We thank the reviewers for lending their expertise and time to provide feedback on our efforts. We are glad that all the reviewers found our insight that action transformation can be seen as an IfO problem novel and interesting. We respond to the biggest questions and comments below and will address all feedback in the paper.

[R2 , R3 , R4] The reviewers are correct in pointing out that, despite the title, we do not include a real robot experiment. Our work is motivated by sim-to-real, but we were unable to conduct real robot experiments due to the current pandemic as R1 and R4 pointed out. If accepted, we will make several changes to moderate the claims as R4 suggested. In particular, we will change the terminology in the paper to align with more general transfer learning, using source and target domains as opposed to sim and "real." Also, we will change the title to "Towards Sim-to-Real Transfer: An Imitation from Observation Approach." Please note that our formulation remains very relevant to the sim-to-real community. We would like to highlight that one of our experiments is indeed an excellent proxy for the sim-to-real problem: In the Minitaur domain (Figure 2), Tan et al. [38] found that while their existing simulator (our source domain) inaccurately represented their robot, the new simulator they crafted (our target domain) *did* enable direct policy transfer from sim to real.

[R2 , R3] Both manipulation domains [7, 24, 26, 39, 40, R2 's suggestions] and locomotion domains [9, 10, 11, 14, 18, 27, 38, 46] are prevalent in the sim-to-real literature. Both are important—but different—problems: manipulation domains are more likely to exhibit observation mismatch, whereas locomotion domains are more typically associated with dynamics mismatch. The scope of our work here is mainly dynamics mismatch, and therefore we focus our experiments on locomotion problems. GARAT solely addresses dynamics mismatch. For locomotion, the observations are usually joint angles and velocities, so observation mismatch is negligible. If accepted, we will make this scope more clear in the camera-ready version of our paper and include the references R2 suggested. Note that in our problem setting, the state spaces are the same in the source and target domains, as is commonly the case in sim-to-real. Specifically, we consider dealing with embodiment mismatch to be beyond the scope of this paper.

[R2] Most domain randomization techniques, and all the papers suggested by R2 , require a modifiable simulator and substantial domain expertise [7]. In this paper we focus on the case where the simulator cannot be modified (black box), and hence it is not appropriate to compare with methods that can adjust the simulator itself. We compare to ANE [20] which is an action randomization technique.

[R2] Respectfully, we strongly disagree with the reviewer's assertion that our approach does not offer significant technical novelty. In this work, we show how tools developed in the imitation learning community can be successfully adapted to sim-to-real problems. Moreover, our adaptation of one such tool actually leads to better performance than alternative applicable approaches. To the best of our knowledge, this is the first time this has been studied in the literature, and therefore our work represents a novel and important connection between two largely separate communities.

[R1] Concerning why GAT was not as effective as GARAT on transfer, perhaps it would be useful to compare the two techniques to their imitation learning equivalents, behavioral cloning (BC) and using inverse RL (IRL). BC suffers from distribution shift while IRL methods are able to learn how to recover from such shifts; likewise, GAT is unable to recover from the shift introduced by an imperfect action transformation while GARAT can correct for such deviations. GAT is myopic, trying to match single transitions, while GARAT matches the whole trajectory (Figure 1b).

[R2 , R3] The curve for GAT cuts off early in Figure 1b. In the InvertedPendulum domain, the episode terminates if the angle of the pendulum exceeds ± 0.2 radians. In the environment with GAT, the action transformation learns to keep close to the target domain's dynamics early on, but this causes instability later in the episode, leading to early termination. GARAT sacrifices initial accuracy to keep the overall trajectory as realistic as possible. We will edit the caption for Figure 1b to make it clear in the camera ready version of the paper.

[R3] We use the loss derived in Section 4.3 in our main results. Our algorithm is agnostic to the RL algorithms used for training. We chose PPO and TRPO for the action transformation function and the agent respectively because that combination worked best in preliminary experiments on the InvertedPendulum domain.

[R3] GARAT should implicitly address process noise due to its adversarial learning procedure. The discriminator in GARAT encourages the action transformation function to learn a distribution of transitions that are similar to the target domain, including any noisy transitions. Moreover, GAT [14] has been shown to be useful in sim-to-real transfer on a real legged humanoid robot, showing that impact dynamics and operational noise do not prevent learning.

[R2 , R3] Figure 3 was normalized in order to compare the performance of different algorithms across different domains. It does not represent the maximum and minimum returns possible. We train π_{real} in the target domain for 1 million time-steps, enough to reach a reasonable policy. These policies may take more training to converge completely (HalfCheetah is usually trained for 10 million timesteps). GARAT manages to learn a policy that does better than the policies trained directly in the target domain for some of these environments.