A Framework for Cyber-Physical Model Creation and Evaluation

Abhijeet Sahu
Electrical Engineering
TAMU, College Station
abhijeet_ntpc@tamu.edu

Hao Huang
Electrical Engineering
TAMU, College Station
hao_huang@tamu.edu

Katherine Davis
Electrical Engineering
TAMU, College Station
katedavis@tamu.edu

Saman Zonouz
Electrical Engineering
Rutgers University
saman.zonouz@rutgers.edu

***Abstract*—In power systems, a cyber-physical model can play a significant role in contingency ranking to assist operators with preventive plans for cyber-related contingencies by identifying the most significant ones. Diverse cyber-physical models based on attack trees and graphs, fault trees, Markov state-space, etc. have been proposed, and are being developed by researches depending on specific objective. However, prior to the deployment of the models in real world, it is essential to evaluate the performance based on their computational bottlenecks, scalability and accuracy. This paper thus introduces a software-based model comparison framework that allows researchers to improve their models and also evaluate new models against existing ones. Additionaly, we present the algorithms of two cyber-physical modeling engines targeted for contingency and critical assets ranking; based on Attack Graph Analysis (AGA) and Markov Decision Process (MDP) and compare their performance. The models are evaluated for three different use cases: IEEE-24, CyPSA 8-substation, and IEEE-300 systems on cyber-physical model parameters such as MDP size, computation time of generation, the number of attack paths, etc. This framework will not only allow us to design and validate models but also provide a platform for researches worldwide to test new models. Further an application is developed for visualization with one-line diagram and ranking of contingencies and critical assets.**

***Index Terms*—Contingency and Critical Asset ranking, Markov Decision Process, Attack Graph Analysis, Depth First Search, Breadth First Search**

I. Introduction

According to the report by the Director of National Intelligence, Daniel Coats, hackers from different parts of the world have the ability to momentarily interrupt mission critical infrastructure of US such as power grid and natural gas pipelines [1]. Historically, two major cyber attacks like Ukranian attack in 2015 and Stuxnet attack in 2008 impacted the economy of Ukraine and Iran drastically. Both attacks were non-trivial from attacker's side to perform and defender's side to prevent. The Stuxnet worm targeted the vulnerability of a network printer spooler services and infiltrated the control network, to modify the *Step7* Siemens' software to tamper the logic of over-speeding the centrifuge of a Uranium enrichment plant, without generating alerts. Similarly, as per SANS report [2], the Ukranian attack was performed in multi stages. In reconnaissance stage, the attacker fixed their target, further weaponizing through the Microsoft Office document by embedding *BlackEnergy3* malware and delivering through a phishing mail to an IT employee, who accidentally installed the malware which enabled the adversary to create a Botnet and escalate privileges and credentials to finally control the circuit breakers through Human Machine Interface (HMI) as well as disrupt communication network.

These scenarios motivate us to design models that consider dependencies between cyber and physical systems to prevent intrusion as well as ensure resiliency of power system under compromised state. These models do not have a standard platform to compare its efficacy, hence the objective of this paper is to present a framework for designing new cyber physical models as well as comparing with the existing ones. This will help validate the cyber physical models before deployment at the utility's Energy Management System (EMS).

The paper proceeds as follows. Section II provides a literature review on different models developed for threat modeling such as attack trees, Bayesian graphs, etc. along with the cyber physical models for critical asset and contingency ranking, developed by our team. Further in Section III, we elaborate the AGA [3] and MDP [4] based approach adopted. The power system use cases: IEEE-24, 8-substation [5] and IEEE-300 system considered for the model comparison are introduced in Section IV. In Section V and VI, we perform analysis on the MDP and AGA model using the use cases.

II. Background

A Cyber Physical System (CPS) model considers models of physical processes along with the software and network models. The model complexity depends on the objective as well as the time and space complexity involved in model creation and its analysis. The complexity of model building further increases when security comes into the picture. Researches have developed many ground-breaking models confined to threat and physical modeling separately. For example, attack trees models are used for analysing threats on systems and possible attack paths to reach those threats. Different types of attack trees are developed by researcher that focuses on a specific objective. Vulnerability trees are proposed in [6], which represents, how different vulnerabilities are hierarchically interdependent in a system. In the paper [7], the author studies how threat trees behave and how the tree computations are done when several interdependent attack parameters are considered. An enhanced attack tree was proposed by [8] that

978-1-7281-3192-4/19/$31.00 ©2019 IEEE

Accepted for publication in IEEE 20th International Conference on Intelligent Systems Applications to Power Systems (ISAP), December 2019

models complex attacks with time dependencies. It has devised a new gate, "Ordered-AND" which considers the attackers subsequential behavior and limitations on the attack paths. The author in [9] proposed attack graph with an example to illustrate how they specify and analyze network attack models. They utilized these models to automatically generate attack graphs by exploring the exploitation of system wide vulnerabilities. Khand [10] utilized different dynamic fault tree gates to attack trees making them more dynamic. Attack trees are utilized in diverse areas of power systems. For example, in the distribution side, the threat analysis of an Advanced Metering Infrastructure (AMI) network is designed in [11] to create an enhanced attack tree that captures the attackers primary objective for example, energy theft and the individual attack steps such as jamming or eavesdropping that are carried out prior to the final stages of attack. In the paper [12]an attack tree formulation based on power system SCADA network is used to evaluate the system, use cases, and vulnerabilities at leaf nodes. The security states are considered for computing the measure of vulnerabilities in the power system. All these techniques have focused on building models considering only pure cyber or physical side.

Modeling trees considering the security of CPS, makes it more challenging to construct attack trees. The paper [13] discusses the challenges in modeling CPS with regards to intrinsic heterogeneity, concurrency and sensitivity to timing. A cyber-physical model for hierarchical control system is proposed in [14] to evaluate the degree to which an inappropriate control command from cyber intrusion can influence power systems. Our previous work on creating cyber physical models for power system contingency and critical asset ranking followed two different approach for addressing the problem. The first work on Security-Oriented Cyber-Physical Contingency Analysis (SOCCA) [4] identifies and ranks the contingencies possible through cyber-side vulnerability exploitation. Basically a Markov Decision Process (MDP) based approach is used to model cyber-physical attack as a finite set of security state [4], that explores all the security states the system can be in. Our second work Cyber Physical Security Assessment (CyPSA) [3] utilizes Attack Graph Analysis (AGA) technique to create possible attack paths from a cyber to physical node in the graph and rank those paths considering the cyber costs as well as impact of the attack on physical sides. Further, an extension to the SOCCA model, a CPMA framework [15] is proposed that builds a partially observable Markov Decision Processes (POMDP) model instead of simple MDP to compute all the possible attack paths. Both these work addresses the issue of finding the critical assets on both cyber and physical side to prioritize the protection schemes.

Many researchers in the area of cyber-physical modeling have cited some of our models. In [16], an information-energy flow model based on matrix-based computation is developed considering the mutual interdependencies of cyber and physical components. Due to challenges associated with selecting security metrics for electric grid, authors in [17] explores diverse security metrics for leveraging attack graphs. A hierarchical control system cyber network is modeled as a directed graph with data nodes and directed branches described using node-branch incidence matrix [18]. ARCADES [19] proposes a technique to explore defense strategies based on contingency ranking in power systems. A multi-resolution model of complex distribution network with five operating states created using time-state machine method [20] is developed and validated using cyber-physical co-simulation. Authors in [21] discusses the effects of cyber coupling on the cascading failure in power system. A stochastic method is used to generate the cascade failures caused by cyber malwares. Most of these works propose a novel model, but there is yet no work on providing platform to compare different models and use cases to deploy them in real world. This framework would provide the researchers, who leveraged our prior work, a platform to design new models and also validate them. In this paper, we summarize both MDP and AGA based approach and use the framework to analyze the performance with regards to scalability and accuracy for varying use-cases. Few algorithms of our SOCCA [4] and CyPSA [3] engine were not presented in detail in the original work, hence here we present them to assist other researchers to explore our models.

III. Analysis of Cyber Physical Models

A. Model Creation and Evaluation Framework

Our previous works *SOCCA* and *CyPSA* were built with each component run separately using different programming language and runtime environment. In this work we developed a framework to run all the modules of the engines using a common platform reducing the bottleneck of inter-application communication and also organising the codes through C# .NET libraries. Hence, we refer the engines as *SOCCA#* and *CyPSA#*. The framework for our cyber physical model generation and evaluation is shown in Fig. 1. Each substation control center has a dedicated firewall to monitor and control incoming and outgoing traffic. We develop a firewall rule generator to automatically generate firewall configuration for the control centers, based on the power system use case. The Network Mapper (NMap) report spawned from the hosts in the control network provides the details on the services running on the hosts. The NPView [22] application then parses the NMap report and firewall rules to generate the host connectivity matrix or the cyber topology, used by the CyPSA# and SOCCA# engine along with the power topology from Power World, for critical asset and contingency ranking. In this section, we will discuss in details on the algorithm used for MDP state creation in SOCCA# and the procedure followed in the CyPSA# along with the AGA algorithm.

B. MDP based SOCCA# engine analysis

SOCCA engine provides an elaborate and insightful state-space exploration of the network by modeling the system states based on the number of compromised hosts of the system and assigning each state a security index reward based on their connectivity with respect to the power components of the cyber-physical network. The state space of the MDP is based

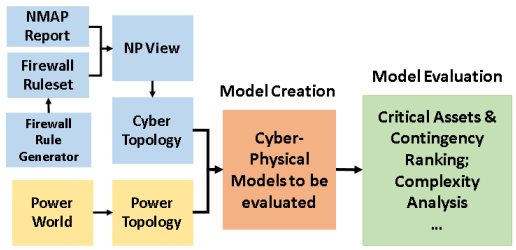

Fig. 1. Framework for the cyber physical model creation and evaluation

on the different combination of hosts that can be compromised based on inter-host reachability. The action spaces are the adversarial vulnerability exploitations. The reward function is based on the susceptibility measure to attacks depended on that state. The transition probability representing the attacker success rate, are computed based on the Common Vulnerability Scoring System (CVSS) scores obtained from the National Vulnerability Database (NVD). For example, a lower CVSS value would have a higher transition probability and zero day attack with no vulnerability will have low transition probability. Each security state is assigned Performance Index (PI) and Security Index (SI) as,

$$PI = \sum_{l \in L} \max\left(\frac{f_s(l)}{f^{MAX}(l)} - 1, 0\right)^2 \quad (1)$$

$$SI(s) = \max_{a \in A(s)} \left(\gamma \cdot \sum_{s' \in S} P(s'|s, a) \left[\Delta PI(s, s') + SI(s') \right] \right) \quad (2)$$

where $f_s(l)$ is the flow in the line l and $f^{MAX}(l)$ is the upper limit on the power flow in the line l in Eq. 1, $P(s'|s, a)$ is the transition probability, γ is the discount factor in Eq. 2.

1) Curse of Scalability: The engine generates the MDP for a network of hosts considering the interconnections among the cyber and physical components in the system. It uses the connectivity matrix to extract the list of the hosts and their neighbors and builds the MDP. With increasing hosts, the MDP creation Algorithm 1 results in state explosion, as it has a time complexity of $O(2^N)$, where N is the number of hosts in the network.

To address the state space explosion problem, the tool considers each subnetwork of hosts as a single node for MDP creation, reducing the state space significantly. For a given set of host-to-host connections, each new network associated with a particular host is considered single *host* in the network. Any connection from one host in one network to another host in a different network is considered a connection between two subnetwork *hosts*. For each subnetwork, the tool iterates through all of the hosts contained in the network, compromising any power components associated with a particular host. The resulting PI from opening all of the subnetworks associated transmission lines through relays, is used to calculate its SI. Although clustering the hosts based on subnetwork made the SOCCA model more scalable, but there is a trade off in the accuracy of the attack paths, because every host has a unique

Algorithm 1 MDP Creation using Dynamic Prog.(SOCCA#)

```
1:  function Create_MDP (mdps, curr_mdp , host)  .
    mdps : list of all mdp states, curr_mdp : index of current
    MDP processed, host: host in the current iteration
2:      for h in host.neigh do
3:          found = if h in mdps[curr_mdp]'s privs
4:          if !found then
5:              Add h to mdps[curr_mdp].privs
6:              mdp_edge_set = false
7:              for mdp in mdps do
8:                  equal = if mdp == mdps[curr_mdp]
9:                  break
10:                 if equal then
11:                     edge_found = check if edge exist
12:                     if edge_found then
13:                         mdps[curr_mdp].neigh add mdp
14:                     end if
15:                     Create_MDP (mdps, index(h), h)
16:                     Create_MDP (mdps, index(h), host)
17:                     mdp_edge_set = true
18:                     break
19:                 end if
20:             end for
21:             if !mdp_edge_set then
22:                 Create a MDP state new_m
23:                 Add new_m to mdps
24:                 new_m.privs = mdps[curr_mdp].privs
25:                 Add an edge for n_mdp added
26:                 Create_MDP (mdps, count(mdps), h)
27:                 Create_MDP (mdps, count(mdps), host)
28:             end if
29:         end if
30:     end for
31: end function
```

set of vulnerabilities, which is well represented in the AGA model.

2) Description of the MDP Generation Algorithm 1: The function *Create_MDP* is called initially with MDP's initial state (∅), representing no host the attacker has privilege over. The connectivity matrix representing a directed graph is traversed recursively starting from the initial entry point of the host, to keep track of the current state of the MDP. Every MDP state comprise of a list *privs* that represent the list of host the attacker has privilege over. It first checks, if the host *h* exist in the current MDP's *privs* list (line 4). If *!found* it adds the host *h* to the list and further searches all the MDP states. When the search meets a graph edge [i,j] that crosses over privilege domains h_i to h_j, an edge or state transition is created (line 13). A new MDP state *new_m* is created with *privs* allocated and added to *mdps* list (line 22-25). Once the MDP is created, ranking of cyber-physical contingencies is performed as per the Algorithm 1 presented in [4].

C. AGA based CyPSA# engine analysis

The CyPSA# based on Attack Graph Analysis of a cyber-physical networks connectivity is implemented as a means of

Algorithm 2 Pseudo Code of CyPSA#

1: Parse Connectivity Matrix obtained from *NP View*
2: Parse Topology File having *nmap* information.
3: Select list of IP of targeted relays (*sel_t*).
4: Build host connectivity graph *G* with each nodes representing the combination of type of services that allows a vulnerability to be exploited on a given host.
5: *attackGraph* = *Generate_Attack_Graph*(*G, L, sel_t*)
6: Compute *PI* by opening a breaker from the relay ∈ *sel_t*.
7: Compute *CC* based on the attack paths in *attackGraph*
$CC(P) = \sum_{p \in P} CC(p)$, *p* are edges in path *P*
8: Rank critical assets based on *SI* computed using Eq 3.

providing a scalable solution for analyzing which attack path contingencies on the system are most critical with respect to safety and security of the network. This engine utilizes the host connectivity paths and vulnerability information to generate an attack graph. The paper [3] demonstrate the application developed for critical assets and contingency ranking. Here we provide the details of the model CyPSA engine through the pseudo code (Algorithm 2) and the AGA (Algorithm 3). The attack graph combines individual host paths chaining system vulnerabilities to produce a list of possible attack paths and rank each of them based on the security index 2. The Security Index(SI) is computed as

$$SI = \frac{Performance Index(PI)}{CyberCost(CC)} \quad (3)$$

where PI is computed using Eq. 1 and CC using line 7 of Algorithm 2. Since PI represent the degree of impact on the physical side, SI is directly proportional to it. But lower CC represent the ease with which the attacker can exploit a vulnerability, so SI is inversely proportional to the CC. The attack path based analysis allows to explore vulnerabilities in different segments of the path in details. For instance, an attacker compromising a web server in the Demilitiarized Zone (DMZ) could get access to a PI server (first segment in the attack path) and then exploit a vulnerability in that PI server to gain local control and connect to a SCADA controller (second segment in the attack path).

D. Firewall rules to connectivity matrix generation

The firewall rules from the substation's control network and main control centre network were collected to form the host connectivity matrix for both CyPSA# and SOCCA# engine. It makes use of Network Perception's NP-View application to build a logical network model by parsing firewall rules of the control network of each substation. Each firewall in a substation is interfaced to two network one that is the Inside control network and the Outside network. Cisco's Object groups were used to classify users, devices and protocols into groups and apply Access Control Lists (ACLs) to create policies for groups rather than individual hosts. The NP View application parses all these grouped ACLs to build the connectivity matrix among the host with their allowed protocols and services. A sample example can be found in the Appendix IX-A.

Algorithm 3 AGA module of CyPSA#

1: **function** *Generate_Attack_Graph* (G, L, sel_t) . *G* : host connectivity graph , *L* : attackers list, *sel_t*: targeted critical assets
2: create empty *attackGraph*
3: **for** attacker *a* in *L* **do**
4: *d,p* = djikstra_shortest _path(*a,G*) . Get the list of distance *d* and predecessor path *p*, based on shortest path from *a* to reachable nodes from *a* in the graph *G*
5: **for** target *t* in *d* **do**
6: **if** *t* in *L* **then**
7: *path* = G(*t*) . get the path from *G*
8: *v_list* = Get vuln list for the path
9: cost = getTotalCost(*path*, *v_list*)
10: Add *path* to *attackGraph*
11: **end if**
12: **end for**
13: **end for**
14: **return** *attackGraph*
15: **end function**

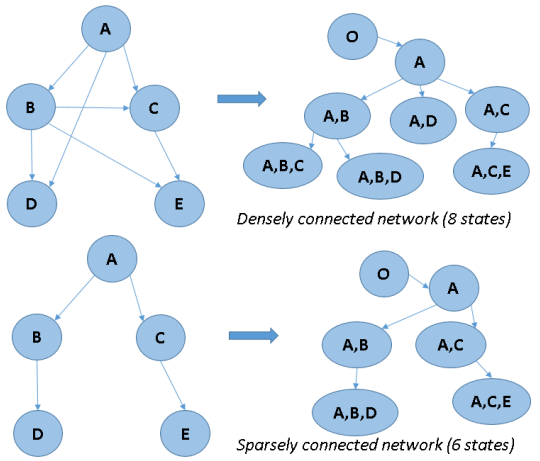

Fig. 2. Impact of sparsity of connectivity matrix on model

IV. Power System Use Cases

The sparsity of the connectivity matrix impacts the computation time of model generation as well as its size. The number of MDP states depends on the inter-connectivity among the hosts, hence a sparse matrix would result in less MDP states. Number of nodes generated using the AGA approach is dependent on the dimension of the connectivity matrix. Hence, for our analysis, we have considered three power system cases with varying sparsity of host connectivity matrix in the control network. Fig. 2 shows how a densely connected network results in higher MDP states.

A. 8-substation Dense Connectivity Model

The 8-substation network uses the node-breaker topology to represent the detailed physical information. Each substation having multiple buses and control devices increases hosts per control room, making the model dense. There are 52 buses and internal substation nodes, relays, breakers, firewalls, and

routers are modeled along with the IPs of the relays and firewall rules of the control network [5]. Relays and breakers in the substations are crucial for cyber-physical modeling. The cyber topology for this case as obtained from NPView.

B. IEEE 24 Sparse Network Model

The IEEE 24-bus reliability test system was developed by the IEEE reliability subcommittee and was published is 1979 [23]. It is a bus-branch topology where 24 bus represent 24 substations. The network topology is sparse in comparison to the 8-substation case since the topology doesn't include the detailed configuration within each substation. On the cyber side, each substation is configured to have its firewall rules to allow specific traffic to go to the main control center.

C. IEEE 300 Large Network model

The IEEE 300 bus test case was developed by Mike Adibi in 1993 [24]. With 300 buses in the network, it represents a large area of power grids with more detailed information. Meanwhile, after expanding the topology into node-breaker topology, the detailed information within the substation can be considered, making the case both large and dense. On the cyber side, utility WAN are connected to these 300 substations using three different communication technologies, they are fiber, cellular, and microwave. A combination of serial and ethernet based firewalls are deployed for the substations.

V. Result and Analysis

A. SOCCA# Engine Analysis

1) Analysis on attack starting point: The number of MDP states generated using Algorithm 1 depended on the attacker's host starting point. Three hosts from three different networks are selected for the analysis for both IEEE 24 and 8-substation case. One host from each network were randomly picked for analyzing the MDP size and computation time. Fig. 3 shows how the attacker's starting point alters the MDP generation computation time and its size. In the IEEE 24 system, when the attacker selects H1, the number of MDP states are 147, with 469 edges and takes 0.068 sec to find the attack paths, which differs from host H2 and H3. The complexity of the attack depends on the size of the MDP, hence this analysis would allow the operators to prioritize the cyber assets based on the complexity. The starting host that generates the least sized MDP needs the highest attention. The dense 8-substation model resulted in more MDP states and edges and a high computational time as shown in Fig. 3 making the model infeasible for the densely connected network.

2) Sensitivity Analysis of Transition Probability on Contingency Ranking: In the SOCCA model, contingencies are defined by the edges between MDP states. The transition probability of these states impacts contingency ranking. These probabilities are the measure of the attacker's capability of compromising one host from another and are based on two types: static and dynamic uncertainty [25]. In static uncertainty the difficulty measure is directly computed from the CVSS score of vulnerabilities exploited. Dynamic uncertainty depends on many factors either on attacker hacking skills

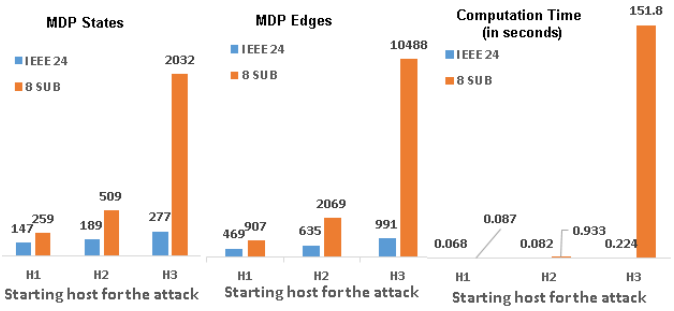

Fig. 3. Comparison of MDP size and computation time for varying starting host for attacker for IEEE 24 and 8-substation case.

TABLE I
Model sensitivity to transition probability , i.e. number of contingency within first N ranks for changing transition probability

N	= 0.15	= 0.25	= 0.35
1	1	2	2
2	1	2	2
3	1	1	1
4	1	2	2

or on the security alerts from Intrusion Detection Systems. Hence we focused on the sensitivity of the model by varying the transition probability $P(s^0|s, a)$ by . In these analyses, the values for each states $P(s^0|s, a)$ is modified by δ, where $\delta \in$ - , . As $P(s^0|s, a)$ impacts the MDPs SI (Eq. 2), we evaluated its effect on the ranking of the contingencies. For a fixed iteration with varying , we calculated the number of contingencies that are ranked less than a rank say $N = 4$. From Table I we can observe, the number of contingencies within rank $N = 4$ is 1 when = 0.15 and 2 when = 0.25. With further rise in there was no impact observed.

B. CyPSA# Engine Analysis

1) Analysis on number of attack paths: We analyse the number of attack paths computed from the algorithm based on varying the number of relays an attacker targets. For example, in 8-substation case, for 21 relays targeted, 87 one hop and 32 two hop paths are obtained(Fig. 4). The 8-substation case being dense, attack paths with multiple hops are observed unlike the IEEE 24 case with attack paths with only one hop (Fig. 4). Even though the IEEE 300 case is a large and dense system, it shows less attack paths in comparison to 8-substation case, denoting less vulnerable hosts in the system.

2) Effect of NMap reports on attack paths: The impact of NMap reports on attack paths computation are analysed for the number of attack paths computed for the 8-substation and IEEE 24 case. We altered the NMap reports of the substations to study how number of open ports and services can escalate access paths to a target device. For the 8-substation case, we observed that as the number of substations that had vulnerabilities from open ports and services increased from 2 to 4, the attack paths increased from 314 to 594 (Fig. 5). Hence, the operator must secure as many hosts as possible to

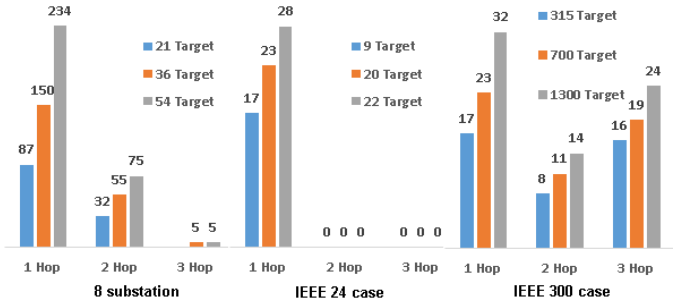

Fig. 4. Attack path count with number of hop distribution with varying target counts for 8-substation, IEEE 24 and 300 case.

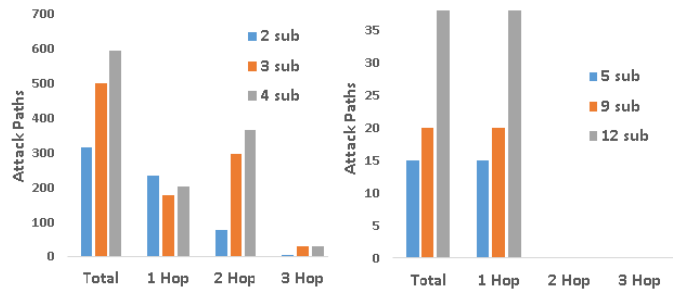

Fig. 5. Effect of number of substations with vulnerabilities on the attack paths for 8-substation (left) and IEEE 24 (right) case .

reduce the number of attack paths. Similar trend followed for IEEE 24 case, but based on the sparseness of its connectivity matrix, there are no attack paths with more than one hop.

3) Analysis of graph traversal algorithms: Graph traversal algorithms for computing shortest path such as Bellman Ford [26], [27], and Djikstra [28], Directed Acyclic Graph(DAG), etc. are compared to determine the best one for AGA approach based on the computation time. The computation required for attack graph analysis depends on the number of starting nodes accessible by the attackers. From Fig. 6 we can observe that as the number of hosts considered for attackers starting point increases, the computation time for graph analysis increases faster in Bellman Ford in comparison to Dijkstra based shortest path computation, for both 8-substation and IEEE 24 case, hence we considered Dijkstra for our CyPSA# engine (line 4 of the Algorithm 3). The attacker may prefer a complex path rather than a shortest path hence it is essential to explore all the possible paths rather than only shortest paths. So we explored Depth First Search(DFS) and Breadth First Search(BFS) based approach for finding attack paths. The time complexity involved in both the techniques depends on the connectivity matrix. Fig. 7 shows that DFS approach resulted in higher computation time in comparison to BFS for the denser 8-substation network. There was no significant difference observed in computation time for sparse IEEE 24 network using both approach.

VI. Comparison of the MDP and AGA approach

MDP based approach for contingency ranking results in higher computation time for the densely connected network

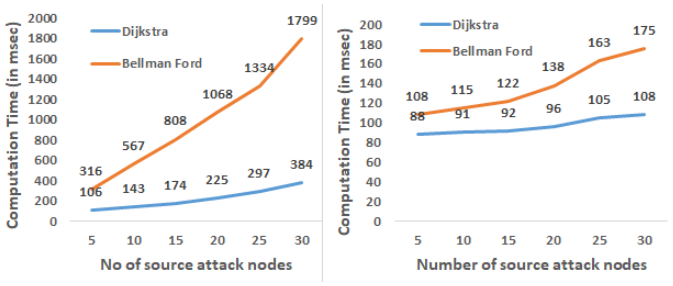

Fig. 6. Comparison of Dijkstra and Bellman Ford on computation time for 8-substation (left) and IEEE 24 case (right)

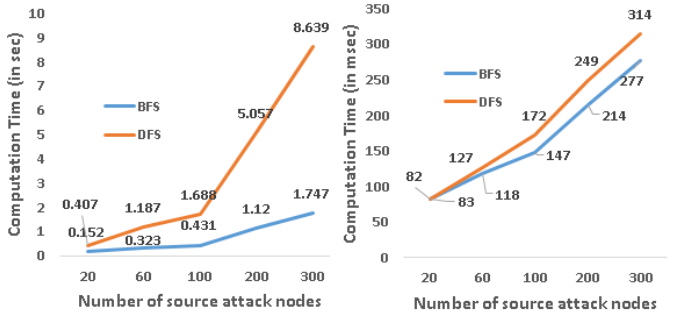

Fig. 7. Computation times using DFS and BFS approach for 8-substation(left) and IEEE 24(right) case

(Fig. 3). AGA based CyPSA# performed the attack graph analysis for IEEE 300 case in 9 mins, but the MDP based approach took more than 10 hours for generating the MDP states and edges. Critical assets ranking is based on the Eq. 3. Hence, we can observe from Fig. 9, due to relatively higher CC (which depends on the type of vulnerability as well as the access path), 10.12.1.101 more critical than 10.14.1.101 though physical impact is more in case of the later. Ranking in MDP approach is dependent on the SI, which depends on transition probability as well as difference of PI between the transitioned states. Contingency ranking in MDP approach refers to the transitions of the MDP states, while in AGA approach, it is referred to the critical path ranking. In both approach, the computation time for model depended on attacker's starting point of intrusion.

VII. Application design for testing SOCCA# and CyPSA# performance

A desktop application was developed to test the contingency rankings as well as visualize the one line diagram that would allow a platform for the cyber players (attackers and defenders) to securely operate the power system under cyber attacks. Fig. 10 shows how an operator can fetch the power system case, select the relays to compromise and compute the critical asset ranking, attack paths, security index, performance index with the list of vulnerabilities that can be exploited. Fig. 11 shows the visualization of the one line diagram for operating the critical components of the 8-substation case.

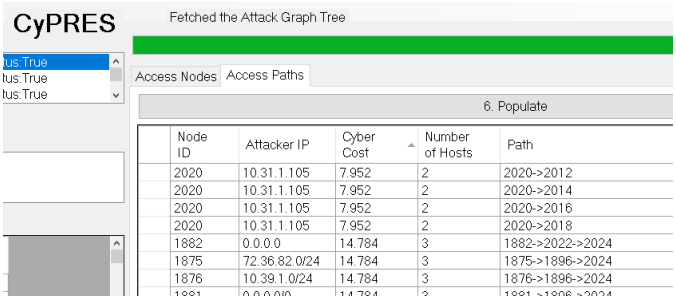

Fig. 8. An application for enumerating attack paths

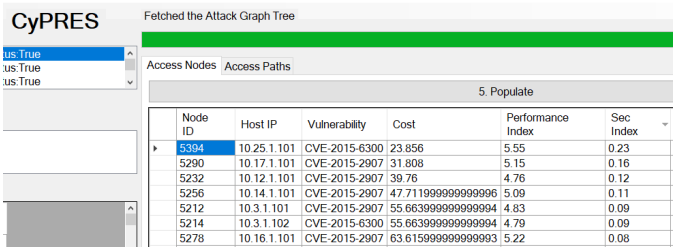

Fig. 9. A desktop application developed for visualizing critical assets and access path ranking

VIII. Conclusion

The framework is successfully utilized to test our cyber physical models for different power system cases, with varying level of cyber network density. The framework was also tested by developing a desktop application for visualization. It will help evaluate the pros and cons of different models to improve existing models and develop new ones for more diverse problems such as cyber-physical state estimation, optimal response, etc. The analysis performed on our two engines will provide situational awareness to the control room operators. Additionally, assist network administrators to fix firewall configuration as well as patch host vulnerabilities. Current work rely on the firewall rules, Nmap reports for cyber modeling. In future we will also explore alerts from other cyber sources such as Snort, Splunk, BRO, Suricata, etc. Further, we will use even larger cases such as Texas 2000 synthetic case [29] in our framework.

Acknowledgment

This research is supported by the US Department of Energy's (DoE) Cybersecurity for Energy Delivery Systems program under award DE-OE0000895.

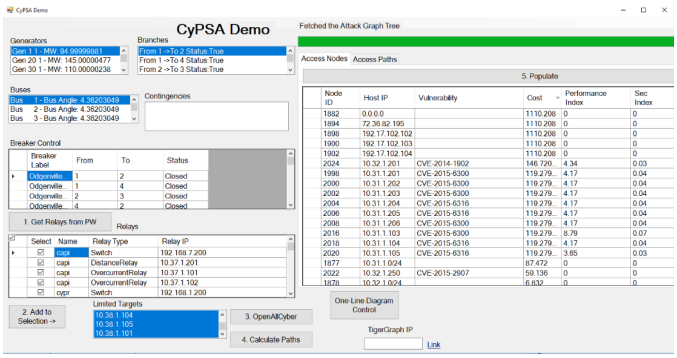

Fig. 10. An application for ranking critical assets

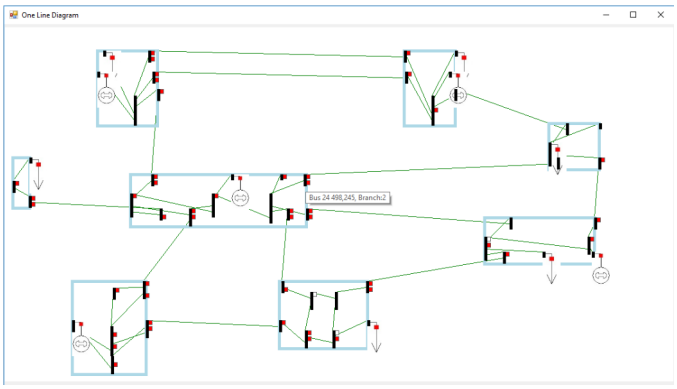

Fig. 11. One Line Diagram for the 8-substation case

References

[1] K. Fazzini, "China and Russia could disrupt US energy infrastructure, intelligence report warns on heels of Huawei indictments ," https://www.cnbc.com/2019/01/29/china-russia-could-disrupt-us-infrastructure-with-cyber-attacks-odni.html.

[2] R. M. Lee, M. J. Assante, and T. Conway, "Analysis of the Cyber Attack on the Ukrainian Power Grid: Defense Use Case by SANS ICS ," https://ics.sans.org/media/E-ISAC_SANS_Ukraine_DUC_5.pdf.

[3] K. R. Davis, R. Berthier, S. Zonouz, G. Weaver, R. B. Bobba, E. Rogers, and D. M. N. P. W. Sauer, "Cyber-physical security assessment for electric power systems," in *IEEE-HKN: The Bridge*, 2017.

[4] S. Zonouz, C. M. Davis, K. R. Davis, R. Berthier, R. B. Bobba, and W. H. Sanders, "Socca: A security-oriented cyber-physical contingency analysis in power infrastructures," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 3–13, Jan 2014.

[5] G. A. Weaver, K. Davis, C. M. Davis, E. J. Rogers, R. B. Bobba, S. Zonouz, R. Berthier, P. W. Sauer, and D. M. Nicol, "Cyber-physical models for power grid security analysis: 8-substation case," in *2016 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Nov 2016, pp. 140–146.

[6] A. Jones, "Using vulnerability trees for decision making in threat assessment," 08 2003.

[7] A. Buldas, P. Laud, J. Priisalu, M. Saarepera, and J. Willemson, "Rational choice of security measures via multi-parameter attack trees," in *Proceedings of the First International Conference on Critical Information Infrastructures Security*, ser. CRITIS'06, Berlin, Heidelberg, 2006.

[8] S. Camtepe and B. Yener, "Modeling and detection of complex network," www.cs.rpi.edu/ yener/PAPERS/ SECURITY/securecom07.pdf.

[9] O. Sheyner and J. Wing, "Tools for generating and analyzing attack graphs," in *IN PROCEEDINGS OF FORMAL METHODS FOR COMPONENTS AND OBJECTS, LECTURE NOTES IN COMPUTER SCIENCE*, 2004, pp. 344–371.

[10] P. A. Khand, "System level security modeling using attack trees," in *2009 2nd International Conference on Computer, Control and Communication*, Feb 2009, pp. 1–6.

[11] D. Grochocki, J. H. Huh, R. Berthier, R. Bobba, W. H. Sanders, A. A. Crdenas, and J. G. Jetcheva, "Ami threats, intrusion detection requirements and deployment recommendations," in *2012 IEEE Third International Conference on Smart Grid Communications (SmartGridComm)*, Nov 2012, pp. 395–400.

[12] C. Ten, C. Liu, and M. Govindarasu, "Vulnerability assessment of cybersecurity for scada systems using attack trees," in *2007 IEEE Power Engineering Society General Meeting*, June 2007, pp. 1–8.
[13] P. Derler, E. A. Lee, and A. Sangiovanni Vincentelli, "Modeling cyberphysical systems," *Proceedings of the IEEE*, 2012.
[14] S. Xin, Q. Guo, H. Sun, B. Zhang, J. Wang, and C. Chen, "Cyber-physical modeling and cyber-contingency assessment of hierarchical control systems," *IEEE Transactions on Smart Grid*, Sep. 2015.
[15] K. R. Davis, C. M. Davis, S. A. Zonouz, R. B. Bobba, R. Berthier, L. Garcia, and P. W. Sauer, "A cyber-physical modeling and assessment framework for power grid infrastructures," *IEEE Transactions on Smart Grid*, pp. 2464–2475, Sep. 2015.
[16] S. Xin, Q. Guo, H. Sun, C. Chen, J. Wang, and B. Zhang, "Information-energy flow computation and cyber-physical sensitivity analysis for power systems," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, June 2017.
[17] P. S. Patapanchala, Chen Huo, R. B. Bobba, and E. Cotilla-Sanchez, "Exploring security metrics for electric grid infrastructures leveraging attack graphs," in *2016 IEEE Conference on Technologies for Sustainability (SusTech)*, Oct 2016, pp. 89–95.
[18] S. Xin, Q. Guo, H. Sun, B. Zhang, J. Wang, and C. Chen, "Cyber-physical modeling and cyber-contingency assessment of hierarchical control systems," *IEEE Transactions on Smart Grid*, vol. 6, no. 5, pp. 2375–2385, Sep. 2015.
[19] M. Touhiduzzaman, A. Hahn, and A. Srivastava, "Arcades: analysis of risk from cyberattack against defensive strategies for the power grid," *IET Cyber-Physical Systems: Theory Applications*, vol. 3, no. 3, pp. 119–128, 2018.
[20] C. Lv, W. Sheng, K. Liu, W. Dong, and X. Meng, "Multi-resolution modelling method based on time-state-machine in complex distribution network," *IET Cyber-Physical Systems: Theory Applications*, vol. 2, no. 4, pp. 172–179, 2017.
[21] X. Zhang, D. Liu, C. Zhan, and C. K. Tse, "Effects of cyber coupling on cascading failures in power systems," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 7, no. 2, pp. 228–238, June 2017.
[22] "NP-View," https://www.network-perception.com/np-view/.
[23] P. M. Subcommittee, "Ieee reliability test system," *IEEE Transactions on power apparatus and systems*, no. 6, 1979.
[24] M. Adibi, "300 bus power flow test case," 1993.
[25] J. Li, X. Ou, and R. Rajagopalan, *Uncertainty and Risk Management in Cyber Situational Awareness*, 09 2010, vol. 46, pp. 51–68.
[26] R. Bellman, "On a routing problem," in *1958 Quaterly of Applied Mathematics*, no. 16, 1958, pp. 87–90.
[27] L. R. Ford, "Network flow theory." in *Rand Corporation*, 1956.
[28] E. W. Dijkstra, "A note on two problems in connexion with graphs." in *Numerische Mathematik.*, no. 1, 1959, pp. 269–271.
[29] A. B. Birchfield, T. Xu, K. M. Gegner, K. S. Shetye, and T. J. Overbye, "Grid structural characteristics as validation criteria for synthetic networks," *IEEE Transactions on power systems*, vol. 32, no. 4, pp. 3258–3265, 2016.
[30] Cisco, "Cisco ASA 5500 Series Configuration Guide using the CLI, 8.2," https://www.cisco.com/c/en/us/td/docs/security/asa/asa82/configuration/guide/config/objectgroups.html.

IX. Appendix

A. Firewall Rule to Host Connectivity

The details on object group and network-object can be found in Cisco's firewall setting documentation [30].

1) Formation of network based object groups for 451 and 421 SEL relays in CAPITAL CITY substation.
   object-group network CAPITALCITY
   network-object host 10.37.1.250
   object-group network CAP CITY 451
   network-object host 10.37.1.101
   network-object host 10.37.1.102
   object-group network CAP CITY 421
   network-object host 10.37.1.201
   object-group network FTP SERVER
   network-object host 72.36.82.194
2) Nesting object groups hierarchically so that one object group can contain other object groups of the same type
   object-group network DIST
   group-object CAPITALCITY
   group-object CAP CITY 451
   group-object CAP CITY 421
3) This example creates a service object group for Inter-control Center Communications Protocol (ICCP) services. It was developed to enable data exchange over between utility control centers, Independent System Operators (ISOs), Regional Transmission Operators (RTOs), and other Generators.
   object-group service ICCP DATA tcp
   port-object eq 102
   port-object eq 8080
4) Similarly another outside network object-group is formed.
   object-group network PEER UTILS
   network-object host 192.17.102.102
   network-object host 192.17.102.103
   network-object host 192.17.102.104
5) Finally ACL is constructed for the inbound traffic as
   access-list FromOUTSIDE extended permit tcp
   object-group PEER UTILS object-group DIST
   object-group ICCP DATA
   Similarly one more ACL that allows outbound traffic of FTP protocol from DIST network to FTP server network.
   access-list FromINSIDE extended permit tcp
   object-group DIST object-group FTP SERVER
   object-group FTP DATA

The connectivity matrix for the above ACL is
1892, 72.36.82.184
1888, 10.37.1.101
1888, 1892, tcp:0-65535:21-21!tcp:0-65535:20-20
The host with ID 1888 and IP 10.37.1.101 can send FTP data port (20) and control port (21) to the FTP server with ID 1892 and IP 72.36.82.184