Adam-based Augmented Random Search for Control Policies for Distributed Energy Resource Cyber Attack Mitigation

Sy-Toan Ngo

Grid Integration Group (GIG), ESDR

SPADES Workshop 17^{th} December 2021

Outline

BERKELEY LAB

Acknowledgements

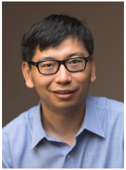
LBNL staffs who supported this work:

Dan Arnold

Sy-Toan Ngo

Sean Peisert

Ciaran Roberts



Yize Chen

BERKELEY LAB

Acknowledgements

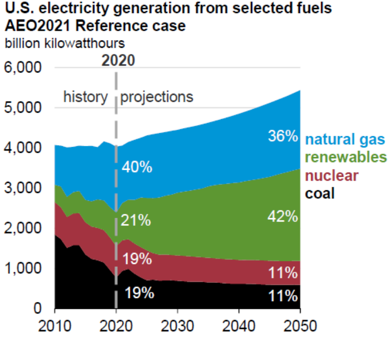
This work was conducted in partnership with the following organizations:

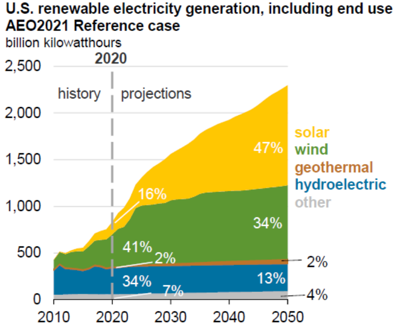
David Pinney
Lisa Slaughter

Anna Scaglione
Ignacio Losada
Tong Wu

Bruno Leao
Tobias Ahlgrim
Siddharth Bhela

Huge growth of solar (PV) as a source of electricity in U.S.





https://www.eia.gov/outlooks/aeo/pdf/04%20AEO2021%20Electricity.pdf

Growth of PV in Distribution Systems

IEEE STANDARDS ASSOCIATION ◆IEEE

IEEE Standard for Interconnection and Interoperability of Distributed Energy Resources with Associated Electric Power Systems Interfaces

IEEE Standards Coordinating Committee 21

Sponsored by the
IEEE Standards Coordinating Committee 21 on Fuel Cells, Photovoltaics, Dispersed Generation, and Energy Storage

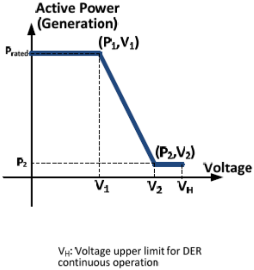
PV resource is highly distributed

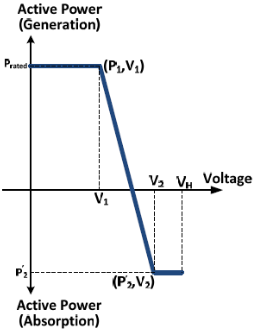
1547 establishes guidelines for PV system voltage and frequency support and ride-through behavior

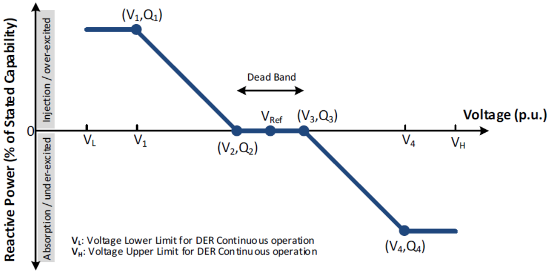
BERKELEY LAB

Smart Inverter Voltage Regulation Controllers

Volt-Watt Curve







Volt-VAR curve

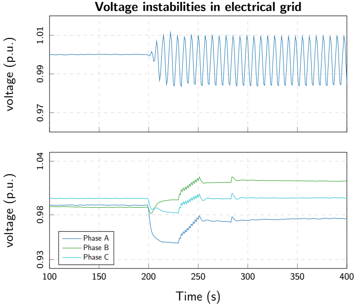
Motivation

- Autonomous control of DERs via Internet, cellular, or power line carrier connectivity exposes the power system to cyber vulnerability.
- In Hawaii (2015), 800,000 micro-inverters are remotely controlled on Oahu in one day
- An increase in the number and type of DERs (PV inverters, batteries, ...) integrate into the power system
- Improper settings in *a portion* of DERs can lead to voltage instabilities
- Voltage instabilities can cause damage to devices, cause device trips, and harm power quality

If the DERs were compromised, what would happened?
How to mitigate potential attacks?

BERKELEY LAB

Motivation



- Bad configuration of inverters can lead to voltage instabilities.
- The system is non-linear, non-convex and dynamic (thousand of DERs).
- Reinforcement learning is a suitable approach for this tasks.

Previous works - Publications

- Deep Reinforcement Learning (DRL) for DER Cyber-Attack Mitigation (SmartGridComm 2020)- Using DRL to mitigate voltage oscillation.
- Deep Reinforcement Learning for Mitigating Cyber-Physical DER Voltage Unbalance Attacks (ACC 2021) - Using DRL to mitigate voltage imbalance.
- Open-source framework PyCIGAR - a reinforcement learning framework to train agents to use non-compromised DER to mitigate voltage instability

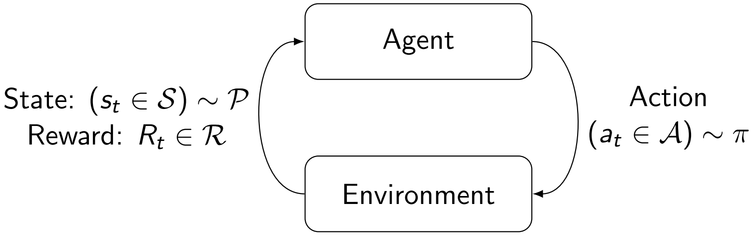
BERKELEY LAB

Previous works

- We trained PPO (a method of DRL) agents to control the DERs to mitigate oscillation voltage and imbalance voltage
- However, reinforcement learning algorithms require a lot of simulations, we need to develop an efficient method.

PPO	Random Search
Value Function, Policy	Policy

Random Search

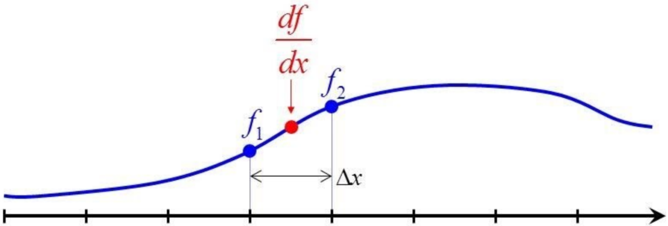


- Rollout simulation multiple times with small fluctuation in the policy parameters θ to approximate the gradient of the return
- Learn the new set of policy parameters θ^* with gradient ascent

Random Search

$$\boxed{\frac{df_{1.5}}{dx} \approx \frac{f_2 - f_1}{\Delta x}}$$

second-order accurate
first-order derivative



Finite-difference approximation

Random Search

Hyperparameters: no. of directions per iteration N,
exploration noise ν, learning rate α

Initialize: π_θ is linear or non-linear policy with parameters 0

while *ending condition not satisfied* **do**

Sample $\delta_1, \delta_2, ..., \delta_N$ i.i.d. standard normal entries

Collect $2N$ rollouts of horizon H and their corresponding rewards using the $2N$ policies.

- Sample the rollouts with policy $\pi_{\theta_j \pm \nu\delta_k}(\tilde{x})$

$$g_j = \frac{\alpha}{N}\sum_{k=1}^{N}[r(\pi_{\theta_j,k,+}) - r(\pi_{\theta_j,k,-})]\delta_k$$

$$\theta_{j+1} = \theta_j + \alpha g_j$$

end

i.i.d: Independently Identically Distributed.

Augmented Random Search

Augmented Random Search proposes 3 improvements:

- Normalization of the states
- Scaling the gradient by the standard deviation of return
- Using top performing directions in mini-batch updates

Augmented Random Search

Hyperparameters: no. of directions per iteration N, exploration noise ν, number of top directions b ($b \leq N$)

Initialize: π_θ is linear or non-linear policy with parameters 0

while *ending condition not satisfied* **do**

Sample $\delta_1, \delta_2, ..., \delta_N$ i.i.d. standard normal entries

Collect $2N$ rollouts of horizon H and their corresponding rewards using the $2N$ policies.

- Normalization of the states $\tilde{x}$
- Sample the rollouts with policy $\pi_{\theta_j \pm \nu\delta_k}(\tilde{x})$

Get b top directions, $\pi_{\theta_j,(k),\pm}, 1 \leq (k) \leq b$ are the policies.

$$g_j = \frac{\alpha}{b\sigma_R} \sum_{k=1}^{b} [r(\pi_{\theta_j,(k),+}) - r(\pi_{\theta_j,(k),-})]\delta_{(k)}$$

$$\theta_{j+1} = \mathrm{ADAM}(\theta_j, g_j, \alpha, \beta_0, \beta_1)$$

end

https://arxiv.org/pdf/1803.07055

Adam Optimizer Overview

Adam Optimizer is the combination of two gradient descent methodologies:

- Momentum: taking into account the moving average of the gradients
- RMSProp: adaptive learning rate - resolve the problem that gradients may vary widely in magnitudes in a batch

Adam Optimizer - Momentum

Momentum accelerate the gradient descent algorithm by taking into account the moving average of the gradients; making the algorithm converge towards the minima faster.

$$w_{j+1} = w_j - \alpha \cdot m_j$$

where,

$$m_{j+1} = \beta_1 \cdot m_j + (1 - \beta_1) \cdot g_{j+1}$$

w_{j+1}: weight at current timestep
w_j: weight at last timestep
g_{j+1}: gradient at the current timestep
α: learning rate
β_1: moving average parameter

Adam Optimizer - RMSProp

RMSprop uses the unit gradients for each weight.

$$w_{j+1} = w_j - \alpha \cdot \frac{g_{j+1}}{\sqrt{v_{j+1}}}$$

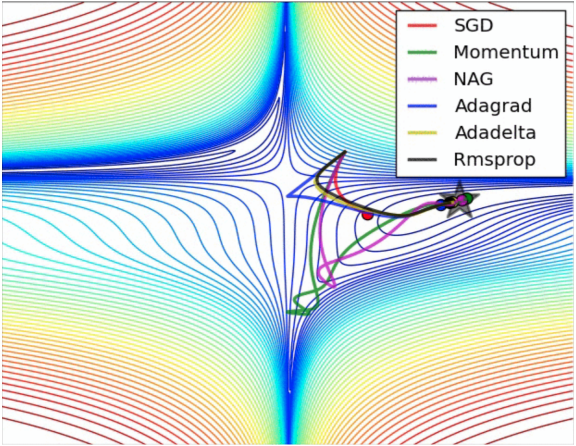
where,

$$v_{j+1} = \beta_2 \cdot v_j + (1 - \beta_2) \cdot g_{j+1}^2$$

w_{j+1}: weight at current timestep
w_j: weight at last timestep
g_{j+1}: gradient at the current timestep
α: learning rate
β_2: moving average parameter

Adam Optimizer - RMSProp visualization

RMS Prop with saddle point and minima



Adam Optimizer

Algorithm 1: Adam Optimization Algorithm 1-step forward

Hyperparameters: Gradient g_t, stepsize α, exponential decay rate β_0, β_1 for moment estimates, tolerance parameter $\lambda_{ADAM} > 0$ for numerical stability. $m_0, v_0 \leftarrow [0, 0, 0]$

Function $\mathtt{ADAM}(\theta_j, g_j, \alpha, \beta_0, \beta_1)$**:**

$m_j \leftarrow \beta_1 \cdot m_{j-1} + (1 - \beta_1) \cdot g_j$ # from momentum

$v_j \leftarrow \beta_2 \cdot v_{j-1} + (1 - \beta_1) \cdot g_j^2$ # from RMSProp

$\hat{m}_j \leftarrow m_j / (1 - \beta_1^j)$

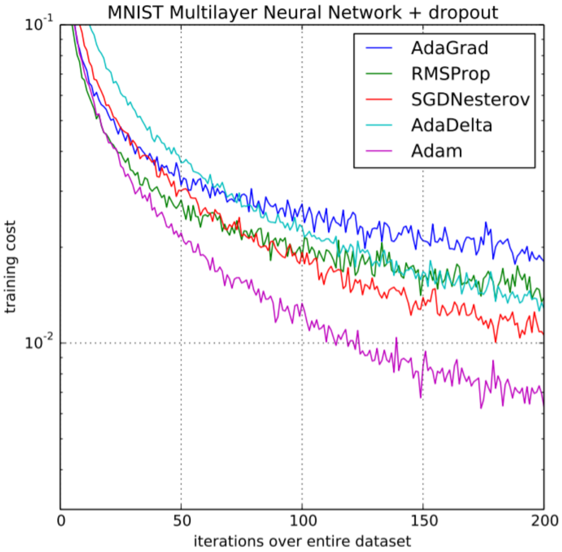
$\hat{v}_j \leftarrow v_j / (1 - \beta_2^j)$

$\theta_{j+1} \leftarrow \theta_j - \alpha \cdot \hat{m}_j / (\sqrt{\hat{v}_j} + \lambda_{ADAM})$

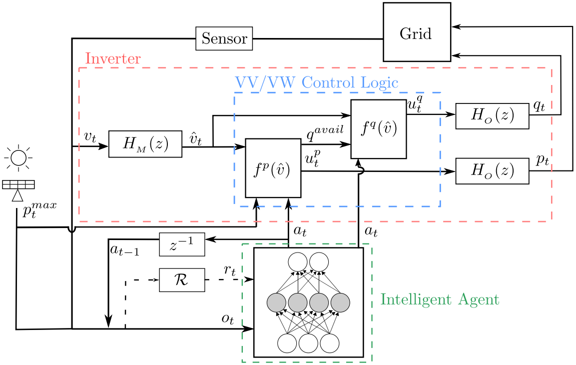
return θ_t

https://arxiv.org/pdf/1412.6980

Adam Optimizer - Training cost comparison



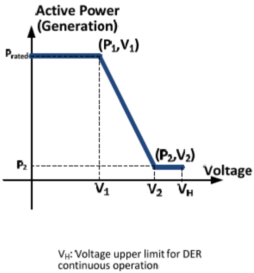
Modeling DER Action Space

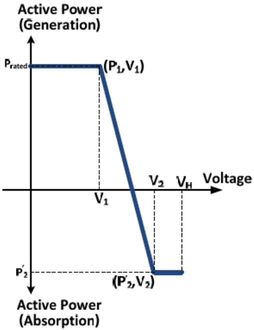


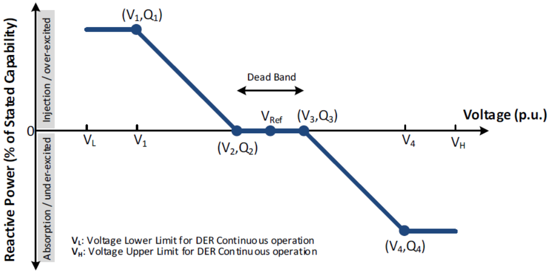
- Voltage measurements are low-pass filtered before active power and reactive power set point calculation
- These set-points are themselves low-pass filtered to ramp rate limit active and reactive power injections

Smart Inverter Voltage Regulation Controllers

Volt-Watt Curve



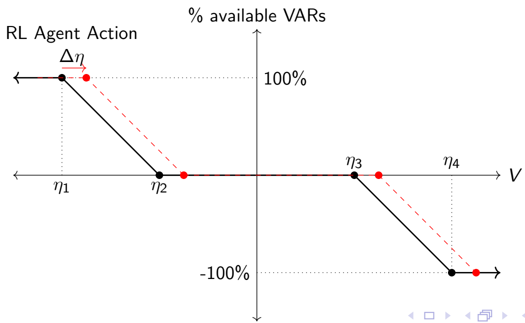




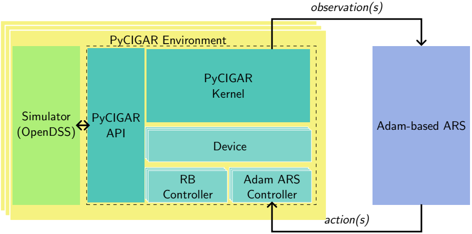
Volt-VAR curve

Modeling Action

- Action is the deviation, i.e. $a_t = \Delta\eta$, from default VV/VW parameterization
- The agent has multi-head output continuous action $a_t^i \forall i \in \{a, b, c\}$ for each phase
- Translating curve was found to be preferred action during training
  - Agent learns to indirectly control reactive power injection/consumption



Training

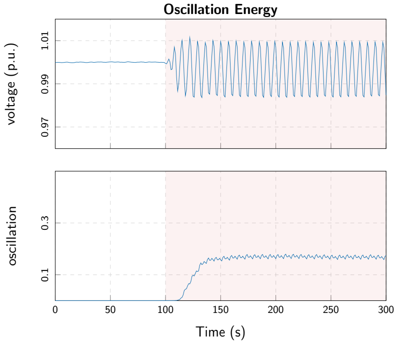


- For training we consider a single ARS agent whose observation input vector is the mean of all DER observation input vectors
- This agent then outputs an action that is applied across all inverters in the system
- Once trained, this policy is deployed and acts only on local measurements

Observation Vector - Oscillation Energy Filter

- We use a simple filter to estimate the energy of the oscillation

$$\xrightarrow{v_{i,t}} \boxed{H_{HP}(z)} \xrightarrow{\Delta v_{i,t}} \boxed{c \cdot ()^2} \longrightarrow \boxed{H_{LP}(z)} \xrightarrow{v_{i,t}}$$



Observation Vector - Unbalance measurement

$$\mathrm{vu}_{i,t} = \frac{\max(|\bar{v}_{i,t} - \bar{v}^a_{i,t}|, |\bar{v}_{i,t} - \bar{v}^b_{i,t}|, |\bar{v}_{i,t} - \bar{v}^c_{i,t}|)}{\bar{v}_{i,t}} \quad (1)$$

- $\bar{v}_i$: the mean measured voltage magnitude at bus i
- $\bar{v}^a_{i,t}$, $\bar{v}^b_{i,t}$, $\bar{v}^c_{i,t}$ are the measured voltage magnitudes on phase a, b, and c respectively.

Observation Vector

The complete observation vector is then given by

- $vo_{i,t}$: the estimation of voltage oscillation energy at node i
- $vi_{i,t}$: the estimation of voltage unbalance energy at node i
- $v_{i,t}^{a,b,c}$: measurement of the phase voltages at bus i
- $q_{i,t}^{\text{avail, nom}}$: the available reactive power capacity without active power curtailment.
- a_{t-1}^{a}, a_{t-1}^{b}, a_{t-1}^{c}: the previous action taken by the agent across each phase.

Reward Function

At a timestep t, the reward function, $R_t(a_t, o_t)$, to be maximized is:

$$R_t = -\Bigg(\sigma_u ||\mathrm{vi}_t||_\infty + \sigma_u ||\mathrm{vo}_t||_\infty + \sum_{i\in\{a,b,c\}} \sigma_a \mathbf{1}_{a_t^i \neq a_{t-1}^i} +$$

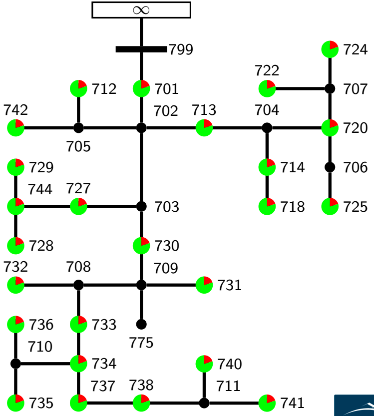
$$\sum_{i\in\{a,b,c\}} \sigma_0 ||a_t^i||_2 + \frac{1}{|\mathcal{U}|}\sum_{j=1}^{|\mathcal{U}|} \sigma_p \left(1 - \frac{p_{j,t}}{p_{j,t}^{\max}}\right)^2\Bigg)$$

This reward seeks to encourage the agent to

- Minimize system maximum voltage oscillations
- Minimize the worst case voltage imbalance
- Minimize number of VV/VW re-configurations
- Encourage the VV/VW parameterizations to remain close to their default values
- Minimize active power curtailment

Simulation Results - Scenario

- *Unbalanced* IEEE 37 node test feeder
- Cyber-attack affects same percentage of inverter capacity at each node
- Red portion of circles represent unstable inverter capacity
- Experiment 1: Compromised inverters create voltage **imbalance**
- Experiment 2: Compromised inverters create voltage **oscillation**



Training Performance

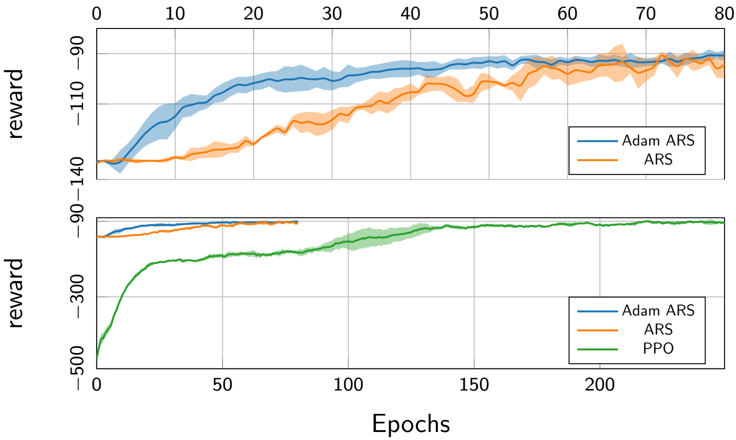
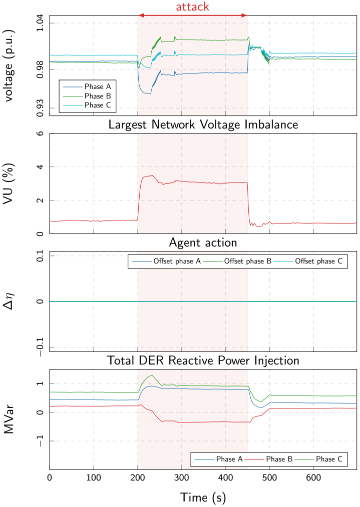
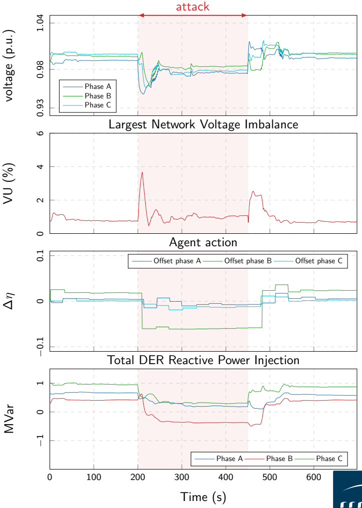


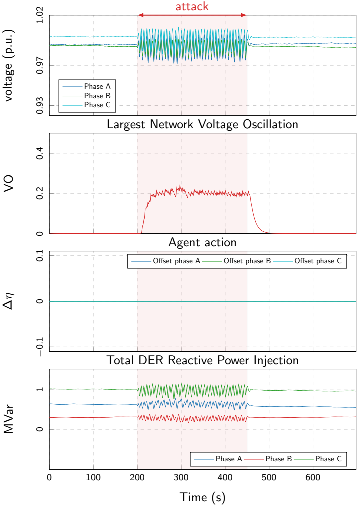
Figure: Average training reward. The shaded area represents the standard deviation over 10 runs.



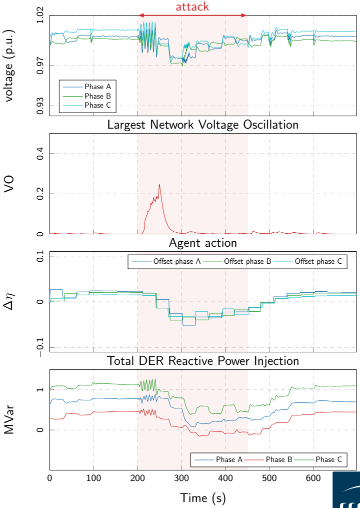
(a) No defense



(b) Defense



(c) No defense



(d) Defense

BERKELEY LAB

Final Remarks

- Oscillation policy is a linear policy
- Voltage bias compliant with IEEE 1547 standard
- Control law is completely local
- Control law requires no knowledge of the system
- No communication required
- Zero-trust control architecture
- Control law generalizes broadly to many different kinds of DER (including demand response)

BERKELEY LAB

Future works

- Optimal device settings under normal condition
- Extend to thermal loads/buildings experiments
- Electric vehicles and batteries

BERKELEY LAB

Final Remarks

This work was sponsored by the Cybersecurity for Energy Delivery Systems (CEDS) program within the Cybersecurity, Energy Security and Emergency Response (CESER) Office at the U.S. Department of Energy

Thank you

sytoanngo@lbl.gov

BERKELEY LAB