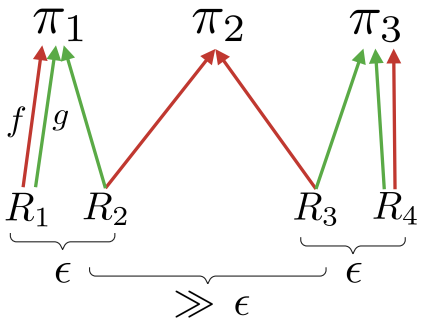
A MOTIVATING OUR DEFINITION OF MISSPECIFICATION ROBUSTNESS

In this section, we provide further discussion and motivation for our formalisation of misspecification robustness, given in Definition 1, beyond the discussion we give in Section 2.1.

A.1 ADDITIONAL COMMENTS ON THE CONDITIONS FOR MISSPECIFICATION ROBUSTNESS

In this section, we make a few additional comments on some of the four conditions in Definition 1. In particular, while the first condition ought to be reasonably clear, we have further comments on each of the remaining three conditions.

Condition 2 says that for all $R_1, R_2 \in \hat{\mathcal{R}}$, if $f(R_1) = f(R_2)$ then $d^{\mathcal{R}}(R_1, R_2) \leq \epsilon$. In other words, any learning algorithm $\mathcal{L}$ based on f is guaranteed to learn a reward function that has a distance of at most ϵ to the true reward function when trained on data generated by f, i.e. when there is no misspecification. It may not be immediately obvious why this assumption is included, since we assume that the data is generated by g, where $f \neq g$. To see this, suppose $\hat{\mathcal{R}} = \{R_1, R_2, R_3, R_4\}$ where $d^{\mathcal{R}}(R_1, R_2) < \epsilon$, $d^{\mathcal{R}}(R_3, R_4) < \epsilon$, and $d^{\mathcal{R}}(R_2, R_3) \gg \epsilon$, and let $f, g : \hat{\mathcal{R}} \to \Pi$ be two behavioural models where $f(R_1) = \pi_1$, $f(R_2) = f(R_3) = \pi_2$, $f(R_4) = \pi_3$, and $g(R_1) = g(R_2) = \pi_1$, $g(R_3) = g(R_4) = \pi_3$. This is illustrated in the diagram below:



In this case, we have that $f(R_2) = f(R_3)$, but $d^{\mathcal{R}}(R_2, R_3) \gg \epsilon$. As such, f violates condition 2 in Definition 1; a learning algorithm $\mathcal{L}$ based on f is *not* guaranteed to learn a reward function that has distance at most ϵ to the true reward function when there is no misspecification, because f cannot distinguish between R_2 and R_3, which have a large distance. However, if $f(R) = g(R')$, it does in this case follow that $d^{\mathcal{R}}(R, R') \leq \epsilon$. In other words, if the training data is coming from g, then a learning algorithm $\mathcal{L}$ based on f *is* guaranteed to learn a reward function that has distance at most ϵ to the true reward function. As such, we could define misspecification robustness in such a way that f would be considered to be robust to misspecification with g in this case. However, this seems unsatisfactory, because g essentially has to be carefully designed specifically to avoid certain blind spots in f. In other words, while condition 1 in Definition 1 is met, it is only met *spuriously*. To rule out these kinds of edge cases, we have therefore included the condition that for all $R_1, R_2 \in \hat{\mathcal{R}}$, if $f(R_1) = f(R_2)$, then it must be that $d^{\mathcal{R}}(R_1, R_2) \leq \epsilon$.

Condition 3 says that there for all $R_1 \in \hat{\mathcal{R}}$ exists an $R_2 \in \hat{\mathcal{R}}$ such that $f(R_2) = g(R_1)$. Stated differently, the image of g on $\hat{\mathcal{R}}$ is a subset of the image of f on $\hat{\mathcal{R}}$. The reason for why this assumption is necessary is to ensure that the learning algorithm can never observer data that is impossible according to its assumed model. For example, suppose f maps each reward function to a deterministic policy; in that case, the learning algorithm $\mathcal{L}$ will assume that the observed policy must be deterministic. What happens if such an algorithm is given data from a nondeterministic policy? This is undefined, absent further details about $\mathcal{L}$, because $\mathcal{L}$ cannot possibly find a reward function that fits the training data under its assumed model. Since we do not want to make any strong assumptions about $\mathcal{L}$, it therefore seems reasonable to say that if f always produces a deterministic policy, and g sometimes produces nondeterministic policies, then f is not robust to misspecification with g. More generally, it has to be the case that any policy that could be produced by g, can be

explained under f. This is encompassed by the condition that there for all $R_1 \in \hat{\mathcal{R}}$ exists an $R_2 \in \hat{\mathcal{R}}$ such that $f(R_2) = g(R_1)$. Of course, in many cases we may have that $\text{Im}(f) = \Pi$, i.e. that f can produce any policy, and in that case this condition is vacuous.

Condition 4 says that there exists $R_1, R_2 \in \hat{\mathcal{R}}$ such that $f(R_1) \neq f(R_2)$; in other words, that $f \neq g$ on $\hat{\mathcal{R}}$. This condition is not strictly necessary – from a mathematical standpoint, very little would change if we were to simply remove it from Definition 1. Indeed, the only effect that this condition has on the results in this paper is that Theorem 1 and Corollary 1 add the condition that $f \neq g$, and that $f \neq g$ is part of the definition of δ-perturbations (Definition 3). Rather, the reason for including the assumption that $f \neq g$ is purely to make Definition 1 more intuitive. If $f = g$, then f is not misspecified, and it would seem odd to say that "f is ϵ-robust to misspecification with itself". As such, there is no deeper significance to this condition besides making our terminology more clear.

A.2 On the Assumption That Behavioural Models Are Functions

Here, we will comment on the fact that behavioural models are assumed to be *functions*; i.e., we assume that a behavioural model associate each reward function R with a unique policy π. This is true for the Boltzmann-rational model and the maximal causal entropy model, but it may not be a natural assumption in all cases. For example, there may in general be more than one optimal policy. Thus, an optimal agent could associate some reward functions R with multiple policies π. This particular example is not too problematic, because the set of all optimal policies still form a convex set. As such, it is natural to assume that an optimal agent would take all optimal actions with equal probability, which is what we have done in the definition of $o_{\tau,\gamma}$.[7] However, we could imagine alternative criteria which would associate some rewards with multiple policies, and where there may not be any canonical way to select a single policy among them. Such criteria may then not straightforwardly translate into a *functional* behavioural model.

There are several ways to handle such cases within our framework. First of all, we may simply assume that the observed agent still has some fixed method for breaking ties between policies that it considers to be equivalent (as we do for $o_{\tau,\gamma}$). In that case, we still ultimately end up with a function from $\mathcal{R}$ to Π, in which case our framework can be applied without modification. We expect this approach to be satisfactory in most cases.

It is worth noting that this approach does not necessarily require us to actually know how the observed agent breaks ties between equivalent policies. To see this, let $G : \mathcal{R} \to \mathcal{P}(\Pi)$ be a function that associates each reward function with a set of policies. We can then say that a behavioural model $g : \mathcal{R} \to \Pi$ *implements* G if $g(R) \in G(R)$ for all $R \in \mathcal{R}$. Using this definition, we could then say that $f : \mathcal{R} \to \Pi$ is robust to misspecification with $G : \mathcal{R} \to \mathcal{P}(\Pi)$ if f is robust to misspecification with each g that implements G, where f being robust to misspecification with g is defined as in Definition 1. In other words, we assume that the observed agent has a fixed method for breaking ties between policies in G, but without making any assumptions about what this method is. Using that definition, our framework can then be applied without modification.

An alternative approach could be to generalise the definition of behavioural models to allow them to return a set of policies, i.e. $f : \mathcal{R} \to \mathcal{P}(\Pi)$. Most of our results can be extended to cover this case in a mostly straightforward manner. However, this approach is somewhat unsatisfactory, because we would then assume that the learning algorithm $\mathcal{L}$ gets to observe all policies in the set $f(R^\star)$. However, in reality, it seems more realistic to assume that $\mathcal{L}$ only gets to observe a single element of $f(R^\star)$, unless perhaps $\mathcal{L}$ gets data from multiple similar agents acting in the same environment.

A.3 On Restricted Spaces of Reward Functions

Our definitions are given relative to a set of reward functions $\hat{\mathcal{R}}$, which in general may be any subset of $\mathcal{R}$. It may not be immediately obvious why this is necessary, and so we will say a few words about that issue here.

[7] Note also that this is equivalent to assuming that an optimal agent would take all optimal actions with *positive* probability, but that the exact probability that it associates with each action does not convey any further information about R.

First of all, we should note that we always allow $\hat{\mathcal{R}} = \mathcal{R}$. This means that the introduction of $\hat{\mathcal{R}}$ makes our analysis strictly more general, in the sense that we always can assume that $\hat{\mathcal{R}}$ is unrestricted. In other words, nothing is lost by giving our definitions and theorem statements relative to a set of reward functions $\hat{\mathcal{R}}$, instead of the set of all reward functions.

Moreover, there are many cases where it is interesting to restrict $\mathcal{R}$. To start with, we use reward functions with type signature $\mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$, but it is quite common to use reward functions with a different type signature, such as for example $\mathcal{S} \times \mathcal{A} \to \mathbb{R}$ or $\mathcal{S} \to \mathbb{R}$. We can ensure that our analysis covers these settings as well, by noting that we can allow $\hat{\mathcal{R}}$ to be equal to $\{R \in \mathcal{R} \mid \forall s, a, s_1, s_2 : R(s, a, s_1) = R(s, a, s_2)\}$, or $\{R \in \mathcal{R} \mid \forall s, a_1, a_2, s_1, s_2 : R(s, a_1, s_1) = R(s, a_2, s_2)\}$, and so on. As such, by using a (potentially restricted) set of rewards $\hat{\mathcal{R}}$, we can make sure that our results do not depend on these design choices.

Additionally, there are many cases where we may have *prior information* about the underlying true reward function $R^\star$, over and above the information provided by the observed policy. For example, we may know that the reward function cannot depend on certain features of the environment, or we may know that it only depends on the state of the environment at the end of an episode, and so on. This information may come from expert knowledge, or from auxiliary data sources, etc. In these cases, it makes sense to restrict $\hat{\mathcal{R}}$ to the set of all reward functions that are viable in light of this prior knowledge. Moreover, restricted reward sets also allow us to handle the case where this information is given in the form of a Bayesian prior, see Appendix A.4.

Another reason for restricting $\mathcal{R}$ is that Definition 1 is existential, in the sense that a single counterexample in principle is enough to prevent f from being ϵ-robust to misspecification with g, even if $f(R_1) = g(R_2)$ implies $d^{\mathcal{R}}(R_1, R_2) \leq \epsilon$ for "most" R_1 and R_2, etc. As such, even if f is not ϵ-robust to misspecification with g, it could in theory still be the case that a learning algorithm $\mathcal{L}$ based on f is guaranteed to learn a reward function R_h that is close to the true reward function $R^\star$ for most choices of $R^\star$. We can rule out this possibility by restricting $\hat{\mathcal{R}}$ in a way that excludes gerrymandered counter-examples.

As such, by giving our definitions relative to a set of reward functions $\hat{\mathcal{R}}$, we make our analysis more versatile and more general.

A.4 On Making the Analysis More Probabilistic

The formalisation of misspecification robustness in Definition 1 is essentially a worst-case analysis, in the sense that it requires each condition to hold for *all* reward functions. For example, a single pair of rewards R_1, R_2 with $f(R_1) = g(R_2)$ and $d^{\mathcal{R}}(R_1, R_2) > \epsilon$ is enough to make it so that f is not ϵ-robust to misspecification with g, even if $f(R_1) = g(R_2)$ implies $d^{\mathcal{R}}(R_1, R_2) \leq \epsilon$ for "most" reward functions. This makes sense if we do not want to make any assumptions about the true reward function $R^\star$, or about the inductive bias of the learning algorithm. However, in certain cases, we may know that $R^\star$ is sampled from a particular distribution $\mathcal{D}$ over $\mathcal{R}$. In those cases, it may be more relevant to know whether $d^{\mathcal{R}}(R^\star, R_h) \leq \epsilon$ with high probability.

To make this more formal, we may assume that we have two behavioural models $f, g : \mathcal{R} \to \Pi$ and a distribution $\mathcal{D}$ over $\mathcal{R}$, that $R^\star$ is sampled from $\mathcal{D}$, and that the learning algorithm $\mathcal{L}$ observes the policy $\pi = g(R^\star)$. We then assume that $\mathcal{L}$ returns the reward function R_h such that $f(R_h) = \pi$, and that $\mathcal{L}$ selects among all such reward functions using some (potentially nondeterministic) inductive bias. We then want to know if $d^{\mathcal{R}}(R^\star, R_h) \leq \epsilon$ with probability at least $1 - \delta$, for some δ and ϵ.

Our framework can, to an extent, be used to study this setting as well. In particular, suppose we pick a set $\hat{\mathcal{R}}$ of "likely" reward functions such that $\mathbb{P}_{R \sim \mathcal{D}}(R \in \hat{\mathcal{R}}) \geq 1 - \delta$, and such that the learning algorithm $\mathcal{L}$ will return a reward function $R_h \in \hat{\mathcal{R}}$ if there exists a reward function $R_h \in \hat{\mathcal{R}}$ such that $f(R_h) = g(R^\star)$. Then if f is ϵ-robust to misspecification with g on $\hat{\mathcal{R}}$, we have that $\mathcal{L}$ will learn a reward function R_h such that $d^{\mathcal{R}}(R^\star, R_h) \leq \epsilon$ with probability at least $1 - \delta$.

So, for example, suppose $\hat{\mathcal{R}}$ is the set of all reward functions that have "low complexity", for some complexity measure and complexity threshold. The above argument then informally tells us that if the true reward function is likely to have low complexity, and if $\mathcal{L}$ will attempt to fit a low-complexity reward function to its training data, then the learnt reward function will be close to the true reward

function with high probability, as long as f is ϵ-robust to misspecification with g on the set of all low-complexity reward functions.

Thus, while Definition 1 gives us a worst-case formalisation of misspecification robustness, it is relatively straightforward to carry out a more probabilistic analysis within the same framework.

B Explaining and Motivating STARC-Metrics

In this section, we will explain Definition 2, and provide the theoretical justification for measuring the difference between reward functions using $d^{\mathrm{STARC}}_{\tau,\gamma}$.

Let us first walk through the definition of $d^{\mathrm{STARC}}_{\tau,\gamma}$, and explain each of the steps. Intuitively speaking, we want to consider R_1 and R_2 to be equivalent if (and only if) they induce the same ordering of policies. Moreover, also recall that R_1 and R_2 have the same ordering of policies if and only if they differ by potential shaping, S'-redistribution, and positive linear scaling (see Proposition 1). For this reason, $d^{\mathrm{STARC}}_{\tau,\gamma}$ first *standardises* each reward function in a way that maps all equivalent rewards to a single representative in their respective equivalence class, before measuring their difference.

To do this, we first use $c^{\mathrm{STARC}}_{\tau,\gamma}$ to map all rewards that differ by potential shaping and S'-redistribution to a single representative. Note that for all R, the set of all rewards that differ from R by potential shaping and S'-redistribution forms an affine subspace. This means that there is a well-defined "smallest" element of each such equivalence class, which is the reward function that $c^{\mathrm{STARC}}_{\tau,\gamma}$ returns. It is also worth noting that $c^{\mathrm{STARC}}_{\tau,\gamma}$ is an orthogonal linear transformation, that maps $\mathcal{R}$ to an $|\mathcal{S}|(|\mathcal{A}|-1)$-dimensional linear subspace of $\mathcal{R}$.

After this, we *normalise* the resulting reward functions, by dividing them by their ℓ^2-norm. This collapses positive linear scaling, which now means that $s^{\mathrm{STARC}}_{\tau,\gamma}(R_1) = s^{\mathrm{STARC}}_{\tau,\gamma}(R_2)$ if and only if R_1 and R_2 have the same ordering of policies. We then measure the distance between the resulting reward functions, and multiply this distance by 0.5 to ensure that the resulting value is between 0 and 1. For more details, see Skalse et al. (2023).

To get an intuitive sense of how $d^{\mathrm{STARC}}_{\tau,\gamma}$ behaves, first note that $d^{\mathrm{STARC}}_{\tau,\gamma}$ is a pseudometric on $\mathcal{R}$. Moreover, as we have already alluded to, $d^{\mathrm{STARC}}_{\tau,\gamma}(R_1, R_2) = 0$ if and only if R_1 and R_2 induce the same ordering of policies under τ and γ. In addition to this, we also have that $d^{\mathrm{STARC}}_{\tau,\gamma}(R_1, R_2) = 1$ if and only if R_1 and R_2 induce the *opposite* ordering of policies under τ and γ. Furthermore, if R_0 is trivial and R is non-trivial, then we have that $d^{\mathrm{STARC}}_{\tau,\gamma}(R, R_0) = 0.5$. More generally, if R_1 and R_2 are approximately orthogonal, then $d^{\mathrm{STARC}}_{\tau,\gamma}(R_1, R_2) \approx 0.5$. As such, $d^{\mathrm{STARC}}_{\tau,\gamma}$ gives each pair of reward functions R_1, R_2 a distance between 0 and 1, where a distance close to 0 means that R_1 and R_2 have approximately the same policy order, a distance close to 1 means that they have approximately the opposite policy order, and a distance close to 0.5 means that they are approximately orthogonal. Almost all reward functions have a distance close to 0.5.

In addition to this, $d^{\mathrm{STARC}}_{\tau,\gamma}$ induces an upper bound on worst-case regret. Specifically:

Definition 6. A pseudometric d on $\mathcal{R}$ is *sound* if there exists a positive constant U, such that for any reward functions R_1 and R_2, if two policies π_1 and π_2 satisfy that $J_2(\pi_2) \geq J_2(\pi_1)$, then

$$J_1(\pi_1) - J_1(\pi_2) \leq U \cdot (\max_{\pi} J_1(\pi) - \min_{\pi} J_1(\pi)) \cdot d(R_1, R_2).$$

Proposition 5. *$d^{\mathrm{STARC}}_{\tau,\gamma}$ is sound.*

For a proof of Proposition 5, see Skalse et al. (2023). Before moving on, let us briefly unpack Definition 6. $J_1(\pi_1) - J_1(\pi_2)$ is the regret, as measured by R_1, of using policy π_2 instead of π_1. Division by $\max_\pi J_1(\pi) - \min_\pi J_1(\pi)$ normalises this quantity to lie between 0 and 1 (though the term is put on the right-hand side of the inequality, instead of being used as a denominator, in order to avoid division by zero when R_1 is trivial. The condition that $J_2(\pi_2) \geq J_2(\pi_1)$ says that R_2 prefers π_2 over π_1. Taken together, this means that a pseudometric d on $\mathcal{R}$ is sound if $d(R_1, R_2)$ gives an upper bound on the (normalised) maximal regret that could be incurred under R_1 if an arbitrary policy π_1 is optimised to another policy π_2 according to R_2. Note that we, as a special case, may assume that π_1 is optimal under R_1, and that π_2 is optimal under R_2. Since $d^{\mathrm{STARC}}_{\tau,\gamma}$ is sound, it induces such a bound.

In addition to this, $d^{\mathrm{STARC}}_{\tau,\gamma}$ also induces a *lower* bound on worst-case regret. It may not be immediately obvious why this property is desirable. To see why this is the case, note that if a pseudometric d on $\mathcal{R}$ does not induce a lower bound on worst-case regret, then there are reward functions that have a low regret, but large distance under d. This would in turn mean that d is not tight, and that it should be possible to find a better way to measure the distance between reward functions. If a pseudometric induces a lower bound on regret, then these kinds of cases are ruled out. When a pseudometric has this property, we say that it is *complete*:

Definition 7. A pseudometric d on $\mathcal{R}$ is *complete* if there exists a positive constant L, such that for any reward functions R_1 and R_2, there exists two policies π_1 and π_2 such that $J_2(\pi_2) \geq J_2(\pi_1)$ and

$$J_1(\pi_1) - J_1(\pi_2) \geq L \cdot (\max_{\pi} J_1(\pi) - \min_{\pi} J_1(\pi)) \cdot d(R_1, R_2),$$

and moreover, if R_1 and R_2 have the same policy order then $d(R_1, R_2) = 0$.

Proposition 6. *$d^{\mathrm{STARC}}_{\tau,\gamma}$ is complete.*

Note that if R_1 and R_2 have the same policy order and $\max_\pi J_1(\pi) - \min_\pi J_1(\pi) > 0$, then $d(R_1, R_2) = 0$; the last condition ensures that this also holds when $\max_\pi J_1(\pi) - \min_\pi J_1(\pi) = 0$. Intuitively, if d is sound, then a small d is *sufficient* for low regret, and if d is complete, then a small d is *necessary* for low regret. Soundness implies the absence of false positives, and completeness the absence of false negatives. As per Proposition 6, we have that $d^{\mathrm{STARC}}_{\tau,\gamma}$ is complete, and hence tight. For a proof, see Skalse et al. (2023).

Moreover, if a pseudometric is both sound and complete, then this implies that it, in a certain sense, is unique. Specifically:

Proposition 7. *If two pseudometrics d_1, d_2 on $\mathcal{R}$ are both sound and complete, then d_1 and d_2 are bilipschitz equivalent.*

For a proof, see Skalse et al. (2023). Note that this means that any pseudometric on $\mathcal{R}$ that is both sound and complete must be bilipschitz equivalent to $d^{\mathrm{STARC}}_{\tau,\gamma}$. As such, $d^{\mathrm{STARC}}_{\tau,\gamma}$ is a canonical pseudometric on $\mathcal{R}$, in the sense that a small $d^{\mathrm{STARC}}_{\tau,\gamma}$-distance is both necessary and sufficient for low worst-case regret, and that any other pseudometric on $\mathcal{R}$ with this property also must be equivalent to $d^{\mathrm{STARC}}_{\tau,\gamma}$. Therefore, we think it is justified to regard $d^{\mathrm{STARC}}_{\tau,\gamma}$ as the "right" way to quantify the difference between reward functions.

Recent literature has proposed other pseudometrics for quantifying the difference between reward functions, namely EPIC (Gleave et al., 2021) and DARD (Wulfe et al., 2022). However, these do not enjoy the same strong theoretical guarantees as $d^{\mathrm{STARC}}_{\tau,\gamma}$. In particular, they are neither sound nor complete in the sense of $d^{\mathrm{STARC}}_{\tau,\gamma}$. For more details, see Skalse et al. (2023).

C Why Not Use EPIC?

Many of our results are invariant to the choice of pseudometric on $\mathcal{R}$, but when we do have to pick a particular metric, we use $d^{\mathrm{STARC}}_{\tau,\gamma}$. Another prominent pseudometric on $\mathcal{R}$ is EPIC, which was first proposed by Gleave et al. (2021), and has since become the most widely used pseudometric on $\mathcal{R}$ (as judged by the number of citations at the time of writing). So why are we not using EPIC in this paper? There is a simple reason for this, namely that EPIC is sensitive to S'-redistribution. Specifically, for any reward function R and any $\delta \in (0, 1]$ there exists two reward functions R_1, R_2 such that R, R_1, and R_2 differ by S'-redistribution, but such that the EPIC-distance between R_1 and R_2 is $1 - \delta$. In other words, starting from an arbitrary reward function R and using only S'-redistribution, we can find reward functions whose EPIC-distance is arbitrarily close to 1.

This is problematic, because essentially any behavioural model of interest is invariant to S'-redistribution (including $o_{\tau,\gamma}$, $b_{\tau,\gamma,\beta}$, and $c_{\tau,\gamma,\alpha}$). This means that any such model will violate condition 2 in Definition 1 for all $\epsilon < 1$ when $d^{\mathcal{R}}$ is the EPIC pseudometric. Moreover, this also means that if f is ϵ-robust to misspecification with g (as defined by the EPIC distance), and g is invariant to S'-redistribution, then it must be the case that $\epsilon \geq 0.5$ (c.f. Lemma 1). Since an EPIC-distance of 0.5 is very large, such results are essentially vacuous. In other words, the EPIC-pseudometric is too loose, and cannot be used to derive any non-trivial results within the setting that we are concerned with in this paper.

In addition to this, $d^{\mathrm{STARC}}_{\tau,\gamma}$ also yields stronger theoretical guarantees than EPIC; see Appendix B and Skalse et al. (2023).

D Why Are Continuous Models Not Robust to Perturbations?

In this section, we give a more in-depth interpretation and explanation of Theorem 3.

Intuitively speaking, the fundamental reason that Theorem 3 holds is because there is a mismatch between ℓ^2-distance and STARC-distance. In particular, if f is continuous, then it must send reward functions that are close under the ℓ^2-norm to policies that are close under the ℓ^2-norm. However, there are reward functions that are close under the ℓ^2-norm but have a large STARC distance. Hence, if f is continuous then it will send some reward functions that are far apart under $d^{\mathrm{STARC}}_{\tau,\gamma}$ (but close under ℓ^2) to policies which are close (under ℓ^2), which in turn means that f is not ϵ/δ-separating.

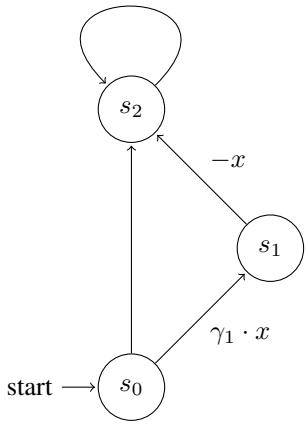
To see this, let R be an arbitrary non-trivial reward function, and let ϵ be any positive constant. We then have that $\epsilon \cdot R$ and $-\epsilon \cdot R$ have the opposite policy ordering, which means that $d^{\mathrm{STARC}}_{\tau,\gamma}(\epsilon \cdot R, -\epsilon \cdot R) = 1$. However, by making ϵ small enough, we can ensure that the ℓ^2-distance between $\epsilon \cdot R$ and $-\epsilon \cdot R$ is arbitrarily small, and hence that $d^{\Pi}(\epsilon \cdot R, -\epsilon \cdot R) < \delta$. Thus, we have two reward functions that have a large STARC-distance that are sent to policies that are close.

This example is not too concerning by itself, because it only demonstrates that we may run into trouble for reward functions that are very close to 0, and we may expect such reward functions to be unlikely (both in the sense that the observed agent is unlikely to have such a reward function, and in the sense that the inductive bias of the learning algorithm is unlikely to generate such a hypothesis). It would therefore be natural to restrict $\hat{\mathcal{R}}$ in some way, for example by imposing a minimum size on the ℓ^2-norm of all considered reward functions, or by supposing that they are normalised. However, Theorem 3 tells us that this will not work either: as long as there is some positive constant c such that if $||R||_2 = c$ then $R \in \hat{\mathcal{R}}$, then we can always find reward functions R_1, R_2 such that their ℓ^2-distance is small but $d^{\mathrm{STARC}}_{\tau,\gamma}(R_1, R_2)$ is large. Theorem 3 thus applies very widely.

E Why Is IRL Sensitive to Misspecified Parameters?

In this section, we give a more in-depth explanation of Theorems 4 and 5.

To start with, the reason that Theorem 4 is true is that we for any reward function R_1 can find a reward function R_2 such that R_1 and R_2 differ by potential shaping with γ_1, but such that R_1 and R_2 have a different policy ordering under γ_2 (when $\gamma_1 \neq \gamma_2$). To see this, consider a simple environment with three states s_0, s_1, s_2, where s_0 is the initial state, and where the agent can choose to either go directly from s_0 to s_2, or choose to first visit state s_1:



Let R_1 be any reward function over this environment, and let R_2 be the reward function that we get if we take R_1 and *increase* the reward of going from s_0 to s_1 by $\gamma_1 \cdot x$, and *decrease* the reward of

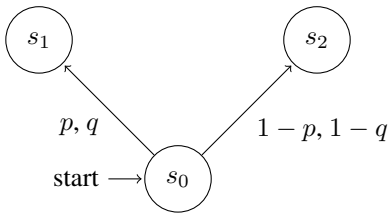
going from s_1 to s_2 by x. Now, the policy order under discounting with γ_1 is completely unchanged. At s_1, the value of every action is changed by the same amount, and so there is no reason to change action. Similarly, at s_0, the value of going to s_1 is changed by $\gamma_1 \cdot x - \gamma_1 \cdot x = 0$, and so there is likewise no reason to change action. This transformation corresponds to potential shaping where $\Phi(s_1) = x$ and $\Phi(s_0) = \Phi(s_2) = 0$. Therefore, if $f : \mathcal{R} \to \Pi$ is invariant to potential shaping with γ_1, then $f(R_1) = f(R_2)$.

However, if we discount with γ_2, then R_1 and R_2 have a different policy order. In particular, the value of going from s_0 to s_1 is changed by $\gamma_1 \cdot x - \gamma_2 \cdot x = (\gamma_1 - \gamma_2) \cdot x \neq 0$. Thus, if the optimal action under R_1 at s_0 is to go to s_1, then by making x sufficiently large or sufficiently small (depending on whether $\gamma_1 > \gamma_2$, or vice versa), then we can create a reward function R_2 for which the optimal action instead is to go to s_2, and vice versa.

Thus, in this environment, for every reward function R_1 and every γ_1, γ_2 such that $\gamma_1 \neq \gamma_2$, we can find a reward function R_2 such that R_1 and R_2 differ by potential shaping with γ_1, but such that they have a different ordering of policies when we discount with γ_2. This in turn means that we cannot be robust to misspecification of γ; if the observed policy is computed using γ_2, then there are reward functions that would lead to the same observed policy (and which hence cannot be distinguished by the learning process) but which nonetheless are a large distance from each other as evaluated by γ_1. This issue is present as long as $\gamma_1 \neq \gamma_2$, and so the degree of misspecification does not matter.

This is the basic mechanism behind Theorem 4, although this theorem additionally shows that the dynamic which we describe above shows up for *any* non-trivial transition function. Intuitively speaking, we can use potential shaping to move reward around in the MDP (so that the agent receives a larger immediate reward at the cost of a lower reward later, or vice versa). However, because of the discounting, later rewards must be made larger than immediate rewards. If the discount values do not match, then this "compensation" will also not match, leading to a distortion of the policy ordering. Indeed, we can make it so that this distortion dominates the rest of the reward function. For the full details, see the proof of Theorem 4.

As for Theorem 4, this theorem is similarly true because we for any reward function R_1 can find a reward function R_2 such that R_1 and R_2 differ by S'-redistribution with τ_1, but such that R_1 and R_2 have a different policy ordering under τ_2 (when $\tau_1 \neq \tau_2$). To see this, suppose we have an MDP with (at least) three states s_0, s_1, s_2, and that taking action a in state s_0 under transition function τ_1 takes you to state s_1 with probability p, and s_2 with probability $1 - p$. Similarly, taking action a in state s_0 under transition function τ_2 takes you to state s_1 with probability q, and s_2 with probability $1 - q$, where $p \neq q$.



Let R_1 be any reward function, and X any real number. Now note that we, regardless of the values of R_1 and X, can find values of $R_2(s_0, a, s_1)$ and $R_2(s_0, a, s_2)$ such that

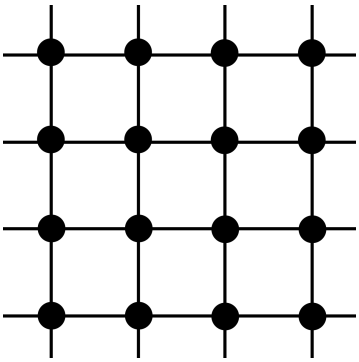
$$p \cdot R_2(s_0, a, s_1) + (1 - p) \cdot R_2(s_0, a, s_2) = p \cdot R_1(s_0, a, s_1) + (1 - p) \cdot R_1(s_0, a, s_2)$$

and such that $q \cdot R_2(s_0, a, s_1) + (1 - q) \cdot R_2(s_0, a, s_2) = X$.

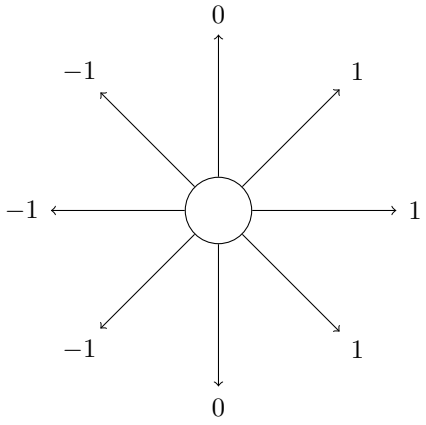
Note that this means that $\mathbb{E}_{S' \sim \tau_1(s_0,a)}[R_2(s_0, a, S')] = \mathbb{E}_{S' \sim \tau_1(s_0,a)}[R_1(s_0, a, S')]$, and that $\mathbb{E}_{S' \sim \tau_2(s_0,a)}[R_2(s_0, a, S')] = X$. Note also that X was selected arbitrarily. In other words, the fact that R_1 and R_2 differ by S'-redistribution under τ_1, leaves the expectation of R_2 under τ_2 *completely unconstrained* for all transitions where $\tau_1 \neq \tau_2$. If $\tau_1 \neq \tau_2$ for all states, then the policy order of R_2 under τ_2 can be literally any possible policy ordering. This in turn means that we cannot be robust to misspecification of τ; if the observed policy is computed using τ_2, then there are reward functions that would lead to the same observed policy (and which hence cannot be distinguished by the learning process) but which nonetheless have an arbitrarily large distance under τ_1.

It is also important to note that Theorem 4 does not require that $\tau_1 \neq \tau_2$ for all states; indeed, it is enough for them to differ at just a single transition s, a. Using the same strategy as above, we can find two reward functions R_1 and R_2 such that R_1 and R_2 differ by S'-redistribution under τ_1, but such that under τ_2, the value of a given policy π under R_1 depends primarily on visiting s, a as many times as possible, but the value of π under R_2 depends primarily on visiting s, a as *few* times as possible. For the full details, see the proof of Theorem 4.

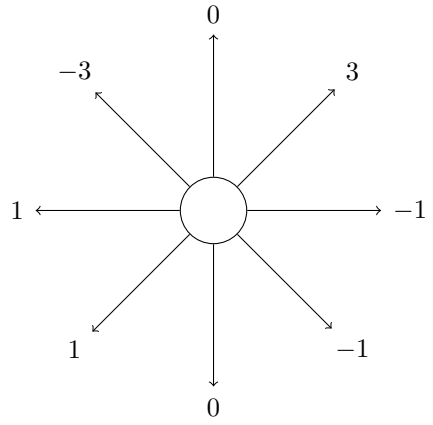
We can also give a somewhat less artificial example, to make this point more intuitive. Consider a simple $N \times N$ gridworld environment. We assume that the agent has four actions, `up`, `down`, `left`, and `right`. We assume that τ_1 is deterministic, so that if the agent takes action `up`, then it moves one step up, etc. Moreover, we assume that τ_2 is slippery, so that if the agent takes action `up`, then it moves up, up-left, and up-right with equal probability, and that if it takes action `right`, then it moves right, up-right, and down-right with equal probability, etc. For simplicity, we will also assume that the environment has a "PacMan-like" border, so that if the agent moves up from the top of the environment, then it ends up at the bottom, etc.[8]



Now suppose that R_1 and R_2 reward each transition depending on how the agent moves, according to the following schemas:



R_1



R_2

These two reward functions are identical under τ_2, and give the agent 1 reward for going right, -1 for going left, and 0 for going up or down. However, under τ_1, they are opposites; R_1 rewards the agent for going right, and R_2 rewards the agent for going left. Thus, if we observe a policy computed under τ_2, then we will not be able to distinguish between R_1 and R_2, even though they have a large distance under τ_1. Similar issues will occur given any discrepancy between τ_1 and τ_2.

[8] In other words, the environment is shaped like a torus.

F PROOFS

Here, we will provide the proofs of all our theorems and other theoretical results. The proofs are split up in three sections, mirroring the three subsections in Section 3.

F.1 NECESSARY AND SUFFICIENT CONDITIONS

Theorem 1. *Let $\hat{\mathcal{R}}$ be a set of reward functions, let $f : \hat{\mathcal{R}} \to \Pi$ be a behavioural model, and let $d^{\mathcal{R}}$ be a pseudometric on $\hat{\mathcal{R}}$. Suppose that $f(R_1) = f(R_2) \implies d^{\mathcal{R}}(R_1, R_2) = 0$ for all $R_1, R_2 \in \hat{\mathcal{R}}$. Then f is ϵ-robust to misspecification with g (as defined by $d^{\mathcal{R}}$) if and only if $g = f \circ t$ for some $t : \hat{\mathcal{R}} \to \hat{\mathcal{R}}$ such that $d^{\mathcal{R}}(R, t(R)) \leq \epsilon$ for all $R \in \hat{\mathcal{R}}$, and such that $f \neq g$.*

Proof. For the first direction, let $t : \hat{\mathcal{R}} \to \hat{\mathcal{R}}$ be a transformation such that $d^{\mathcal{R}}(R, t(R)) \leq \epsilon$ for all $R \in \hat{\mathcal{R}}$, and let $g = f \circ t$. To show that f is ϵ-robust to misspecification with g, we need to show that:

1. For all $R_1, R_2 \in \hat{\mathcal{R}}$, if $f(R_1) = g(R_2)$ then $d^{\mathcal{R}}(R_1, R_2) \leq \epsilon$.
2. For all $R_1, R_2 \in \hat{\mathcal{R}}$, if $f(R_1) = f(R_2)$ then $d^{\mathcal{R}}(R_1, R_2) \leq \epsilon$.
3. For all $R_1 \in \hat{\mathcal{R}}$ there exists an $R_2 \in \hat{\mathcal{R}}$ such that $f(R_2) = g(R_1)$.
4. There exists $R_1, R_2 \in \hat{\mathcal{R}}$ such that $f(R_1) \neq g(R_2)$.

For the first condition, suppose $f(R_1) = g(R_2)$, which implies that $f(R_1) = f \circ t(R_2)$. By assumption, we have that if $f(R) = f(R')$, then $d^{\mathcal{R}}(R, R') = 0$. This implies that $d^{\mathcal{R}}(R_1, t(R_2)) = 0$. Moreover, we have that $d^{\mathcal{R}}(R, t(R)) \leq \epsilon$ for all R; this implies that $d^{\mathcal{R}}(R_2, t(R_2)) \leq \epsilon$. By the triangle inequality, we then have that $d^{\mathcal{R}}(R_1, R_2) \leq 0 + \epsilon = \epsilon$. Since R_1 and R_2 were chosen arbitrarily, this means that condition 1 holds. For condition 2, note that we by assumption have that if $f(R_1) = f(R_2)$, then $d^{\mathcal{R}}(R_1, R_2) = 0$. Since $0 \leq \epsilon$, this implies that condition 2 holds. For condition 3, let R_1 be any reward function, and let $R_2 = t(R_1)$. Now $f(R_2) = g(R_1)$. Since R_1 was chosen arbitrarily, this means that condition 3 is satisfied. Condition 4 is satisfied by direct assumption. We have thus shown that if $g = f \circ t$ for some $t : \hat{\mathcal{R}} \to \hat{\mathcal{R}}$ such that $d^{\mathcal{R}}(R, t(R)) \leq \epsilon$ for all $R \in \hat{\mathcal{R}}$ and such that $f \neq g$, then f is ϵ-robust to misspecification with g (as defined by $d^{\mathcal{R}}$).

For the other direction, let f be ϵ-robust to misspecification with g (as defined by $d^{\mathcal{R}}$). For each $y \in \mathrm{Im}(g)$, let $R_y \in \hat{\mathcal{R}}$ be some reward function such that $f(R_y) = y$; since $\mathrm{Im}(g) \subseteq \mathrm{Im}(f)$, such an $R_y \in \hat{\mathcal{R}}$ always exists. Now let t be the function that maps each $R \in \hat{\mathcal{R}}$ to $R_{g(R)}$. Since by construction $g(R) = f(R_{g(R)})$, and since f is ϵ-robust to misspecification with g on $\hat{\mathcal{R}}$, we have that $d^{\mathcal{R}}(R, R_{g(R)}) \leq \epsilon$. Since by construction $t(R) = R_{g(R)}$, this means that $d^{\mathcal{R}}(R, t(R)) \leq \epsilon$. Thus $t : \hat{\mathcal{R}} \to \hat{\mathcal{R}}$ satisfies the condition that $d^{\mathcal{R}}(R, t(R)) \leq \epsilon$. Moreover, since f is ϵ-robust to misspecification with g, we have that $f \neq g$. Finally, note that $g = f \circ t$. This completes the proof of the other direction, which means that we are done. □

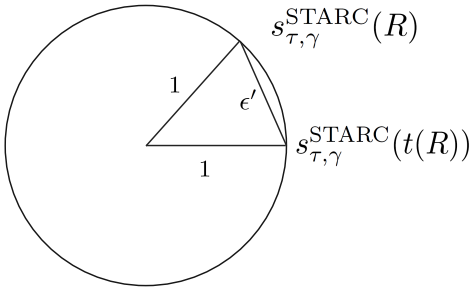
Proposition 3. *A transformation $t : \mathcal{R} \to \mathcal{R}$ satisfies that $d^{\mathrm{STARC}}_{\tau,\gamma}(R, t(R)) \leq \epsilon$ for all $R \in \mathcal{R}$ if and only if t can be expressed as $t_1 \circ \cdots \circ t_{n-1} \circ t_n \circ t_{n+1} \circ \cdots \circ t_m$ for some n and m where*

$$||R - t_n(R)||_2 \leq ||c^{\mathrm{STARC}}_{\tau,\gamma}(R)||_2 \cdot \sin(2 \arcsin(\epsilon/2))$$

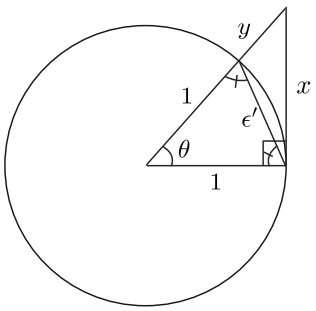
for all R, and for all $i \neq n$ and all R, we have that R and $t_i(R)$ differ by potential shaping (with γ), S'-redistribution (with τ), or positive linear scaling.

Proof. For the first direction, suppose $d^{\mathrm{STARC}}_{\tau,\gamma}(R, t(R)) \leq \epsilon$ for all $R \in \mathcal{R}$, and let R be an arbitrarily selected reward function. We will show that it is possible to navigate from R to $t(R)$ using the described transformations.

Recall that $d^{\mathrm{STARC}}_{\tau,\gamma}(R_1, R_2)$ is computed by first applying $c^{\mathrm{STARC}}_{\tau,\gamma}$ to both R_1 and R_2, then normalising the resulting vectors, and finally measuring their ℓ^2-distance. This means that $s^{\mathrm{STARC}}_{\tau,\gamma}(R)$ and $s^{\mathrm{STARC}}_{\tau,\gamma}(t(R))$ can be placed in the following diagram, where $\epsilon' \leq \epsilon$:



Now, elementary trigonometry tells us that $\theta = 2\arcsin(\epsilon'/2)$. Moreover, suppose we extend $s^{\mathrm{STARC}}_{\tau,\gamma}(R)$ to make the triangle a right triangle, as follows:



Here elementary trigonometry again tells us that $x/(1+y) = \sin(2\arcsin(\epsilon'/2))$, or that $x = (1+y)\sin(2\arcsin(\epsilon'/2))$. This means that we can go from R to $t(R)$ as follows:

1. Apply $c^{\mathrm{STARC}}_{\tau,\gamma}$. Since R and $c^{\mathrm{STARC}}_{\tau,\gamma}(R)$ differ by potential shaping and S'-redistribution, this transformation can be expressed as a combination of potential shaping and S'-redistribution. Call the resulting vector R'.
2. Normalise R', so that its magnitude is 1. This transformation is an instance of positive linear scaling. Call the resulting vector R''.
3. Scale R'' until it forms a right triangle with $s^{\mathrm{STARC}}_{\tau,\gamma}(t(R))$. This transformation is an instance of positive linear scaling. Call the resulting vector R'''.

4. Move from R''' to $s^{\mathrm{STARC}}_{\tau,\gamma}(t(R))$. This will move R''' by $(1+y)\sin(2\arcsin(\epsilon'/2))$, where $(1+y) = ||R'''||_2$. Moreover, since R''' is in the image of $c^{\mathrm{STARC}}_{\tau,\gamma}$, we have that $R''' = c^{\mathrm{STARC}}_{\tau,\gamma}(R''')$, and so $||R'''||_2 = ||c^{\mathrm{STARC}}_{\tau,\gamma}(R''')||_2$. This means that R''' is moved by $||c^{\mathrm{STARC}}_{\tau,\gamma}(R''')||_2 \cdot \sin(2\arcsin(\epsilon'/2))$. Since $\epsilon' \leq \epsilon \leq \pi/2$, this means that $||R''' - s^{\mathrm{STARC}}_{\tau,\gamma}(t(R))||_2 \leq ||c^{\mathrm{STARC}}_{\tau,\gamma}(R''')||_2 \cdot \sin(2\arcsin(\epsilon/2))$.

5. Move from $s^{\mathrm{STARC}}_{\tau,\gamma}(t(R))$ to $c^{\mathrm{STARC}}_{\tau,\gamma}(t(R))$. Since $s^{\mathrm{STARC}}_{\tau,\gamma}(t(R))$ is simply a normalised version of $c^{\mathrm{STARC}}_{\tau,\gamma}(t(R))$, this is an instance of positive linear scaling.

6. Move from $c^{\mathrm{STARC}}_{\tau,\gamma}(t(R))$ to $t(R)$. Since $t(R)$ and $c^{\mathrm{STARC}}_{\tau,\gamma}(t(R))$ differ by potential shaping and S'-redistribution, this transformation can be expressed as a combination of potential shaping and S'-redistribution.

Thus, for an arbitrary reward function R, we can find a series of transformations that fit the given description. This completes the first direction.

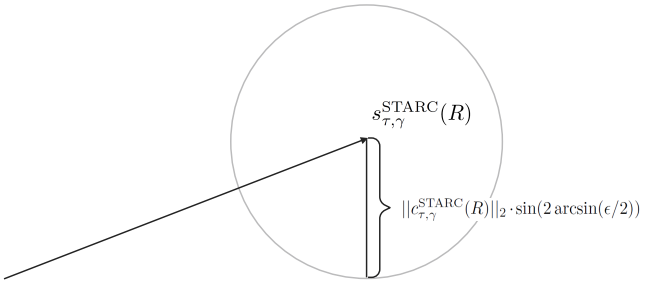
For the other direction, suppose t can be expressed as $t_1 \circ \cdots \circ t_{n-1} \circ t_n \circ t_{n+1} \circ \cdots \circ t_m$ where

$$||R - t_n(R)||_2 \leq ||c^{\mathrm{STARC}}_{\tau,\gamma}(R)||_2 \cdot \sin(2\arcsin(\epsilon/2))$$

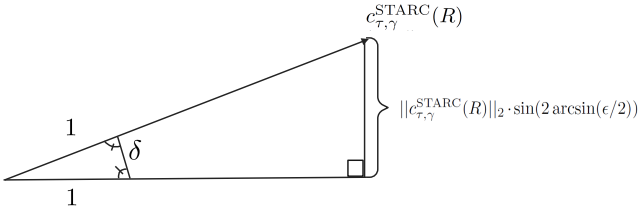
for all R, and for all $i \neq n$ and all R, we have that R and $t_i(R)$ differ by potential shaping (with γ), S'-redistribution (with τ), or positive linear scaling.

Recall that $d^{\mathrm{STARC}}_{\tau,\gamma}$ is invariant to potential shaping (with γ), S'-redistribution (with τ), and positive linear scaling; this means that $d^{\mathrm{STARC}}_{\tau,\gamma}(R, t_i(R)) = 0$ for $i \neq n$.

For t_n, recall that $c^{\mathrm{STARC}}_{\tau,\gamma}$ is a linear orthogonal projection; this means that $||c^{\mathrm{STARC}}_{\tau,\gamma}(R_1) - c^{\mathrm{STARC}}_{\tau,\gamma}(R_2)||_2 \leq ||R_1 - R_2||_2$. As such, if $||R - t_n(R)||_2 \leq ||c^{\mathrm{STARC}}_{\tau,\gamma}(R)||_2 \cdot \sin(2\arcsin(\epsilon/2))$, then $||c^{\mathrm{STARC}}_{\tau,\gamma}(R) - c^{\mathrm{STARC}}_{\tau,\gamma}(t_n(R))||_2 \leq ||c^{\mathrm{STARC}}_{\tau,\gamma}(R)||_2 \cdot \sin(2\arcsin(\epsilon/2))$ as well. Consider the following diagram:



Now $c^{\mathrm{STARC}}_{\tau,\gamma}(t_n(R))$ is located within circle in the diagram above. The vector within this circle that maximises the distance to $c^{\mathrm{STARC}}_{\tau,\gamma}(R)$ *after normalisation* lies on the tangent of the circle:



Elementary trigonometry now tells us that

$$\sin(\theta) = \frac{||c^{\mathrm{STARC}}_{\tau,\gamma}(R)||_2 \cdot \sin(2\arcsin(\epsilon/2))}{||c^{\mathrm{STARC}}_{\tau,\gamma}(R)||_2},$$

which gives that $\theta = 2\arcsin(\epsilon/2)$. From this, we have that $\delta = \epsilon$, and so $d^{\mathrm{STARC}}_{\tau,\gamma}(R, t(R)) \leq \epsilon$. This completes the other direction, and hence the proof. □

Corollary 1. *Let $\hat{\mathcal{R}}$ be a set of reward functions, τ be a transition function, γ a discount factor, β a temperature parameter, and α a weight parameter. Moreover, let $\hat{T}_\epsilon$ be the set of all functions $t : \mathcal{R} \to \mathcal{R}$ that satisfy Proposition 3, and additionally satisfy that $t(R) \in \hat{\mathcal{R}}$ for all $R \in \hat{\mathcal{R}}$. Then $b_{\tau,\gamma,\beta} : \hat{\mathcal{R}} \to \Pi$ is ϵ-robust to misspecification with g (as defined by $d^{\mathrm{STARC}}_{\tau,\gamma}$) if and only if $g = b_{\tau,\gamma,\beta} \circ t$ for some $t \in \hat{T}_\epsilon$, and $c_{\tau,\gamma,\alpha} : \hat{\mathcal{R}} \to \Pi$ is ϵ-robust to misspecification with g (as defined by $d^{\mathrm{STARC}}_{\tau,\gamma}$) if and only if $g = c_{\tau,\gamma,\alpha} \circ t$ for some $t \in \hat{T}_\epsilon$.*

Proof. Immediate from Theorem 1 and Proposition 3. □

Proposition 4. *Unless $|\mathcal{S}| = 1$ and $|\mathcal{A}| = 2$, then for any τ and any γ there exists an $E > 0$ such that for all $\epsilon < E$, there is no behavioural model g such that $o_{\tau,\gamma}$ is ϵ-robust to misspecification with g (as defined by $d^{\mathrm{STARC}}_{\tau,\gamma}$).*

Proof. We will first show that unless $|\mathcal{S}| = 1$ and $|\mathcal{A}| = 2$, there exists reward functions R_1, R_2 and an $E > 0$ such that $o_{\tau,\gamma}(R_1) = o_{\tau,\gamma}(R_2)$, but $d^{\mathrm{STARC}}_{\tau,\gamma}(R_1, R_2) = E$. In particular, note that if $|\mathcal{S}| \geq 2$ or $|\mathcal{A}| \geq 3$, then there exists uncountably many reward functions that do not have the same ordering of policies. Moreover, also note that $\mathrm{Im}(o_{\tau,\gamma})$ is finite. By the pigeonhole principle, this means that there must exist a policy $\pi \in \mathrm{Im}(o_{\tau,\gamma})$ and reward functions R_1, R_2 such that $o_{\tau,\gamma}(R_1) = o_{\tau,\gamma}(R_2) = \pi$, and such that R_1 and R_2 do not have the same ordering of policies. Moreover, recall that $d^{\mathrm{STARC}}_{\tau,\gamma}(R_1, R_2) = 0$ if and only if R_1 and R_2 have the same ordering of policies. Thus, unless $|\mathcal{S}| = 1$ and $|\mathcal{A}| = 2$, there exists reward functions R_1, R_2 such that $o_{\tau,\gamma}(R_1) = o_{\tau,\gamma}(R_2)$, but $d^{\mathrm{STARC}}_{\tau,\gamma}(R_1, R_2) = E > 0$. Thus $o_{\tau,\gamma}$ violates condition 2 of Definition 1 for all $\epsilon < E$. □

F.2 Perturbation Robustness

Theorem 2. *Let $\hat{\mathcal{R}}$ be a set of reward functions, let $f : \hat{\mathcal{R}} \to \Pi$ be a behavioural model, let $d^{\mathcal{R}}$ be a pseudometric on $\hat{\mathcal{R}}$, and let d^{Π} be a pseudometric on Π. Then f is ϵ-robust to δ-perturbation (as defined by $d^{\mathcal{R}}$ and d^{Π}) if and only if f is ϵ/δ-separating (as defined by $d^{\mathcal{R}}$ and d^{Π}).*

Proof. For the first direction, suppose f is ϵ/δ-separating, and let g be a δ-perturbation of f with $\mathrm{Im}(g) \subseteq \mathrm{Im}(f)$. We will show that f and g satisfy the conditions of Definition 1. For the first

condition, let R_1, R_2 be two arbitrary reward functions in $\hat{\mathcal{R}}$ such that $f(R_1) = g(R_2)$. Since g is a δ-perturbation of f, we have that $d^{\Pi}(g(R_2), f(R_2)) \leq \delta$. Since $f(R_1) = g(R_2)$, straightforward substitution thus gives us that $d^{\Pi}(f(R_1), f(R_2)) \leq \delta$. Since f is ϵ/δ-separating, this means that $d^{\mathcal{R}}(R_1, R_2) \leq \epsilon$. Since R_1 and R_2 were chosen arbitrarily, this means that if $f(R_1) = g(R_2)$ then $d^{\mathcal{R}}(R_1, R_2) \leq \epsilon$. Thus, the first condition of Definition 1 holds. For the second condition, note that if $f(R_1) = f(R_2)$, then $d^{\Pi}(f(R_1), f(R_2)) = 0 \leq \delta$. Since f is ϵ/δ-separating, this means that $d^{\mathcal{R}}(R_1, R_2) \leq \epsilon$, which means that the second condition is satisfied as well. The third condition is satisfied, since we assume that $\mathrm{Im}(g) \subseteq \mathrm{Im}(f)$, and the fourth condition is satisfied by the definition of δ-perturbations. This means that f and g satisfy all the conditions of Definition 1, and thus f is ϵ-robust to misspecification with g. Since g was chosen arbitrarily, this means that f is ϵ-robust to misspecification with any δ-perturbation g such that $\mathrm{Im}(g) \subseteq \mathrm{Im}(f)$. Thus f is ϵ-robust to δ-perturbation.

For the second direction, suppose f is *not* ϵ/δ-separating. This means that there exist $R_1, R_2 \in \hat{\mathcal{R}}$ such that $d^{\mathcal{R}}(R_1, R_2) > \epsilon$ and $d^{\Pi}(f(R_1), f(R_2)) \leq \delta$. Now let $g : \hat{\mathcal{R}} \to \hat{\mathcal{R}}$ be the behavioural model where $g(R_1) = f(R_2)$, $g(R_2) = f(R_1)$, and $g(R) = f(R)$ for all $R \notin \{R_1, R_2\}$. Now g is a δ-perturbation of f. However, f is not ϵ-robust to misspecification with g, since $g(R_1) = f(R_2)$, but $d^{\mathcal{R}}(R_1, R_2) > \epsilon$. Thus, if f is not ϵ/δ-separating then f is not ϵ-robust to δ-perturbation, which in turn means that if f is ϵ-robust to δ-perturbation, then f is must be ϵ/δ-separating. □

Theorem 3. *Let $d^{\mathcal{R}}$ be $d^{\mathrm{STARC}}_{\tau,\gamma}$, and let d^{Π} be a pseudometric on Π which satisfies the condition that for all δ_1 there exists a δ_2 such that if $||\pi_1 - \pi_2||_2 < \delta_2$ then $d^{\Pi}(\pi_1, \pi_2) < \delta_1$. Let c be any positive constant, and let $\hat{\mathcal{R}}$ be a set of reward functions such that if $||R||_2 = c$ then $R \in \hat{\mathcal{R}}$. Let $f : \hat{\mathcal{R}} \to \Pi$ be continuous. Then f is not ϵ/δ-separating for any $\epsilon < 1$ or $\delta > 0$.*

Proof. Let R be a non-trivial reward function that is orthogonal to all trivial reward functions. Since the set of all trivial reward functions form a linear subspace, such a reward function R exists. Note that R must not necessarily be contained in $\hat{\mathcal{R}}$.

We now have that for any positive constant ϵ, it is the case that ϵR and $-\epsilon R$ have the opposite ordering of policies, and thus $d^{\mathrm{STARC}}_{\tau,\gamma}(\epsilon R, -\epsilon R) = 1$. Next, let R_Φ be some potential-shaping reward function such that $||\epsilon R + R_\Phi||_2 = c$. Since potential shaping does not change the ordering of policies, we have that $\epsilon R + R_\Phi$ and $\epsilon R + R_\Phi$ must have the opposite ordering of policies as well, and so $d^{\mathrm{STARC}}_{\tau,\gamma}(\epsilon R + R_\Phi, -\epsilon R + R_\Phi) = 1$. Moreover, since R_Φ is trivial, we have that both ϵR and $-\epsilon R$ are orthogonal to R_Φ, and so $||-\epsilon R + R_\Phi||_2 = c$ as well. Since $||\epsilon R + R_\Phi||_2 = ||\epsilon R + R_\Phi||_2 = c$, we have that both $\epsilon R + R_\Phi$ and $-\epsilon R + R_\Phi$ are contained in $\hat{\mathcal{R}}$.

Let δ_1 be any positive constant. By assumption, there exists a δ_2 such that if $||\pi_1 - \pi_2||_2 < \delta_2$ then $d^{\Pi}(\pi_1, \pi_2) < \delta_1$. Moreover, since f is continuous, there exists an ϵ_1 such that if $||R_1 - R_2||_2 < \epsilon_1$, then $||f(R_1) - f(R_2)||_2 < \delta_2$. Next, note that by making ϵ sufficiently small, we can ensure that the ℓ^2-distance between $\epsilon R + R_\Phi$ and $\epsilon R + R_\Phi$ is arbitrarily small (and, in particular, less than ϵ_1).

Thus, for any positive δ there exist reward functions $\epsilon R + R_\Phi$ and $-\epsilon R + R_\Phi$ that are both contained in $\hat{\mathcal{R}}$, such that $d^{\Pi}(f(\epsilon R + R_\Phi), f(-\epsilon R + R_\Phi)) < \delta$, and such that $d^{\mathrm{STARC}}_{\tau,\gamma}(\epsilon R + R_\Phi, -\epsilon R + R_\Phi) = 1$. Thus f is not ϵ/δ-separating for any $\delta > 0$ and any $\epsilon < 1$. □

F.3 Misspecified Parameters

Lemma 1. *Let $\hat{\mathcal{R}}$ be a set of reward functions, let $f, g : \hat{\mathcal{R}} \to \Pi$ be two behavioural models, and let $d^{\mathcal{R}}$ be a pseudometric on $\hat{\mathcal{R}}$. Suppose f is ϵ-robust to misspecification with g (as defined by $d^{\mathcal{R}}$). Then if $g(R_1) = g(R_2)$, we have that $d^{\mathcal{R}}(R_1, R_2) \leq 2\epsilon$.*

Proof. Let $f, g : \hat{\mathcal{R}} \to \hat{\mathcal{R}}$ be two behavioural models, and suppose f is ϵ-robust to misspecification with g. Let $R_1, R_2 \in \hat{\mathcal{R}}$ be any two reward functions such that $g(R_1) = g(R_2)$. From condition 3 in Definition 1, we have that there must be a reward function R_3 such that $f(R_3) = g(R_1) = g(R_2)$. From condition 1 in Definition 1, we have that $d^{\mathcal{R}}(R_3, R_1) \leq \epsilon$ and $d^{\mathcal{R}}(R_3, R_2) \leq \epsilon$. The triangle inequality then implies that $d^{\mathcal{R}}(R_1, R_2) \leq 2\epsilon$. □

Lemma 2. *If $f_\gamma : \mathcal{R} \to \Pi$ is invariant to potential shaping with γ, then for all τ and all γ_1, γ_2 such that $\gamma_1 \neq \gamma_2$ and τ is non-trivial, then there exists a reward function $R^\dagger$ such that $f_{\gamma_1}(R) = f_{\gamma_1}(R + \alpha R^\dagger)$ for all $R \in \mathcal{R}$ and all $\alpha \in \mathbb{R}$.*

Proof. Analogous to the proof of Lemma A.18 in Skalse & Abate (2023). □

Lemma 3. *If $f_\tau : \mathcal{R} \to \Pi$ is invariant to S'-redistribution with τ, then for all γ and all τ_1, τ_2 such that $\tau_1 \neq \tau_2$, there exists a reward function $R^\dagger$ that is non-trivial under τ_2 and γ, such that $f_{\tau_1}(R) = f_{\tau_1}(R + \alpha R^\dagger)$ for all $R \in \mathcal{R}$ and all $\alpha \in \mathbb{R}$.*

Proof. Since f_{τ_1} is invariant to S'-redistribution with τ_1, we have that $f_{\tau_1}(R_1) = f_{\tau_1}(R_2)$ for any two reward functions R_1, R_2 such that

$$\mathbb{E}_{S' \sim \tau_1(s,a)} \left[R_1(s, a, S') \right] = \mathbb{E}_{S' \sim \tau_1(s,a)} \left[R_2(s, a, S') \right].$$

Note that R_1 and R_2 satisfy this condition if and only if

$$\mathbb{E}_{S' \sim \tau_1(s,a)} \left[(R_2 - R_1)(s, a, S') \right] = 0.$$

That is to say, if R' is a reward function such that $\mathbb{E}_{S' \sim \tau_1(s,a)} \left[R'(s, a, S') \right] = 0$, then $f_{\tau_1}(R) = f_{\tau_1}(R + R')$ for all R. Next, note that the set of all such reward functions R' form a linear subspace of $\mathcal{R}$, with $|\mathcal{S}||\mathcal{A}|(|\mathcal{S}| - 1)$ dimensions. We will show that this subspace contains reward functions that are non-trivial under γ and τ_2.

Since $\tau_1 \neq \tau_2$, we have that there exists a state s and action a such that $\tau_1(s, a) \neq \tau_2(s, a)$. Let $R^\dagger$ be a reward function that is 0 everywhere, except that $\mathbb{E}_{S' \sim \tau_1(s,a)} \left[R'(s, a, S') \right] = 0$, and $\mathbb{E}_{S' \sim \tau_2(s,a)} \left[R'(s, a, S') \right] = 1$. Note that there is always a solution to this system of linear equations. In particular, the values of $R^\dagger(s, a, s')$ for each transition s, a, s' form a $|\mathcal{S}|$-dimensional vector space. The set of all values for these variables that satisfy $\mathbb{E}_{S' \sim \tau_1(s,a)} \left[R'(s, a, S') \right] = 0$ form an $(|\mathcal{S}| - 1)$-dimensional linear subspace, and the set of all values that satisfy $\mathbb{E}_{S' \sim \tau_2(s,a)} \left[R'(s, a, S') \right] = 1$ form an $(|\mathcal{S}| - 1)$-dimensional affine subspace. These two sets must intersect, unless they are parallel. However, since $\sum_{s'} \mathbb{P}(\tau_1(s, a) = s') = \sum_{s'} \mathbb{P}(\tau_2(s, a) = s') = 1$, they cannot be parallel. Thus, such a reward function $R^\dagger$ must exist.

It is clear that $R^\dagger$ is non-trivial under γ and τ_2. To spell it out; since all states are reachable under τ_2 and μ_0, there exists a policy π that visits state s with positive probability. Let π_1 and π_2 be two policies that are identical to π everywhere, except that π_1 takes action a with probability 1 in state s, and π_2 takes action a with probability 0 in state s. Then $J^\dagger(\pi_1) > J^\dagger(\pi_2)$. Moreover, since $\mathbb{E}_{S' \sim \tau_1(s,a)} \left[R^\dagger(s, a, S') \right] = 0$ for all s and a, we have that R and $R + \alpha R^\dagger$ differ by S'-redistribution (with τ_1) for all reward functions $R \in \mathcal{R}$ and all scalars $\alpha \in \mathbb{R}$. Thus $f_{\tau_1}(R) = f_{\tau_1}(R + \alpha R^\dagger)$.

Thus, if $f_\tau : \mathcal{R} \to \Pi$ is invariant to S'-redistribution with τ, then for all γ and all τ_1, τ_2 such that $\tau_1 \neq \tau_2$, there exists a reward function $R^\dagger$ that is non-trivial under τ_2 and γ, such that $f_{\tau_1}(R) = f_{\tau_1}(R + \alpha R^\dagger)$ for all $R \in \mathcal{R}$ and all $\alpha \in \mathbb{R}$. □

Theorem 4. *If $f_\gamma : \mathcal{R} \to \Pi$ is invariant to potential shaping with γ, and $\gamma_1 \neq \gamma_2$, then f_{γ_1} is not ϵ-robust to misspecification with f_{γ_2} under $d^{\mathrm{STARC}}_{\tau,\gamma_3}$ for any non-trivial τ, any γ_3, and any $\epsilon < 0.5$.*

Proof. If $\gamma_1 \neq \gamma_2$, then either $\gamma_1 \neq \gamma_3$ or $\gamma_2 \neq \gamma_3$.

If $\gamma_1 \neq \gamma_3$, then Lemma 2 implies that there exists a reward function $R^\dagger$ that is non-trivial under τ and γ_3, such that $f_{\gamma_1}(R) = f_{\gamma_1}(R + \alpha R^\dagger)$ for all $R \in \mathcal{R}$ and all $\alpha \in \mathbb{R}$. This means that $f_{\gamma_1}(R^\dagger) = f_{\gamma_1}(-R^\dagger)$ and $d^{\mathrm{STARC}}_{\tau,\gamma_3}(R^\dagger, -R^\dagger) = 1$. Thus f_{γ_1} violates condition 2 of Definition 1 for all $\epsilon < 1$.

If $\gamma_2 \neq \gamma_3$, then Lemma 2 implies that there exists a reward function $R^\dagger$ that is non-trivial under τ and γ_3, such that $f_{\gamma_2}(R) = f_{\gamma_2}(R + \alpha R^\dagger)$ for all $R \in \mathcal{R}$ and all $\alpha \in \mathbb{R}$. This means that $f_{\gamma_2}(R^\dagger) = f_{\gamma_2}(-R^\dagger)$ and $d^{\mathrm{STARC}}_{\tau,\gamma_3}(R^\dagger, -R^\dagger) = 1$. Then Lemma 1 implies that there can be no f that is ϵ-robust to misspecification with f_{γ_2} (as defined by $d^{\mathrm{STARC}}_{\tau,\gamma_3}$) for any $\epsilon < 0.5$. □

Theorem 5. *If $f_\tau : \mathcal{R} \to \Pi$ is invariant to S'-redistribution with τ, and $\tau_1 \neq \tau_2$, then f_{τ_1} is not ϵ-robust to misspecification with f_{τ_2} under $d^{\mathrm{STARC}}_{\tau_3,\gamma}$ for any τ_3, any γ, and any $\epsilon < 0.5$.*

Proof. If $\tau_1 \neq \tau_2$, then either $\tau_1 \neq \tau_3$ or $\tau_2 \neq \tau_3$.

If $\tau_1 \neq \tau_3$, then Lemma 3 implies that there exists a reward function $R^\dagger$ that is non-trivial under τ_3 and γ, such that $f_{\tau_1}(R) = f_{\tau_1}(R + \alpha R^\dagger)$ for all $R \in \mathcal{R}$ and all $\alpha \in \mathbb{R}$. This means that $f_{\tau_1}(R^\dagger) = f_{\tau_1}(-R^\dagger)$ and $d^{\mathrm{STARC}}_{\tau_3,\gamma}(R^\dagger, -R^\dagger) = 1$. Thus f_{τ_1} violates condition 2 of Definition 1 for all $\epsilon < 1$.

If $\tau_2 \neq \tau_3$, then Lemma 3 implies that there exists a reward function $R^\dagger$ that is non-trivial under τ_3 and γ, such that $f_{\tau_2}(R) = f_{\tau_2}(R + \alpha R^\dagger)$ for all $R \in \mathcal{R}$ and all $\alpha \in \mathbb{R}$. This means that $f_{\tau_2}(R^\dagger) = f_{\tau_2}(-R^\dagger)$ and $d^{\mathrm{STARC}}_{\tau_3,\gamma}(R^\dagger, -R^\dagger) = 1$. Then Lemma 1 implies that there can be no f that is ϵ-robust to misspecification with f_{τ_2} (as defined by $d^{\mathrm{STARC}}_{\tau_3,\gamma}$) for any $\epsilon < 0.5$. □

G Connecting Our Analysis To Earlier Proposals

In this section, we will explain how to connect the results of Skalse & Abate (2023) to our results in a rigorous way. Skalse & Abate (2023) assume that we have a partition P on $\mathcal{R}$, which of course corresponds to an equivalence relation $\equiv_P$, and say that two reward functions R_1, R_2 should be considered to be "close" if $R_1 \equiv_P R_2$. Like us, they consider functions $f, g : \mathcal{R} \to \Pi$ that take a reward function and return a policy. They then say that f is "P-robust to misspecification with g" if each of the following conditions hold:

1. $f(R_1) = g(R_2) \implies R_1 \equiv_P R_2$.
2. $f(R_1) = f(R_2) \implies R_1 \equiv_P R_2$.
3. For all R_1 there exists an R_2 such that $f(R_2) = g(R_1)$.
4. $f \neq g$.

Note that this definition is analogous to Definition 1, except that an equivalence relation P plays the role that a pseudometric $d^{\mathcal{R}}$ does in our framework. Next, note that we for any pseudometric $d^{\mathcal{R}}$ can define an equivalence relation $\equiv_P$ such that $R_1 \equiv_P R_2$ if and only if $d^{\mathcal{R}}(R_1, R_2) = 0$. In that case, we would have that f is P-robust to misspecification with g (in the terminology of Skalse & Abate, 2023) if and only if f is 0-robust to misspecification with g (as evaluated by $d^{\mathcal{R}}$) in our terminology (i.e. Definition 1). Moreover, if f is 0-robust to misspecification with g, then it of course follows that f is ϵ-robust to misspecification with g for all $\epsilon \geq 0$. In this way, their results can be expressed within our more general framework.

Next, also recall that $d^{\mathrm{STARC}}_{\tau,\gamma}(R_1, R_2) = 0$ if and only if R_1 and R_2 induce the same ordering of policies (under τ and γ). Skalse & Abate (2023) use "$\mathrm{ORD}^{\mathcal{M}}$" to denote this equivalence relation. Thus, if f is $\mathrm{ORD}^{\mathcal{M}}$-robust to misspecification with g (as defined by Skalse & Abate, 2023) then f is ϵ-robust to misspecification with g (as evaluated by $d^{\mathrm{STARC}}_{\tau,\gamma}$) for all $\epsilon \geq 0$.