J. Mod. Power Syst. Clean Energy (2018) 6(5):860–871
https://doi.org/10.1007/s40565-018-0432-2

Graph theoretical defense mechanisms against false data injection attacks in smart grids

Mohammad Hasan ANSARI[1,2], Vahid Tabataba VAKILI[1,2], Behnam BAHRAK[1,2], Parmiss TAVASSOLI[1,2]

Abstract This paper addresses false data injection, which is one of the most significant security challenges in smart grids. Having an accurately estimated state is of great importance for maintaining a stable running condition of smart grids. To preserve the accuracy of the estimated state, bad data detection (BDD) mechanisms are utilized to remove erroneous measurements due to meter failures or outsider attacks. In this paper we use a graph-theoretical formulation for false data injection attacks in smart grids and propose defense mechanisms to mitigating this type of attacks. To this end we discuss characteristics of a typical smart grid graph such as planarity. Then we propose three different approaches for finding optimal protected meters set: a fast and efficient heuristic algorithm that works well in practice, an approximation algorithm that provides guarantee for the quality of the protected set, and an exact algorithm that find the optimal solution. Our extensive simulation results show that our algorithms outperform similar existing solutions in terms of different performance metrics.

Keywords Smart grids, False data injection, Attack, Minimum Steiner tree problem

CrossCheck date: 21 June 2018

Received: 12 June 2017 / Accepted: 21 June 2018 / Published online: 6 September 2018
© The Author(s) 2018

✉ Mohammad Hasan ANSARI
mh-ansari@elec.iust.ac.ir

Vahid Tabataba VAKILI
vakily@iust.ac.ir

Behnam BAHRAK
bahrak@ut.ac.ir

Parmiss TAVASSOLI
p.tavassoli@ut.ac.ir

[1] Electrical Engineering Department, Iran University of Science and Technology, Tehran, Iran

[2] School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran, Iran

1 Introduction

The smart grid is one of the fastest growing, most complex, and vitally important cyber-physical systems in the world. Smart grid is envisioned to fully integrate high-speed and two-way communication technologies into millions of power equipment to establish a dynamic and interactive infrastructure with new energy management capabilities, such as advanced metering infrastructure (AMI) and demand response [1]. At the same time, smart grids heavily rely on information and communication technology to achieve efficient and reliable operation. The reliable operation of smart grids depend on the security and robustness of their information infrastructure against attacks and failures [2, 3]. With the incorporation of information technology, the smart grids would be exposed to potential cyber attacks. These attacks target the availability, integrity, and confidentiality of smart grids' functionalities such as monitoring and control systems [4].

This paper addresses false data injection (FDI) attack, which is the most well-known and critical data integrity vulnerability in smart grids. Successful detection of FDI attacks is essential for ensuring secure operation of smart grids. Recent works focused on FDI attacks have demonstrated how an opponent can bias measurements and bypass residual-based bad data detection (BDD) systems [5]. This type of attack is called undetectable FDI [6]. State of the art studies have shown that FDI attacks could be unobservable

STATE GRID
STATE GRID ELECTRIC POWER RESEARCH INSTITUTE

in presence of an intruder who coordinately controls a small number of smart meters. In other words, there is no defense mechanism against these malicious attacks in the existing state estimation processes. An adversary can perform a man-in-middle attack on the communication channel between the control center and measurement units and send malicious false data. The only step that might detect the attack during the state estimation process is using a BDD system.

The BDD is typically used to ensure the integrity of state estimation and filter faulty measurements introduced by device malfunctions or malicious attacks. But almost all bad data detectors are unable to detect a certain type of attack, which is known as the undetectable FDI attack. Experimental results on the supervisory control and data acquisition (SCADA) systems under FDI attack demonstrate that the attack is successful at least 50% of the time without alerting the BDD system [7]. Therefore, it is important to understand that FDI attacks can circumvent BDD and insert bias into the value of the estimated state stealthily and affect the system's stability [8]. Biased estimates could directly lead to serious social and economic consequences and even manipulate the electricity price in the power market [9], and even result in widespread blackouts.

The main contributions of this paper are:

1) Optimal protection problem in smart grids is modeled as a Steiner tree problem, taking into account the practical concerns of critical bus importance and financial considerations regarding the growth strategy.
2) We determine the critical nodes in the smart grid using social network metrics and use these nodes as terminals in a weighted Steiner tree to find the optimal protection strategy.
3) Since Steiner tree problem might be solved faster and simpler in planar graphs, we test the planarity of the smart grid network. We also use a maximal edge planar subgraph detection algorithm to extend our algorithm for solving Steiner tree problem in planar graphs to a heuristic algorithm for non-planar smart grid graphs.
4) We propose an efficient heuristic algorithm to find the minimum Steiner tree and its corresponding optimal protection set which outperforms existing heuristic algorithms for this purpose.
5) To guarantee the quality of the protection set, we propose two approaches: an exact but slow algorithm that finds the optimal solution and an approximation algorithm which guarantees that the cost of its solution is at most 1.39 times the optimal solution, but is much faster than the exact algorithm.

The rest of the paper is organized as follows. Section 2 discusses the related work. In Section 3, we describe the system model that is used throughout the paper. Section 4 presents the smart grid graph and explains the graph-based state estimation problem. In Section 5, we propose our approaches for determining the protection set of measurements. In Section 6, we evaluate the proposed approaches and provide simulation results. Finally, some concluding remarks and directions of future research are presented in Section 7.

2 Related work

A wide range of studies are related to the security challenges in smart grids. In this section, we briefly cover two lines of research that their results are mostly related to our work. At first we survey important types of FDI attacks that are proposed in the literature. Then, we focus on protection mechanisms and attack detection methods in smart grids.

2.1 FDI attacks in smart grids

FDI attacks on smart grids were first introduced in 2009 by [5] and expanded in [10]. Following this work, many researchers tried to come up with more realistic attacks on smart grid state estimation. In [11], a comprehensive review of state of the art in FDI attacks against modern power systems is presented. In [12], authors proposed an attack strategy for smart grid state estimation and considered two regimes of attacks: the strong attack regime where the adversary attacks a sufficient number of meters so that the attack becomes unobservable by the control center and the weak attack regime where the adversary controls only a small number of meters. The problem is investigated from a decision theoretic perspective for both the control center and the adversary.

In [13], authors point out that a realistic FDI attack is essentially an attack with incomplete information due to the attacker's lack of real-time knowledge with respect to various grid parameters and attributes such as the position of circuit breaker switches and transformer tap changers, and also because of the attacker's limited physical access to grid facilities. They also characterize FDI attacks mathematically with incomplete information from both the attacker's and grid operator's viewpoints. In [14], the case of a bi-level hierarchical state estimator that provides only partial observability to lower-tier state estimators, is considered and an attack model is proposed which presents a formulation for identifying minimal sets of additional measurements to tolerate k-measurement attacks in this hierarchical state estimator.

An algorithm to construct unobservable FDI attacks for an AC state estimator is proposed in [15]. In [16], the problem of finding the optimal attack strategy, i.e. a data-injection attack strategy that selects a set of meters to manipulate in order to cause the maximum damage is studied. A bi-level mixed integer linear programming (MILP) model for determining the optimal measurements set protection is presented in [17]. The stealthy deception attack in remote state estimation, which is a typical attack in cyber physical system (CPS), is investigated in [18]. A two-stage attack scheme to demonstrate the practical feasibility of unobservable FDI attacks in smart grids is described in [19].

In [20], the authors studied the general problem of blind FDI attack using the principal component analysis (PCA) approximation method without the knowledge of Jacobian matrix and assumptions on the distribution of state variables. The proposed attack is proven to be approximately stealthy. An efficient strategy for determining the optimal attack region that requires reduced network information is proposed in [21]. Reference [22] proposes an imperfect FDI attack model and its corresponding forecasting-aided implementation method against the nonlinear power system state estimation by introducing an attack vector relaxing error. An end-to-end case study of how to instantiate real FDI attacks to the AC state estimation process presented in [23]. Authors in [24] use two-stage sparse cyber-attack models for smart grid with complete and incomplete network information and then propose a novel detection mechanism based on dual optimization problem and stacked auto-encoder (SAE) typical deep learning method.

2.2 Detection and protection mechanisms against FDI

Researches focused on devising mechanism against FDI attacks can be categorized in two classes: detection mechanisms and protection mechanisms. Here we briefly survey the two categories.

2.2.1 Detection mechanisms for FDI attack

Reference [6] proposed a detection procedure with protecting a critical selected set of sensor measurements in smart grid. The problem of FDI attack detection is intended as a framework that characterizes the attack as an optimization problem through a security metric in [7]. Since distributed state estimation will become an important alternative to centralized and hierarchical solutions in smart grid, a fully distributed attack detection algorithm is introduced in [25]. In [26], the attack detection problem in the smart grid is formulated as a statistical learning problem for different attack scenarios in which the measurements are observed in batch or on-line settings and a machine learning algorithm is used to classify the measurements and detect the attack. A graph-theoretic algorithm for localizing the target of FDI attacks in large power system networks using real-time synchrophasor measurements is proposed in [27]. In [28], the problems of state estimation and attack detection in smart grids when the measurements are corrupted by colored Gaussian noise are considered.

2.2.2 Protection mechanisms against FDI

In this category, researchers try to find necessary and sufficient conditions for selecting the protection measure set and analyzing the properties of the optimal solution with minimum cost and number of measurements. Considering the typical large size of electrical grids, the selection of subsets of measurement units for protection is a highly complex combinatorial optimization problem. Defense mechanisms against false-data injection attacks for the first time investigated in [29]. A fast greedy algorithm to select a subset of measurements to be protected is presented in [30]. A protection approach based on dividing the large system to several subsystems in order to improve the sensitivity of BDD system is proposed in [31]. A relationship between the system stability indices and FDI attacks extracted in [32]. In [33], a graphical method is introduced as a defending mechanism against FDI attacks on power system. They have proven if a subset of meter measurements selected securely, no FDI attack can be launched. Also, in [34] using covert power network topological information and graph concepts, a novel defense mechanism is presented. In a practical scenario to protect smart grid against unknown attackers with unpredictable and dynamic behaviors, a novel adaptive Markov strategy (AMS) is proposed in [35].

3 System model and assumptions

In order to investigate the FDI attack in smart grid, we first describe the linear and simplified DC state estimation model that is used in our work. Let us consider the DC state estimation problem in a network with $n+1$ nodes. The network states are comprised of bus phase and voltage amplitude. Often voltage amplitude can be measured directly, while to obtain the phase angle, state estimation is needed [36]. In general, a network with a large number of phasor measurement units (PMUs) produces more accurate state estimations. The major part of Jacobian matrix $\boldsymbol{H}$ has a linear relation with PMU observations. In other words, if the number of PMUs that are installed in the network is

STATE GRID
STATE GRID ELECTRIC POWER RESEARCH INSTITUTE

sufficient, such that the network is observable from these PMUs, then $\boldsymbol{H}$ is linear. In this case, linearization error is decreased significantly and thus using DC model is optimal. We assume that the attacker use linear approximation DC estimation model but the network operator uses accurate model. In DC model, voltage amplitude and phase angle estimations are calculated based on active power of transmission line and injection power into the buses [36]. An arbitrary bus with zero phase angle is selected as the reference bus and states are n-dimensional vector phase voltage angles $\boldsymbol{x} = (\theta_1, \theta_2, \ldots, \theta_n)'$. If $\boldsymbol{z} = (z_1, z_2, \ldots, z_m)'$ be the m-dimensional measurement vector, then the relation between $\boldsymbol{z}$ and $\boldsymbol{x}$ can be modeled as [36]:

$$\boldsymbol{z} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{e} \tag{1}$$

where $\boldsymbol{e} \sim N(0, \boldsymbol{R})$ is the measurement Gaussian noise with covariance matrix $\boldsymbol{R}$ and zero mean. In a fully observable system equipped with PMUs, the matrix $\boldsymbol{H}$ in the control center is complete and has no linearization error. But in practice and especially from the attacker's perspective, the system is partially observable, i.e. $\boldsymbol{H} = [\boldsymbol{H}_1, \boldsymbol{H}_2]$, where $\boldsymbol{H}_1$ is an exact linear model for the observable part that obtained from PMUs and $\boldsymbol{H}_2$ is a linear approximation for the unobservable part. If $\boldsymbol{H}$ is full rank (i.e. $rank(\boldsymbol{H}) = n$), then the weighted linear square (WLS) estimate $\hat{\boldsymbol{x}}$ is given by [29]:

$$\hat{\boldsymbol{x}} = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{z} \tag{2}$$

Since $rank(\boldsymbol{H}) \leq m$, at least n measurements are required to extract the optimal state estimation and $n - m$ measurement units should be used to enhance resilience against errors.

3.1 FDI attack model

The BDD algorithm in SCADA systems compares the norm-2 residuals (i.e. $\| \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}} \|$) with a predefined threshold τ to make decisions [33]:

$$\| \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}} \| < \tau \tag{3}$$

It has been proven that if an attacker use cleverly driven attack vector $\boldsymbol{a} = \boldsymbol{H}\boldsymbol{c}$, where $\boldsymbol{c} = [c_1, c_2, \ldots, c_n]^{\mathrm{T}}$ is a non-zero vector that the attacker designed to cheat the BDD and launch an undetectable attack as follows [5]:

$$\begin{aligned} \| \boldsymbol{z} + \boldsymbol{a} - \boldsymbol{H}(\hat{\boldsymbol{x}} + \boldsymbol{c}) \| &= \| \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}} + (\boldsymbol{a} - \boldsymbol{H}\boldsymbol{c}) \| \\ &= \| \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}} \| \leq \tau \end{aligned} \tag{4}$$

Then the estimated state is given by:

$$\begin{aligned} \hat{\boldsymbol{x}}_a &= (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}(\boldsymbol{z} + \boldsymbol{a}) \\ &= (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}(\boldsymbol{z} + \boldsymbol{H}\boldsymbol{c}) \\ &= \hat{\boldsymbol{x}} + \boldsymbol{c} \end{aligned} \tag{5}$$

But, it is impossible for the attacker to obtain exact $\boldsymbol{H}$ in practice, which in turn makes it impossible to approximate $\boldsymbol{a} = \boldsymbol{H}\boldsymbol{c}$. Therefore, if we assume that attacker can deceive the BDD system based on residual norm-2, then the attacker should be able to perform the attack by computing $\epsilon = \| \boldsymbol{a} - \boldsymbol{H}\boldsymbol{c} \| \leq \tau - \| \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}} \|$, and then $\| \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}} \| \leq \tau$. In an attack scenario that utilizes $\boldsymbol{z}_a = \boldsymbol{z} + \boldsymbol{a}$, the estimated state vector is $\hat{\boldsymbol{x}}_{FDI} = \hat{\boldsymbol{x}} + \boldsymbol{c}$. Therefore, residual norm-2 in the BDD system is given by:

$$\begin{aligned} \| \boldsymbol{z}_a - \boldsymbol{H}\hat{\boldsymbol{x}}_{FDI} \| &= \| \boldsymbol{z} + \boldsymbol{a} - \boldsymbol{H}(\hat{\boldsymbol{x}} + \boldsymbol{c}) \| \\ &\leq \| \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}} \| + \| \boldsymbol{a} - \boldsymbol{H}\boldsymbol{c} \| \end{aligned} \tag{6}$$

As a result we always have $\| \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}}_{FDI} \| \leq \tau$ and the attack is undetectable until $\epsilon \leq \tau - \| \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}} \|$. The full attack vector assumption in [5, 6, 29, 33, 34] is a special case where $\epsilon = 0$.

In practice, attackers are unable to access the exact Jacobian matrix $\boldsymbol{H}$, because of limited access to operational information and restricted physical access to the network. In other words, the attacker's information about the system is incomplete and can be modeled using an error matrix $\boldsymbol{\delta}$: $\boldsymbol{H} \longrightarrow \boldsymbol{H} + \boldsymbol{\delta}$. Thus the attack vector produced by the attacker is $\boldsymbol{a} = (\boldsymbol{H} + \boldsymbol{\delta})\boldsymbol{c} = \boldsymbol{H}\boldsymbol{c} + \boldsymbol{\delta}\boldsymbol{c}$. In this situation, estimated state vector given by:

$$\begin{aligned} \hat{\boldsymbol{x}}_a &= (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}(\boldsymbol{z} + \boldsymbol{a}) \\ &= (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}(\boldsymbol{z} + \boldsymbol{H}\boldsymbol{c} + \boldsymbol{\delta}\boldsymbol{c}) \\ &= (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{z} \\ &\quad + (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}(\boldsymbol{H}\boldsymbol{c}) \\ &\quad + (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}(\boldsymbol{\delta}\boldsymbol{c}) \\ &= \hat{\boldsymbol{x}} + \boldsymbol{c} + \boldsymbol{A}\boldsymbol{\delta}\boldsymbol{c} \end{aligned} \tag{7}$$

where $\boldsymbol{A} = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{R}^{-1}$. Due to the incomplete information, the attack vector amplitude is changed to $\bar{\boldsymbol{c}} = \boldsymbol{c} + \boldsymbol{A}\boldsymbol{\delta}\boldsymbol{c}$ which increases the attack detection probability. In this case, norm-2 of measurement residuals in the BDD system is given by:

$$\begin{aligned} \boldsymbol{r}_a = \boldsymbol{z}_a - \boldsymbol{H}\hat{\boldsymbol{x}}_a &= \boldsymbol{z}_a - \boldsymbol{H}(\hat{\boldsymbol{x}} + \bar{\boldsymbol{c}}) \\ &= \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}} + \boldsymbol{a} - \boldsymbol{H}\bar{\boldsymbol{c}} \\ &= \boldsymbol{r} + \boldsymbol{a} - \boldsymbol{H}\bar{\boldsymbol{c}} \\ &= \boldsymbol{r} + (\boldsymbol{H}\boldsymbol{c} + \boldsymbol{\delta}\boldsymbol{c}) - \boldsymbol{H}(\boldsymbol{c} + \boldsymbol{A}\boldsymbol{\delta}\boldsymbol{c}) \\ &= \boldsymbol{r} + (\boldsymbol{I} - \boldsymbol{H}\boldsymbol{A})\boldsymbol{\delta}\boldsymbol{c} \\ &= \boldsymbol{r} + \boldsymbol{B}\boldsymbol{\delta}\boldsymbol{c} \end{aligned} \tag{8}$$

where $\boldsymbol{B} = \boldsymbol{I} - \boldsymbol{H}\boldsymbol{A}$. Therefore the attack is undetectable if:

$$\begin{aligned}\epsilon &= ||\boldsymbol{a} - \boldsymbol{H}\bar{\boldsymbol{c}}|| = ||(\boldsymbol{H}\boldsymbol{c} + \boldsymbol{\delta c}) - \boldsymbol{H}(\boldsymbol{c} + \boldsymbol{A}\boldsymbol{\delta c})|| \\ &= ||\boldsymbol{I} - \boldsymbol{H}\boldsymbol{A}||||\boldsymbol{\delta c}|| \\ &\leq \tau - ||\boldsymbol{\delta z} - \boldsymbol{H}\hat{\boldsymbol{x}}||\end{aligned} \tag{9}$$

Upper bound on $\boldsymbol{\delta c}$ indicates the trade-off between the achievable information accuracy and the attack amplitude.

4 Smart grid graph and state estimation protection

In general, system operators try to make sure that it is impossible to launch undetectable attacks to falsify a certain set of state variables $D \subseteq I$, where I is the set of unknown estimated states. The system operators can decrease the rate of undetectable FDI attacks using secure measurements [9, 29, 34, 37]. In fact the probability of an undetectable attack will be negligible if and only if the set of secure measurements P protects state variables D such that [29]:

$$rank(\boldsymbol{H}_{P,*}) = rank(\boldsymbol{H}_{P,I\backslash D}) + |D| \tag{10}$$

where $\boldsymbol{H}_{P,*}$ is a sub-matrix of $\boldsymbol{H}$ that contains rows corresponding to P and $\boldsymbol{H}_{P,I\backslash D}$ is a sub-matrix of $\boldsymbol{H}_{P,*}$ that contains the columns corresponding to D. We need to minimize the protection cost of state variables D:

$$\begin{cases} \min\limits_{P\subseteq M} |P| \\ \text{s.t.} \quad rank(\boldsymbol{H}_{P,*}) = rank(\boldsymbol{H}_{P,I\backslash D}) + |D| \end{cases} \tag{11}$$

where M is a meters set. It has been proved that solving (11) is an NP-hard problem [33]. Note that in order to simplify the problem, we have assumed for each measurement the cost of labor and monitoring systems installation is fixed.

4.1 Observability and state variable protection

In this section, we first introduce graph-theoretical specifications of the power network and then explain the equivalence relationship between the observability and the state estimation protection. Next, we use graph theory to formulate the state estimation protection (11).

Power networks can be modeled as an undirected graph where vertices and edges correspond to buses and transmission lines, respectively. For each edge e_i, the head and tail vertices are denoted as e_i^h and e_i^t, respectively. We use N_j to represent the set of edges connected to vertex v_j. The power network observability analysis investigates whether it is possible to find a unique estimation from the unknown state variables or not [38]. In [39], the concept of network observability is extended to the state variable observability. Based on the results of [29], the state variables set $D \subseteq I$ is observable from the measurements set $P \subseteq M$ if and only if the unique estimation of D can be obtained using the vector measurements of P. In other words for each arbitrary measurement $\boldsymbol{z}_P$ and two different vectors $\bar{\boldsymbol{\theta}} \neq \boldsymbol{\theta}$ we always have:

$$\boldsymbol{H}_{P,*}\bar{\boldsymbol{\theta}} = \boldsymbol{H}_{P,*}\boldsymbol{\theta} = \boldsymbol{z}_P \Longrightarrow \bar{\boldsymbol{\theta}} = \boldsymbol{\theta} \qquad \forall k \in I \tag{12}$$

Therefore the measurement subnetwork $\bar{G}(P) = (v', \varepsilon')$ is observable if and only if all the unknown state variables S is observable from P, i.e.

$$rank(\boldsymbol{H}_{P,S}) = |S| \tag{13}$$

where $S = v'\backslash R$ and R is the reference bus. It is clear from (13) that while $|D| \leq |P|$, D is observable from P. Thus the set of state variables D is protected against undetectable FDI attacks using the protected measurement set P if and only if D is observable from P [33].

4.2 Optimal protection graph

Relation between network observability and spanning trees has been established in [39]. A spanning tree for a connected graph G is a tree containing all the vertices of G. The complete measurement network $G = (v, \varepsilon)$ is observable if and only if the graph of G contains a spanning tree where each edge of which is mapped to a meter on it or an injection meter that measures it. The measurement subnetwork $\bar{G}(M) = (\bar{v}, \bar{\varepsilon})$ is observable if and only if the graph defined on $G(P)$ contains a tree that connects all vertices in $\bar{v}$ and each edge is uniquely mapped to a meter in P.

It can be seen that the tree which is made of a protected meter set consists of the protected unknown state variables. Thus it is possible to formulate the optimal state protection problem as a minimum measurement Steiner tree (MMST) problem in the graph of G [34]. A minimum Steiner tree is a tree in a graph G which spans a given subset of vertices with the minimal total distance on its edges. In other words, to protect the set of state variables D with minimum cost, it is enough to find minimum weighted Steiner tree (MWST) $T^* = (V^*, \varepsilon^*)$ with meter set $P^* \subseteq M$ under the following conditions: ① V^* is a set of all vertices measured by P^*; ② $D \subset V^*$; ③ each edge in ε^* is uniquely mapped to a meter in P^*.

The measurement set P^* is an optimal solution of (11). We consider MWST problem instead of the spanning tree problem, because T^* only depends on a subgraph of the full measurement graph and all unknown state variables in T^* and D are observable from P^*. The MWST problem is an NP-hard problem and all known exact algorithms for

STATE GRID
STATE GRID ELECTRIC POWER RESEARCH INSTITUTE

solving it have exponential time complexities with respect to $|D|$ or $|I| - |D|$ [40].

5 Proposed algorithms

In this section we present our approaches for solving optimal protection problem, which formulates protecting the state variables in a smart grid with minimum number of measures against FDI attacks.

5.1 Optimal state variable protection problem in a smart grid with critical buses

In real smart grids, buses have fundamental differences in terms of level of importance. In other words, the priorities of protecting different buses against undetectable FDI attacks are not the same. In node priority selection process, practical issues such as the cost of implementing protection processes and the challenges of budget limit must be considered. Therefore, we consider different priorities for buses, lines and areas in formulating the optimal protection problem based on their importance. Next, we propose a novel approach to find the protected meter set using weighted Steiner tree. In order to determine the importance of a node in the smart grid we use centrality measures.

Since we have buses with various roles and importance in a power system, the following assumptions are considered in modeling the network graph. The graph of power network is defined as $G = (V, E)$ where V and E are the set of vertices and edges, respectively, with $|V| = n$ and $|E| = m$. The terminal nodes of Steiner tree denoted as $K \subseteq V, |K| = k$. The importance of each bus in the network is modeled by a cost function $\omega : V \longrightarrow R^{+}$. The most critical nodes have a value of 100, and the least valuable nodes have a value of 1. The criticality of a node is determined by centrality metrics as described in Section 5.3. Since we are dealing with terminal nodes of different values in the Steiner tree, our optimization problem must be modeled as a MWST problem. In other words we are formulating the state variables protection problem as a MWST $T = (V_T, E_T)$ problem with cost function $\omega : V \longrightarrow R^{+}$ such that $K \subseteq V_T \subseteq V$ and $E_T \subseteq E$. The objective function is to find minimum cost Steiner tree, i.e. the tree that covers all the terminal nodes K and its total cost $C = \sum v \in V(T)\omega(v)$ is minimized:

$$\min_{P \subseteq M} C = \sum_{v \in V_T} \omega(v) \tag{14}$$

It has been proved that solving (14) is an NP-hard problem [41].

5.2 Testing planarity of power network

As mentioned earlier, finding an optimal weighted Steiner tree is an NP-hard problem and there is no existing exact algorithm that solves this problem in polynomial time. But the problem of finding the minimum cost Steiner tree in a planar graph can be solved in polynomial time [42]. Therefore, we first test the planarity of power network using Boyer-Myrvold algorithm [43], to improve the performance of our proposed approaches in case of the planarity of the smart grid graph. If the graph is planar, protected node determination might be solved in polynomial time. Otherwise, we use edge-maximal planar subgraph detection algorithm to compute maximal planar subgraph [44]. The edge-maximal planar subgraph algorithm works as follows:

1) Compute spanning tree T on $G = (V, E)$ and then initialize $E(T) \longrightarrow E_p$.
2) For each $e \in E \backslash E_p$ check whether $(V, E_p \bigcup \{e\})$ is planar or not.
3) If e exists in the new planar graph, then update $E_p \bigcup \{e\}$.
4) Consider(V, E_p) as an edge-maximal planar subgraph of (V, E) .

The Steiner tree obtained on this planar subgraph is a suboptimal solution for the original graph $G = (V, E)$ [45].

5.3 Centrality criteria and weighting nodes

There are critical nodes in all networks that removing them makes the network vulnerable and may cause in cascading failure. A lot of works has been done to protect network nodes against FDI attacks so far, but considering varying node importance, which is a more realistic scenario, is not considered in any of them for determining the optimal protection strategy [8]. It is obvious that identification, monitoring and protection of critical nodes will improve the network reliability considerably. In case of having the full knowledge about the network structure, we may use social network metrics which are designed to measure the influence of different nodes in a large network [46]. The degree centrality, closeness centrality and betweenness centrality are three metrics that have been used widely in social network analysis [47]. In [48], the centrality metrics are used for power network analysis in order to determine the importance of nodes. Degree centrality is the simplest form of centrality in networks. Although this metric is simple, it is important for representing a node's role in a network [49]. It is defined as:

$$C_D(k) = \frac{deg(k)}{n-1} \tag{15}$$

where $deg(k)$ is the degree of node k, i.e. the number of edges connected to k. Another centrality metric, which is usually used in smart grids, is electrical degree centrality which is considered as power flow in adjacent node links:

$$C_D^E(k) = \frac{\sum P_{kt}}{n-1} \tag{16}$$

where P_{kt} is a power flow in connection line between node k and node t. Another social network centrality measure is closeness centrality, which is used as one of the most applied metrics. It is defined as the average of shortest paths between vertex k and all accessible vertices through it. It can be used to quantify the isolation of nodes in a network. In general, closeness centrality of vertex k in a network with n vertices is defined as:

$$C_c(k) = \frac{\sum\limits_{t \in V \setminus k} d(k,t)}{n-1} \tag{17}$$

where $d(k, t)$ is the length of the shortest path between k and t. Some references use the inverse of the shortest path length to compute the closeness centrality [49]. In smart grids we usually define electrical closeness centrality based on the inverse shortest electrical path as follows:

$$C_{cz}(k) = \frac{n-1}{\sum\limits_{t \in V \setminus k} d_Z(k,t)} \tag{18}$$

where $d_Z(k,t)$ is the length of a shortest electrical path (lower impedance path) between vertices k and t. It is also common to normalize (18) as follow:

$$C_c^E(k) = \frac{1}{\sum\limits_{t \in V \setminus k} d_i(k,t)} \tag{19}$$

where $d_i(k,t)$ is the length of the shortest electrical path (lower impedance path) between k and all reachable vertices accessible from it. The betweenness centrality is another popular metric that is widely used in social network analysis. It shows number of the shortest paths in a network that passes through a specific node [46]. The betweenness centrality of vertex k is denoted by $C_B(k)$ is computed using:

$$C_B(k) = \sum_{s=1}^{n} \sum_{t=1, t \neq s \neq k}^{n} \frac{\sigma_{st}(k)}{\sigma_{st}} \tag{20}$$

where σ_{st} is the number of shortest paths from s to t and $\sigma_{st}(k)$ is number of shortest paths from s to t that include k. For smart grids, betweenness centrality of vertex k is defined as follows:

$$C_B^E(k) = \sum_{s=1}^{n} \sum_{t=1, t \neq s \neq k}^{n} \frac{P_{st}(k)}{P_{st}} \tag{21}$$

where P_{st} is the maximum power flow in the shortest path from s to t and $P_{st}(k)$ is the maximum power flow input and output to vertex k in the shortest paths from s to t based on Kirchhoff law.

5.4 Proposed algorithms for finding protected meter measurements

In this section we propose three algorithms with different characteristics to determine the protected meter measurements based on the structure of smart grid graph. At first, we propose an efficient heuristic algorithm to find the minimum Steiner tree and the corresponding optimal protection set, which outperforms existing heuristic algorithms that are designed for this purpose. Then, to guarantee finding the optimal protection set, we propose an exact algorithm that finds the optimal solution. Finally, we propose an approximation algorithm, which guarantees to find a solution with a cost that is at most 1.39 times the cost of the optimal solution, and is much faster than the exact algorithm. In all approaches node weight and terminal set is determined based on betweenness centrality criteria.

5.4.1 Steiner tree heuristic (STH) algorithm

As we mentioned earlier, finding the optimal protection set of buses is equivalent to a minimum Steiner tree problem in the smart grid graph. The minimum Steiner tree problem is one of the most fundamental NP-hard problems that given a weighted undirected graph and a subset of terminal nodes, find a minimum-cost tree spanning the terminals. Since minimum Steiner tree problem is NP-hard, we propose a novel and efficient heuristic algorithm to determine the protected set of nodes.

In Algorithm 1, for n_{num} times, we consider a random ordering of terminals S_{term} and find the shortest path (P_{path}) between the root vertex and next vertex in S_{term}, then add the path in the resulted graph. In order to prevent using the edges of this path in the next iterations of the algorithm, we make the weight of all these edges equal to zero. In this algorithm, V_{res} and R_{res} are the vertices and edges of the resulted Steiner tree, respectively. Note that the algorithm's performance depends on the vertex ordering in S_{term}, thus considering more orderings (i.e. larger n_{num}) will improve the result of the algorithm and the probability of finding optimal solution will increase.

STATE GRID
STATE GRID ELECTRIC POWER RESEARCH INSTITUTE

Algorithm 1 STH

Input: graph $G(V, E)$, terminal set, and number of orderings n_{num}
Output: minimum cost Steiner tree
Initialization :
$R_{res} \longleftarrow E$
for each S_{term} **do**
  $r \longleftarrow S_{term}(0)$; $R_{term} \longleftarrow \emptyset$; $W' \longleftarrow w$
  for $i = 1$ to length(S_{term}) **do**
    $T_{term} \longleftarrow S_{term}(i)$;
    $P_{path} \longleftarrow dijkstra(G, W', r, T_{term})$;
    Add P_{path} to R_{term};
    for $e = uv \in P_{path}$ **do**
      $W'(e) \longleftarrow 0$;
      if $(\sum_{e \in R_{term}} w(e) < \sum_{e \in R_{res}} w(e))$ **then**
        $R_{res} \longleftarrow R_{term}$
      end if
    end for
  end for
end for
$V_{res} \longleftarrow f_{nodes}(R_{res})$
return V_{res}, R_{res}

5.4.2 *Approximation approach for finding protected set of meter measurements*

Although our simulation results show that STH performs very well in practice, but since it is a heuristic algorithm and does not provide any guarantee on the cost of the output solution, we also propose using an approximation algorithm for finding a minimum Steiner tree to extract the optimal protected set. Minimum Steiner tree problem is known to be APX-complete, even when the graph is complete and all edge costs are either 1 or 2. On the other hand, the problem admits a constant factor approximation algorithm.

In general, several different techniques have been proposed to design approximation algorithms for minimum Steiner tree problem, including a greedy approach that gives a 2-approximation algorithm, a local search approach which leads to an 11/6-approximation algorithm [50] and an iterative rounding approach to make a 1.39-approximation algorithm [51]. In this paper, we use the 1.39-approximation algorithm proposed in [51] which is the best known approximation algorithm for solving the minimum Steiner tree problem at the moment. This approximation algorithm is based on linear programming (LP) and iterative randomized rounding techniques. In this method we solve an LP-relaxation of the MSTP problem, possibly obtaining a non-integer solution. Then we iteratively round some LP variables and resolve the modified LP, until obtain an approximate solution to the original problem. The cost of the algorithm's output is at most $\ln(4) + \epsilon < 1.39$ times the cost of the optimal Steiner tree. Our approximation algorithm for finding the protected set of meter measurements is as follows:

Algorithm 2 Minimum protected set of meter measurements algorithm with 1.39-approximation

Input: graph $G(V, E)$ and terminal set
Output: minimum cost protected set of meter measurements
Steiner tree initialization:
for $i = 1$ to S_{term} **do**
  $C_k \longleftarrow$all components on at most k terminal, each generated using Dreyfus-Wagner algorithm;
  Solve the k-DCR;
  Steiner tree round condition:
  Select one component C_i, where $C_i = C$ with probability $x_C / \sum_{C' \in C_k} x_{C'}$;
  Contract terminals of C_i into its root
  if only one terminal remains **then**
    $i_{\max} \longleftarrow i$
    Steiner tree stop condition:
    Exit loop
  end if
end for
Steiner tree set solution:
return $\bigcup_{i=1}^{i_{\max}} C_i$

The core of Algorithm 2 is based on an LP-relaxation known as the directed-component cut relaxation (DCR). As the size of the set of all directed-components is exponential, the authors of [51] restrict it to a set of directed components that contain at most k terminals C_k. In Algorithm 2, Steiner tree initialization stage generates the k-DCR set C_k and initializes the LP. In [52], in order to generate C_k, the Dreyfus-Wagner algorithm [53] is used for finding the optimal Steiner tree on each subset. Then, Steiner tree round condition stage selects one random directed-component C with probability $x_C / \sum_{C' \in C_k} x_{C'}$, connects terminals of C into its root, updates the metric distances, and reinitializes C_k and the LP (using Steiner tree initialization). In the following Steiner tree stop condition stage checks whether the number of remaining terminals is equal to 1 or not. Finally Steiner tree set solution stage joints the sets of Steiner vertices from directed-components selected in each phase and determines the optimum protected set of buses.

5.4.3 *Exact algorithm for finding optimal protected set of buses*

In this section, we present an exact algorithm which guarantees to find the optimal solution for protected set of nodes in an edge weighted graph and outperforms existing exact algorithms such as SVE [33] for solving this problem.

The algorithm computes a minimum-weight tree that is equivalent to real smart grid network and contains all the terminal nodes (the buses that must be protected). In exact algorithm, we used Dijkstra-Steiner algorithm [54] as the core technique that achieves a significantly better practical performance comparing to similar exact algorithms via pruning and future costs, a generalization of a well-known concept to speed up shortest path computations [54]. The algorithm matches the best known worst-case run time and has a fast practical performance on smart grid. Let (G, c, K) define the space of a Steiner tree problem, where G is the graph of smart grid network, $c : E(G) \rightarrow R_{\geq 0}$ is the edge cost function, and K is a subset of terminal nodes, i.e. $K \in T$ and T is the terminal set. Our exact algorithm for finding minimum cost protected set of buses is as follows:

Algorithm 3 Exact minimum buses protected set algorithm based on Dijkstra Steiner algorithm

Input: connected undirected graph $G(V, E)$, costs $c : E(G) \rightarrow R_{>0}$ a terminal set $K \subseteq V(G)$, a root terminal $r_0 \in K$ and a valid lower bound $\mathcal{L} : V(G) \times 2^K \rightarrow R_{\geq 0}$
Output: minimum cost protected set buses
Initialization :
$l(v, I) := \infty$ for all $(v, I) \in V(G) \times 2^{K \backslash \{r_0\}}$
$l(s, \{s\}) := 0$ for all $s \in K \backslash \{r_0\}$
$l(v, \emptyset) := 0$ for all $v \in V(G)$
$bl(v, I) := \emptyset$ for all $(v, I) \in V(G) \times 2^{K \backslash \{r_0\}}$
$N := \{(s, \{s\}) | s \in K \backslash \{r_0\}\}$
$P := V(G) \times \{\emptyset\}$
while $(r_0, K \backslash \{r_0\}) \notin P$ **do**
  Choose $(v, I) \in N$ minimizing $l(v, I) + \mathcal{L}(v, K \backslash I)$
  $N := N \backslash \{(v, I)\}$
  $P := P \cup \{(v, I)\}$
  for all edges $e = \{v, w\}$ incident to v **do**
    if $l(v, I) + c(e) < l(w, I)$ and $(w, I) \notin P$ **then**
      $l(w, I) := l(v, I) + c(e)$
      $b(w, I) := \{(v, I)\}$
      $N := N \cup \{(w, I)\}$
    end if
  end for
  for all $\emptyset \notin J \subseteq (K \backslash \{r_0\}) \backslash I$ with $(v, J) \in P$ **do**
    if $l(v, I) + l(v, J) < l(v, I \cup J)$ and $(v, I \cup J) \notin P$ **then**
      $l(v, I \cup J) := l(v, I) + l(v, J)$
      $b(v, I \cup J) := \{(v, I), (v, J)\}$
      $N := N \cup \{(v, I \cup J)\}$
    end if
  end for
end while
return $T_{back}(r_0, K \backslash \{r_0\})$
Procedure $T_{back}(v, I)$
if $b(v, I) = \{(w, I)\}$ **then**
  return $\{\{v, w\}\} \cup T_{back}(w, I)$
else
  return $\bigcup_{(w, I') \in b(v, I)} T_{back}(w, I')$
end if

In Algorithm 3, T_{back} is backtrack function and each label (v, I) represents the best found Steiner tree for $\{v\} \cup I$ at the current iteration. $J \subseteq (K \backslash \{r_0\}) \backslash I$ and the algorithm checks whether the Steiner tree for (v, I) and (v, J) could be combined to make a tree for $(v, I \cup J)$ which leads to a better solution than the solution found in the previous iteration. $b(v, I) \subseteq V(G) \times 2^{K \backslash r_0}$ is the backtracking data which is used to construct the Steiner tree represented by each label $(v, I) \in V(G) \times 2^{R \backslash r_0}$. Finally, the valid lower bounds in (G, c, K) is a function $\mathcal{L} : V(G) \times 2^K \rightarrow R_0$ defined as [54]:

$$\begin{cases} \mathcal{L}(r_0, \{r_0\}) = 0 \\ \mathcal{L}(v, I) = \mathcal{L}(w, I') + smt((I \backslash I') \cup \{v, w\}) \end{cases} \tag{22}$$

where $v, w \in V(G)$ and $r_0 \subseteq I' \subseteq I \subseteq K$. Also $smt(X)$ is the cost of an optimal Steiner tree for the terminal set $X \subseteq V(G)$.

6 Evaluation

Extensive simulation is performed to examine the efficiency of our proposed approaches against the FDI attacks. We use Matpower6 package [55] to build IEEE test case graphs, MatlabBGL4.0 library [56] for graph analysis and PAAL library [52] to implement the algorithms for solving the weighted Steiner tree problem. The following results obtained on a 2 GHz Intel Core i7 machine with 8 GB RAM. To evaluate the performance of the proposed approaches, IEEE test cases that are observable with respect to measurement placements are considered. Table 1 shows the characteristics of these test cases.

6.1 Finding critical nodes using centrality metrics

Regarding budget challenges, to protect smart grid network against the FDI attack realistically, node importance and stage progress strategy should be considered. Here, we determine the critical nodes for IEEE 30-bus, IEEE 57-bus and IEEE 118-bus test cases. Since the stability of a smart grid depends on the communication between different sections control system, we use betweenness centrality as an efficient metric to determine critical nodes [57]. Table 2 shows top ten critical nodes in IEEE test case based on betweenness centrality metric. The critical nodes are

Table 1 Characteristics of IEEE test cases

Test case	Number			
	Line	Injection measurement	Flow measurement	Unmeasured line
30-bus	41	16	25	3
57-bus	80	30	50	2
118-bus	186	70	110	7

STATE GRID
STATE GRID ELECTRIC POWER RESEARCH INSTITUTE

Table 2 Top 10 critical nodes based on betweenness centrality metric

Node of 30-bus	C_B^E of 30-bus	Node of 57-bus	C_B^E of 57-bus	Node of 118-bus	C_B^E of 118-bus
2	0.610	1	0.620	12	0.700
1	0.550	2	0.480	7	0.440
6	0.310	17	0.400	11	0.430
4	0.300	3	0.340	2	0.350
3	0.290	15	0.241	3	0.080
5	0.180	16	0.150	6	0.063
7	0.054	4	0.140	14	0.035
8	0.050	6	0.061	117	0.033
9	0.041	14	0.055	13	0.028
10	0.040	5	0.050	4	0.210

considered as terminals for the Steiner tree and have larger weights in the node weighted Steiner tree problem.

6.2 Solving optimal protection problem

We simulated our three proposed algorithms along with SVE and TPH algorithms introduced in [33] as the reference methods for comparison. Figure 1 shows the time required to find the protected node set in three test cases, using each of these algorithms. One can observe that SVE algorithm which uses a brute force mechanism to find the optimal protected set takes a significant time for networks with more than 30 buses. For example, we estimate that the required time to find the optimal protection set in IEEE 118-bus using an ordinary processor is about a year.

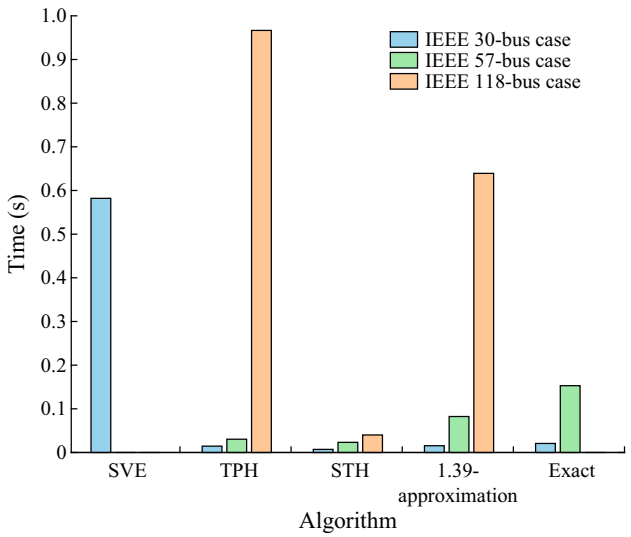

Fig. 1 Protected set time cost

Table 3 Cost comparison of Steiner trees found by different algorithms

Algorithm	30-bus	57-bus	118-bus
SVE [33]	70	–	–
TPH [33]	70	121	237
STH	70	119	235
1.39-approximation	70	154	240
Exact	70	113	–

Among heuristic algorithms, our proposed STH algorithm is faster than TPH algorithm proposed in [33], while the cost of its output solution is also less than TPH. Figure 1 also shows that our exact algorithm which is based on Dijkstra-Steiner algorithm cannot find the optimal solution for IEEE 118-bus test case in a reasonable amount of time, but unlike SVE it is capable to reach the optimal solution for IEEE 57-bus test case, and is much faster than SVE in finding the optimal solution for IEEE 30-bus test case. Finally, note that though the our 1.39-approximation algorithm does not perform very well, but since, unlike the heuristic algorithms, it can provide a guarantee on the quality of the final solution, and runs faster than the exact algorithm, it might be preferable in some cases. Table 3 shows the cost of Steiner tree found by different proposed and existing algorithms. The cost of a Steiner tree is the total weight of the Steiner tree edges. It is observed that our proposed heuristic algorithm has a cost that is less than other methods.

7 Conclusion

We analyzed the structure of the smart grid graph in order to formulate the FDI attack from a realistic point of view. To find the optimal set of nodes that must be protected against FDI attack in smart grids, we mapped the problem to a minimum Steiner tree problem in the smart grid graph. We used social network criteria to determine critical nodes as the terminal set in Steiner tree problem. We proposed three different algorithms to solve the problem of finding optimal protected set: a fast heuristic algorithm, a 1.39-approximation algorithm, and an efficient exact algorithm. Thorough simulation we showed that our heuristic and exact algorithms outperform reference algorithms proposed in [33]. In future we plan to focus on two main direction for completing the proposed initiatives: ① definition of the proposed methods for the AC estimation model; ② considering the real development methods based on prize collection Steiner tree problem-solving schemes for certain cost and prize. Using this, the network

protection designers can calculate the optimal plan for a real network, taking into account the actual circumstances of the budget constraints and resource schedule.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

[1] Wang W, Lu Z (2013) Cyber security in the smart grid: survey and challenges. Comput Electr Eng 57(5):1344–1371
[2] Weerathunga PE (2012) Security aspects of smart grid communication. The School of Graduate and Postdoctoral Studies Western University London, Ontario, Canada, 2012, 98 p
[3] Vukovi O (2014) Cyber-security in smart grid communication and control. Dissertation, Royal Institute of Technology
[4] Chen J, Liang G, Cai Z et al (2016) Impact analysis of false data injection attacks on power system static security assessment. J Mod Power Syst Clean Energy 4(3):496–505
[5] Liu Y, Ning P, Reiter MK (2009) False data injection attacks against state estimation in electric power grids. In: Proceedings of the 16th ACM conference on computer and communications security, Chicago, USA, 9–13 November 2009, pp 21–32
[6] Bobba RB, Rogers KM, Wang Q et al (2010) Detecting false data injection attacks on DC state estimation. In: Proceedings of preprints of the first workshop on secure control systems, Stockholm, Sweden, 12 April 2010
[7] Teixeira A, Dan G, Sandberg H et al (2011) A cyber security study of a SCADA energy management system: stealthy deception attacks on the state estimator. IFAC Proc Vol 44(1):271–277
[8] Deng R, Xiao G, Lu R et al (2017) False data injection on state estimation in power systems attacks, impacts, and defense: a survey. IEEE Trans Ind Inf 13(2):411–423
[9] Deka D, Member S, Baldick R et al (2014) Hidden attacks on power grid: optimal attack strategies and mitigation. https://arxiv.org/abs/1401.3274. Accessed 14 January 2014
[10] Liu Y, Ning P, Reiter MK (2011) False data injection attacks against state estimation in electric power grids. In: Proceedings of the 16th ACM conference on computer and communications security, Chicago, USA, 9–13 November 2009, pp 21–32
[11] Liang G, Zhao J, Luo F et al (2017) A review of false data injection attacks against modern power systems. IEEE Trans Smart Grid 8(4):1630–1638
[12] Kosut O, Jia L, Thomas RJ et al (2011) Malicious data attacks on the smart grid. IEEE Trans Smart Grid 2(4):645–658
[13] Rahman MA, Mohsenian-Rad H (2012) False data injection attacks with incomplete information against smart power grids. In: Proceedings of IEEE global telecommunications conference, Anaheim, USA, 3–7 December 2012, pp 3153–3158
[14] Feng Y, Foglietta C, Baiocco A et al (2013) Malicious false data injection in hierarchical electric power grid state estimation systems. Int Conf Future Energy Syst 13(91):183–192
[15] Liang J, Kosut O, Sankar L (2014) Cyber attacks on AC state estimation: unobservability and physical consequences. In: Proceedings of IEEE power and energy society general meeting, National Harbor, USA, 27–31 July 2014, pp 1–5
[16] Yang Q, Yang J, Yu W et al (2014) On false data-injection attacks against power system state estimation: modeling and countermeasures. IEEE Trans Parallel Distrib Syst 25(3):717–729
[17] Liu X, Li Z, Member S et al (2016) Optimal protection strategy against false data injection attacks in power systems. IEEE Trans Smart Grid 8(4):1802–1810
[18] Zhang H, Cheng P, Wu J et al (2014) Online deception attack against remote state estimation. IFAC Proc Vol 47(3):128–133
[19] Yang J, Yu R, Liu Y et al (2015) A two-stage attacking scheme for low-sparsity unobservable attacks in smart grid. In: Proceedings of IEEE international conference on communications (ICC), London, UK, 8–12 June 2015, pp 7210–7215
[20] Yu ZH, Chin WL (2015) Blind false data injection attack using PCA approximation method in smart grid. IEEE Trans Smart Grid 6(3):1219–1226
[21] Liu X, Bao Z, Lu D (2015) Modeling of local false data injection attacks with reduced network information. IEEE Trans Smart Grid 6(4):1686–1696
[22] Zhao J, Zhang G, Dong ZY (2016) Forecasting-aided imperfect false data injection attacks against power system nonlinear state estimation. IEEE Trans Smart Grid 7(1):6–8
[23] Konstantinou C, Maniatakos M (2016) A case study on implementing false data injection attacks against nonlinear state estimation. In: Proceedings of the 2nd ACM workshop on cyber-physical systems security and privacy, Vienna, Austria, 28 October 2016, pp 81–92
[24] Wang H, Ruan J, Wang G et al (2018) Deep learning based interval state estimation of AC smart grids against sparse cyber attacks. IEEE Trans Ind Inf. https://doi.org/10.1109/TII.2018.2804669
[25] Vukovic O, Dan G (2014) Security of fully distributed power system state estimation: detection and mitigation of data integrity attacks. IEEE J Sel Areas Commun 32(7):1500–1508
[26] Ozay M, Esnaola I, Vural FTY et al (2016) Machine learning methods for attack detection in the smart grid. IEEE Trans Neural Netw Learn Syst 27(8):1773–1786
[27] Nudell TR, Nabavi S, Chakrabortty A (2015) A real-time attack localization algorithm for large power system networks using graph-theoretic techniques. IEEE Trans Smart Grid 6(5):2551–2559
[28] Tang B, Yan J, Kay S et al (2016) Detection of false data injection attacks in smart grid under colored Gaussian noise. In: Proceedings of IEEE conference on communications and network security (CNS), Philadelphia, USA, 17–19 October 2016, pp 172–179
[29] Bi S, Zhang YJ (2011) Defending mechanisms against false-data injection attacks in the power system state estimation. In: Proceedings of IEEE GLOBECOM Workshops (GC Wkshps), Houston, USA, 5–9 December 2011, pp 1162–1167
[30] Kim TT, Poor HV (2011) Strategic protection against data injection attacks on power grids. IEEE Trans Smart Grid 2(2):326–333
[31] Liu T, Gu Y, Wang D et al (2013) A novel method to detect bad data detection injection attack in smart grid. In: Proceedings of 2013 IEEE conference on computer communications workshops, Turin, Italy, 14–19 April 2013, pp 3423–3428
[32] Anwar A, Mahmood AN, Tari Z (2014) Identification of vulnerable node clusters against false data injection attack in an AMI based smart grid. Inf Syst 53:201–212
[33] Bi S, Zhang YJ (2014) Graphical methods for defense against false-data injection attacks on power system state estimation. IEEE Trans Smart Grid 5(3):1216–1227
[34] Bi S, Zhang YJ, Member S (2014) Using covert topological information for defense against malicious attacks on DC state estimation. IEEE J Sel Areas Commun 32(7):1471–1485

[35] Hao J, Kang E, Sun J et al (2018) An adaptive Markov strategy for defending smart grid false data injection from malicious attackers. IEEE Trans Smart Grid 9(4):2398–2408
[36] Abur A, Exposito AG (2004) Power system state estimation: theory and implementation. Marcel Dekker, New York
[37] Kim TT, Poor HV (2011) Strategic protection against data injection attacks on power grids. IEEE Trans Smart Grid 2(2):326–333
[38] Krumpholz G, Clements K, Davis P (1980) Power system observability: a practical algorithm using network topology. IEEE Trans Power Appar Syst 99(4):1534–1542
[39] Bargiela A, Irving M, Sterling M (1986) Observability determination in power system state estimation using a network flow technique. IEEE Trans Power Syst 1(2):108–112
[40] Hauptmann M, Karpinski M (2013) A compendium on Steiner tree problems. Dissertation, University of Bonn
[41] Bondy JA, Murty USR (1982) Graph theory with applications, 5th edn. Elsevier, Amsterdam
[42] Demaine ED, Hajiaghayi M, Klein PN (2009) Node-weighted Steiner tree and group Steiner tree in planar graphs. In: Proceedings of international colloquium on automata, languages, and programming, Rhodes, Greece, 5–12 July 2009, pp 328–340
[43] Boyer JM, Myrvold WJ (2004) On the cutting edge: simplified o(n) planarity by edge addition. J Graph Algorithms Appl 8(2):241–273
[44] Faria L, de Figueiredo CMH, Gravier S et al (2006) On maximum planar induced subgraphs. Discrete Appl Math 154(13):1774–1782
[45] Calinescu G (1998) A better approximation algorithm for finding planar subgraphs. J Algorithms 27(2):269–302
[46] Kivimaki I, Lebichot B, Saramaki J et al (2016) Two betweenness centrality measures based on randomized shortest paths. Sci Rep. https://doi.org/10.1038/srep19668
[47] Opsahl T, Agneessens F, Skvoretz J (2010) Node centrality in weighted networks: generalizing degree and shortest paths. Social Netw 32(3):245–251
[48] Ban D (2011) A flow-based centrality measure through resistance distances in smart-grid networks. In: Proceedings of IEEE global telecommunications conference, Kathmandu, Nepal, 5–9 December 2011, pp 1–5
[49] Hanneman RA, Riddle M (2005) Introduction to social network methods. Dissertation, University of California Riverside
[50] Zelikovsky AZ (1993) An 11/6-approximation algorithm for the network Steiner problem. Algorithmica 9(5):463–470
[51] Byrka J (2013) Steiner tree approximation via iterative randomized rounding. J ACM 60(1):883–892
[52] Wygocki P (2016) PAAL. http://paal.mimuw.edu.pl/. Accessed 5 Oct 2016
[53] Dreyfus SE, Wagner RA (1971) The steiner problem in graphs. Networks 1(3):195–207
[54] Hougardy S, Silvanus J, Vygen J (2017) Dijkstra meets Steiner: a fast exact goal-oriented Steiner tree algorithm. Math Program Comput 9(2):135–202
[55] Zimmerman RD, Murillo-Sanchez CE, Thomas RJ (2011) Matpower: steady-state operations, planning, and analysis tools for power systems research and education. IEEE Trans Power Syst 26(1):12–19
[56] Gleich D (2008) MatlabBGL: A Matlab graph library. https://www.cs.purdue.edu/homes/dgleich/packages/matlab_bgl/. Accessed 22 Oct 2008
[57] Wang Z, Scaglione A, Thomas RJ (2010) Electrical centrality measures for electric power grid vulnerability analysis. In: Proceedings of 49th IEEE conference on decision and control (CDC), Atlanta, USA, 15–17 December 2010, pp 5792–5797

Mohammad Hasan ANSARI received his Bachelor degree in electrical engineering, from KNTU and Master degree of power system engineering and communication system engineering from Iran University of Science and Technology (IUST), Tehran, Iran, in 2007, 2010 and 2012, respectively. He is currently an Ph.D. student in electrical engineering at IUST. His main research interests include network security, graph theory, artificial intelligence and big data analysis.

Vahid Tabataba VAKILI received his Bachelor degree in electrical engineering, from Sharif University of Technology, Tehran, Iran and Master degree in communication systems from Bradford University, England in 1970 and 1973, respectively. He received the Ph.D. degree in communication systems, Bradford University, England in 1978. He is currently a full professor of electrical and computer engineering at Iran University of Science and Technology (IUST).

Behnam BAHRAK received his Bachelor and Master degrees both in electrical engineering, from Sharif University of Technology, Tehran, Iran, in 2006 and 2008, respectively. He received the Ph.D. degree from the Bradley Department of Electrical and Computer Engineering at Virginia Tech in 2013. He is currently an assistant professor of Electrical and Computer Engineering at University of Tehran.

Parmiss TAVASSOLI received her Bachelor degree in information technology from University of Tehran, Iran, in 2017. Her main research interests include network science and artificial intelligence.