2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence
and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress

Approximate Power Grid Protection Against False Data Injection Attacks

Kelvin Ly*, Kevin Kwiat†, Charles Kamhoua‡, Laurent Njilla§, and Yier Jin¶
*Department of Electrical and Computer Engineering, University of Central Florida
†Haloed Sun TEK, LLC
‡US Army Research Laboratory, Network Security Branch
§Air Force Research Lab., Information Directorate, Cyber Assurance Branch
¶Department of Electrical and Computer Engineering, University of Florida
rangertime@knights.ucf.edu, kwiatk@sunyit.edu, charles.a.kamhoua.civ@mail.mil, laurent.njilla@us.af.mil,
yier.jin@ece.ufl.edu

***Abstract*—A family of new attacks have been found against power grid systems recently which are capable of affecting power grids without being detectable via conventional means. Though powerful, the attacks rely on the attacker having complete knowledge of the power grid system. This work will evaluate the power grid resiliency countering such attacks when an attacker does not have such complete knowledge. More specifically, this paper examines the rerouting that already commonly occurs in power systems as an inherent defense against this particular class of attacks by increasing the power grid topology complexity. An algorithm is developed to calculate the probability of a successful attack given a particular topology and configuration of circuit breakers. The experimental results demonstrate that the existence of defense circuit breakers in a power grid system can lead to substantial improvements in security.**

I. Introduction

The power grid is a vital component of modern life, and acts as an meritorious example of large scale cyber physical systems (CPS) integrating into vital infrastructure successfully. The power grid is also a leading example of a large network consisting of smart devices doing valuable work in the real world today. These smart devices, connected over large distances, provide the important service of monitoring and automating aspects of the power grid's operation.

However, like many CPS devices in use today, the modern power grid suffers from vulnerabilities to an acutely greater extent than traditional computing platforms. The use of widely distributed devices relying on wireless communication, using embedded devices with few hardware security features leads to a system with much greater surface area for attack and fewer resources per device to defend with. Researchers have already discussed the possibility of cascading attacks, where an attack on a few nodes has the possibility of causing a much larger fault in the system [1], [2]. The attacks on the power system in Ukraine demonstrate that these scenarios are well in the realm of possibility [3].

Additionally, a highly powerful attack, known as false data injection, has been developed that is capable of affecting operation without being detectable using traditional fault detection methods. It works using a property of the common approach towards state estimation, showing that any attack satisfying particular properties cannot be detected [4]. Research has shown that the attacker may be capable of staging this sort of attack with control over as few as four sensor nodes in some of the simulated systems. The attack can further be refined to affect only particular state variables, again, without being detected using traditional methods.

Although this type of attacks becomes a main threat to power grid, the success of this attack relies on two major assumptions: 1) The attacker has control over some number of sensor nodes; 2) The attacker has complete knowledge of the system, in that the attacker knows the exact topology of the system at all moments of the attack. Therefore, one straightforward solution to counter the fault data injection attack is to invalidate the assumptions. Note that the second assumption is less certain as many modern power systems are highly dynamic thanks to the large amount of automation the current power grid makes possible. The need to protect itself from faults and to optimize power flow mean that the topology of the modern power grid is constantly changing to meet the demands and needs of the system. This provides some inherent protection against the false data injection attack.

Based on this observation, this paper, for the first time, tries to evaluate the power grid security and resiliency countering fault injection attacks under the more realistic situation that the power grid's configuration will be changed dynamically. The main contribution of this paper is the development and the evaluation of a dynamic power grid topology which can prevent the state-of-the-art fault injection attacks.

The rest of the paper is organized as follows: Section II introduces the existing power grid protection methods countering the state-of-the-art false data injection attacks. Section III elaborates an enhanced protection method to thwart the attacks through the increased power grid topology flexibility. Experimentation setup and results are presented in

Approved for Public Release; Distribution Unlimited : 88ABW-2016-4600, Dated 19 Sep 2016

978-1-5386-1956-8/17 $31.00 © 2017 IEEE
DOI 10.1109/DASC-PICom-DataCom-CyberSciTec.2017.97

IEEE computer society

Section IV with a discussion on the experimental results in Section V. Finally, the conclusion is drawn in Section VI.

II. Related Work

The false data injection attack was introduced by Liu et al [4]. In that work the false data injection attack was formulated, and proved to be effective and possible with as few as four nodes in all of the tested simulated systems. It is also proven that there is a threshold number of compromised nodes, above which the attacker is guaranteed to be able to perform an undetectable attack. The paper also developed a variant of the false data injection attack, demonstrating that the attack can be made more specific so that it affected chosen state variable estimations. Another paper by Liu et al extended this concept of attack into a generalized false data injection attack, where the attack is allowed to cause some detectable error within some specified margin [5].

The essential equation to understanding how this is possible is the relation

$$\mathbf{Hc} = \mathbf{a} \tag{1}$$

where $\mathbf{H}$ is the state matrix of a power system, $\mathbf{c}$ is some arbitrary vector, and $\mathbf{a}$ is the attack vector. Every nonzero element of the attack vector will require the attacker to gain access to the respective sensor node to allow the attacker to change the value. For each of these compromised sensor nodes, the attacker will change the measurement by adding the corresponding element of the attack vector to it. Consequently, the new measurement vector is $\mathbf{m} + \mathbf{a}$, where $\mathbf{m}$ is the original set of measurements.

In DC state estimation, the measurements and current state are approximately related by the equation

$$\mathbf{Hx} = \mathbf{m} \tag{2}$$

where $\mathbf{x}$ is the state of the system, $\mathbf{m}$ are the measurements from the sensors, and $\mathbf{H}$ is the state matrix. There are some complications due to the fact that the system will rarely follow this relationship precisely due to noise and errors in sensor measurements, but the state estimation algorithm will attempt to produce a state vector that closely approximates this relationship. If the system's actual state was in fact $\mathbf{x} + \mathbf{c}$, then by the equation above the measurements would be $\mathbf{H}(\mathbf{x} + \mathbf{c}) = \mathbf{Hx} + \mathbf{Hc} = \mathbf{m} + \mathbf{a}$. Conversely, if the measurements the system collects equal $\mathbf{m} + \mathbf{a}$, then it will estimate its state as $\mathbf{x} + \mathbf{c}$. Thus, an attacker who successfully alters the measurement following the equation listed above will successfully spoof the system into estimating an incorrect state. Liu et al continue to demonstrate that the attacker can always successfully find an attack vector with as few as $m - n$ nonzero attack elements, due to the full rankedness of the $\mathbf{H}$ matrix and some linear algebra [5].

Bobba et al examined the effect of protecting a subset of sensor nodes to improve detection of false data injection attack [6]. Dan and Sandberg characterized the logic behind the false injection attack in the basis of graph theory rather than linear algebra, while also relating it to the observability problem in control theory [7].

Kosut et al examined the effects that attacks would have on the power market, and also developed a security metric that provides a measure of how difficult a given power grid system is to attack [8]. Chaojun et al described a defense against the attack in AC power state estimation systems [9]. They use a statistic based on the Kullback-Leibler distance metric, along with historical data, to measure the current state of the system and determine whether it is similar to previous states.

Additionally, Hug and Giampapa characterized these attacks in AC state estimation systems, which are notably more complicated than the DC state estimation systems on which Liu et al developed their attacks [10]. The work demonstrated that AC state estimation attacks were possible, albeit requiring the attacker to know much more about the system, as the AC state model is more complicated than the DC state model.

More recent works discussed the potential of attacks or defenses using a lack of knowledge of the system as a given assumption. These are more closely related to the work done here, but either reduce the problem to the specific case of the attacker not knowing about certain connected regions of the power grid, or apply statistical, inexact methods to the problem rather than attempting to find an exact attack vector as in Liu's paper. Rahman and Mohsenian-Rad examined the effect of line impedance uncertainty on the false data injection attack [11]. Liu and Li examined potential attacks on local branches of the power system without knowing the global system topology [12]. Yu and Chin, develop an approach to formulating a false data injection attack without any knowledge of the underlying power system [13]. Instead, they apply principle component analysis (PCA) on power system data to derive the values for the attack. The resulting attack does not reach the perfect undetectability of the original attack, but it does have some probability of providing a perfect attack; the attack vector calculated here approximates that found given perfect knowledge of the power system. Tang et al examine the effect of colored Gaussian noise on the measurements on detecting attacks in 2016 [14].

III. Research Approach

Our approach begins by precisely defining the problem discussed above, starting with the formulations developed for the false data injection attack. These formulations are then extended to include the unknown topology aspect of the problem. This formalization is then characterized using linear algebra, and then converted into a usable formula for simulation.

A. Threat Model

This research focuses on a specific subset of the unknown topology false data injection attack problem in DC state estimation power systems. Consequently, our threat model follows that of Liu's paper, but with relaxation of a few constraints to provide more realism; therefore, it is designed to mimic an enhanced protection scenario where the power grid is of high flexibility to invalidate one of the assumptions of an attacker. Specifically, in the enhanced framework, the attacker understands the base topology but cannot determine if certain circuit breakers are on or off. Further, the attacker has access to a certain number of sensor nodes but cannot collect information about the power grid topology, aside from knowing the structure of the power system. Besides the enhanced protection scenario, the attacker's goal is the same as in the false data injection model, i.e., compromising the power grid through fault data injection without being detected.

B. Formalized Approach

Based on this threat model, the attacker will choose attack vectors that are undetectable under both the original system without any changes and all potential system configurations where one or more circuit breakers are switched to a different state. Thus, for any state the power grid's switching can take, the attack will still be undetectable. Assuming the attacker has no additional knowledge of the system, this is a necessary and sufficient condition for a valid attack. The DC state estimation model is based on that provided by MATPOWER, a common MATLAB power systems library. Besides this, the power grid system is treated as a black box, with a circuit breaker configuration being its input and a state matrix as its output. Aside from assuming that these changes lead to differences in the resulting matrices, no other assumptions are made of the system.

The problem is then formalized as systems of homogeneous linear equations which can be simply described as

$$\mathbf{Hc} - \mathbf{Ia} = \mathbf{0} \tag{3}$$

where $\mathbf{H}$ is the n-by-m state matrix of the power system, $\mathbf{c}$ is the arbitrary vector of length n, and $\mathbf{a}$ is the attack vector of length m. $\mathbf{H}$ is assumed to be full rank. It is also assumed that the system contains m sensor nodes and is represented by n state variables.

Other features of the problem are modelled by augmenting this set of equations to include the constraints they cause. For example, adding in the constraint that an attacker cannot predict the state of one circuit breaker results in the additional equation

$$\mathbf{H}'\mathbf{c}' - \mathbf{Ia} = \mathbf{0} \tag{4}$$

where $\mathbf{H}$ represents the state matrix of the power system when the circuit breaker is in one state, and $\mathbf{H}'$ represents the other.

Limiting the attacker's ability to only attacking a subset of nodes is represented as

$$a_i = 0, \forall i \in \mathbb{A} \tag{5}$$

where $\mathbb{A}$ is the set of the sensor indices whose sensors are protected.

Permuting the components of the attack vector so that the protected nodes are the first components, this can be represented as

$$\left[\begin{array}{c|c|c} \mathbf{H} & \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & \mathbf{H}' & -\mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}|\mathbf{0} \end{array}\right] \begin{bmatrix} \mathbf{c} \\ \mathbf{c}' \\ \mathbf{a} \end{bmatrix} = \mathbf{0} \tag{6}$$

ignoring permutation. This matrix can be further extended to more cases by simply augmenting it further.

For a homogeneous linear system of equations, the solution space is equivalent to the null space of the coefficient matrix. In our cases, this solution space is also exactly equal one point in the attack space, because the coefficient matrix is assumed to be full rank, meaning that the equations (3) and (4) imply an injective mapping from $\mathbf{c}$ and $\mathbf{c}'$ to $\mathbf{a}$, Thus, assuming each solution vector has unique state space components implies that the solution vector has a unique attack component, meaning each solution vector has an injective projection into the attack vector space. Consequently, the order of the attack space is exactly equal to the order of the solution space of this system of equations.

The rank nullity theorem states that when the dimension of the null space is added to the rank of the coefficient matrix the sum must equal the dimension of the solution space [15]:

$$rk(\mathbf{A}) + null(\mathbf{A}) = n \tag{7}$$

Thus, in this representation, the potential to create an effective attack is related to the rank of the resulting matrix of coefficients, and is thus related to the linear independence of the rows and columns of the matrix.

C. Intuitive Results

Some intuitive results arise fairly directly from this relationship. The attacker's capabilities are constrained by the solution space of the above equation. The defender's capabilities, on the other hand, are described through the augmentation of the coefficient matrix. This augmented coefficient matrix places further constraints upon the system of equations such that the available attack space is reduced.

When the rank of the augmented coefficient is at its minimum, i.e., there are no addition constraints added to the base case, the rank of the coefficient matrix is exactly m, as the identity matrix addition forces the matrix to have exactly m linearly independent columns. This case has an input dimension of $n+m$. This means the solution will have a dimension space of n, which makes sense as the attacker has complete control over all the nodes in this case, and should be able to change the state without any restrictions. Similarly, when the rank of the augmented coefficient matrix is full and equal to the dimension of the input vector, as is the case when the attacker does not control any nodes, then the dimension of the solution space is zero, implying that the only solution to the question is the trivial case of all zeros, meaning an attack is impossible.

It can also be demonstrated that adding more constraints will lead to a monotonic decrease in the dimension of the attack space. Adding in one such constraint increases the dimension of the solution space by n, but could potentially increase the rank by as much as m, as it results in adding m potentially linearly independent rows. The added rows will always be at least of rank n, as it will always add n linearly independent columns due to the state matrix component part of the addition. Consequently, the security of the system against an attack increases monotonically with the addition of unknown circuit breakers until it reaches a maximum where an attacker cannot modify the state in any way, regardless of the order of addition.

IV. Experimentation

In this section, we will discuss our attempts at implementing and simulating the false data injection attack based on the research approach discussed in the previous section. This was performed in two steps, first following the previous formulations directly, and then focusing on improving the performance of the algorithm to generate experimental results in a reasonable time frame.

A. Intuitive Solution

Our first implementation precisely followed the algorithm described above, simulating the addition of unknown circuit breakers and uncompromisable nodes by augmenting the coefficient matrix appropriately. This first implementation was to sanity check this approach against the results in Liu's paper, calculating the probability of a brute force search finding a suitable combination of nodes to compromise that would allow an attacker to execute a false data injection attack [5]. This would ensure the validity of the assumptions made in our formalization, and provide some experimental insight into the properties of the algorithm. The coefficient matrix was shuffled, the uncompromised node constraint was added, and the rank of the matrix, subtracted from $n + m$, was used to determine whether that iteration resulted in a valid attack, for the three smallest test cases. This showed good parity with Liu's results.

The resulting matrix was highly sparse, but still experienced drastic increase in computational time as a result of the time complexity of rank finding algorithms. Holmes et al cite that the time complexity of exact singular value decomposition, which the method used by MATLAB to calculate matrix rank, has a time complexity of $O(min(mn^2, m^2n))$ [16]. This implies sextic growth with the number of circuit breakers constraint, along with cubic growth with test case size, multiplied by the already quadratic growth in the number of test cases, as the algorithm iterates over both unknown circuit breakers and number of protected nodes. This high degree growth rate meant that the larger test cases would impose significantly increased runtime overhead using this approach.

B. Optimizations

The sextix growth in computational time, discussed in the previous section, was too steep to allow the algorithm to perform sufficiently quickly for many of the larger test cases. Consequently, the later implementations only approximate this result, resulting in an overall speedup of 100X compared to the naive approach, with an error rate of around 5%. The main cost in the algorithm was in calculating the rank of the coefficient matrix. Consequently the optimization involved finding ways to reduce the size of this matrix wherever possible.

The first step towards the optimized implementation used the projection matrix representation of the problem, so that the equation was only in terms of $\mathbf{a}$:

$$\mathbf{Ba} = \mathbf{0} \tag{8}$$

$$\text{where } \mathbf{B} = \mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T - \mathbf{I} \tag{9}$$

which is equivalent to equation (3).

The addition of protected nodes was still done by appending an identity matrix, but the addition of unknown circuit breaker constraints was done by adding row wise their equivalent $\mathbf{B}$, $\mathbf{B}'$, instead of a much larger increase in matrix size of the preceding approach. This leads to only linear growth in matrix size, and hence only cubic growth in time complexity per test case. The results here matched the results in first test case, but the resulting implementation was still too slow to calculate values for the larger test cases on a practical time scale.

Consequently, the coefficient matrix, regardless of augmentation, can only have up to m linearly independent rows. Rows past the mth row do not contribute to the rank or to the solution of the problem because they are simply a linear combination of some set of rows already in the matrix. Thus, theoretically, the coefficient matrix should be reducible to m linearly independent rows without losing any information, because all the rows past there will always be solved if the first m rows are solved. Calculating the linearly independent set of rows through reduction to row echelon form was costly, as instead an approximation was done. The singular value decomposition (SVD) decomposes a matrix into three components:

$$\mathbf{B_{tot}} = \mathbf{U\Sigma V^\star} \tag{10}$$

where B_{tot} is the augmented matrix described above, U and V are unitary matrices, and Σ is a diagonal matrix. There will only be $rk(\mathbf{B_{tot}})$ non zero diagonal elements in Σ, as any more or any less would imply a different rank for $\mathbf{B_{tot}}$ The SVD operation of MATLAB ensures that V's diagonal elements are ordered in descending order, so all zero elements will be at the bottom-right of the matrix. Looking back at the equation we're trying to solve, we get

$$\mathbf{Ba} = \mathbf{U\Sigma V^\star a} = \mathbf{0} \tag{11}$$

premultiplying both sides by $\mathbf{U}^{-1}$ results in

$$\mathbf{U}^{-1}\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\star}\mathbf{a} = \mathbf{\Sigma}\mathbf{V}^{\star}\mathbf{a} = \mathbf{U}^{-1}\mathbf{0} = \mathbf{0} \tag{12}$$

$$\mathbf{\Sigma}\mathbf{V}^{\star}\mathbf{a} = \mathbf{0} \tag{13}$$

The diagonal component can be removed by truncating the rows corresponding to zeroed diagonals and then multiplying each of the remaining rows by the reciprocal of their corresponding diagonal element, leaving

$$\mathbf{V}^{\star}\mathbf{a} = \mathbf{0} \tag{14}$$

Thus, any coefficient matrix can be reduced to a form with dimensions m by r through singular value decomposition. However, there appears to be a loss of information during the process, as the results differ against those of the previous algorithm with an error rate of around 5% for our test cases of the IEEE 18-bus and 30-bus models.

C. Sources of Errors

The errors are likely due to some numerical instability in the algorithm, a result of repeated singular value decomposition and potentially some loss while removing the $\mathbf{\Sigma}$ and $\mathbf{U}$ components.

In the case of removing $\mathbf{\Sigma}$, the truncation actually takes place at a small positive value rather than zero, as MATLAB does not, in practice, appear to produce perfectly zeroed diagonals. In the case of removing $\mathbf{U}$, there is a difference in the meaning of the system of equations before and after removing it. With the $\mathbf{U}$ (Case 1), the system of equations means that the attack vector is undetectable if a linear combination of the vector $\mathbf{\Sigma}\mathbf{V}^{\star}\mathbf{a}$ is zero, whereas the form (Case 2) without it states that the attack vector is undetectable if the vector $\mathbf{\Sigma}\mathbf{V}^{\star}\mathbf{a}$ is zero.

The Case 1 states that a linear combination of the components of that vector must be zero, whereas the Case 2 implies a tighter bound, in that all the components of that vector must be vector. The Case 2 implies the Case 1, but not vice versa, meaning the solutions of the Case 2 will always form a subset of the solutions of the original equation.

To reduce the numerical degradation caused by repeated SVD decomposition and truncation, each iteration checks the rank of $\mathbf{V}^{\star}$ against the raw augmented matrix. If they are equal, then the shortened $\mathbf{V}^{\star}$ is used for the next iteration; else, the augmented matrix is used. This provides a weak, but still valuable, check to ensure that at least discrepancies caused by a single iteration are prevented.

Overall, this allows the matrix to remain approximately constant in size, a drastic improvement over the linear or quadratic growth symptomatic of the preceding approaches. This allows for the 10X speedup relative to the original approach for the IEEE 30-bus case, and it is conjectured (but unmeasured, as the larger cases are not practical to run for the older algorithms) that this speed improvement is even greater for the other test cases. This approximation overall works fairly well in practice, given the empirically found 5% error rate.

Algorithm 1 Algorithm used to calculate successful attack rate

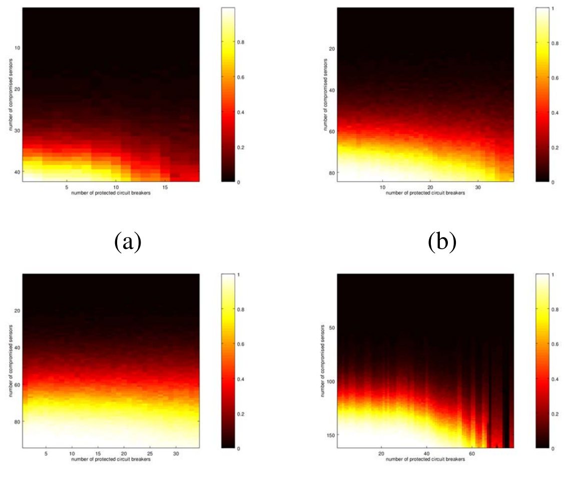
```
$H \leftarrow H(branches)$
$(U, \Sigma, V) \leftarrow SVD(H)$
$[NumRows, NumCols] \leftarrow rows(H)$
$Rows \leftarrow sum(V > threshold)$
$B_{tot} \leftarrow V(1 : Rows, :)$
for all $i \in [0..size(branches))$ do
    $branches(i) \leftarrow 0$
    $H_{tmp} \leftarrow H(branches)$
    $B_{aug} \leftarrow [B_{tot}; H_{tmp}]$
    $(U, \Sigma, V) \leftarrow SVD(H_{aug})$
    $Rows \leftarrow sum(V > threshold)$
    $B_{tmp} \leftarrow V(1 : Rows, :)$
    if $rk(B_{tmp}) < rk(B_{aug})$ then
        $B_{tot} \leftarrow B_{aug}$
    else
        $B_{tot} \leftarrow B_{tmp}$
    end if
    for all $k \in [1..Rows - Cols + 1)$ do
        $SuccessRate \leftarrow 0$
        for all $j \in [0..NumTrials)$ do
            $shuffled \leftarrow Shuffle(B_{tot})$
            $AttackAugment \leftarrow [\mathbf{I}(\mathbf{k})\mathbf{0}(\mathbf{k}, \mathbf{Rows} - \mathbf{k})]$
            $tmp \leftarrow [shuffled; AttackAugment]$
            $result \leftarrow rows - rk(tmp)$
            if $result \leq 0$ then
                $SuccessRate \leftarrow SuccessRate + 1$
            end if
        end for
        $Print(SuccessRate/NumTrials)$
    end for
end for
```

The final algorithm, shown in Algorithm 1, adds an inner loop to the algorithm described in the preceding sections to allow different orders of compromised nodes to be tried. This will allow it to calculate the probability of an attacker's brute force attempt successfully finding a valid attack vector, while varying both the number of unknown circuit breakers in the system and the number of nodes the attacker can compromise. Unfortunately, the order of circuit breakers was not shuffled, as this would have increased the time requirements too greatly. Each parameter set was sampled a thousand times; $NumTrials$ in the above is 1000.

D. Experimental Results

Similar to the work in [5], we perform our experiments in 14-bus, 30-bus, 39-bus, and 57-bus scenarios.

As the algorithm involves two independent variables, the results are best represented using a heat map. The color in these figures represents the probability of an attacker finding an attack vector.



(a)

(b)

(c)

(d)

Fig. 1: Successful attack search probability for the (a) 14-bus, (b) 30-bus, (c) 39-bus, and (d) 57-bus test cases under the enhanced power grid protection.

As the figures in Figure 1 show, the variances with the effect of unknown circuit breakers on the attacker are significant. The color represents the probability of the attacker's brute force search finding a successful attack vector. In all cases, as expected, increasing circuit breaker count correlates to the reduction of successful attack probability. Both the lowest and highest bus count cases show more significant effect from unknown circuit breaker topology than the moderate bus count cases, especially near the higher values.

The effect of circuit breaker count is highest when the attacker controls more nodes; the change in color is less noticeable at lower node counts. This implies that the circuit breakers provide higher defense when the attacker controls more sensor nodes.

There is substantial variance in the effect of the circuit breakers between the test cases. This implies that the network topology has a substantial effect on its effectiveness. In all cases there is a downward trend, which demonstrates the previously asserted property that circuit breakers always improve security.

V. Discussions

The artifacts visible in the IEEE 57-bus case (Figure 1d) indicates that the algorithm cannot scale to higher numbers with high accuracy due to the increased computation complexity. The large spikes seen there are not consistent with what is expected, as the attack probability intuitively should not increase with an increase in circuit breaker count. Moreover, the output from the MATLAB script indicated that the code did not to use the truncating optimization the majority of the code, in contrast to the other models. The non-truncating path was meant to hopefully reduce errors, as the inequality should in most cases signify that the reduced form has the same rank as the non-truncated form. Thus, the algorithm may need to be modified to provide better accuracy at higher bus counts. More testing will need to be performed to better understand this behavior.

VI. Conclusion

An algorithm was developed which approximated the probability of attack within around 5% of the values available through previous solutions. Additionally, how this optimized approach allowed the effect of unknown states of circuit breakers on the attacker's ability to perform an attack has been analyzed. The results indicate that leveraging circuit breakers can help enhance the security of the power grid countering fault injection attacks. Nevertheless, based on the currently collected data, simply adding more breakers may not be sufficient to defend against attacks if the attacker can compromise a large set of sensor nodes.

More research will be done to improve the algorithm developed here to ensure the accuracy of the results. For example, the attacker may not mount a static attack, as is assumed in the paper, but may dynamically change their attack vectors based on some Bayesian analysis of the state variables of the system. Further research may work to examine optimum strategies from an attacker's perspective to estimate the the state matrix of the system, given knowledge of some subset of it.

Acknowledgements

This work was supported in part by Air Force Research Lab (AFRL) Visiting Faculty Research Program and Florida Center of Cybersecurity (FC2) 2016 Collaborative Seed Grant Program. Mr. Kelvin Ly was partly supported by the Semiconductor Research Corporation (SRC/Intel) Graduate Fellowship Program.

References

[1] R. Kinney, P. Crucitti, R. Albert, and V. Latora, "Modeling cascading failures in the north american power grid," *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 46, no. 1, pp. 101–107, 2005.

[2] J.-W. Wang and L.-L. Rong, "Cascade-based attack vulnerability on the us power grid," *Safety Science*, vol. 47, no. 10, pp. 1332–1336, 2009.

[3] R. M. Lee, M. J. Assante, and T. Conway, "Analysis of the cyber attack on the ukrainian power grid," 2016.

[4] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, ser. CCS '09. New York, NY, USA: ACM, 2009, pp. 21–32. [Online]. Available: http://doi.acm.org/10.1145/1653662.1653666

[5] ——, "False data injection attacks against state estimation in electric power grids," *ACM Transactions on Information and System Security (TISSEC)*, vol. 14, no. 1, p. 13, 2011.

[6] R. B. Bobba, K. M. Rogers, Q. Wang, H. Khurana, K. Nahrstedt, and T. J. Overbye, "Detecting false data injection attacks on dc state estimation," in *Preprints of the First Workshop on Secure Control Systems, CPSWEEK*, vol. 2010, 2010.

[7] G. Dan and H. Sandberg, "Stealth attacks and protection schemes for state estimators in power systems," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*. IEEE, 2010, pp. 214–219.

[8] O. Kosut, L. Jia, R. J. Thomas, and L. Tong, “Malicious data attacks on the smart grid,” *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 645–658, 2011.

[9] G. Chaojun, P. Jirutitijaroen, and M. Motani, “Detecting false data injection attacks in ac state estimation,” *IEEE Transactions on Smart Grid*, vol. 6, no. 5, pp. 2476–2483, Sept 2015.

[10] G. Hug and J. A. Giampapa, “Vulnerability assessment of ac state estimation with respect to false data injection cyber-attacks,” *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1362–1370, Sept 2012.

[11] M. A. Rahman and H. Mohsenian-Rad, “False data injection attacks with incomplete information against smart power grids,” in *Global Communications Conference (GLOBECOM), 2012 IEEE*, Dec 2012, pp. 3153–3158.

[12] X. Liu and Z. Li, “Local load redistribution attacks in power systems with incomplete network information,” *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1665–1676, July 2014.

[13] Z. H. Yu and W. L. Chin, “Blind false data injection attack using pca approximation method in smart grid,” *IEEE Transactions on Smart Grid*, vol. 6, no. 3, pp. 1219–1226, May 2015.

[14] B. Tang, J. Yan, S. Kay, and H. He, “Detection of false data injection attacks in smart grid under colored gaussian noise,” *arXiv preprint arXiv:1607.06015*, 2016.

[15] C. D. Meyer, *Matrix analysis and applied linear algebra*. Siam, 2000, vol. 2.

[16] M. Holmes, A. Gray, and C. Isbell, “Fast svd for large-scale matrices,” in *Workshop on Efficient Machine Learning at NIPS*, vol. 58, 2007, pp. 249–252.