Frame Skip Is a Powerful Parameter for Learning to Play Atari

Alex Braylan, Mark Hollenbeck, Elliot Meyerson and **Risto Miikkulainen**
Computer Science Department, The University of Texas at Austin
2317 Speedway, Austin, TX 78712

Abstract

We show that setting a reasonable frame skip can be critical to the performance of agents learning to play Atari 2600 games. In all of the six games in our experiments, frame skip is a strong determinant of success. For two of these games, setting a large frame skip leads to state-of-the-art performance.

The rate at which an agent interacts with its environment may be critical to its success. In the Arcade Learning Environment (ALE) (Bellemare et al. 2013) games run at sixty frames per second, and agents can submit an action at every frame. *Frame skip* is the number of frames an action is repeated before a new action is selected. Existing reinforcement learning (RL) approaches use static frame skip: HNEAT (Hausknecht et al. 2013) uses a frame skip of 0; DQN (Mnih et al. 2013) uses a frame skip of 2-3; SARSA and planning approaches (Bellemare et al. 2013) use a frame skip of 4. When action selection is computationally intensive, setting a higher frame skip can significantly decrease the time it takes to simulate an episode, at the cost of missing opportunities that only exist at a finer resolution. A large frame skip can also prevent degenerate super-human-reflex strategies, such as those described by Hausknecht et al. for Bowling, Kung Fu Master, Video Pinball and Beam Rider.

We show that in addition to these advantages agents that act with high frame skip can actually learn faster with respect to the number of training episodes than those that skip no frames. We present results for six of the seven games covered by Mnih et al.: three (Beam Rider, Breakout and Pong) for which DQN was able to achieve near- or superhuman performance, and three (Q*Bert, Space Invaders and Seaquest) for which all RL approaches are far from human performance. These latter games were understood to be difficult because they require 'strategy that extends over long time scales.' In our experiments, setting a large frame skip was critical to achieving state-of-the-art performance in two of these games: Space Invaders and Q*Bert. More generally, the frame skip parameter was a strong determinant of performance in all six games.

Our learning framework is a variant of Enforced Subpopulations (ESP) (Gomez and Miikkulainen 1997), a neuroevolution approach that has been successfully implemented and extended to train agents for a variety of complex behaviors and control tasks (Gomez and Schmidhuber 2005; Schmidhuber et al. 2007, e.g.). In contrast to conventional neuroevolution (CNE) which evolves networks directly, ESP maintains a distinct population of neurons for each hidden node in the network, which enables hidden nodes to coevolve to take on complementary roles. ESP can also add hidden nodes to provide a boost when learning stagnates. In the experiments below, all networks are feedforward. The input layer is the object representation introduced by Hausknecht et al. The output layer has one node for each of the nine joystick positions and one indicating whether or not to fire. For each game we trained agents at four frame skips: 0, 20, 60 and 180. For each of these 24 setups, to maintain comparison to Hausknecht et al., we averaged scores over five runs, simulated 100 episodes per generation, and capped each episode at 50,000 frames. To further speed up training, on all games except Seaquest (which has particularly sparse rewards) we stop agents when they have not received a positive reward in 30 game seconds. Each run lasts 200 generations. The score of a run at a given generation is the highest total reward an agent has achieved in an episode by that generation. Figure 1 depicts the training progress for each setup.

ESP performed better with a high frame skip for Beam Rider, Q*Bert, Seaquest and Space Invaders. Seaquest achieved top performance when skipping 180 frames, that is, when pausing for a full three seconds between decisions. Space Invaders and Beam Rider achieved their top performance when skipping 60 frames. Agents that use high frame skip do not learn action selection for states that are skipped, and thus have a greater capacity to learn associations between more temporally distant states and actions. This could help deal with the non-Markovian nature of some of these games. For example, as noted by Mnih et al., in Space Invaders lasers are not visible every fourth frame. If an agent only commits to longterm actions when lasers are visible, it will not be confused by this peculiarity. These longer-term decisions also lead to broader exploration of the behavior space resulting in increased population diversity. On the other hand, it is not surprising that Pong and Breakout perform best with low frame skip, since these games require fine actions to reach every state necessary to block the ball. For Pong, the performance difference would be even larger if we did not stop agents that did not score for 30 seconds.

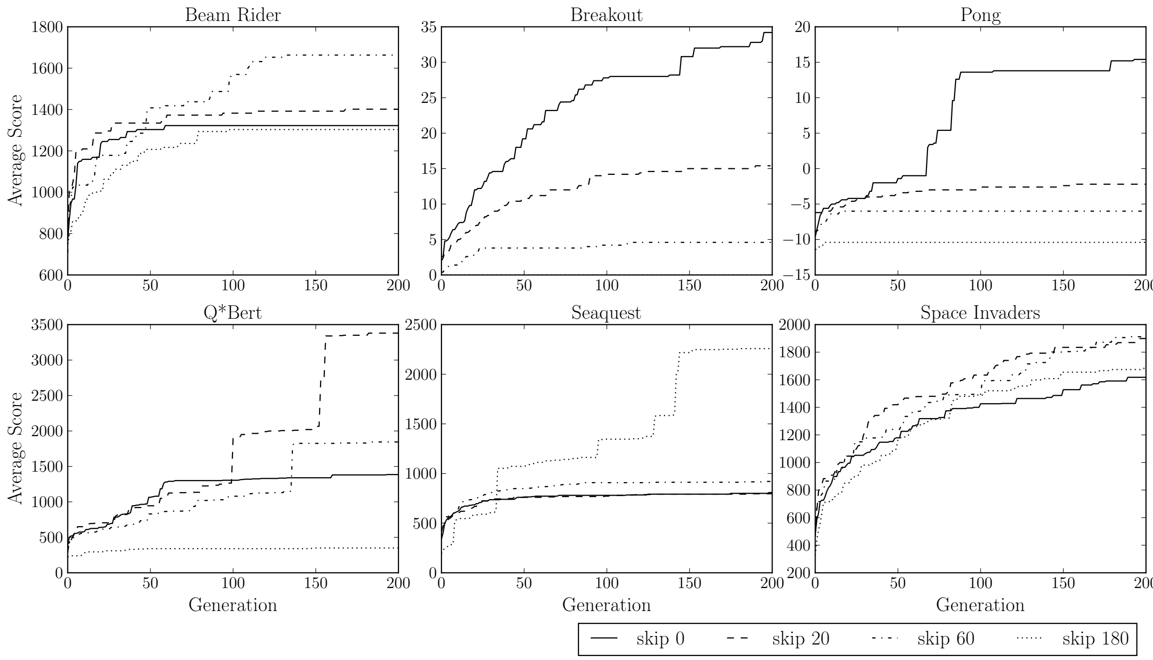

Figure 1: ESP average scores over five runs by generation for each of the six games.

	E15	E20	HNEAT	DQN
Beam Rider	1663.2_{60}	1663.2_{60}	1736.8_{0}	$\mathbf{4092}_{2}$
Breakout	30.8_{0}	34.2_{0}	43.6_{0}	$\mathbf{168}_{2}$
Pong	13.8_{0}	15.4_{0}	15.2_{0}	$\mathbf{20}_{2}$
Q*Bert	2020_{20}	$\mathbf{3380}_{20}$	2165_{0}	1952_{2}
Seaquest	2218_{180}	2258_{180}	$\mathbf{2508}_{0}$	1705_{2}
S. Invaders	1835_{20}	$\mathbf{1912}_{60}$	1481_{0}	581_{3}

Table 1: Average scores for ESP v. existing approaches. The frame skip used is subscripted for each. E15 and E20 are average ESP scores after 150 and 200 generations. E15 is used for comparison to HNEAT, which ran 150 gens.; E20 shows that scores continue to improve with more training.

Breakout with frame skip 180 always scored 0. Table 1 compares our results to previous approaches.

Parameter search techniques could be used to find a ‘good enough’ frame skip for each game, but perhaps for some games there is no single best static frame skip. A more adaptive possibility is for the algorithm to adjust the frame skip based on learning progress. Taking this one step further, RL agents could be extended to specify, each time they interact with ALE, *both* an action and the number of frames they would like to skip before the next interaction. A related idea has been investigated in the Atari domain with respect to Monte-Carlo Tree Search (Vafadost 2013), in which the planner can take an action repeated k times as a macro-action. In neuroevolution, one approach to this problem could be to include an additional output node whose output is mapped into a range of possible frame skips. The experiments presented above are by no means exhaustive, but they lead us to conclude that frame skip is a powerful parameter for learning to play Atari. It is currently intractable for general methods to achieve human performance on all Atari 2600 games at 60Hz. Harnessing frame skip could be a key ingredient to tractability and future success.

Acknowledgements We’d like to thank Matthew Hausknecht and Joel Lehman for their help and useful discussions. This work was supported in part by an NPSC Fellowship sponsored by NSA.

References

Bellemare, M. G.; Naddaf, Y.; Veness, J.; and Bowling, M. 2013. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research* 47:253–279.

Gomez, F., and Miikkulainen, R. 1997. Incremental evolution of complex general behavior. *Adaptive Behavior* (5):317–342.

Gomez, F. J., and Schmidhuber, J. 2005. Co-evolving recurrent neurons learn deep memory pomdps. In *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation (GECCO)*, 491–498.

Hausknecht, M.; Lehman, J.; Miikkulainen, R.; and Stone, P. 2013. A neuroevolution approach to general atari game playing. In *IEEE Transactions on Computational Intelligence and AI in Games*.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; and Riedmiller, M. 2013. Playing atari with deep reinforcement learning. *CoRR* abs/1312.5602.

Schmidhuber, J.; Wierstra, D.; Gagliolo, M.; and Gomez, F. 2007. Training recurrent networks by evolino. *Neural computation* 19(3):757–779.

Vafadost, M. 2013. Temporal abstraction in monte carlo tree search. Master’s thesis, Department of Computer Science, University of Alberta.