We would like to thank the reviewers for their detailed comments and suggestions.

Scalability: As noted by Reviewer 1, our method scales well to much larger domains. For example, our method can optimize a robust policy for a machine replacement problem with 5,000 states in only 163 seconds. We can optimize a robust policy for a 60-by-60 gridworld (3,600 states) in under two minutes. For comparison, a state-of-the-art CVaR optimization approach for MDPs with no uncertainty over the reward function takes 2 hours for a similar-sized gridworld (see [1] Section 5, last paragraph). We will add experiments demonstrating the scalability of our method to the appendix and will add an experiment where we transfer a learned reward function to a new environment (as suggested by Reviewer 1).

Reviewer #1

> *Right now I think that the description of the contributions hides the most useful contribution.*

Thank you for the suggestion, we will clarify the contributions as suggested.

> *The robust implementation of the method (4.d) doesn't match the demonstration for one state.*

The reason is that Bayesian IRL does not assume demonstrator optimality, only Boltzman rationality. We used a relatively small inverse temperature resulting in reward function hypotheses that allow for occasional demonstrator errors. Using a larger inverse temperature will cause the robust policy to match all the demonstrator's actions.

> *The paper is fairly well written, but there is room for improvement in the paper presentation.*

Thank you for the detailed and constructive suggestions. We will make the notation more consistent (for example by sticking to μ and w as much as possible) and add notation reminders throughout the paper. We will also introduce the examples as a motivation earlier in the paper.

Reviewer #2

> *Is conditional value-at-risk the best approach for adjustable risk-sensitivity?*

We used CVaR because of its popularity and interpretability, but it is true that it is not always the best metric. BROIL actually works with any convex risk measure, such as EVaR or entropic risk, the only modification is that the linear program would need to be replaced by a convex optimization problem. We will make this clear in the paper.

> *The key issue is how to handle misspecification of the Bayesian prior.*

This is a good point and something we would really like to tackle in a followup work. The Bayesian statistics community has devoted a lot of effort to addressing this problem; we will add appropriate pointers.

Reviewer #3

> *The method seems to rely heavily on the quality of prior/posterior distribution of the reward …*

Yes, this is true for all Bayesian methods.

> *The linear structure of the reward brings computational convenience, however it is hard for the reward to satisfy this structure in real-world applications.*

We agree that linear approximation methods (including linear regression, and linear value function approximation) have limits, but their simplicity, speed, and generally smaller data needs (bias-variance tradeoff) make them often very useful.

> *Any generalization of the method when we could parametrize the policy? … deep RL or healthcare experiments?*

These are good suggestions. The BROIL objective is convex and nearly everywhere differentiable so it could also be used in place of expected return in a policy gradient-style approach. We judged this to be beyond the scope of this paper, but will mention this idea in the paper as an important and interesting area for future work.

Reviewer #4

> *Assuming the samples are not exact, how do approximations in MCMC propagate onto optimization of BROIL?*

Thank you, this is an important question. It has been investigated in the stochastic programming community in the context of the SAA method. We will include pointers to relevant literature on this topic in the revision.

> *The method is presented in the context of inverse RL. Has it already been addressed within ordinary (non-inverse) RL?*

Several similar methods have been studied in the context of RL, but the key difference is that most RL work considers uncertain transition probabilities, while in IRL it is rewards that are uncertain. This difference has a major impact on the type of algorithms that are appropriate for the two settings. Most relevant papers in ordinary RL, which address robustness/risk aversion to *model error*, are distributionaly robust MDPs (Xu & Mannor 2012), percentile optimization (Delage & Mannor 2010), and epistemic risk aversion (Eriksson & Dimitrakakis 2019).

References

[1] Yinlam Chow, Aviv Tamar, Shie Mannor, and Marco Pavone. Risk-sensitive and robust decision-making: a cvar optimization approach. In *Advances in Neural Information Processing Systems*, pages 1522–1530, 2015.