ACCELERATED DISTRIBUTED REINFORCEMENT LEARNING

SAAIF AHMED

REINFORCEMENT LEARNING

- Want to improve actions by using observations recorded. Take actions into a “(FIFO) replay buffer” that then passes to an algorithm for learning
- Difficult due to scale of data needed to learn. Millions of data points to accurately display the problem.
    - Other issues associated with this. Noisy data since useful since is larger than supervised size
    - Must be fresh data that is up to date with “policy” of the actions

WHAT DOES DISTRIBUTED REINFORCEMENT LEARNING SOLVE?

- Feed faster into the replay buffer for learning improvement.
- Train larger data models due to parallel gradient descent. Reduce noise in the gradient
- The field of RL is highly parallelizable

USING RAPTOR ON THE PREFIX CIRCUITS

- Raptor is a powerful tool from NVIDIA for Distributed Learning as it can extend computation across multiple GPUs, has a powerful replay buffer, generate new learned events and then send that updated gradient to other GPUs for further computation/prediction.
- Apply Raptor to Prefix Circuits
  - Circuits within GPUs that perform arithmetic. Change order of operations for processes like binary addition to better suit our needs. Model correctness is maintained.

APPLYING RAPTOR

- Distribute computation with Raptor synchronously.
- Then use many “reward CPUs”, other machines, that calculate rewards for each state of the learning. Then this can be prepared and made ready for training.
- The Raptor loop: CPU environments take outputs to the Raptor buffer which then generates batches of training, trains them on multiple GPUs, then does prediction with them on another set of GPU’s (NCCL Broadcasting). Sends computation back and waits for the previous states rewards for further training.

SCALING

- Raptor solves the problem of being able to refresh data as fast as possible.
- Larger gains are to be made by increasing actions per second.
- Limit replay ratio. Reinforcement Learning reaches convergence through many learned experiences rather than a hyper tuned small set.

REFERENCES

- Jonathan Raiman (2022), *Accelerating Distributed Reinforcement Learning. NVIDIA*. Retrieved April 3, 2022.