Leveraging Communication Topologies Between Learning Agents in Deep Reinforcement Learning

Extended Abstract

Dhaval Adjodah
MIT Media Lab
dval@mit.edu

Dan Calacci
MIT Media Lab
dcalacci@media.mit.edu

Abhimanyu Dubey
MIT Media Lab
dubeya@mit.edu

Anirudh Goyal
MILA/Université de Montréal
anirudhgoyal9119@gmail.com

P. M. Krafft
Oxford Internet Institute
p.krafft@oii.ox.ac.uk

Esteban Moro
MIT Media Lab
Universidad Carlos III de Madrid
emoro@mit.edu

Alex Pentland
MIT Media Lab
pentland@mit.edu

ACM Reference Format:
Dhaval Adjodah, Dan Calacci, Abhimanyu Dubey, Anirudh Goyal, P. M. Krafft, Esteban Moro, and Alex Pentland. 2020. Leveraging Communication Topologies Between Learning Agents in Deep Reinforcement Learning. In *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, May 9–13, 2020,* IFAAMAS, 3 pages.

A common technique to improve learning performance in deep reinforcement learning (DRL) and many other machine learning algorithms is to run multiple learning agents in parallel [7, 11]. A neglected component in the development of these algorithms has been how best to arrange the learning agents involved to improve distributed search [1–3, 13]. Here we draw upon results from the networked optimization literatures [4–6] suggesting that arranging learning agents in communication networks other than fully connected topologies (the implicit way agents are commonly arranged in) can improve learning. As shown in Fig. 2, our intuition is that decentralized communication topologies will lead to clusters of agents searching different parts of the landscape simultaneously.

Given that network effects are generally only significant with large numbers of agents, we choose to build upon one of the DRL algorithms most oriented towards parallelizability and scalability: Evolution Strategies (ES) [8–10, 12]. To ensure that none of the modifications we implemented to the ES paradigm to create our novel Networked Evolution Strategies (NetES) algorithm are causing improvements in performance, we run a careful ablation study to control for each modification separately.

Empirically, using the MuJoCo benchmark tasks with 100 agents, we evaluate NetES on each of the 4 families of communication topology: Erdos-Renyi, scale-free, small-world and the standard fully-connected network. As seen in Fig. 1A, Erdos-Renyi strongly outperforms the other topologies.

Given the superiority of Erdos-Renyi networks, we focus on them for all other empirical results going forward - this choice is supported by our theoretical results which indicate that Erdos-Renyi would do better on any task.

We run larger networks of 1000 agents on all 5 benchmark results. As can be seen in Table 1, our Erdos-Renyi networks outperform fully-connected networks on all benchmark tasks, resulting in improvements ranging from 9.8% on MuJoCo Ant-v1 to 798% on MuJoCo Humanoid-v1. All results are statistically significant (based on 95% confidence intervals).

Finally, we investigate whether organizing the communication topology using Erdos-Renyi networks can outperform larger fully-connected networks. We choose one of the benchmarks that had a small difference between the two algorithms at 1000 agents, Roboschool Humanoid-v1. As shown in Fig. 1B, an Erdos-Renyi network with 1000 agents provides comparable performance to 3000 agents arranged in a fully-connected network.

Theoretically, we complement these empirical results with an investigation of why our alternate topologies perform better. We formalize the capacity of a communication topology to explore the parameter space as the diversity of parameter updates during each iteration, which can be measured by the variance of parameter updates. We provide the proof in the full paper[1].

Theorem. *In a NetES update iteration t for a system with N agents with parameters $\Theta = \{\theta_1^{(t)}, ..., \theta_N^{(t)}\}$, agent communication matrix $A = \{a_{ij}\}$, agent-wise perturbations $\mathcal{E} = \{\epsilon_1^{(t)}, ..., \epsilon_N^{(t)}\}$, and parameter update $u_i^{(t)} = \frac{\alpha}{N\sigma^2} \sum_{j=1}^{N} a_{ij} \cdot \big(R(\theta_j^{(t)} + \sigma\epsilon_j^{(t)}) \cdot ((\theta_j^{(t)} + \sigma\epsilon_j^{(t)}) - (\theta_i^{(t)}))\big)$, the following relation holds:*

$$\mathrm{Var}_i[u_i^{(t)}] \leq \frac{\max^2 R(\cdot)}{N\sigma^4} \Big\{ \Big(\frac{\|A^2\|_F}{(\min_l |A_l|)^2} \Big) \cdot f(\Theta, \mathcal{E}) - \Big(\frac{\min_l |A_l|}{\max_l |A_l|} \Big)^2 \cdot \frac{\sigma^2}{N} \big(\sum_{i,j} \epsilon_i^{(t)} \epsilon_j^{(t)} \big) \Big\} \quad (1)$$

Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), B. An, N. Yorke-Smith, A. El Fallah Seghrouchni, G. Sukthankar (eds.), May 9–13, 2020, Auckland, New Zealand.

[1] Code, JSON experiment files and supplementary available at github.com/d-val/NetES

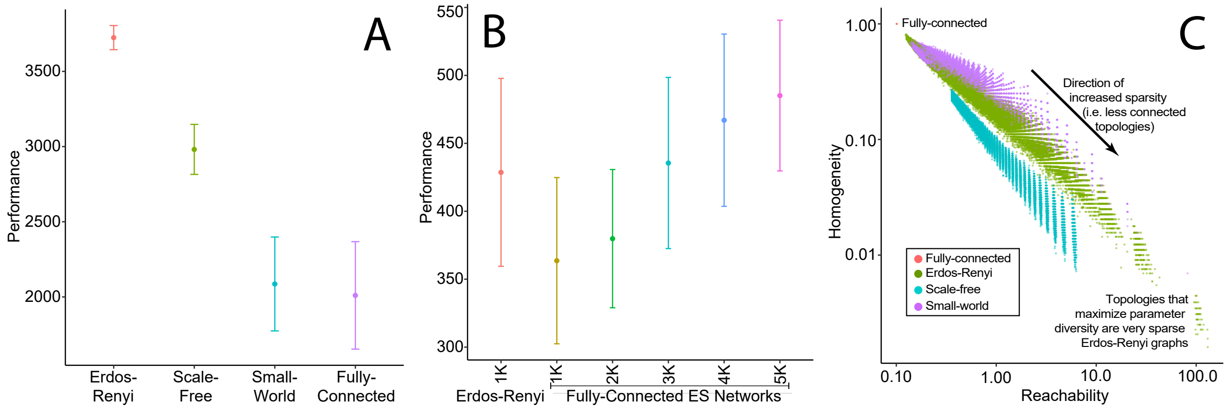

Figure 1: A: Erdos-Renyi graphs do best, fully-connected graphs do worst. B: Erdos-Renyi graphs with 1000 agents compared to varying size fully-connected networks. C: Sparser Erdos-Renyi graphs maximize diversity of parameter updates.

Here, $|A_l| = \sum_j a_{jl}$, *and* $f(\Theta, \mathcal{E}) =$
$$\sqrt{(\sum_{j,k,m}((\theta_j^{(t)} + \sigma\epsilon_j^{(t)} - \theta_m^{(t)}) \cdot (\theta_k^{(t)} + \sigma\epsilon_k^{(t)} - \theta_m^{(t)}))^2)}.$$

In this work, our hypothesis is to test if some networks *could* do better than the de facto fully-connected topologies used in state-of-the-art algorithms. We leave to future work the important question of optimizing the network topology for maximum performance. Doing so would require a lower bound, as it would provide us the *worst-case* performance of a topology.

Instead, in this section, we are interested in providing insights into why some networks *could* do better than others, which can be understood through our upper-bound, as it allows us to understand the *capacity* for parameter exploration of a network topology.

From this result, we see that the diversity of exploration across agents is likely affected by two quantities that involve the connectivity matrix A: the first being the term $(\|A^2\|_F/(\min_l |A_l|))^2$ (henceforth referred to as the *reachability* of the network), which according to our bound we want to maximize, and the second being $(\min_l |A_l|/\max_l |A_l|)^2$ (henceforth referred to as the *homogeneity* of the network), which according to our bound we want to be as small as possible in order to maximize the diversity of parameter updates across agents. Reachability and homogeneity are not independent, and are statistics of the degree distribution of a graph. It is interesting to note that the upper bound *does not depend on the reward landscape* $R(\cdot)$ *of the task at hand*, indicating that our theoretical insights should be independent of the learning task.

Using the above definitions for reachability and homogeneity, we generate random instances of each network family, and plot them in Fig. 1C. Two main observations can be made from this simulation: 1) Erdos-Renyi networks maximize reachability and minimize homogeneity, which means that they likely maximize the diversity of parameter exploration. 2) Fully-connected networks are the single worst network in terms of exploration diversity (they minimize reachability and maximize homogeneity, the opposite of what would be required for maximizing parameter exploration according to the suggestion of our bound).

These theoretical results agree with our empirical results: Erdos-Renyi networks perform best, followed by scale-free networks, while fully-connected networks do worse.

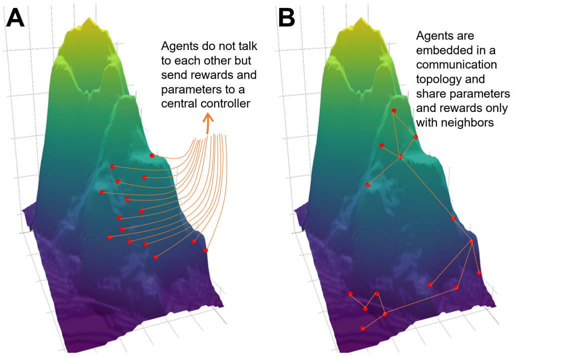
A
Agents do not talk to each other but send rewards and parameters to a central controller
B
Agents are embedded in a communication topology and share parameters and rewards only with neighbors

Figure 2: A: In DRL, agents search the same local area and communicate in a fully connected network. B: In NetES, agents are embedded in a sparser topology where clusters of agents search different parts of the landscape.

Type	Task	Fully-connected	Erdos	Improv. %
MuJoCo	Ant-v1	4496	4938	**9.8**
MuJoCo	HalfCheetah-v1	1571	7014	**346.3**
MuJoCo	Hopper-v1	1506	3811	**153.1**
MuJoCo	Humanoid-v1	762	6847	**798.6**
Roboschool	Humanoid-v1	364	429	**17.9**

Table 1: Improvements from Erdos-Renyi networks with 1000 nodes compared to fully-connected networks.

In summary, we extended ES, a DRL algorithm, to use alternate network topologies and empirically showed that the de facto fully-connected topology performs worse in our experiments. We also performed an ablation study by running controls on all the modifications we made to the ES algorithm, and we showed that the improvements we observed are not explained away by modifications other than the use of alternate topologies (ablation results in full paper). Finally, we provided theoretical insights into why alternate topologies may be superior, and observed that our theoretical predictions are in line with our empirical results. Future work could explore the use of dynamical topologies where agent connections are continuously rewired to adapt to the local terrain of the research landscape.

REFERENCES

[1] Ricardo M Araujo and Luis C Lamb. 2008. Memetic Networks: Analyzing the Effects of Network Properties in Multi-Agent Performance.. In *AAAI*, Vol. 8. 3–8.

[2] Tang Jing, Meng Hiot Lim, and Yew Soon Ong. 2004. Island model parallel hybrid-ga for large scale combinatorial optimization. In *Proceedings of the 8th International Conference on Control, Automation, Robotics and Vision, Special Session on Computational Intelligence on the Grid.*

[3] Sergio Valcarcel Macua, Aleksi Tukiainen, Daniel García-Ocaña Hernández, David Baldazo, Enrique Munoz de Cote, and Santiago Zazo. 2017. Diff-DAC: Distributed Actor-Critic for Multitask Deep Reinforcement Learning. *arXiv preprint arXiv:1710.10363* (2017).

[4] Angelia Nedic. 2011. Asynchronous broadcast-based convex optimization over a network. *IEEE Trans. Automat. Control* 56, 6 (2011), 1337–1351.

[5] Angelia Nedić, Alex Olshevsky, and Michael G Rabbat. 2017. Network Topology and Communication-Computation Tradeoffs in Decentralized Optimization. *arXiv preprint arXiv:1709.08765* (2017).

[6] Angelia Nedic and Asuman Ozdaglar. 2010. 10 cooperative distributed multi-agent. *Convex Optimization in Signal Processing and Communications* 340 (2010).

[7] OpenAI. 2018. OpenAI Five. https://blog.openai.com/openai-five/. (2018).

[8] Ingo Rechenberg. 1973. Evolution Strategy: Optimization of Technical systems by means of biological evolution. *Fromman-Holzboog, Stuttgart* 104 (1973), 15–16.

[9] Tim Salimans, Jonathan Ho, Xi Chen, and Ilya Sutskever. 2017. Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864* (2017).

[10] Hans-Paul Schwefel. 1977. *Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie: mit einer vergleichenden Einführung in die Hill-Climbing-und Zufallsstrategie.* Birkhäuser.

[11] Oriol Vinyals, Igor Babuschkin, Junyoung Chung, Michael Mathieu, Max Jaderberg, Wojtek Czarnecki, Andrew Dudzik, Aja Huang, Petko Georgiev, Richard Powell, Timo Ewalds, Dan Horgan, Manuel Kroiss, Ivo Danihelka, John Agapiou, Junhyuk Oh, Valentin Dalibard, David Choi, Laurent Sifre, Yury Sulsky, Sasha Vezhnevets, James Molloy, Trevor Cai, David Budden, Tom Paine, Caglar Gulcehre, Ziyu Wang, Tobias Pfaff, Toby Pohlen, Dani Yogatama, Julia Cohen, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy Lillicrap, Chris Apps, Koray Kavukcuoglu, Demis Hassabis, and David Silver. 2019. AlphaStar: Mastering the Real-Time Strategy Game StarCraft II. https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/. (2019).

[12] Daan Wierstra, Tom Schaul, Tobias Glasmachers, Yi Sun, Jan Peters, and Jürgen Schmidhuber. 2014. Natural evolution strategies. *The Journal of Machine Learning Research* 15, 1 (2014), 949–980.

[13] Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Başar. 2018. Fully decentralized multi-agent reinforcement learning with networked agents. *arXiv preprint arXiv:1802.08757* (2018).