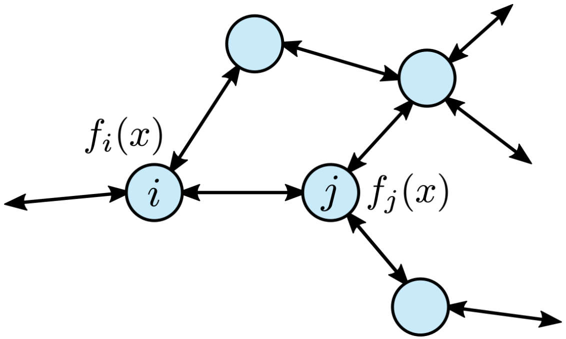
Distributed Zero-Order Algorithms for Nonconvex Optimization

Yujie Tang, Junshan Zhang and Na Li

Distributed Zero-Order Optimization



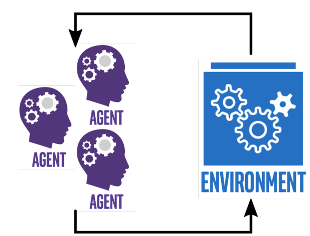
- Minimize $\frac{1}{n}\sum_i f_i(x)$
- One only inquires **local objective values,** not gradient
- **Nonconvex** objectives

→ ***distributed*** + ***zero-order*** **optimization**

robot swarms

wind farm operation

networking



multi-agent reinforcement learning

Existing Literature (Selected)

distributed first-order

- decentralized gradient descent

  [Tsitsiklis 1986] [Nedic 2009] [Chen 2012] [Lian 2017]

- push-sum algorithms

  [Nedic 2016] [Tatarenko 2017]

- EXTRA / gradient tracking

  [Shi 2015] [Xu 2015] [Di Lorenzo 2016] [Nedic 2017] [Qu 2018] [Pu 2018]

- ADMM / method of multipliers

  [Boyd 2011] [Hong 2017] [Wang 2019]

centralized zero-order

- deterministic, two points

  [Nesterov 2017]

- stochastic/online, two points

  [Duchi 2015] [Shamir 2017] [Liu 2018]

- stochastic/online, single point

  [Flaxman 2005] [Bach 2016]

distributed zero-order

- [Hajinezhad 2017]

  nonconvex, unconstrained, method of multipliers

- [Yu 2019]

  convex, constrained, simple consensus

- [Sahu 2018]

  strongly convex, unconstrained, simple consensus, noisy zero-order info

Distributed + Zero-Order = ?

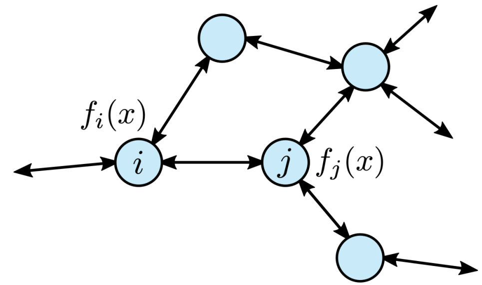
		smooth	smooth & strongly convex
distributed first-order	DGD/push-sum	$O\left(\frac{\log t}{\sqrt{t}}\right)$ (convex) $O\left(\frac{1}{\sqrt{T}}\right)$ (nonconvex)	$O\left(\frac{\log t}{t}\right)$
	gradient tracking	$O\left(\frac{1}{t}\right)$	$O\left(\left[1-c(1-\rho)^2\left(\frac{\mu}{L}\right)^{\frac{3}{2}}\right]^t\right)$
centralized zero-order	noiseless, two-point	$O\left(\frac{d}{N}\right)$	$O\left(\left[1-\frac{c}{d}\frac{\mu}{L}\right]^N\right)$

- How do distributed and zero-order affect each other?
- Can we keep **fundamental** structural properties by tuning the way of combination?

Distributed First-Order Methods

$$\min_{x\in\mathbb{R}^d} \frac{1}{n}\sum_i f_i(x)$$

smooth objectives



Gossip matrix

$$W = [W_{ij}] \in \mathbb{R}^{n\times n}$$

- doubly stochastic

Decentralized Gradient Descent (DGD)

$$x_t^i = \underbrace{\sum_j W_{ij}\, x_{t-1}^j}_{\text{averaging}} - \underbrace{\eta_t \nabla f_i(x_{t-1}^i)}_{\text{local gradient}}$$

convergence rates:

- smooth: $O\left(\log t/\sqrt{t}\right)$
- smooth & strongly convex: $O(1/t)$

DGD with Gradient Tracking

$$s_t^i = \sum_j W_{ij}\, s_{t-1}^j + \nabla f_i(x_{t-1}^i) - \nabla f_i(x_{t-2}^i)$$

$$x_t^i = \sum_j W_{ij}\, x_{t-1}^j - \eta s_t^i$$

gradient tracking

convergence rates:

- smooth: $O(1/t)$
- smooth & strongly convex: $O(\lambda^t)$

Zero-Order Optimization: Gradient Estimation

$f : \mathbb{R}^d \to \mathbb{R}$ differentiable.

$2d$-point estimator:

$$\mathsf{G}_f^{(2d)}(x;u) := \sum_{k=1}^{d} \frac{f(x+ue_k)-f(x-ue_k)}{2u} e_k$$

$\{e_k\}_{k=1}^d$: standard basis

- A straightforward estimator
- Works well when dimension is small
- Does not scale well as dimension becomes large

2-point estimator:

$$\mathsf{G}_f^{(2)}(x;u,z) := d\,\frac{f(x+uz)-f(x-uz)}{2u}\,z$$

where $z \sim \mathsf{Uni}(\mathbb{S}_{d-1})$

- Property [Flaxman2005]:

$$\mathbb{E}_z\left[\mathsf{G}_f^{(2)}(x;u,z)\right] := \nabla f^u(x)$$

where $f^u(x) := \frac{1}{|\mathbb{B}_d|}\int_{\mathbb{B}_d} f(x+uy)\,dy$

$\mathsf{G}_f^{(2)}$ gives a **stochastic gradient** of "smoothed" f

Zero-Order Optimization: Gradient Estimation

$f : \mathbb{R}^d \to \mathbb{R}$ differentiable.

$2d$-point estimator:

$$\mathsf{G}_f^{(2d)}(x; u) := \sum_{k=1}^{d} \frac{f(x+ue_k) - f(x-ue_k)}{2u} e_k$$

$\{e_k\}_{k=1}^d$: standard basis

- A straightforward estimator
- Works well when dimension is small
- Does not scale well as dimension becomes large

2-point estimator:

$$\mathsf{G}_f^{(2)}(x; u, z) := d\, \frac{f(x+uz) - f(x-uz)}{2u}\, z$$

where $z \sim \mathsf{Uni}(\mathbb{S}_{d-1})$

- Scales well with the dimension
- Variance cannot be arbitrarily small
- Achieves comparable convergence rates with first-order counterparts
  - GD + $\mathsf{G}_f^{(2)} \implies O(d/N)$

Algorithm 1 (based on 2-point + DGD)

for $t = 1, 2, 3, \ldots$ **do**

  foreach $i \in \{1, 2, \ldots, n\}$ **do**

  1. Generate $z_t^i \sim \mathsf{Uni}(\mathbb{S}_{d-1})$
  2. Update x_t^i by

$$
\begin{aligned}
g_t^i &= \mathsf{G}_{f_i}^{(2)}(x_{t-1}^i; u_t, z_t^i) \\
&= d \cdot \frac{f_i(x_{t-1}^i + u_t z_t^i) - f_i(x_{t-1}^i - u_t z_t^i)}{2u_t} z_t^i
\end{aligned}
$$

$$
x_t^i = \sum_{j=1}^{n} W_{ij}(x_{t-1}^j - \eta_t g_t^j)
$$

  end

end

2-point estimator

+

DGD

simple, but possibly *slow* convergence
[recall $O(\log t/\sqrt{t})$ for DGD]

Algorithm 2 (based on $2d$-point + gradient tracking)

Set $s^i(0) = g^i(0) = 0$ for each $i \in \{1, \ldots, N\}$.
for $t = 1, 2, 3, \ldots$ **do**
  foreach $i \in \{1, 2, \ldots, n\}$ **do**

1. Update s_t^i by

$$g_t^i = \mathsf{G}_{f_i}^{(2d)}(x_{t-1}^i; u_t) = \sum_{k=1}^{d} \frac{f_i(x_{t-1}^i + u_t e_k) - f_i(x_{t-1}^i - u_t e_k)}{2u_t} e_k$$

$$s_t^i = \sum_{j=1}^{n} W_{ij} \left(s_{t-1}^j + g_t^j - g_{t-1}^j \right)$$

2. Update x_t^i by

$$x_t^i = \sum_{j=1}^{n} W_{ij} (x_{t-1}^j - \eta s_t^j)$$

  end
end

$2d$-**point** estimator

+

DGD with
gradient tracking

- possibly ***faster*** convergence [recall $O(1/t)$ for gradient tracking]
- # of zero-order queries per iteration has ***worse dependence*** on d

Distributed Zero-Order Optimization: Algorithms

- Alg. 1

$$g_t^i = \mathsf{G}_{f_i}^{(2)}(x_{t-1}^i; u_t, z_t^i) = d \cdot \frac{f_i(x_{t-1}^i + u_t z_t^i) - f_i(x_{t-1}^i - u_t z_t^i)}{2u_t} z_t^i$$

$$x_t^i = \sum_{j=1}^{n} W_{ij}(x_{t-1}^j - \eta_t g_t^j)$$

2-point estimator

+

DGD

- Alg. 2

$$g_t^i = \mathsf{G}_{f_i}^{(2d)}(x_{t-1}^i; u_t) = \sum_{k=1}^{d} \frac{f_i(x_{t-1}^i + u_t e_k) - f_i(x_{t-1}^i - u_t e_k)}{2u_t} e_k$$

$$s_t^i = \sum_{j=1}^{n} W_{ij} \left(s_{t-1}^j + g_t^j - g_{t-1}^j \right)$$

$$x_t^i = \sum_{j=1}^{n} W_{ij}(x_{t-1}^j - \eta s_t^j)$$

$2d$-point estimator

+

DGD with gradient tracking

Convergence Rates

$$\min_{1\le\tau\le t} \mathbb{E}\left[\|\nabla f(\bar{x}_\tau)\|^2\right] + \sum_i \mathbb{E}\left[\|x_t^i - \bar{x}_t\|^2\right]$$

		smooth	gradient-dominated
this work (nonconvex)	Alg. 1	$O\left(\sqrt{\frac{d}{N}}\log N\right)$	$O\left(\frac{d}{N}\right)$
	Alg. 2	$O\left(\frac{d}{N}\right)$	$O\left(\left[1-c(1-\rho^2)^2\left(\frac{\mu}{L}\right)^{\frac{4}{3}}\right]^{N/d}\right)$
distributed first-order	DGD	$O\left(\frac{\log t}{\sqrt{t}}\right)$	$O\left(\frac{1}{t}\right)$ (str. convex)
	gradient tracking	$O\left(\frac{1}{t}\right)$	$O\left(\left[1-c(1-\rho)^2\left(\frac{\mu}{L}\right)^{\frac{3}{2}}\right]^{t}\right)$ (str. convex)
centralized zero-order	noiseless, two-point	$O\left(\frac{d}{N}\right)$	$O\left(\left[1-\frac{c}{d}\frac{\mu}{L}\right]^{N}\right)$ (str. convex)

d: problem dimension, N: number of inquiries of function values

$f:\mathbb{R}^d\to\mathbb{R}$ is called **gradient-dominated** if

$$f(x)-f(x^*)\le\frac{1}{2\mu}\|\nabla f(x)\|^2$$

for some $\mu>0$ where x^* is a global minimizer

- nonconvex counterpart of **strong convexity**
- eg: LQR cost as a function of feedback gain [Fazel 2018]

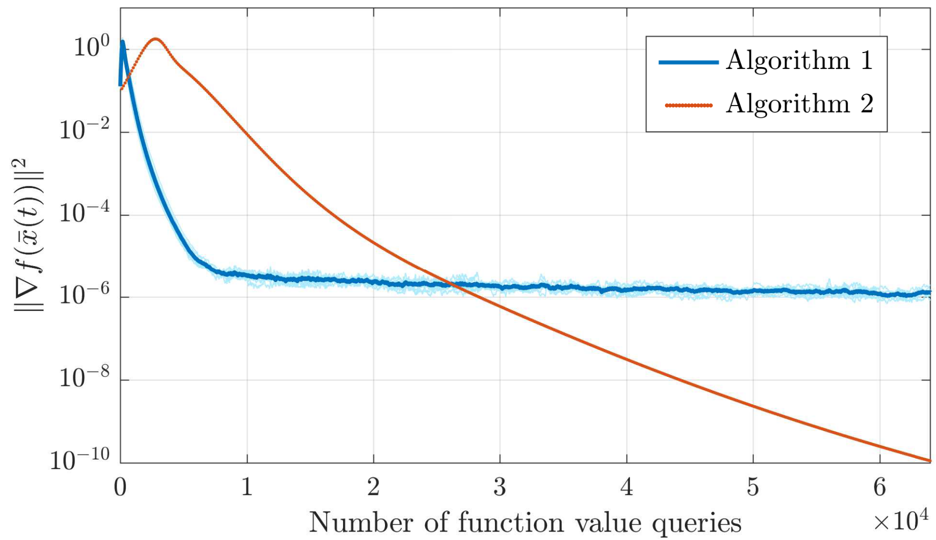
Numerical Examples

- Synthesized distributed phase retrieval:

$$\min_{x\in\mathbb{R}^d} \quad \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$

$$f_i(x) := \frac{1}{m}\sum_{k=1}^{m}\left(y_{ik}^2 - |a_{ik}^T x|^2\right)^2$$

- $a_{ik} \in \mathbb{C}^d$ are complex standard Gaussian
- dimension: $d = 64$
- # of agents: $n = 50$
- # of samples per agent: $m = 30$



Summary

- Two **distributed zero-order** algorithms: deterministic zero-order information
- Convergence rates for **nonconvex** objectives
    - general smooth / gradient dominated
- Dependence on **problem dimension**
- Comparison with 1) distributed first-order and 2) centralized zero-order

Open Questions

- Recall for centralized zero-order:
  - ➢ constant # of zero-order queries per iteration ➡ Alg. 1 only
  - ➢ $O(d/N)$ convergence rate ➡ Alg. 2 only

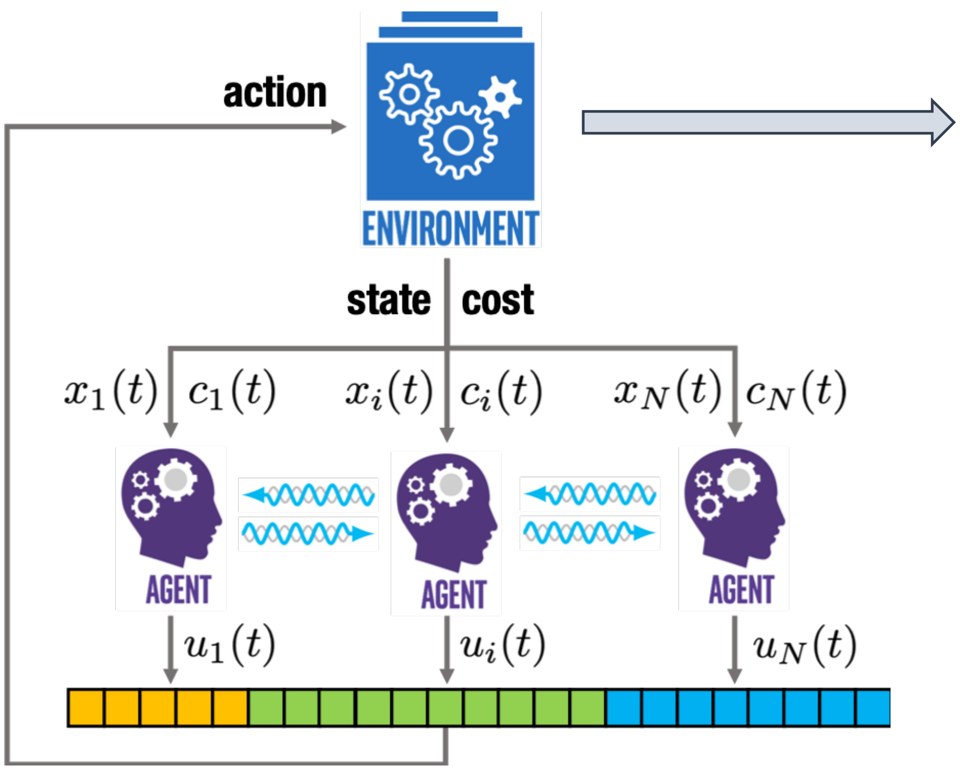
  Is it possible to achieve both for distributed zero-order methods?

- Noisy zero-order information: a much harder problem!
  - **Single-point estimator** [Flaxman 2005] and its variants

$$\mathsf{G}_f^{(1)}(x;u,z) = d\,\frac{f(x+uz)+\varepsilon}{u}\,z \qquad z \sim \mathsf{Uni}(\mathbb{S}_{d-1})$$

  - Large variance, slower asymptotic convergence
    [Flaxman 2005], [Shamir2013], [Bach 2016]

Multi-Agent Reinforcement Learning of LQR



LTI dynamics

$$x(t+1) = Ax(t) + Bu(t) + w(t)$$

state action noise

cost

$$c_i(t) = x(t)^\top Q_i\, x(t) + u(t)^\top R_i\, u(t)$$

$$c(t) = \tfrac{1}{N} \textstyle\sum_i c_i(t)$$

control policy

$$u_i(t) = K_i x_i(t)$$

$$\min_{K_1,\ldots,K_N} J(K) := \lim_{T\to\infty} \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} c(t)\right]$$

Policy Gradient

- If we know $\nabla J(K)$, run policy gradient

$$K(s+1) = K(s) - \eta \nabla J(K(s))$$

starting from some stabilizing controller known a priori

- What each agent can actually do:
  1. Apply a policy
  2. Observe local state $x_i(t)$ and cost $c_i(t)$ for an episode of finite length
  3. Update the policy and iterate

How to bridge this gap?

zero-order optimization

Gradient estimation: multiple single-point estimators

$$\hat{g}(K) = \frac{1}{T_B} \sum_{b=1}^{T_B} n_K \frac{\tilde{J}(K + u z_b)}{u} z_b \qquad z_1, \ldots, z_{T_B} \sim \mathrm{Uni}(\mathbb{S}_{n_K - 1})$$

Multi-Agent Zero-Order Policy Gradient

- Ensure stability during the learning process?
- Sample complexity?
- Comparison to indirect learning methods?
  - Learn dynamics from partial observations then design the controller

Y. Li, Y. Tang and N. Li, “Multiagent reinforcement learning based on zero-order policy gradient,” coming soon.

References

Yujie Tang, Na Li, “Distributed zero-order algorithms for nonconvex multi-agent optimization”, arXiv: 1908.11444, 2019.

[Tsitsiklis 1986] J. Tsitsiklis, D. Bertsekas and M. Athans, “Distributed asynchronous deterministic and stochastic gradient optimization algorithms,” IEEE Transactions on Automatic Control, vol. 31, no. 9, pp.803–812, 1986.

[Nedic 2009] A. Nedic and A. Ozdaglar, “Distributed subgradient methods for multiagent optimization,” IEEE Transactions on Automatic Control, vol. 54, no. 1, pp. 48–61, 2009.

[Chen 2012] I.-A. Chen, “Fast distributed first-order methods,” Master’s thesis, Massachusetts Institute of Technology, 2012.

[Lian 2017] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu, “Can decentralized algorithms outperform centralized algorithms? A case study for decentralized parallel stochastic gradient descent,” in Proceedings of the 31st International Conference on Neural Information Processing Systems, ser. NIPS’17, 2017, pp. 5336–5346.

[Nedic 2016] A. Nedic and A. Olshevsky, “Stochastic gradient-push for strongly convex functions on time-varying directed graphs,” IEEE Transactions on Automatic Control, vol. 61, no. 12, pp. 3936–3947, 2016.

[Tatarenko 2017] T. Tatarenko and B. Touri, “Non-convex distributed optimization,” IEEE Transactions on Automatic Control, vol. 62, no. 8, pp. 3744–3757, 2017.

[Shi 2015] W. Shi, Q. Ling, G. Wu and W. Yin, “Extra: An exact first-order algorithm for decentralized consensus optimization,” SIAM Journal on Optimization, vol. 25, no. 2, pp.944–966, 2015.

References

[Xu 2015] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, “Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes,” in Proceedings of the 54th IEEE Conference on Decision and Control (CDC), 2015, pp. 2055–2060.

[Di Lorenzo 2016] P. Di Lorenzo and G. Scutari, “NEXT: In-network nonconvex optimization,” IEEE Transactions on Signal and Information Processing over Networks, vol. 2, no. 2, pp. 120–136, 2016.

[Nedic 2017] A. Nedic, A. Olshevsky, and W. Shi, “Achieving geometric convergence for distributed optimization over time-varying graphs,” SIAM Journal on Optimization, vol. 27, no. 4, pp. 2597–2633, 2017.

[Qu 2018] G. Qu and N. Li, “Harnessing smoothness to accelerate distributed optimization,” IEEE Transactions on Control of Network Systems, vol. 5, no. 3, pp. 1245–1260, 2018.

[Pu 2018] S. Pu and A. Nedic, “A distributed stochastic gradient tracking method,” in Proceedings of the 57th IEEE Conference on Decision and Control (CDC), 2018, pp. 963–968.

[Boyd 2011] S. Boyd, N. Parikh, E. Chu, B. Peleato and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” Foundations and Trends® in Machine learning, vol. 3, no. 1, pp.1–122, 2011.

[Hong 2017] M. Hong and Z. Q. Luo, “On the linear convergence of the alternating direction method of multipliers,” Mathematical Programming, vol. 162, no. 1-2, pp.165–199, 2017.

[Wang 2019] Y. Wang, W. Yin and J. Zeng, “Global convergence of ADMM in nonconvex nonsmooth optimization,” Journal of Scientific Computing, vol. 78, no. 1, pp.29–63, 2019.

References

[Nesterov 2017] Y. Nesterov and V. Spokoiny, “Random gradient-free minimization of convex functions,” Foundations of Computational Mathematics, vol. 17,no. 2, pp. 527–566, 2017.

[Duchi 2015] J. C. Duchi, M. I. Jordan, M. J. Wainwright, and A. Wibisono, “Optimalrates for zero-order convex optimization: The power of two function evaluations,” IEEE Transactions on Information Theory, vol. 61, no. 5,pp. 2788–2806, 2015.

[Shamir 2017] O. Shamir, “An optimal algorithm for bandit and zero-order convex optimization with two-point feedback,” Journal of Machine Learning Research, vol. 18, no. 52, pp. 1–11, 2017.

[Liu 2018] S. Liu, J. Chen, P.-Y. Chen, and A. Hero, “Zeroth-order online alternating direction method of multipliers: Convergence analysis and applications,” in Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics, ser. Proceedings of Machine Learning Research, vol. 84. PMLR, 2018, pp. 288–297.

[Flaxman 2005] A. D. Flaxman, A. T. Kalai, and H. B. McMahan, “Online convex optimization in the bandit setting: gradient descent without a gradient,” in Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms, 2005, pp. 385–394.

[Bach 2016] F. Bach and V. Perchet, “Highly-smooth zero-th order online optimization,” in 29th Annual Conference on Learning Theory, ser. Proceedings of Machine Learning Research, vol. 49. PMLR, 2016, pp. 257–283.

[Shamir 2013] O. Shamir. “On the complexity of bandit and derivative-free stochastic convex optimization,” In Conference on Learning Theory, pp. 3–24. 2013.

References

[Hajinezhad 2017] D. Hajinezhad, M. Hong, and A. Garcia, “Zeroth order nonconvex multi-agent optimization over networks,” 2017, arXiv preprint arXiv:1710.09997.

[Yu 2019] Z. Yu, D. W. C. Ho, and D. Yuan, “Distributed randomized gradient-free mirror descent algorithm for constrained optimization,” 2019, arXiv preprint arXiv:1903.04157.

[Sahu 2018] A. K. Sahu, D. Jakovetic, D. Bajovic, and S. Kar, “Distributed zeroth order optimization over random networks: A Kiefer-Wolfowitz stochastic approximation approach,” in Proceedings of the 57th IEEE Conference on Decision and Control (CDC), 2018, pp. 4951–4958.

[Fazel 2018] M. Fazel, R. Ge, S. Kakade and M. Mesbahi, “Global Convergence of Policy Gradient Methods for the Linear Quadratic Regulator,” in Proceedings of the 35th International Conference on Machine Learning, 2018, pp. 1467–1476.