An Influence Model Approach to Failure Cascade Prediction in Large Scale Power Systems

Xinyu Wu, Dan Wu, Eytan Modiano

**Laboratory for Information and Decision Systems,
Massachusetts Institute of Technology**

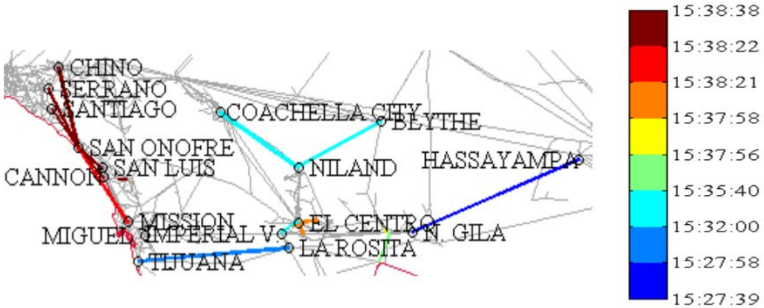
Massachusetts Institute of Technology

Outline

- Motivation
- Influence Model
- Hybrid Learning Framework
- Performance Evaluation
- Conclusion and Future Directions

Motivation

Large power blackouts



San Diego Power Blackout on Sept. 8, 2011[1]

A 500 kV line tripped off by a mistake.
Wrongly cut off 2 generators in Mexico.
Finally separated into three islands which collapsed afterwards.

Manhatten City Blackout on Jul. 13, 2019[2]

A transformer fired at West 64th Street and West End Avenue

- Starting from extrinsic effects (weather, man-made error, etc.) on few nodes or links.
- After some point, the blackout propagates broadly within minutes.
- The failure of a component may affect another remote component.

[1] G.Zussman et.al., "Power Grid Vulnerability to Geographically Correlated Failures: Analysis and Control Implications", IEEE INFOCOM 2014
[2] https://en.wikipedia.org/wiki/Manhattan_blackout_of_July_2019

Motivation

Failure Cascade in Power System

Motivation

Question: Given the initially failed components, can we predict the cascade process?

- **How many** components will fail?
- **Which** component will fail?
- If it will fail, **when** will it fail once initial failure occurs?
- ...

To solve the problem

- a model is needed to capture such cascade process
- historical cascade records may help us to refine the model.

Related Works

DC and AC calculation[3][4]

- heavy calculation burden (solving linear equations iteratively, load shedding, etc.)
- not applicable to broad cascade analysis in large system (exp: 3000~ nodes, all possible M-3 initial contingencies)
- hard to analyze the cascade properties

Influence Model[5][6]

- proposed to capture underlying correlation among network components, including remote ones [5]
- applied to power system to capture cascade process and predict failure size based on historical records [6]

[3] H. Cetinay, S. Soltan, F. A. Kuipers, G. Zussman, and P. Van Mieghem, "Comparing the effects of failures in power grids under the ac and dc power flow models," IEEE Transactions on Network Science and Engineering, vol. 5, no. 4, pp. 301–312, 2017.
[4] S. Soltan, A. Loh, and G. Zussman, "Analyzing and quantifying the effect of k-line failures in power grids," IEEE Transactions on Control of Network Systems, vol. 5, no. 3, pp. 1424–1433, 2017.
[5] C. Asavathiratham, S. Roy, B. Lesieutre, and G. Verghese, "The influence model," IEEE Control Systems Magazine, vol. 21, no. 6, pp. 52–64, 2001.
[6] P. D. Hines, I. Dobson, and P. Rezaei, "Cascading power outages propagate locally in an influence graph that is not the actual grid topology," IEEE Transactions on Power Systems, vol. 32, no. 2, pp. 958–967, 2016.

Contributions

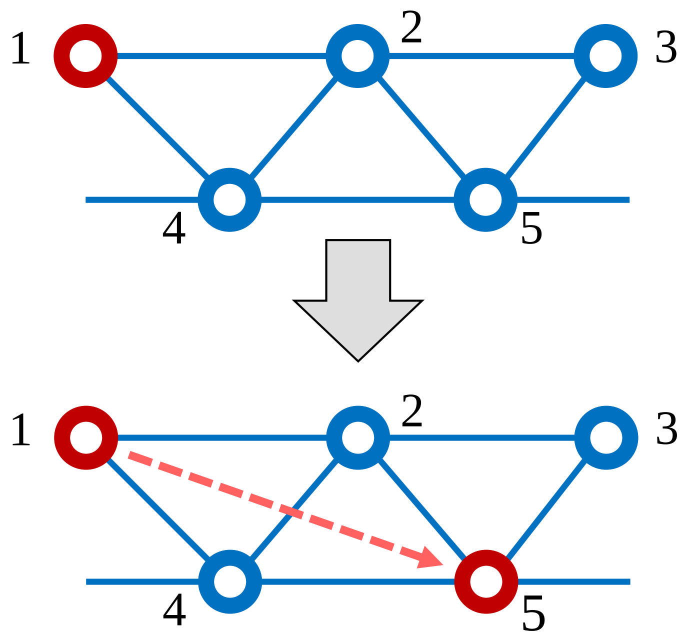
We applied influence model to model failure cascade.

- proposed a **hybrid learning framework** that can efficiently train the influence model for very large systems.

- applied the influence model to a few large scale power systems to predict their failure cascade sequences, whose performance is thoroughly evaluated at **different levels of granularity**.

- the influence model can predict the cascade sequence two orders of magnitude faster than simulation based on power flow calculation, with small compromise in accuracy

Influence Model



Failure of node-1 induces failure of node-5.

We can estimate a 2-by-2 influence matrix $\mathbf{A}_{15}$ for the pair of node-1 and node-5.

Prob of node-5 on if node-1 on

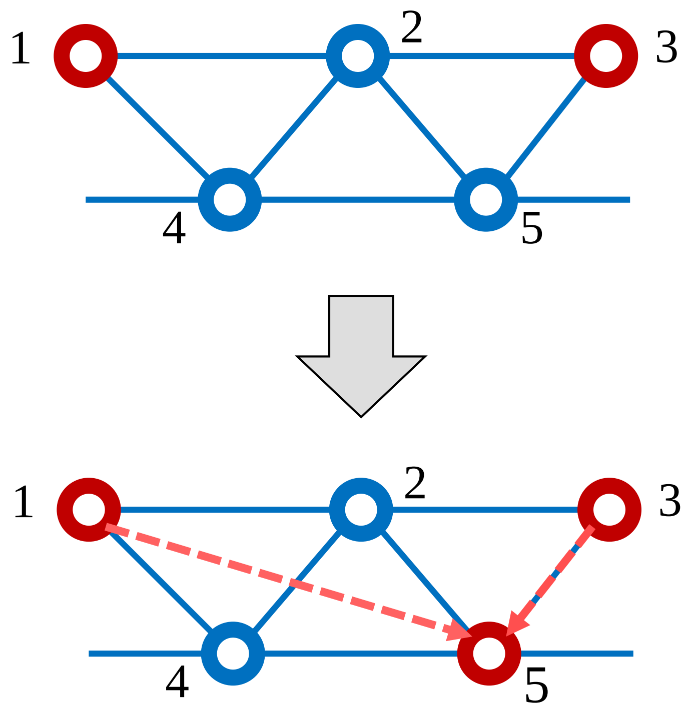
Prob of node-5 off if node-1 on

$$A_{15} = \begin{bmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{bmatrix}$$

Prob of node-5 on if node-1 off

Prob of node-5 off if node-1 off

Influence Model



Node 3 may cause the failure of Node 5 as well.

The state of Node 5 depends on both Node 1 and Node 3. ($\mathbf{A}_{15}$, $\mathbf{A}_{35}$)

The total influence on Node 5 should be a weighted combination of $\mathbf{A}_{15}$, $\mathbf{A}_{35}$ (maybe involving other nodes)

The weights: $\boldsymbol{d}_{15}, \boldsymbol{d}_{35}$

If only Node 1 and 3 will influence Node 5, then we have

$$\boldsymbol{d}_{15} + \boldsymbol{d}_{35} = 1, \quad \boldsymbol{d}_{25} = \boldsymbol{d}_{45} = \boldsymbol{d}_{55} = 0$$

$$A_{15} = \begin{bmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{bmatrix}$$

Influence Model Formulation

In our setting we consider the **link** failure cascade

Given a system with *N* buses and *M* transmission links.

The state of link i at time t: $s_i[t] \in \{0,1\}$ 1 normal; 0 failed

Each pair of link (i, j) induces an influence matrix $\mathbf{A}_{ij}$

$$\mathbf{A}_{ij} \in [0,1]^{2\times 2} \quad \text{and} \quad \begin{matrix} \mathbf{A}_{ij}(1,1) + \mathbf{A}_{ij}(1,2) = 1 \\ \mathbf{A}_{ij}(2,1) + \mathbf{A}_{ij}(2,2) = 1 \end{matrix}$$

Hence for $\mathbf{A}_{ij}$ only 2 independent components exist

- only consider $\mathbf{A}_{ij}(1,1)$ and $\mathbf{A}_{ij}(2,1)$ for each link pair (i,j)

Influence Model Formulation

$$\mathbf{A}_{ij} \in [0,1]^{2\times 2}$$

Build **pairwise influence matrices**: $\mathbf{A}^{11}, \mathbf{A}^{01} \in [0,1]^{M\times M}$

where $\mathbf{A}^{11}_{ji} = \mathbf{A}_{ji}(1,1) = P(s_i[t+1]=1 \mid s_j[t]=1)$

$$\mathbf{A}^{01}_{ji} = \mathbf{A}_{ji}(2,1) = P(s_i[t+1]=1 \mid s_j[t]=0)$$

Weight of the pairwise influence

$$d_{ij} \geq 0 \qquad \sum_{j=1}^{M} d_{ij} = 1$$

Therefore we have the following iteration

$$\tilde{s}_i[t+1] = \sum_{j=1}^{M} d_{ij}\left(\mathbf{A}^{11}_{ji} s_j[t] + \mathbf{A}^{01}_{ji}(1 - s_j[t])\right)$$

an estimated value of s_i**[t+1]**

Influence Model Formulation

$$\tilde{s}_i[t+1] = \sum_{j=1}^{M} d_{ij}(\mathbf{A}_{ji}^{11} s_j[t] + \mathbf{A}_{ji}^{01}(1 - s_j[t]))$$

$\tilde{s}_i[t+1]$ can be viewed as probability that node i takes '1' at t+1

Hence an intuitive way to predict $s_i[t+1]$

$$\hat{s}_i[t+1] = \begin{cases} 1, & \text{w.p.} \quad \tilde{s}_i[t+1] \\ 0, & \text{w.p.} \ 1 - \tilde{s}_i[t+1] \end{cases}$$

However in our prediction

- given an initial failure, the cascade process should be deterministic under fixed power injection
- we need to predict the whole process rather than only 1 generation.

Influence Model Formulation

We use the **threshold-**based prediction mechanism

$$\hat{s}_i[t+1] = \begin{cases} 1, & \tilde{s}_i[t+1] \geq \varepsilon_i \\ 0, & \tilde{s}_i[t+1] < \varepsilon_i \end{cases}$$

Hence the iteration process is

$$\tilde{s}_i[t+1] = \sum_{k=1}^{M} d_{ik} (\mathbf{A}_{ji}^{11} s_j[t] + \mathbf{A}_{ji}^{01}(1 - s_j[t]))$$

$$\hat{s}_i[t+1] = \begin{cases} 1, & \tilde{s}_i[t+1] \geq \varepsilon_i \\ 0, & \tilde{s}_i[t+1] < \varepsilon_i \end{cases}$$

$$t \leftarrow t+1$$

Given K historical sequences $\{s^k[t]\}_{t=0}^{T_k},\ k = 1,2,\ldots,K$

We need to learn $\{\mathbf{A}^{11}, \mathbf{A}^{01}, d, \varepsilon\}$

Hybrid Learning Framework

1. Learning Pairwise Influence Matrices $\{\mathbf{A}^{11}, \mathbf{A}^{01}\}$

Monte-Carlo based method

Example: 2 training cascade sequences and 2 links

$$s^1 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad s^2 = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

$$\mathbf{A}_{21}^{11} = \boldsymbol{P}(\boldsymbol{s}_1[\boldsymbol{t}+1]=1 \mid \boldsymbol{s}_2[\boldsymbol{t}]=1) = \frac{\boldsymbol{P}(\boldsymbol{s}_1[\boldsymbol{t}+1]=1,\ \boldsymbol{s}_2[\boldsymbol{t}]=1)}{\boldsymbol{P}(\boldsymbol{s}_2[\boldsymbol{t}]=1)} = \frac{\boxed{2}+\boxed{2}}{\boxed{2}+\boxed{3}} = \frac{4}{5}$$

$$s^1 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad s^2 = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

$$\mathbf{A}_{21}^{01} = \boldsymbol{P}(\boldsymbol{s}_1[\boldsymbol{t}+1]=1 \mid \boldsymbol{s}_2[\boldsymbol{t}]=0) = \frac{\boldsymbol{P}(\boldsymbol{s}_1[\boldsymbol{t}+1]=1,\ \boldsymbol{s}_2[\boldsymbol{t}]=0)}{\boldsymbol{P}(\boldsymbol{s}_2[\boldsymbol{t}]=0)} = \frac{2+0}{3+0} = \frac{2}{3}$$

Hybrid Learning Framework

2. Learning weights $\boldsymbol{d}$

Convex quadratic programming

$$\min_{d} \ \frac{1}{K}\sum_{k=1}^{K}\sum_{t=1}^{T_k}\sum_{i=1}^{M}\left(s_i^k[t+1]-\tilde{s}_i^k[t+1]\right)^2$$

$$\text{s.t.} \quad \tilde{s}_i^k[t+1]=\sum_{j=1}^{M} d_{ij}\left(\mathbf{A}_{ji}^{11} s_j^k[t]+\mathbf{A}_{ji}^{01}(1-s_j^k[t])\right)$$

$$d_{ij}\ge 0 \qquad \sum_{k=1}^{M} d_{ik}=1$$

Note that this problem can be solved in parallel for each $i\in\{1,2,\ldots,M\}$

Hybrid Learning Framework

3. Threshold ε

$$\boldsymbol{s}_i[\boldsymbol{t}+1] = \begin{cases} 1, & \tilde{\boldsymbol{s}}_i[\boldsymbol{t}+1] \geq \varepsilon_i \\ 0, & \tilde{\boldsymbol{s}}_i[\boldsymbol{t}+1] < \varepsilon_i \end{cases}$$

The most naive way is to set $\varepsilon_i = 0.5, \ \boldsymbol{i} \in \{1,2,\ldots,\boldsymbol{M}\}$

But in real case, this may cause prediction error

Observation:

- Threshold for each link varies
- Estimating thresholds adaptively based on different initial contingencies increases accuracy

Hybrid Learning Framework

$$s_i[t+1]=\begin{cases}1, & \tilde{s}_i[t+1]\ge \varepsilon_i\\ 0, & \tilde{s}_i[t+1]<\varepsilon_i\end{cases}$$

Example: one cascade sequence with 3 links

Link 1	0	0	0	0	0	0	No Way to Estimate!
$\tilde{s}_1[t]$	0	0.45	0.36	0.29	0.28	0.26	
Link 2	1	1	1	$\boxed{1}$	$\boxed{0}$	0	
$\tilde{s}_2[t]$	1	0.78	0.71	$\boxed{0.67}$	$\boxed{0.63}$	0.62	(0.67+0.63)/2=0.65
Link 3	1	1	1	1	1	$\boxed{1}$	
$\tilde{s}_3[t]$	1	0.91	0.85	0.80	0.77	$\boxed{0.76}$	0.8*0.76=0.608
$t \rightarrow$	0	1	2	3	4	5	

For the k-th sequence, we can estimate a threshold for each link i $\hat{\varepsilon}_i^k$

Idea:

- Forming the threshold pool of each link from all sample sequences
- To estimate the threshold values for each link under a new initial contingency, we use the threshold values of the training cascade sequence with initial state closest to the new one.

$$k^* = \arg\min_{k\in\{1,2,\ldots,K\}} \left\| s^{\text{new}}[0]-s^k[0]\right\|_1 \qquad \varepsilon^{\text{new}} = \varepsilon^{k^*}$$

Performance Evaluation

Dataset:

TABLE I: Default Cascade Sample Information

System	1354-Bus	2383-Bus	3012-Bus
#Generators	260	327	297
#Links	1710	2886	3566
#Eff. Links	762	2088	2083
Eff. Rate	44.6%	72.4%	58.4%
Avg. Fail Size	179	598	263
Max Fail Size	314	862	792
Min Fail Size	2	110	11

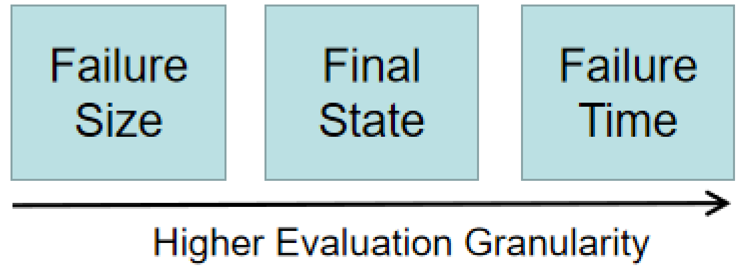
- consider M-2 initial contingencies
- training set size: at most 50,000
- training set accounts for very small portion $\frac{50{,}000}{\binom{1710}{2}} = 3.4\%$

Generate synthetic cascade sequence
- DC simulation with load shedding
- Fixing the power injection values

Performance Evaluation

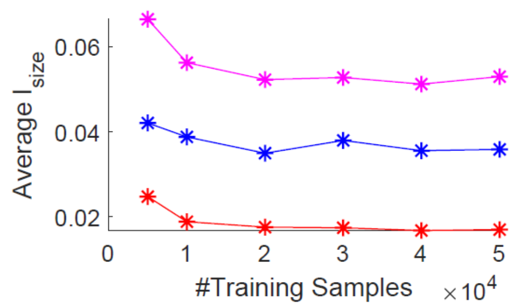
Metrics:

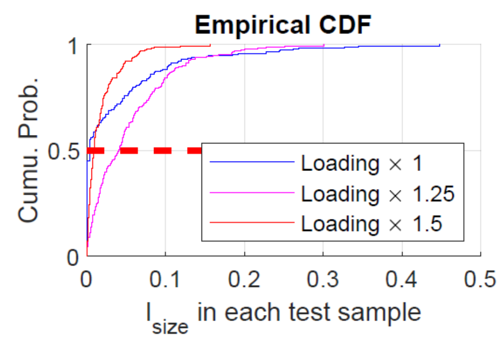
- I_{size}: Avg. Failure size error rate
- I_f: Avg. Final state error rate
- I_t: Avg. Failure time error



We consider 3 different values of loading:
- default loading & 1.5 times & 2 times

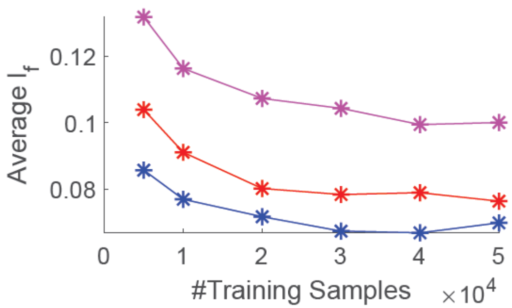
$\mathbf{I_{size}}$ (3012-node system)

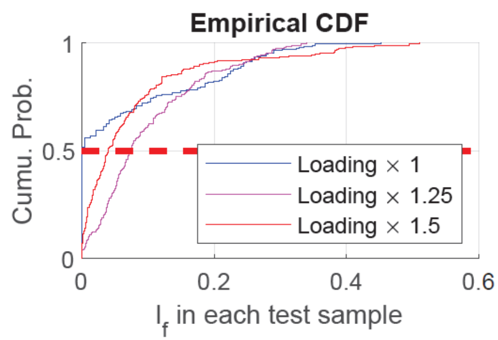




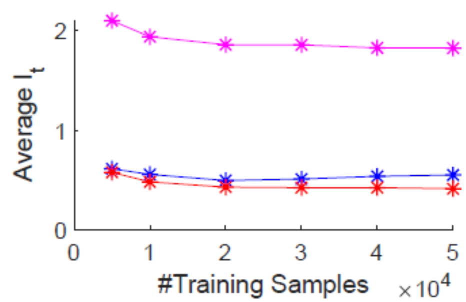
Performance Evaluation

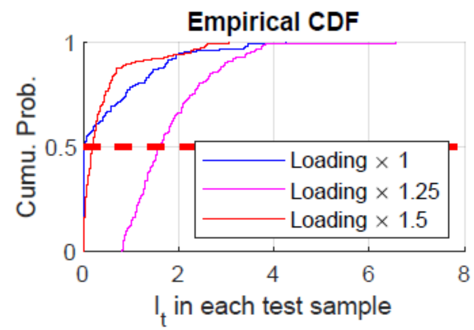
l_f (3012-node system)





l_t (3012-node system)





Performance Evaluation

Prediction time cost:

- compare time cost of our prediction and simulation by DC power flow calculation
- a | b | **c**, where 'a' is the time cost of the DC simulation, 'b' is that of our prediction, 'c' is b/a, the gain.

TABLE II: Prediction Time Cost on 1,000 Samples

	Low Load	Medium Load	High Load
1354-bus	808 | 21.3 | **38**	1930 | 19.8 | **97**	1740 | 19.6 | **89**
2383-bus	2597 | 43.3 | **60**	3490 | 37.8 | **92**	3603 | 34.7 | **104**
3012-bus	3891 | 59.4 | **66**	8020 | 58.9 | **136**	5864 | 46.9 | **125**

Conclusion and Future Directions

Conclusion:
- Proposed a hybrid learning framework over influence model to predict cascade sequences
- Evaluated in 3-layer granularity and time cost

Future Directions:
- Applying this method to AC cases
- Showing its applicability under fluctuating power injections
- Using the learned influence model to identify critical components and initial contingencies