An Adversarially Robust Approach to Security Constrained Optimal Power Flow

Neeraj Bedmutha, Priya L. Donti, J. Zico Kolter
{nbedmuth, pdonti, zkolter} @andrew.cmu.edu

We propose the use of adversarial robustness in neural networks to provide solutions for security constrained power flow optimization problems.

Introduction

It is of great importance for a power system to schedule electricity generation in an optimal and robust manner, but doing so via traditional security-constrained optimal power flow (SCOPF) techniques [1] can be expensive.

$\underset{P_g}{\textbf{minimize}}\ f_c(P_g)$	→	Optimize power costs
s.t. grid constraints	→	Power flow equations (non-convex), equipment limits
possible failures	→	Generator/line outages or contingencies (computationally expensive to incorporate)

Bi-level Optimization Approach

We solve SCOPF via adversarially robust optimization [2] by viewing (continuous relaxations of) contingencies as *attacks* α on the power grid, and power generation P_g as *parameters* that we can adjust to defend against attacks.

Bi-level optimization problem

$$\underset{P_g \in \mathbb{R}^g}{\textbf{minimize}}\ \underset{\alpha \in [0,1]^k}{\textbf{maximize}}\ \ell(P_g, \alpha) + \ell(P_g, 0),$$

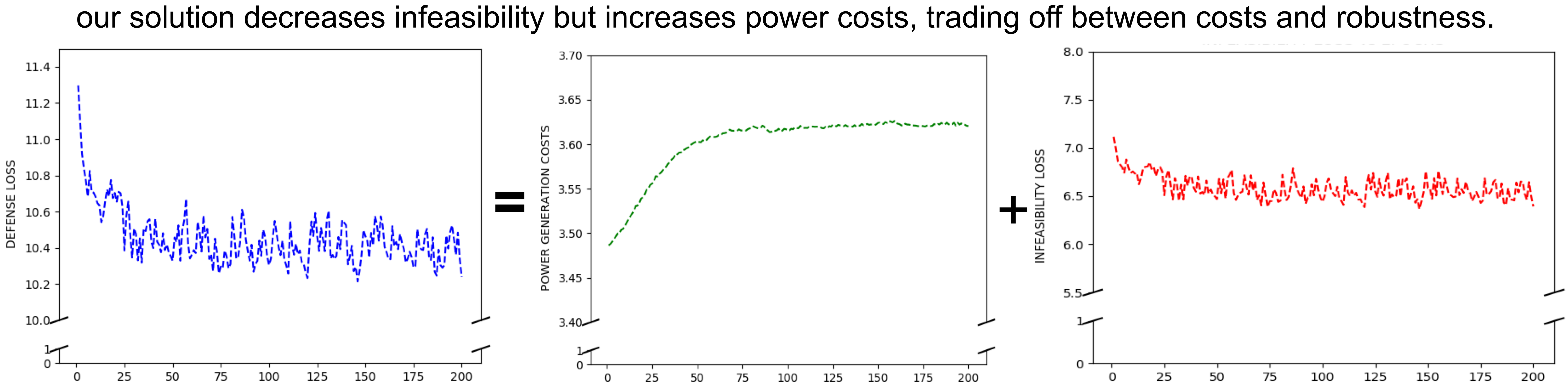
$$\text{where } \ell(P_g, \alpha) = \underbrace{f_c(P_g)}_{\text{Power costs}} + \underbrace{\ell_{line}(P_g, \alpha)}_{\text{Line infeasibilities}} + \underbrace{\ell_{gen}(P_g, \alpha)}_{\text{Generator infeasibilities}}$$

Optimization via gradient descent

- Initialize some $P_g = P_{init}$
- Until convergence:
  1. Solve the inner maximization problem to find an argmax α^*, i.e., the worst case failure for the current setting P_g
  2. Calculate the loss $\ell(P_g, \alpha^*) + \ell(P_g, 0)$
  3. Update $P_g = P_g - \beta \nabla_{P_g} ((\ell(P_g, \alpha^*) + \ell(P_g, 0))$ for some learning rate β
     - Potentially requires implicit differentiation [3] through computation of line flows

Results

Compared to the “base case” optimal power flow solution (that does not account for contingencies), we find that our solution decreases infeasibility but increases power costs, trading off between costs and robustness.



[1] X. Pan, T. Zhao, and M. Chen, “Deepopf: A deep neural network approach for security-constrained dc optimal power flow,”arXiv preprint arXiv:1910.14448, 2019.
[2] Z. Kolter and A. Madry, “Tutorial: Adversarial robustness - theory and practice.”
[3] B. Amos and J. Z. Kolter, “OptNet: Differentiable Optimization as a Layer in Neural Networks,” in Proceedings of the 34th International Conference on Machine Learning-Volume 70, pp. 136–145, JMLR. org, 2017.