf{P}(x,a,y)$. The Bellman equation now comes

$$\mathbf{v}^* = (\mathbf{I} - \gamma \boldsymbol{\pi}^*[\mathsf{P}])^{-1} \boldsymbol{\pi}^*[\mathbf{R}]. \tag{6}$$

On the other hand, we also have $\mathbf{v}^* \geq \mathbf{R}_a + \gamma \mathsf{P}_a \mathbf{v}^*$ or, equivalently

$$(\mathbf{I} - \gamma \mathsf{P}_a)\mathbf{v}^* \geq \mathbf{R}_a, \tag{7}$$

where $\mathbf{R}_a$ is the ath column of $\mathbf{R}$ and the inequalities are taken component-wise. Replacing (6) into (7) finally yields

$$(\mathbf{I} - \gamma \mathsf{P}_a)(\mathbf{I} - \gamma \boldsymbol{\pi}^*[\mathsf{P}])^{-1} \boldsymbol{\pi}^*[\mathbf{R}] \geq \mathbf{R}_a. \tag{8}$$

The result in [6] arises from considering in (8) a reward function that is only state-dependent.

The above expression provides a set of linear constraints on the set of possible reward functions that yield π^* as an optimal policy. Unfortunately, this set does not uniquely determine one such reward function for a given policy – in particular, it includes trivial solutions such as the all-zeros reward function, $r(x,a) \equiv 0$. Therefore, to solve the IRL problem, it is necessary to consider some additional selection criterion that disambiguates among all functions in the set defined by (8) and, if possible, eliminates trivial solutions. Most such criteria considered in the literature, however, are empirically motivated and lack theoretical support. It is possible to consider a stricter definition of "optimal policy" that successfully eliminates some of the ambiguity in the solution space defined by (8).

3.3 IRL as Parameter Estimation

We conclude this section by briefly reviewing the common ideas in [4, 7]. Unlike the approach in [6], these works assume that the learner is provided with samples of a perturbed version of the optimal policy associated with the desired reward (a *demonstration*). Some of these samples may not correspond to optimal actions but to "perturbations" of the optimal policy (to be detailed in the continuation), and there is no assumption on the *completeness* of the demonstration, *i.e.*, the demonstration may not include samples in all states.

Both works assume that the demonstration is generated according to a particular distribution that depends non-linearly on the reward function. This reduces the IRL problem to that of finding the reward function yielding a distribution that most closely matches the empirical distribution of the demonstration. In other words, the reward function *parameterizes* a family of distributions, and IRL consists in fitting this parameter to minimize some empirical loss function.

Both aforementioned works [4, 7] assume that the states in the demonstration are sampled uniformly in an i.i.d. fashion, and that the corresponding actions are sampled according to the distribution:

$$\mathbb{P}\left[\mathsf{A}_i = a \mid \mathsf{X}_i = x\right] = \frac{e^{\eta Q^*(x,a)}}{\sum_b e^{\eta Q^*(x,b)}}, \tag{9}$$

where X_i is the r.v. corresponding to the ith sampled state and A_i the corresponding sampled action. In the expression above, Q^* represents the optimal Q-function associated with the reward function to be estimated, and η is a confidence parameter determining the spread of the distribution around the optimal actions.

The choice of this particular distribution is motivated by several observations. First of all, it translates the intuition that, even if making mistakes, the demonstrator is more prone to choose better actions than worse ones. Secondly, as long as the optimal actions are observed more often than non-optimal actions, the above distribution will yield a reward function whose associated optimal actions will also be optimal to those of the target reward function. In particular, if all optimal actions are sampled equally often, the recovered optimal actions will match those of the target reward function, and the obtained reward function will be equivalent to the desired reward function in terms of optimal policies.[4]

We conclude by noting that in [7] this distribution is used as the *likelihood function* in a Bayesian setting. The paper proceeds by estimating the posterior distribution over possible reward functions given the demonstration using a variant of the Monte-Carlo Markov chain algorithm [3]. In [4], on the other hand, the authors adopt a gradient approach to recover the reward function that minimizes the loss w.r.t. some target policy in terms of empirical distributions.

4 Characterization of the IRL Solution Set

In this section we present the contributions of this paper. We provide an analysis of the solution space for the IRL problem when the learner has only access to imperfect/incomplete information concerning the optimal policy for the MDP whose reward is to be estimated, complementing the results in [6] reviewed in Section 3.2. Our analysis also leads to a refinement of the results in [6] when the agent has perfect information concerning the optimal policy.

We start by proposing a stricter interpretation of the IRL problem that settles some of the solution-space ambiguity issues identified in Section 3.2. We then describe the IRL solution space for different possible situations, and conclude with the discussion of several related results that follow from our analysis. In particular, we derive an analytical, closed-form solution to the IRL problem.

4.1 Restricted Optimal Policies in IRL

Let $\mathcal{M} = (\mathcal{X}, \mathcal{A}, \mathsf{P}, r, \gamma)$ be an MDP whose reward function, r, we want to estimate from the corresponding optimal policy or a perturbation thereof. The relation between the reward r and the corresponding optimal policy is, in a sense, "encoded" by the optimal Q-function associated with r, Q^*. In fact, Q^* is determined uniquely from r, and a policy pi^* is optimal if and only if, for every $x \in \mathcal{X}$,

$$\pi^*(x,a) > 0 \Rightarrow a \in \mathcal{A}^{Q^*}(x).$$

However, the above dependence of π^* on Q^* is only w.r.t. the sets $\mathcal{A}^{Q^*}$. In fact, any other function Q' for which $\mathcal{A}^{Q'}(x) = \mathcal{A}^{Q^*}(x)$ for every $x \in \mathcal{X}$ has exactly the same set of optimal policies, and the corresponding reward function

$$r'(x,a) = Q'(x,a) - \gamma \sum_{y \in \mathcal{X}} \mathsf{P}(x,a,y) \max_{b \in \mathcal{A}} Q'(y,b)$$

is *equivalent to* r in terms of optimal policies – *i.e.*, any policy that is optimal for r is optimal for r' and vice-versa. One example of one such function is the advantage function A^*, corresponding to the reward function

$$r^*(x,a) = A^*(x,a) - \gamma \sum_{y \in \mathcal{X}} \mathsf{P}(x,a,y) \max_{b \in \mathcal{A}} A^*(y,b) = A^*(x,a).$$

Indeed, we note that any two reward functions r_1 and r_2 are equivalent in terms of optimal policies if the zeros of the corresponding advantage functions match. Equivalently, two reward functions r_1 and r_2 are equivalent if the corresponding greedy action sets, $\mathcal{A}^{Q_1^*}$

[4] This notion of *equivalence* between reward functions is further explored in Section 4.

and $\mathcal{A}^{Q_2^*}$, match. We take this opportunity to note that this notion of equivalence between reward functions can be used to alleviate some of the degenerate solutions discussed in Section 3.2. Given a policy π, we will restrict our attention to those reward functions whose corresponding greedy action set *exactly matches* the support of π. Within this "stricter" formulation of IRL, we note that the trivial reward function $r(x,a) \equiv 0$ is no longer considered a solution to the IRL problem, except in the degenerate case in which all actions are *simultaneously* optimal in all states.

For our purposes, it is more convenient to reformulate the above restriction in terms of a more strict definition of "optimal policy".

Definition 1 (Optimal Policy) *The optimal policy for an MDP $\mathcal{M} = (\mathcal{X}, \mathcal{A}, \mathsf{P}, r, \gamma)$ is defined, for each state $x \in \mathcal{X}$, as the uniform distribution over the set $\mathcal{A}^{Q^*}(x)$.*

4.2 Analysis of IRL Solutions

In seeking a general description for the IRL solution set, we start by providing a characterization of the latter in terms of Q-functions. In other words, given the optimal policy for an MDP or a perturbed/incomplete version thereof, we are interested in computing the set of Q-functions for which the provided policy is optimal. Once this is achieved, we can use (5) to trivially obtain the corresponding solution set in terms of rewards. In tackling this problem, we denote by $\mathcal{Q}$ the set of *all* functions $Q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ and parameterize any such function as

$$Q(x,a) = V(x) + A(x,a), \tag{10}$$

with $V(x) = \max_{b \in \mathcal{A}} Q(x,b)$ and $A(x,a) = Q(x,a) - V(x)$. Although the discussion in Section 4.1 regarding the equivalence of rewards in terms of optimal policies already hints at some of the appealing properties of this particular parameterization, it will soon become apparent that this representation of Q-functions is indeed most useful in our analysis.

Also, following the discussion in Section 3.3, when dealing with perturbed policies for an MDP $(\mathcal{X}, \mathcal{A}, \mathsf{P}, r, \gamma)$, we adopt the general form portrayed in (9), where Q^* is taken as the optimal Q-function for the desired reward function r. As discussed in Section 3.3, this is not a very restrictive assumption since, as long as the optimal actions are observed more often than the non-optimal actions, the corresponding reward function will still yield an optimal policy for the desired policy (although possibly not in the restricted sense of Definition 1).

Given a policy π, we want to compute the subset $\mathcal{Q}_\pi \subset \mathcal{Q}$ that is *consistent* with π, meaning that any $Q \in \mathcal{Q}_\pi$ generates the given policy π according to (9). Noting that the distribution in (9) is specified in a state-wise manner, it is possible to also detail the relation between a policy π and a Q-function Q in a state-wise manner. As such, in the continuation, we consider a fixed "query" state $x_q \in \mathcal{X}$ and derive the Q-function in that state that corresponds to the provided policy π^*. In our analysis, we consider three distinct situations:

- The learner is provided with a perturbed version of the optimal policy at state x_q – corresponding to a finite value of η in (9). In this case, $\pi(x_q, a)$ is specified to the learner for all $a \in \mathcal{A}$ and belongs to the interval $(0, 1)$;
- The learner is provided with the optimal policy at state x_q – corresponding to a the situation in which $\eta \to \infty$. In this case, $\pi(x_q, a)$ is specified and is either 0 or $1/\left|\mathcal{A}^{Q^*}(x)\right|$;
- The learner receives no information about the optimal policy at state x_q. In this case, $\pi(x_q, a)$ is unspecified (free) for all $a \in \mathcal{A}$.

Resorting to the representation in (10), we now show how each of the above situations translates in terms of constraints in terms of $V(x_q)$ and $A(x_q, \cdot)$.

Perturbed Policy Observed

In this scenario, the learner is provided with a perturbed version of the optimal policy at state x_q. Then, given the probability distribution in (9) computed from the (unknown) optimal Q-function at x_q, one possible solution is given by

$$Q(x_q, a) = \frac{\ln(\pi(x_q, a))}{\eta}, \tag{11}$$

which can easily be confirmed by replacing in (9). This solution, however, is not unique, as seen from the following result.

Lemma 2 *Let $p_{xa}(Q)$ denote the probability in* (9) *at (x, a), seen as a function of Q. Given any two Q-functions, Q_1 and Q_2, $p_{xa}(Q_1) = p_{xa}(Q_2)$ if and only if $Q_1(x,a) = Q_2(x,a) + \phi(x)$ for all (x,a), where ϕ is any real valued function that is constant over actions.*

Proof: On one hand, if $Q_1(x,a) = Q_2(x,a) + \phi(x)$ for every $(x,a) \in \mathcal{X} \times \mathcal{A}$, direct substitution on (9) immediately yields $p_{xa}(Q_1) = p_{xa}(Q_2)$. On the other hand, we can write, for a general Q,

$$p_{xa}(Q) = \frac{e^{\eta Q(x,a)}}{e^{-\phi(x)}} = e^{\eta Q(x,a) + \phi(x)},$$

where $\phi(x) = -\ln(\sum_b e^{\eta Q(x,b)})$. If $p_{xa}(Q_1) = p_{xa}(Q_2)$, then

$$e^{\eta Q_1(x,a) + \phi_1(x)} = e^{\eta Q_2(x,a) + \phi_2(x)},$$

and the result immediately follows. □

It follows from Lemma 2 that, at state $x_q \in \mathcal{X}$, the solution in (11) is unique up to an additive term. Using our previous parameterization we have, for every $Q \in \mathcal{Q}_\pi$,

$$A(x_q, a) = \frac{1}{\eta}\Big[\ln(\pi(x_q, a)) - \max_b \ln(\pi(x_q, b))\Big].$$

and V^* is arbitrary. In this case, $Q(x_q, \cdot)$ is defined up to a single "degree of freedom" arising from the value of $V(x_q)$.

Optimal Policy Observed

In this case, the learner is provided with the optimal policy at state x_q. In the case of an optimal policy, (9) degenerates as $\eta \to \infty$ and the policy $\pi(x_q, a)$ is non-zero only in those entries where $Q(x_q, a) = \max_b Q(x_q, b)$. This implies that for a given policy π, any function Q that contains the maximizing actions at x_q in the same positions as the non-zero entries of $\pi(x_q, \cdot)$ is consistent with this policy. In terms of our representation, this means that, for every $Q \in \mathcal{Q}_\pi$, $A(x_q, a) = 0$ if $\pi(x_q, a) > 0$ and $A(x_q, a) < 0$ (but otherwise arbitrary) if $\pi(x_q, a) = 0$, and V is arbitrary. We now have several degrees of freedom arising from the value of $V(x_q)$ and the negative components of $A(x_q, \cdot)$.

Policy Unobserved

In this case, no constraints apply and $Q(x_q, a)$ can be arbitrary. It can be written as in (10) with several degrees of freedom arising both from the value of $V(x_q)$ and the components of $A(x_q, \cdot)$, now constrained only to be non-positive and to have at least one zero element (there is always at at least one optimal action per state).

Now given the solution set $\mathcal{Q}_\pi$ associated with the given policy π^*, we can apply (5) to obtain the corresponding set in reward space. In particular, for each $Q \in \mathcal{Q}_\pi$, we have for all $a \in \mathcal{A}$,

$$r(x,a) = V(x) - \gamma \sum_{a \in \mathcal{A}} \mathsf{P}(x,a,y)V(y) + A(x,a), \qquad (12)$$

where V and A are as in (10). It is worth noting at this point that the optimal policy associated with r is solely defined by the component A, in the sense that changes to V have no effect on the corresponding policy. In fact, this holds both for the unperturbed and the perturbed cases, as seen in Lemma 2. Also, from the analysis above, it is always possible to build a Q-function from a given policy (perturbed or not) from which a reward function can, in turn, be computed. This means that it is always possible to compute a non-trivial reward function for any optimal policy.

4.3 Ng and Russel Revisited

We now revisit the result in [6] in light of the results in the previous subsection. In particular, we show that our results are in accordance with those derived in [6]. This is summarized in the following result.

Theorem 3 *Given an optimal policy*[5] π*, let* $\mathcal{R}_\pi$ *be the reward space described in Section 4.2 and* $\hat{\mathcal{R}}_\pi$ *be the set of reward functions verifying* (8)*. Then* $\hat{\mathcal{R}}_\pi = \mathrm{cl}(\mathcal{R}_\pi)$*, where* $\mathrm{cl}(\cdot)$ *denotes set closure.*

Proof: In the proof we adopt the vector notation from Section 3.2. We start by showing that $\mathcal{R}_\pi \subset \hat{\mathcal{R}}_\pi$. From Section 4.2, $\mathbf{R}_a = (\mathbf{I} - \gamma \mathsf{P}_a)\mathbf{v} + \mathbf{A}_a$, where $\mathbf{v}$ is a vector corresponding to the arbitrary function V and $\mathbf{A}_a$ denotes column a of matrix $\mathbf{A}$, corresponding to the function A. By definition, $\mathbf{A} \leq 0$ (component-wise) and $\boldsymbol{\pi}[\mathbf{A}] = \mathbf{0}$. Replacing in (8), yields

$$(\mathbf{I} - \gamma \mathsf{P}_a)\mathbf{v} \geq (\mathbf{I} - \gamma \mathsf{P}_a)\mathbf{v} + \mathbf{A}_a,$$

which trivially holds. This means that $\mathcal{R}_\pi \subset \hat{\mathcal{R}}_\pi$. It remains to show that $\hat{\mathcal{R}}_\pi \subset \mathrm{cl}(\mathcal{R}_\pi)$. From the Bellman equation, given a reward function r and the corresponding optimal policy π, it holds that

$$\boldsymbol{\pi}[\mathbf{R}] = (\mathbf{I} - \gamma \boldsymbol{\pi}[\mathsf{P}])\mathbf{v}, \qquad (13)$$

for some vector $\mathbf{v}$. Defining $\mathbf{u} = (\mathbf{I} - \gamma \boldsymbol{\pi}[\mathsf{P}])^{-1}\boldsymbol{\pi}[\mathbf{R}]$, (8) becomes, for each $a \in \mathcal{A}$,

$$\big(\mathbf{I} - \gamma \mathsf{P}_a\big)\mathbf{u} = \mathbf{R}_a - \mathbf{Z}_a,$$

for some non-positive slack matrix $\mathbf{Z}$. Applying π to the expression above for all a yields

$$\boldsymbol{\pi}[\mathbf{R}] = (\mathbf{I} - \gamma \boldsymbol{\pi}[\mathsf{P}])\mathbf{u} + \boldsymbol{\pi}[\mathbf{Z}].$$

From (13), $\boldsymbol{\pi}[\mathbf{Z}]$ must be of the form $(\mathbf{I} - \gamma \boldsymbol{\pi}[\mathsf{P}])\mathbf{u}'$, implying that $\mathbf{Z}_a = (\mathbf{I} - \gamma \mathsf{P}_a)\mathbf{u}' + \mathbf{A}_a$, for some matrix $\mathbf{A}$ such that $\boldsymbol{\pi}[\mathbf{A}] = 0$. Putting everything together, we have

$$\mathbf{R}_a = \big(\mathbf{I} - \gamma \mathsf{P}_a\big)(\mathbf{u} + \mathbf{u}') + \mathbf{A}_a,$$

and the result follows. □

It is also interesting to consider the parallel between Lemma 2 and some of the results in [5]. The analysis of *reward shaping* in [5] essentially concludes a similar set of invariances in terms of the optimal policy as those identified in Lemma 2. To see this, note that the functions ϕ in Lemma 2 correspond to *shaping potential* in [5]. In turn, these potentials affect only the value of V in the parameterization (10), which do not affect the corresponding optimal policy.

[5] Here, optimal policy is taken in the sense of Definition 1.

4.4 Parameter Estimation Approach Revisited

Our work shares with the approaches in [4, 7] the assumption that the policy provided to the agent is generated from some unknown reward function according to the distribution in (9). In the aforementioned works, the learner is then provided with a demonstration consisting of a set $\mathcal{D} = \{(x_i, a_i), i = 1, \ldots, N\}$ of state-action pairs, where the states are sampled in an i.i.d. manner and the corresponding actions sampled according to (9). Both works then propose a loss function and a method to compute the reward function that minimizes it. In their essence both methods seek to approximate – within the same parameterized family of distributions – the empirical distribution of the data.

Therefore, it is no surprise that the maximum likelihood solution in the formulation of [7] matches the distribution that minimizes the loss in [4] and this is, in turn, the solution considered in Section 4.2. Computing the empirical distribution of the data at each state x, $\hat{\pi}(x, \cdot)$, we simply set

$$Q(x,a) = \frac{\ln(\hat{\pi}(x,a))}{\eta} + V(x), \qquad (14)$$

for some arbitrary V. The expression (14) constitutes a closed-form solution for the problems addressed therein. In other words, the solutions described in our paper are (global) maximizers of both the maximum likelihood criterion in the formulation of [7] and the criterion considered in [4].

In a more general setting, we may have situations in which the leaner is provided the optimal policy at some states, a perturbed policy in other states, and no policy at all in the remaining states. Using the results in Section 4.2 we can immediately compute from this policy information one possible Q-function that is compatible with the provided policy, from which a reward can be extracted trivially using (5). In Section 5 we illustrate this process with a simple example and discuss further use for our results.

5 A Simple Example

In this section we present a simple example in which we use our results within a broader estimation setting to compute analytically in closed-form the solution to an IRL problem.

We consider the general architecture depicted in the diagram of Fig. 2. In this diagram, the learner is provided with a demonstration consisting of a set $\mathcal{D} = \{(x_i, a_i), i = 1, \ldots, N\}$ generated as discussed in Section 4.2. This demonstration is combined with some prior information on the policy to yield a representative estimate policy $\hat{\pi}$. From our results from Section 4.2, we can use this estimate to compute the corresponding set of Q-functions, $\mathcal{Q}_{\hat{\pi}}$. Using (5), we can compute the corresponding set of reward functions, $\mathcal{R}_{\hat{\pi}}$, from which an individual reward can be selected according to some criterion. Note that this specific reward selection is a problem outside the IRL problem, since the selection of one particular reward function from $\mathcal{R}_{\hat{\pi}}$ implies absolutely no change on the corresponding policy.

The prior information included in the first block of Fig. 2 is particularly useful when the number of samples is very small since, to some extent, it makes up for insufficient samples. Many different priors can be used and the best one must be judged according to the specific problem at hand.

To illustrate our application, consider the simple 4-state scenario in Fig. 3, where the target reward assigns the agent with a reward of $+1$ whenever it reaches the shaded cell. The agent has 5 actions available at each state, 4 of which move it in one of the four directions, and the fifth corresponding to the "NoOp" action. For the MDP thus obtained (considering the target reward function) we computed

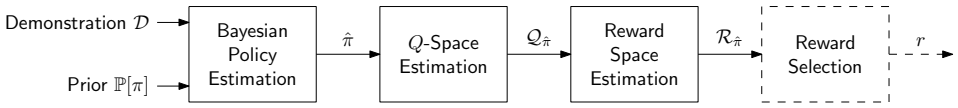

Figure 2. Overview of the general approach to the IRL problem considered in this paper.

(a) Actual sampled distribution. (b) Estimated policy.

Figure 3. Results of the policy estimation step.

the optimal Q-function and provided the learner with a demonstration consisting of 20 random state-action pairs sampled according to the distribution in (9), with $\eta = 2$. This corresponds to the distribution depicted in Fig. 3(a), where in each state the height of the bars is proportional to the corresponding sampling probability. From the demonstration, we computed the policy corresponding to the maximum a posteriori, given a uniform prior for the parameters of the policy. Notice that, since we have such a small demonstration, it is only natural that the estimates for the policy are not too precise. For comparison, we depicted these in Fig. 3(b).

From the estimated policy, $\hat{\pi}$, we immediately compute the set of Q-functions associated with the estimated policy $\hat{\pi}$ as the set of all Q-functions verifying (14), for an arbitrary real-valued function V. Finally, we compute the corresponding reward solution space from the above expression using (5). In our case, we computed one representative reward function obtained by setting $V = 0$ in (14) and the corresponding optimal policy, which matched the optimal for the original reward function, as expected. Our closed-form solution does not require running time-consuming optimization routines to output a solution for the IRL problem. In this aspect, our analysis is also distinct from that in [6], in that it is amenable to such straightforward computation.

6 Concluding Remarks

We conclude with several remarks concerning the general applicability of the results in this paper. First of all, our results feature discrete state and action spaces. While the ideas should carry without change to more general settings, the associated computations are not amenable to a straightforward generalization. Considering, for example, an MDP with an infinite state-space implies that the corresponding transition probabilities cannot be explicitly represented and, hence, expressions such as (5) cannot easily be generalized.

Another very important aspect to take into account is the fact that we are dealing with general reward functions that depend on state and action. In this setting, as seen by our results, it is always possible to recover a non-degenerate reward function that yields any given policy as optimal. However, this fact is not generally true if we consider more restricted classes of reward functions. For example, when considering a reward function r that depend only on x, it may happen that no solution exists for (5). When this is the case, no exact solution exists for the IRL problem and, therefore, some form of approximation must be adopted. In such situation, the approaches in [4, 7] appear naturally.

We conclude with two observations. First, it follows from the results in Section 4.2 that, for a given policy π, there is one reward function for which the policy π is optimal *independently of the particular dynamics of the problem*. This reward is obtained by setting $V(x)$ to zero in (12). In this case, the corresponding value function is identically zero and $r(x, a) = A(x, a)$. This corresponds to the "ideal reward" situation discussed in [5], in which precisely the shaping potential is chosen so as to ensure that $r(x, a) = Q(x, a)$.

Secondly, our results clearly show that the degrees of freedom in the solution set $\mathcal{R}_\pi$ arise from

1. The unspecified components of π. These are associated with the "free" entries of A in (12).
2. The invariance of π described in Lemma 2. These are associated with V in (12).

In choosing one particular reward function from the set $\mathcal{R}_\pi$ (corresponding to the dashed block in Fig. 2), we argue that these two "types" of degrees of freedom should be dealt with differently. Concerning those in 1, a particular choice of A determines how the agent should act in those states not specified by the demonstration. A criterion to choose among the possible A basically determines what the policy of the agent should be "by default". Concerning those in 2, these don't affect the policy. Therefore, a particular choice of $\mathbf{v}$ simply determines a particular form for $\mathbf{R}$, without affecting the corresponding optimal policy. In a sense, this is precisely the problem of reward shaping [5].

Acknowledgements

The authors acknowledge the useful comments by the anonymous reviewers. This work was supported by the Portuguese Fundaccao para a Ciência e a Tecnologia (INESC-ID and ISR multiannual funding) through the PIDDAC Program funds. M. Lopes was also partially supported by the PTDC/EEA-ACR/70174/2006 project and the EU Project Handle (EU-FP7-ICT-231640).

References

[1] P. Abbeel, *Apprenticeship learning and reinforcement learning with application to robotic control*, Ph.D. thesis, Dep. Computer Science, Stanford Univ., 2008.
[2] P. Abbeel and A. Ng, 'Apprenticeship learning via IRL', in *Int. Conf. Machine Learning*, pp. 1–8, (2004).
[3] C. Andrieu, N. Freitas, A. Doucet, and M. Jordan, 'An introduction to MCMC for machine learning', in *Machine Learning*, vol. 50, pp. 5–43, (2003).
[4] G. Neu and C. Szepesvári, 'Apprenticeship learning using IRL and gradient methods', in *Conf. Uncertainty in Artificial Intelligence*, pp. 295–302, (2007).
[5] A. Ng, D. Harada, and S. Russel, 'Policy invariance under reward transformations: Theory and application to reward shaping', in *Int. Conf. Machine Learning*, pp. 278–287, (1999).
[6] A. Ng and S. Russel, 'Algorithms for IRL', in *Int. Conf. Machine Learning*, pp. 663–670, (2000).
[7] D. Ramachandran and E. Amir, 'Bayesian IRL', in *Int. Joint Conf. Artificial Intelligence*, pp. 2586–2591, (2007).
[8] S. Russell, 'Learning agents for uncertain environments (extended abstract)', in *Adv. Neural Information Proc. Systems*, vol. 10, (1998).
[9] V. Silva, A. Costa, and P. Lima, 'IRL with evaluation', in *IEEE Int. Conf. Robotics and Automation*, pp. 4246–4251, (2006).
[10] U. Syed and R. Schapire, 'A game-theoretic approach to apprenticeship learning', in *Adv. Neural Information Proc. Systems*, vol. 20, pp. 1449–1456, (2008).
[11] U. Syed, R. Schapire, and M. Bowling, 'Apprenticeship learning using linear programming', in *Int. Conf. Machine Learning*, pp. 1032–1039, (2008).