

HAL: HARMONIC LEARNING IN HIGH-DIMENSIONAL MDPs

Anonymous authors

Paper under double-blind review

1 ADDITIONAL RESULTS AND STATE REPRESENTATIONS WITH SPECTRAL PROPERTIES

Figure 4 reports stable basis robustness analysis results for the inverse Q -learning policy and deep neural policies trained via harmonic learning policy in SpaceInvaders and Breakout. The results reported in Figure 4 once more demonstrate that harmonic learning yields learning of more robust policies. The results reported in Figure 5 of the main body of the paper further demonstrate that the robustness and resilience properties of harmonic learning further extends substantially to distributional shift.¹ One of the concrete reasons for the level of achieved robustness by harmonic learning lies in Figure 4 and Table 2 of the main body of the paper. The results reported in Figure 4 from the main

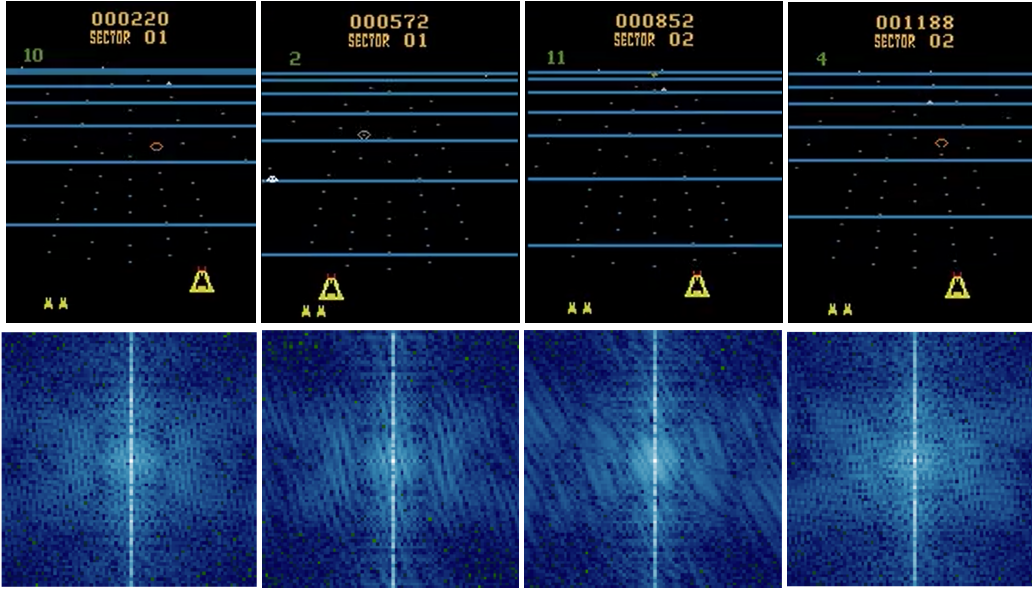


Figure 1: State observations of the high-dimensional state representation MDPs and the spectral representations in BeamRider.

body of the paper demonstrate that the inverse Q -learning policy further suffers from overfitting of the state-action value function, and harmonic learning addresses this problem resulting in learning a more correct estimation of the state-action values. More clearly, Figure 5 of the main body of the paper demonstrates the sensitivities of the deep inverse reinforcement learning policy, and further harmonic learning provides a concrete theoretically well-founded solution for these sensitivities and vulnerabilities of the deep inverse reinforcement learning policies. Also note that the methods focusing on robustification of the policies (i.e. adversarial training) results in **sample inefficiency**.

¹Some recent work also argued that state-of-the-art certified robustification methods for deep reinforcement learning policies results in learning policies that have worse reaction to distributional shift than straightforward vanilla trained deep reinforcement learning policies Korkmaz (2023). However, our proposed harmonic learning method, i.e. HAL, demonstrates that deep neural policies trained via harmonic learning are more robust to both distributional shift and the overfitting problem, while further increasing the sample efficiency.

However, our theoretically well-founded method demonstrates that while harmonic learning results in learning robust policies, it also further substantially increases the sample efficiency.

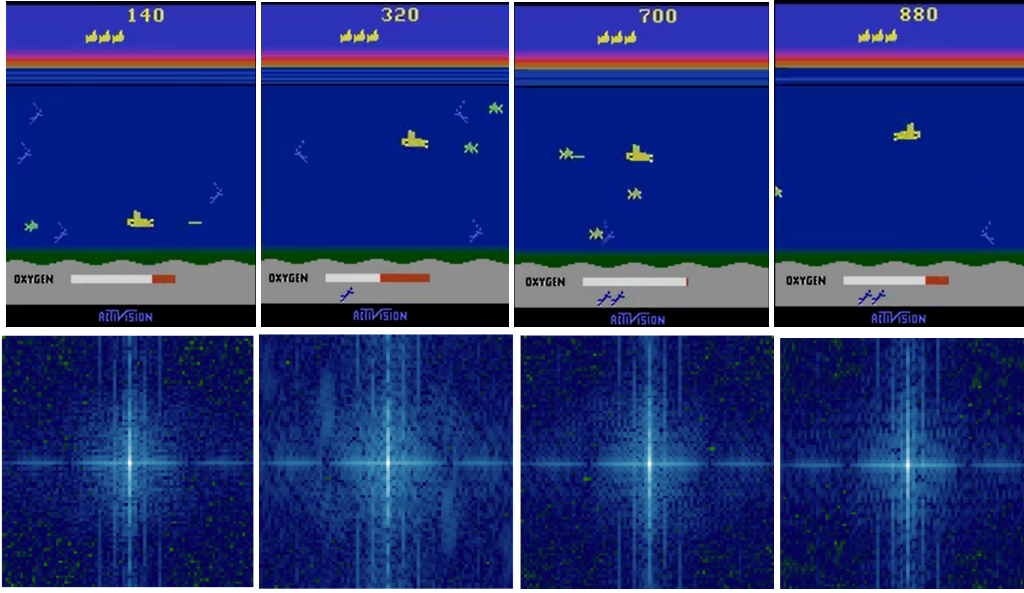


Figure 2: State observations of the high-dimensional state representation MDPs and the spectral representations in Seaquest.

Figure 2 and Figure 1 demonstrate state observations and the corresponding spectral representations of these high-dimensional state representations.

2 CODE FOR HARMONIC LEARNING ALGORITHM

```

for episode_step in range(EPISEODE_STEPS):
    if steps < args.num_seed_steps:
        action = env.action_space.sample()
    else:
        with train_mode(agent):
            dim_obs = np.size(state)[0,: ,0]
            fourier_obs_initial = np.fft.fft2(np.array(state))
            LQ = np.random.randint(dim_obs/2, size=1)
            HQ = dim_obs-LQ-1
            fourier_obs_initial[:,LQ,LQ:HQ] = 0
            fourier_obs_initial[:,HQ,LQ:HQ] = 0
            fourier_obs_initial[:,LQ:HQ,LQ] = 0
            fourier_obs_initial[:,LQ:HQ,HQ] = 0
            state = np.fft.ifft2(fourier_obs_initial)
            action = agent.choose_action(state, sample=True)
        next_state, reward, done, _ = env.step(action)
        episode_reward += reward
        steps += 1

```

Figure 3: Harmonic learning code. Harmonic learning is easy to implement and only takes 9 lines of code with substantial sample-efficiency gains resulting in more generalizable and robust policies.

Figure 3 reports the code for harmonic learning. Our algorithm does not require any gradient and function evaluations, and only takes 9 lines of code. Harmonic learning is an extremely fast and efficient algorithm.

3 HYPERPARAMETERS AND TRAINING SETUP

Table 1 describes the hyperparameter settings for inverse- Q learning and the harmonic learning algorithm. Note that the hyperparameters are fixed exactly to the inverse- Q learning algorithm setting (Garg et al., 2021) for transparency and fairness. The expert demonstrations are obtained from a DQN trained policy (Mnih et al., 2015) as also has been described in detail in (Garg et al., 2021). The activation functions for the connected layers are exponential linear units (ELU). Further note that all of the experiments conducted in our paper are in MDPs with high-dimensional state representations (i.e. Arcade Learning Environment) (Bellemare et al., 2013). Some of the expert demonstrations are obtained by using the (Raffin, 2020) pipeline as also has been described in detail in (Garg et al., 2021).

Table 1: Hyperparameter settings for the inverse- Q learning and the harmonic learning algorithm.

Hyperparameters	Settings
Target Update Frequency	1000
Critic Learning Rate	10^{-4}
Initial Temperature	0.01
Critic τ	0.1
Subsampling frequency	1
Replay Memory	150000
Initial Memory	5000
Demos	20
α	0.5
ϵ steps	1000
ϵ window	100
Batch Size	64
Discount factor	0.99
Observation size	(84, 84)
Evaluation steps	5000
Q -Network channels	32,64,64
Q -Network filter size	$8 \times 8, 4 \times 4, 3 \times 3$
Q -Network stride	(4, 4), (2, 2), (1, 1)
Q -Network hidden units	512

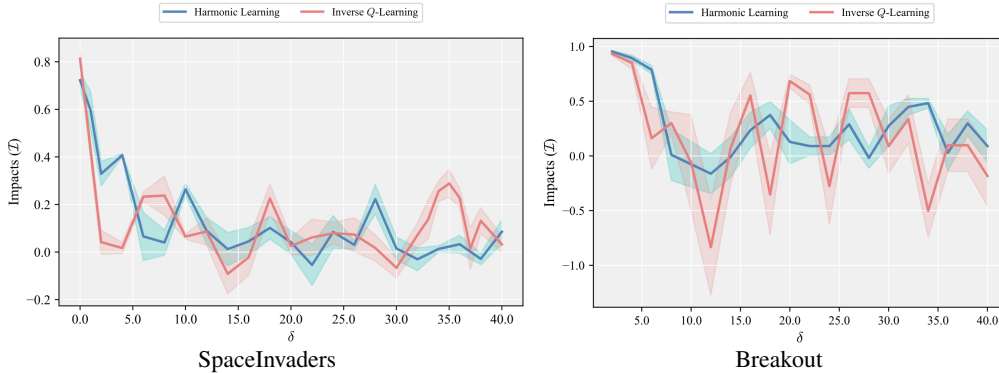


Figure 4: Stable Basis Robustness Analysis (SBRA) results for the inverse Q -learning policy and deep neural policies trained via harmonic learning in SpaceInvaders and Breakout.

Table 2: Raw scores obtained by harmonic learning policy, inverse Q -learning policy and the expert policy in Pong, Breakout, Seaquest, SpaceInvaders and BeamRider.

Training Method	Harmonic Learning	Inverse Q -learning	Expert Policy
Pong	19.0 ± 1.89736	8.0 ± 5.3814	21 ± 0.0
Seaquest	906.0 ± 53.2202	864.0 ± 42.0285	2393 ± 291.0
SpaceInvader	609.0 ± 14.5223	470.555 ± 23.6812	823.0 ± 272.0
BeamRider	1023.6 ± 140.974	909.6 ± 65.392	4295.0 ± 1173.0
Breakout	228.8 ± 35.4606	108.9 ± 29.7198	376.0 ± 34.0

4 EXPERT DEMONSTRATIONS AND HARMONIC LEARNING

Another important metric that we can measure and provide is the performance of the inverse- Q learning algorithm and the harmonic learning algorithm when compared to the performance level of an expert in a data-limited setting.

Table 2 reports results of raw scores obtained by the harmonic learning policy, inverse Q -learning policy and the expert policy in Pong, Breakout, Seaquest, SpaceInvaders and BeamRider.

Table 3 reports the percentage of the expert policy performance that harmonic learning and the inverse- Q learning policy reach with only **50K environment interactions**. The percentage of the expert policy results once more demonstrate clearly the sample efficiency gains achieved by the harmonic learning algorithm.

5 THEORETICAL BASIS AND EMPIRICAL ANALYSIS

Note that Definition 3.1 introduces the notion of a stable basis, which is intuitively a basis in feature space such that the optimal policy depends approximately equally on each basis vector. Definition 3.4 extends the notion of stable-basis to the general function approximation setting, and Proposition 3.5 implies a natural method for testing if a given basis is a stable-basis by measuring the rewards obtained when removing the component of the observations along each stable basis vector. On the experimental side, the stable basis robustness analysis via Algorithm 3 is an operationalization of the test for a stable basis under general function approximation given by Proposition 3.5. In particular, by removing the observation components of basis vectors corresponding to particular frequencies.

The results of stable basis robustness analysis in Figure 2 demonstrate that the vanilla trained policy is robust across the spectrum to the removal of these Fourier frequency components, and hence provides support to the claim that the Fourier basis is a stable-basis for the high-dimensional state observation MDPs considered. Furthermore, Proposition 3.3 implies that removing the observation components

Table 3: Percentage of the expert policy performance achieved by the harmonic learning policy and the inverse Q -learning policy in Pong, Breakout, Seaquest, SpaceInvaders and BeamRider.

Training Method	Harmonic Learning	Inverse Q -learning
Pong	$90.47\% \pm 9.03\%$	$38.09\% \pm 25.23\%$
Seaquest	$37.86\% \pm 2.22\%$	$36.10\% \pm 1.756\%$
SpaceInvader	$73.99\% \pm 1.764\%$	$57.16\% \pm 2.877\%$
BeamRider	$26.15\% \pm 3.282\%$	$21.17\% \pm 0.125\%$
Breakout	$60.64\% \pm 9.43\%$	$28.71\% \pm 7.90\%$

along randomly selected stable-basis vectors during training (i.e. Algorithm 2) corresponds to adding noise to the value function estimate proportional to a natural uncertainty measure given by Definition 3.2. Thus, training with Algorithm 2 has an interpretation that is analogous to randomized least-squares value iteration and other related methods for value-function randomization that yield provable regret bounds. Therefore, because the stable basis robustness analysis results indicate that the Fourier basis is a stable-basis for our setting of high-dimensional state observations, Proposition 3.3 provides theoretical motivation for the use of harmonic learning via Algorithm 3 to improve

sample-efficiency. In connection to this the experimental results in Table 1 demonstrate notably improved sample-efficiency via harmonic learning, providing support for the theoretical justifications in Section 3.

REFERENCES

- Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael. Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research.*, pp. 253–279, 2013.
- Divyansh Garg, Shuvam Chakraborty, Chris Cundy, Jiaming Song, and Stefano Ermon. Iq-learn: Inverse soft-q learning for imitation. *Neural Information Processing Systems (NeurIPS) [Spotlight Presentation]*, 2021.
- Ezgi Korkmaz. Adversarial robust deep reinforcement learning requires redefining robustness. *AAAI Conference on Artificial Intelligence*, 2023.
- Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Antonoglou, Helen King, Dhharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518: 529–533, 2015.
- Antonin Raffin. RL baselines3 zoo. <https://github.com/DLR-RM/rl-baselines3-zoo>, 2020.