COVER LETTER FOR "BENCHMARKING CONSTRAINT INFERENCE IN INVERSE REINFORCEMENT LEARNING": IMPROVEMENTS UPON NEURIPS-2022 SUBMISSION

Anonymous authors
Paper under double-blind review

ABSTRACT

This cover letter provides details on our efforts to substantially improve the ICLR 2023 submission of the paper "Benchmarking Constraint Inference in Inverse Reinforcement Learning" upon our previous Neurips 2022 submission. We addressed all major comments of the Neurips 2022 reviewers and in fact resolved every major issue raised by them. We also provide clarifications and justifications of all other points raised by the reviewers. we include a response to points raised by each reviewer. **The reference in this letter is consistent with our Neurips version, which we have included in Appendix**.

1 REVIEWER 1

- "**For benchmarking, I noticed that in Figures 3 and 4, there is only one Lagrange method(PPO-LAG). As far as I know, there seem to be many other excellent constrained algorithms, such as FocOps, Pcpo and so on. I seriously from the experimental level, this comparison is not sufficient.**"

  Response: Thank you for the feedback. There is a misunderstanding. Let us clarify. The point of Figures 3 and 4 is not to compare PPO and PPO-Lag nor to evaluate constrained RL techniques (such as FocOps, Pcpo, and so on), but to demonstrate that the constraints matter in the sense that they have an effect on the optimal policy. Figures 3 and 4 could have been generated by comparing the output of any constrained RL technique to the output of any unconstrained RL technique. The choice of algorithm doesn't matter and there is no suggestion that the chosen algorithms (PPO and PPO-Lag) are better than other algorithms. Again, we are not evaluating constrained RL nor unconstrained RL techniques, but simply showing that the constraints have an impact which becomes clear when there is a gap between the optimal unconstrained policy and optimal constrained policy.

- "**The author says in line53 that Each of these environments is charmed with a demonstration dataset generated by an expert agent trained by the Reward Constrained Policy Optimization (RCPO) [11]. In Line 163, Training Expert Agent with PPO-Lag. This is confusing. At the same time, the author did not explain whether the important process of the PPO-LAG algorithm, such as the update of the lambda multiplier, is similar or the same with the implementation of OpenAI [1]. [1]https://github.com/openai/safety-starter-agents**"

  Response: Thank you for the question. As explained on Line 166 of the paper, we optimize the PPO-Lag objective with the RCPO algorithm. So there is really only one algorithm: RCPO. This algorithm can be used to optimize various objectives. As explained in the paper we optimize the PPO-Lag objective. The RCPO implementation that we used is not from https://github.com/openai/safety-starter-agents. As mentioned in Footnote 1 of Page 6, our implementation is based on https://github.com/DLR-RM/stable-baselines3. Please see the code in our GitHub repository for more details. The initial value of lambda and its learning rate is specified in the config files of each domain inside the repository.

- "**I observe that the author has a Stable Baseline3 folder directly in his Github repository, and then at the end of Page 6, the paper says PPO may models the policy distribution with a Gaussian, for which we follow a stable implementation in https://github.com/DLR-RM/stable-baselines3, I think the author's work is largely dependent on Stable Baseline3, which is an excellent open source project. But the author also does not refer to the reference connection*

*in the ReadME of Stable Basline3. @articlestable-baselines3, author = Antonin Raffin and Ashley Hill and Adam Gleave and Anssi Kanervisto and Maximilian Ernestus and Noah Dormann, title = Stable-Baselines3: Reliable Reinforcement Learning Implementations, journal = Journal of Machine Learning Research, year = 2021, volume = 22, number = 268, pages = 1-8, url = http://jmlr.org/papers/v22/20-1364.html **"

Response: Thank you for catching this omission. We added the reference to the paper in the footonote besides the url (see newly uploaded version). We did not mean any disrespect. This was simply an oversight on our part.

2 REVIEWER 2

- "**I would like to see at least a few more types of tasks. That is, currently, one type of task is a robotic locomotion task with a constraint on body movement. This task has not been well-motivated as important/relevant in the current task. Why are these good tasks to study for this setting?***"

  Response: Thanks for raising this important concern. Our virtual environment mainly studies the robot control task with a constraint on the location. In practice, this type of constraint captures the locations of obstacles in the environment. For example, the agent observes that none of the expert demonstrations has visited some places in the environment. Then it is reasonable to infer that these locations must be unsafe even if the robot sensors cannot detect what happens there. These unsafe locations can be represented by constraints. In our virtual environment, the added constraints mark some backward locations (e.g., x-Coordinate ¡ -3) in the environment as unsafe. Although the real-world task might require more complicated constraints, our benchmark, as the first benchmark for constraint inverse reinforcement learning, could serve as a stepping stone for these tasks. We have clarify this in Section 3.1 (lines 111 - 117)

- "**Is there a reason these tasks are much more realistic inverse constrained RL problems than the grid-world tasks of prior work?***"

  Response: Thanks for mentioning this. Grid-worlds are environments with discrete action and state spaces. The size of the action and state spaces are often limited, e.g., a previous work [2] used x-y coordinate to represent states of a 9-by-9 grid-world, for a total of 81 states. On the other hand, our tasks have continuous action and state spaces. Some of our tasks (e.g., Blocked Ant and HighD) use high-dimensional state spaces, which are similar to observations (e.g., signals from lidar and other sensors) in real-world tasks.

  [2] Dexter R. R. Scobee and S. Shankar Sastry. Maximum likelihood constraint inference for inverse reinforcement learning. In 8th International Conference on Learning Representations, (ICLR). OpenReview.net, 2020.

- "**First, as I've already described, the tasks need further explanation as to why they are well-suited. The text says "Since the quality of expert trajectories can significantly influence model performance, it is difficult to compare different CIRL algorithms without a consistent expert dataset." Can you show this experimentally? This would help solidify the usefulness of this dataset. ***"

  Response: We have added an experiment in Section B.1. We study the influence on model performance by utilizing noisy demonstration datasets that record some random actions during data generations. Please check the results (Figure B.1) and discussion.

- "**Another thing I am very curious about regarding the demonstrations is what the performance of using imitation learning methods (so this would look like simply adding a behavior cloning baseline) is–do these tasks require the RL phase? Can it successfully satisfy constraints and solve the tasks at hand by imitating the demonstrations?***"

  Response: Thanks for mentioning about our comparison method. We agree that more clarifications are required. Our comparison methods include GACL, which is based on Generative Adversarial Imitation Learning (see Section 4.2). GACL alternates between updating an imitating policy and a constraint function under an adversarial framework. On the other hand, Behavior Cloning (BC) directly imitates the action at each observed state without inferring any constraints. Note that directly imitating an expert without inferring constraints is a different task than inferring constraints. Our paper focuses on inferring

constraints. We work with an industrial partner who specifically mandated us to design algorithms to infer constraints. Our industrial partner plans to reuse the constraints in different environments to optimize policies that are different from the expert policy due to the different environments. While we can also use the inferred constraints in the same environment to optimize a constrained policy that mimics the expert, this is not the goal of the paper.

- "**Next, it is unclear whether the oracle performance PPO-Lag can be achieved.**"

  Response: Thanks for raising this concern. We have included the performance of PPO-Lag in our paper (See Figures 5 and 6). Ideally, if the ground-truth constraints are accurately captured, these algorithms can reproduce the performance PPO-Lag. In fact, we do observe such cases in the Blocked Half-Cheetah and the Blocked Ant environments. An intriguing finding is the baselines' performance in the Blocked Ant environment is better than that of PPO-Lag. We believe the main reason is that PPO-Lag applies a **hard** constraint (e.g., 1-0 constraints) for the policy update, whereas our baselines apply a soft constraint (continuous values). The hard constraint limits the exploration of PPO agents.

- "**Can you show the effect of the number of demonstrations used – in this case it would be easier to understand what the possible scope of progress is. For example, algorithmic improvement can achieve the same performance as current methods with half the number of demonstrations.**"

  Response: We have added another experiment in Appendix B.1. We study the influence on model performance by utilizing smaller demonstration datasets that cover only parts of expert trajectories in the original data. Please check the results (Figure B.1) and discussion.

- "**Or, can you outline what kinds of improvements concretely we can expect to make as a community by using this benchmark? To improve clarity, please describe more about the tasks and why they are important (even moving some of the information from the appendix to the main paper would be helpful)**"

  Response: Yes, you raise an important concern for our research. We are expecting the following contributions to our community: 1) CIRL is an emerging research topic without a common benchmark for examining the model. Our benchmark offers a consistent testbed for CIRL algorithms. We expect the community can examine new algorithms in our environment by comparing the baseline models implemented in the benchmark. 2) The tasks defined in the benchmark, including robot-control and auto-driving, are among the most direct and important applications of CIRL in the industry. We have explained the applications of location constraints in robot control. In terms of auto-driving, our benchmark evaluates whether CIRL algorithms can learn the constraints added to the environment. If they pass the tests, our industrial partner will apply them as data mining tools to discover the constraints of human-driving demonstrations. Adding these constraints to an auto-driving system can facilitate a more natural policy that resembles human preferences. Our benchmarks are based on OpenAI for ease of implementation. Although there is a gap between the constraints in our benchmark and the constraints that our industrial partners are interested in, these benchmarks are valid testbeds in the pipeline of constraint inference and discovery, which supports building more mature CIRL algorithms. We have expanded the introduction of our environments by adding more details in Section 3 (lines 102-108).

- "**Can you add ppo-lag to the plots for a comparison to what CIRL methods should aim to achieve?**"

  Response: Yes, we agree adding ppo-lag to the plots can clarify our results. we have included PPO-Lag into the plots (Figure 5 and 6).

- "**Please detail the baselines (and what evaluating them should show) much more thoroughly.**"

  Response: Thanks for mentioning this. We agree that providing a more detailed introduction is important. Our baselines are inspired by the common algorithms for Inverse Reinforcement Learning (IRL), and we extend them to CIRL by following some previous works. Striving for scalability in real-world environments, these baselines are based on the model-free RL setting (without relying on transition dynamics). To be more specific, 1) MECL assumes the policy maximizes its entropy $H(\pi_E)$ during learning to solve the unidentifiable issues of classic apprenticeship learning. As one of the most generalizable

frameworks, the maximum entropy framework acts as the foundation of many recent IRL methods. MECL extends the classic maximum entropy framework to constraint inference. 2) GACL extends MECL by incorporating adversarial training into IRL. These combinations have been very popular in IRL, and previous works [Ho2016,Fu2017,Qureshi2019] showed adversarial networks can accelerate training and improve imitation performance. Here we study this combination in the CIRL tasks. 3) BC2L replaces the loss in the constraint learning step of MECL with a binary classification loss. These updates act as a simplification of MECL, which, in the meantime, incurs some loss of identifiability. We study whether a binary classifier is sufficient for capturing the ground-truth constraints. These baselines are proposed for the environment with deterministic dynamics, and thus we proposed a variational inference method that can learn constraints from stochastic environments. We have expanded the introduction of our baselines by adding more details in Section 4.2.

[Ho2016] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In Neural Information Processing Systems (Neurips), pages 4565–4573, 2016

[Fu2017] Justin Fu, Katie Luo and Sergey Levine. Learning Robust Rewards with Adversarial Inverse Reinforcement Learning. In CoRR, abs/1710.11248, 2017

[Qureshi2019] Ahmed Hussain Qureshi, Byron Boots and Michael C. Yip. Adversarial Imitation via Variational Inverse Reinforcement Learning. In International Conference on Learning Representations (ICLR), 2019

- *"*Regarding claims, the abstract writes: "As an emerging research topic, CIRL does not have common benchmarks, and previous works tested their algorithms with hand-crafted environments (e.g., grid worlds)." However, the ICRL paper proposes some of the same MuJoCo tasks as you use. Perhaps reflect this more accurately?*"*

  Response: Thanks for mentioning this. We have mentioned their contribution in related works (Section 7) by explicitly mentioning that they have proposed the Mujoco environments for CIRL, but they studied only the half-cheetah and ant environments. We will make sure their credits can be accurately reflected in our paper.

3 REVIEWER 3

- "*A benchmarking paper should have benchmarks designed with "room to grow" in mind. In other words it should clearly demonstrate that existing methods have a performance gap to be overcome by new algorithms people will develop and use with the benchmark. This is explicitly discussed in the text for Biased Pendulum and Blocked Walker, but a) it would be clearer to see if the expert performance were included in the graphs, b) Most of the rest of the benchmarks (except, perhaps, Blocked Swimmer) don't seem to have much of a gap. In particular, HighD not having much of a gap seems like a critical problem, as this is your more advanced environment. Can it be made harder?*"

  Response: Thanks for mentioning this. There are few things we need to clarify here.

  In this work, the virtual (robot) environment is based on Mujoco, which often does not have a performance upper bound. In terms of the realistic (HighD) environments, we include the performance upper bound into the plot. The *highest success rate* (i.e., reaching the destination without collision, going off-road, breaking time limits, or the added constraints) among all baselines in our HighD environments is around 68% (we have added the plots of success rate in Figure 6). Since the environments are created based on real data and all the ego cars can drive safely in the beginning of a game under the added constraints, we can make sure these environments are solvable, and the upper bound of success rate is 100% (Our appendix A.6 has shown some examples of invalid constraints, with which games might not be solvable). We believe the room for improvement should be sufficient. We have illustrated the causes of some failing cases by showing the collision rate, time-out rate, and off-road rate in Appendix B.1. These visualizations can support the development of future algorithms to overcome the limitations of current baselines. We have add these discussions to section 5 (lines 275 - 279).

  We also want to clarify that although our benchmark uses PPO-Lag to generate expert demonstrations, PPO-Lag is **not** optimal in terms of maximizing the rewards or satisfying constraints (Section 6 briefly mentions the issues with Lagrange methods). In fact, to handle the sub-optimality of PPO-Lag, we filter the generated trajectories with state-action

pairs that violate constraints (Section 3.3), which makes sure the demonstration data is optimal in terms of satisfying the constraints. However, there is no guarantee that the demonstration data is optimal in terms of maximizing cumulative rewards. In other words, The performance of PPO-Lag should not be treated as the performance upper bound in our benchmark. Despite these issues, we can still apply PPO-Lag for generating demonstration data. This is because humans are not always driving to maximize rewards (i.e., driving to a destination as fast as they can) but they often make sure that driving constraints are not violated (i.e., driving safe under the traffic rules). We have add these discussions to section 3.3 (lines 176-180).

When it comes to CIRL algorithms, since the agent can interact with the environment and receive feedback, its policy could outperform PPO-Lag in terms of collecting rewards. Most of the time, this improvement is accompanied by a higher constraint violation rate, but we do observe in some cases the agent can collect more rewards with a constraint violation rate that is similar to PPO-Lag's violation rate. In order to better illustrate our points, we have added the performance of PPO-Lag to our plots (Figure 5 and 6).

- *"*It seems as though PPO-Lag is what is being used for expert demonstrations for HighD, based on the section starting at line 176 and the provided dataset supplementary materials. However, the environment was introduced by saying, "following the constraints learned from human drivers' trajectories". If PPO-Lag is what was used, why? Real data would be the preferred benchmark, in my opinion.*"*

  Response: Thanks for raising this concern. We do use PPO-Lag for generating demonstration data for the ego vehicle (i.e., the vehicle that our agent controls or the blue car in Figure 3), but the trajectories of other vehicles follow the human demonstrations in the HighD dataset. We apologize if our writing induced some confusion. In fact, there are several issues when directly using human demonstrations as the ego vehicle's demonstrations: 1) the underlying constraints in human demonstrations are unknown. This is problematic since we will not be able to determine whether the ground-truth constraint is broken or not. The constraint violation rate cannot be calculated. We are not sure what will be a proper metric for evaluation. 2) the rewards followed by the human demonstrations are unknown. We can use existing reward-recovering algorithms (e.g, apprenticeship learning, IRL) to learn the rewards, but these algorithms often assume the MDP is not constrained or the underlying constraints are known (e.g., CMDP) in the environment, which contradicts the goal of CIRL. Considering the above issues, we train a PPO-Lag agent to generate trajectories as expert demonstration. During the training of PPO-Lag, we can simply use manually-defined constraints and rewards and this information can be used for testing.

  Having said that, we must admit your concern is very important and in fact, it is closely related to the ultimate goal of CIRL: discovering constraints from human demonstrations. However, before applying CIRL algorithms to real-world tasks, we must first make sure they are working properly. Our benchmark serves as a tool for validating CIRL algorithms. If they pass the test, we can trust the constraints they discover from the real data. Knowing these constraints could help build a more robust AI control system. We have added the above ideas into Section 3.3 (lines 157-162).

- *"*My understanding is that Mujoco environments are intended to be deterministic by default, and no added stochasticity was described for HighD. Particularly given the focus of VCIRL on stochasticity, it seems as though the benchmarks presented are not an adequate testbed for the method presented.*"*

  Response: Thanks for mentioning this. We agree more clarifications are required. In fact, the HighD environment is stochastic since it is constructed by following the human demonstration in the dataset. As mentioned above, although the states of the environments do not reflect the demonstrations of ego vehicles (the blue car in Figure 3), they do follow the demonstrations of all other vehicles (the red cars in Figure 3). To build the environment, we randomly select trajectory data from a total of 3041 scenarios in the HighD dataset and construct the states with our game engine. The distributions of the vehicles' features (e.g., speeds, locations, and steering angles) are different from one trajectory to another. Even when some vehicles share the same starting points and destinations, human drivers might behave differently depending on the road conditions, their preferences, and driving skills. Moreover, the states in the HighD environment are partially observable, and predicting the

location of unobserved vehicles is impossible. Considering these properties, we believe the HighD environment is stochastic. We have clarified it in the Section 3.2 (lines 150 - 153).

- *"*Based on the plots, it seems as though HighD was divided up into two separate tasks: one with a velocity constraint, and one with a distance constraint. If this is accurate, why are they separate? It seems as though a more accurate, challenging benchmark would be for an agent to need to deduce both together. Additionally, if accurate, it needs to be made more clear in the text. If it is not accurate, the plots need to be clarified that all data was collected during the same run, and simply different aspects of the data are being analyzed.*"*

  Response: We apologize for the potential misunderstanding. In fact, we do separate the velocity constraint and the distance constraint in our benchmark. We will clarified it. As the reviewer has suggested, we will add experiment for an environment with both constraints. The experiment is running on our machine and we will update the results once it is done.

- *"*Section 2.2 mentions a method that incorporates constraints into the reward function. It would be nice to see this as a baseline as well, though I will not make it a requirement since baseline implementation can be non-trivial.*"*

  Response: Thanks for mentioning this. Our baseline model GACL implements the idea of incorporating constraints into the reward function. We have clarified it in line 97.

- *"*The benchmarks in Section 3 are all presented with PPO and PPO-Lag(range) results. Since PPO-Lag is not properly introduced until Section 3.3, it was not clear during the earlier sections that PPO-Lag is an expert method with direct access to the constraints, and its purpose in the paper is to be used for demonstration generation. It would be significantly clearer, in my opinion, to describe the benchmarks first, then describe how demonstrations are collected, with Figures 2 and 4 as justification for PPO-Lag as expert. My first impression was that PPO-Lag was your proposed method for solving the environments.*"*

  Response: Thanks for your suggestion. As you have mentioned, PPO and PPO-Lag are introduced to justify the proposed constraint in the environment and to provide demonstration data. We have revised the paper according to your suggestion (see Section 3).

- *"* $r(\tau)$ wasn't defined that I could see. I'm assuming observed rewards, but I believe the missing definition to simply be an oversight. Additionally, could a consistent font be used for ϕ?*"*

  Response: Thanks for mentioning this. $r(\tau)$ defines the reward of a trajectory τ. This definition is popular in IRL. Φ indicates a random variable while ϕ defines an instance (or value) of the random variable. We have clarified these definitions in the revised version.