Blockchain-based Distributed Key Management Approach Tailored for Smart Grid

mohamed

mostafafoda

Most of the existing key management protocols adopt a centralized architecture, which depends on a single server to distribute keys and update them. In this paper, Leveraging the blockchains that allow us to have a distributed peer-to-peer network where non-trusting entities can interact with each other without the need to a trusted intermediary, in a verifiable manner, we developed an efficient distributed multicast key management scheme. In the proposed scheme, group members can manage the group communication in a contributory way. Furthermore, we apply our proposed approach to new critical infrastructure paradigm such as smart grid microgrids, which can be considered as a small-scale system where no fixed trusted third party to control the key management process. The security and performance analysis of our proposed approach demonstrates its effectiveness and scalability by taking into account the computation and communication costs which are main concerns in the big data era.

# Introduction

The Smart Grid (SG), also referred to as the intelligent grid, the future grid, or the so-called intelligrid, is the convergence of information technology, communications, and power system engineering to provide a more robust, efficient and flexible electrical power system (1). It enables two-way flows of both electricity and real-time energy related information that are issued by smart devices and smart meters. Some of the expected benefits of SG are preventing blackouts and providing greater availability of electricity to homes at a lower cost, opening the door to new grid paradigms such as microgrids and Plug-in Hybrid Electric Vehicles (PHEVs) (2), transforming the power grid into a bi-directional power system in which customers can receive as well as supply power from/to the grid. Thus, the SG paradigm dramatically transforms the state-of-the-art energy grid into a distributed power generation system which reduces the cost of the electricity generation and distribution (3).

The previously mentioned benefits of deploying SG arise through the communication between smart meters and the utility via Advanced Metering Infrastructure (AMI) system (4). AMI is a key enabler of the SG system, role of which is to collect the metering data and all information from loads and consumers in order to efficiently perform the required operations. Moreover, it is responsible for maintaining Demand Response (DR) tasks, which are an essential part of the SG system, to reduce the consumption of electricity or shift it from on-peak to off-peak periods depending on choices of the end users. A typical AMI involves Smart Meters (SMs), Home Area Networks (HANs), wide area network infrastructure, and Meter Data Management Systems (MDMSs).

In order to secure communication between SG-entities, data confidentiality, integrity, and availability should be guaranteed. Confidentiality for SG system indicates that the data exchanged should be protected from unauthorized access of other people or systems. Integrity indicates that the data transmitted (e.g., DR commands and so forth) should be authentic, with no modifications or additions. Meanwhile, availability deals with the data that should be accessible at any time due to the high availability of electrical power. To meet these security requirements in the smart grid, several organizations (e.g., Cyber Security Working Group (CSWG) led by National Institute of Standards and Technology (NIST) (5)) are interested in the deployment of smart grid security and mitigating the cyber attacks affecting the communications within the whole system such as man-in-the-middle (MITM) attack, Denial of Service (DoS) attack, sniffing on smart meters, impersonation attack, spoofing and replay attacks (missing citation); 7; (missing citation), and so forth.

To pursue both confidentiality and integrity in the SG system, the key management remains a critical yet open research issue (9). The key management of the SG system needs to effectively deal with the process of key generation, establishment, storage, revocation, and update (10). Most of existing schemes depends on a traditional client-server model for the key management process in the SG. However, such client-server model may fail in the SG for many reasons. First, SG is made a large number of smart meters, whose number may be at magnitude of millions. As a result, depending on a client server model may impose huge overhead on the server-end. Second, securing multicast communications needs the group key to be refreshed and redistributed (rekeying) securely whenever a group member changes to achieve forward and backward secrecy. This approach poses challenges of rekeying efficiency, especially in a heterogeneous complex network such as the SG. Third, future SG will enable distinct network structures such as microgrids which can be defined as a small-scale SG system that may be isolated from the main grid and can supply the SG system with electricity. However, existing key management schemes do not consider the possibility of isolation of the microgrid.

Recently, blockchain has gained notable attention from various fields (11). The blockchain is a synchronised and distributed *ledger* which stores a list of blocks. No more central managers are required to maintain the blockchain structure, and instead, the public ledger is secured by all the network participants. Each peer acts as a node of the network and can participate in calculating the solution to a hash-based mathematical problem assuring the integrity of transactions. Each transaction record is encapsulated as a block and added to the existing block chains. All information is then updated synchronously to the entire network so that each peer keeps a record of the same ledger. Moreover, the distributed structure of blockchain network performs better robustness under the single point of failure.

In this paper, we contribute by taking into account the shortcoming of the existing key management schemes used in the SG systems leveraging blockchain technology. The main contributions of our work are as follows. (1) We study the smart grid architecture and propose an efficient end to end key agreement protocol, which is based on the Diffie-Hellman (DH) key establishment with less computation overhead. (2) We propose a blockchain-based self-organized or distributed multicast key management for secure communication through the SG. A one-purpose blockchain is presented to allow group members in a distributed way to securely manage the group rekeying operations whenever a meter joins or leaves the group. Then, using Diffie-Hellman (DH) cryptography with the symmetric cryptosystem, we show how the group members can achieve forward and backward secrecy. (3) We conduct security and performance analysis for our proposed schemes and compare them with existing schemes to prove that our proposal is efficient and scalable in the context of key management.



 Illustration for the system model under consideration. Three Different NANs exist. (1) A group of industrial users, (2) A group of home users (3) And, a microgrid.

The rest of contents of this paper is organized as follows. In Section ???, a review of the previous efforts on key management in smart grids is presented. In Section ???, we describe our considered system model of SG and the microgrid with the relevant security considerations. In Section ???, we first propose a key agreement protocol. Then, we introduce a distributed multicast key management. Then, in Section ???, we provide a security and performance analysis of our proposed scheme. Finally, the paper is concluded in Section ???.

# Related Work

Many key management schemes have been proposed for securing communications through the different elements of the smart grid. In this section, we review these efforts and critically analyze several relevant ones.

Fouda et al. (missing citation) proposed a light-weight authentication scheme based on the DH (13) protocol. This scheme provides a solution for authenticated key agreement between the Home Area Network (HAN) gateway and the Building Area Network (BAN) gateway. The target behind this scheme is to reduce the computation cost on the resource-constrained HAN gateways. However, the scheme still needs further computation efforts.

Nicanfar et al. (14) proposed a key management for unicast and multicast communication in the SG system based on the concept of ID-based public/private key pair model (15). Although it accomplishes efficiency from the computation cost point of view, it is vulnerable to impersonation attacks (16).

Kamto et al. (17) developed a light-weight key management protocol based on the DH key agreement (13). Kamto scheme achieves a low computation overhead, especially on the SMs. However, it suffers from inefficient group key management and the MITM attack in which an adversary is able to easily share a key with both the SM and gateway as illustrated in (16).

Xia and Wang (missing citation) proposed a key distribution scheme with a low computation overhead. However, it has a high communication overhead due to the high number of exchanged messages. It also does not support a means for multicast key management. Later, it has been proven that this scheme is vulnerable to MITM attack and a common type of DoS attack (i.e., desynchronization attack) (19).

Liu et al. (20) proposed a key management method to provide secure unicast, multicast, and broadcast communication based on the key graph technique (21). Their scheme depends on generating the session keys based on previously stored keys and additional counters. The authors suggests to use simple cryptographic algorithms for key generation and refreshing to overcome the computation and storage constraints of smart meters. However, it also suffers from desynchronization attacks. In other words, once an attacker blocks the path of data, the counters on both the SM and the management side will be different. As a consequence, both parties are unable to agree on the session key. Also, the scheme proposed in (20) suffers from inefficient group key update (10).

Inspired from the work conducted by Liu et al., another scalable key management scheme was envisioned by Wan et al. in (10). First, Wan et al. proposed an end-to-end key establishment protocol between the SMs and the head end based on the bilinear pairing (22). However, the end-to-end key agreement method suffers from a substantially high computation overhead, especially on the resource constrained SMs due to the pairing operation. Then, Wan et al. used the session keys created by the key establishment protocol to provide a centralized multicast key management adopted from the One-way Function Tree (OFT) approach for key management (23). Although that scheme provides a less computation overhead on the rekeying process, the SM has a significantly high storage overhead to maintain the binary tree in a balanced manner (24).

In (missing citation), we introduced the idea of using self-organized or distributed multicast key management in which no central server is used to manage the group communication in the microgrids. However, the scheme suffers from false data injection attack since there is no trusted third party to secure the rekying process. In this paper, we provide a full enhanced framework to achieve secure key management in the SG. We first propose reliable distributed multicast comprising a light-weight end-to-end key establishment scheme, which is customized to the specific needs of the SG. Also, leaveraging blockchain to provide a synchronised and distributed ledger which stores the required cryptographic materials to reliably distribute or update group keys among all the communicating participants.

# System Model and Problem Formulation

In this section, we provide a review of the SG system structure followed by pointing out the problem statement to identify the basic requirement to secure communication within the microgrid.

## SG System Structure

As shown in Fig ???, the main components of the smart grid are the smart meters, the Wide Area Network (WAN), the utility management system and the microgrids.

**Smart Meters (SMs):** SMs are digital devices, which are responsible for many functions such as measuring not only electricity usage but also maximum demand, current, voltage, dynamic pricing policies, remote turn on/off, and so forth. Moreover, they can allow Demand Response (DR) programs (3) to increase the efficiency of SG (26). In our scheme, we assume that meters are able to communicate with each other through wired or wireless communication channels.

**Wide Area Network (WAN):** It allows the two-way communication between different SG elements (i.e., consumers) and the utility. Various communication technologies can be deployed such as optical fiber, power line carrier, copper, radio frequency cellular networks or Internet Protocol (IP)-based networks (27).

**Utility Management System:** It includes the “brain” of the smart grid system, particularly the Meter Data Management System (MDMS) which has a database of the whole system with further analytical tools (10). Furthermore, it includes an AMI Head End (AHE) to communicate with SG end users (i.e., customers or consumers).

**Neighborhood area networks (NAN):** The NAN is formed by a large number of smart meters (SM) in a certain neighborhood. NANs can be divided into different groups. One possible way is according to their electricity consumption type, or to the coverage area. In addtion, By exploiting the users having the ability to generate electricity locally, recently a new concept called the microgrid is gaining significant research attention. A Microgrid (28) is generally defined as a low voltage network with distributed generation sources, together with local storage devices and controllable loads (e.g. water heaters and air conditioning). The microgrid may be either connected to the main power grid or in an isolated (i.e., “island”) mode. In the normal situation, users in the microgrid can use Distributed Energy Resources such as wind turbines, solar panels, and fuel cells to generate low voltage electricity and provide the power grid with electricity. On the other hand, it may operate in an “island” mode whereby it can still generate electricity but without exchanging electricity with the main grid (9). Communication through the NAN can be faciliated through the Fourth Generation (4G) cellular technologies such as Long Term Evolution (LTE), WiMAX, and so forth (29) as shown in Fig. ???. In addition, the notations and abbreviations with their definitions through our paper are listed in Table. ???.

## Problem Formulation

Securing the microgrid is a key element to increase the reliability and efficiency of SG. According to the report in (30), several vulnerabilities may exist in the microgrid system including attacks like MITM, DoS, eavesdropping, masquerading, message replay, message modification, traffic analysis, and unauthorized access. In order to combat such malicious attacks and, at the same time, to pursue data confidentiality and integrity in the microgrid system, a key management is required to secure communications through it.

Most of the existing related key management schemes are centralized which rely on a single entity to manage keys. However, key management techniques can take other forms than being centralized (31). In this paper, we extend our previous work (missing citation) which introduces the idea of using distributed key management protocol in which no central server is required to secure the communication. Therefore, in the following section, we first propose a key agreement protocol through which a shared key is shared between the AHE and SMs in different NANs. Secondly, we propose a blockchain-based distributed multicast key management tailored for securing communications within the NANs in the SG.



 Our proposed end-to-end key establishment scheme.



 Our proposed end-to-end key establishment scheme.

System Notations.

|  |  |
| --- | --- |
| Symbol | Description |
| $Z\_{p}^{\*}$ | A multiplicative group. |
| $⊕$ | An exclusive OR operation. |
| $n$ | Number of Smart Meters (SMs) in a multicast group. |
| $H1(.)$ | A secure hash function. |
| $K\_{G}$ | The group key. |
| $K\_{NG}$ | The new group key after a member change. |
| $∥$ | A concatenation operator. |
| $f(g)$ | A one way function. |
| $σ\_{d}(M)$ | The signature of $M$ using a private key $d$. |
| Kgen$(1^{b})$ | A secure $b$-bit key generation algorithm. |
| $Ti$ | The recorded time instance of sending the message. |
| $mi$ | The message. |
| PKT | A Public Key Table contains the parent binary code |
|  | associated with the member public key. |
| $HMAC\textsubscript{$K\_i$}$ | A hash-based Message Authentication Code (MAC) |
|  | generation algorithm by using the shared session key $K\_{i}$. |



 A group of eight SMs that consists a multicast group which is organized in a binary tree. Each node has two codes, i.e., a decimal one and a binary one. Also, the intermediate codes are shown in green boxes. Moreover, D-H keys are assigned to the leaf nodes.

# Our Proposed Scheme

In this section, we present our key management scheme tailored for securing communications in the SG. Our key management consists of two phases. First, an end-to-end key agreement is presented to secure unicast communication between the AMI Head-End (AHE) and smart meters in the NAN. Second, a blockchain based distributed key multicast management is envisioned to secure group communications within the NAN.

## End-to-End Key Establishment

We adopt our key establishment from DH key agreement (13). Let $G=⟨g⟩$ be a group of large prime order $p$. The following preliminaries and assumptions should be considered.

* The discrete logarithm assumptions should hold, i.e., given $g^{xi}$, it is computationally hard to calculate $xi$ given $g^{xi}$. Also, the assumptions of Computational Diffie-Hellman (CDH) hold, i.e., given $g^{xa}$ and $g^{xb}$ where $xa$, $xb$ $\in $ $Z\_{p}^{\*}$, it should be computationally hard to calculate $g^{xaxb}$.
* AMI Head-End should have a private/public key pair ($e,d$) which is used for the signing RSA algorithm (32).
* Each SM should be preloaded by the parameters ($Z\_{p}^{\*}$, $g$, $p$, $e$).

**The proposed end-to-end key agreement protocol:**
The AHE with identity $(ID\_{A})$ and smart meter $SM\_{i}$ with identity $(ID\_{i})$ in a NAN run the following authenticated key agreement with only three messages.

**Step 1.** First, the meter $SM\_{i}$ chooses a random number, $xi$ $\in $ $Z\_{p}^{\*}$ to compute its public key $PK\_{i}=g^{xi}$ then it calculates $h\_{i}=H1(0,ID\_{i}||PK\_{i})$. Finally it sends $M\_{1}=(h\_{i}||IDi||PK\_{i}$) to the AHE.

**Step 2.** Upon receiving the initiating message from $SM\_{i}$, AHE does the following.

1. Verifies the received $h\_{i}$.
2. Generates a random number $xj$ $\in $ $Z\_{p}^{\*}$ and computes $PK\_{G}=g^{xj}$.
3. Computes the shared key $K\_{AHE\rightarrow SM\_{i}}=(PK\_{i})^{xj}$.
4. Computes $C\_{AHE}=H1(1,K\_{AHE\rightarrow SM\_{i}})$ and calculates $sign\_{C\_{AHE}}$ which is the signature of $C\_{AHE}$ using its private key $(d)$.
5. Sends $M\_{2}=(PK\_{AHE}||ID\_{A}||C\_{AHE}||sign\_{C\_{AHE}})$ to $SM\_{i}$.

**Step 3.** The SM verifies the signature of the AHE using its public key $(e)$. Then, if the verification is “ok”, it calculates $K\_{SM\_{i}\rightarrow AHE}=(PK\_{AHE})^{xi}$. After that, it computes $C\_{SM\_{i}}=H1(1,K\_{SM\_{i}\rightarrow AHE})$. It checks if $C\_{SM\_{i}}$ is equal to the received $C\_{AHE}$, then it can authenticate the AHE. At the end, it sets $K=H1(ID\_{A}||ID\_{i}||K\_{SM\_{i}\rightarrow AHE})$ as the shared session key, and then computes $M\_{3}=H1(2,K\_{SM\_{i}\rightarrow AHE})$ and sends it to AHE.

**Step 4.** Upon receiving the message $M\_{3}$, AHE computes $M′\_{3}=H1(2,K\_{AHE\rightarrow SM\_{i}})$. If $M\_{3}$=$M′\_{3}$, it authenticates the SM and sets $K=H1(ID\_{A}||ID\_{i}||K\_{AHE\rightarrow SM\_{i}})$ as the shared session key.

## Blockchain-based Multicast Key Management

For efficient management of multicast keys in the smart grid, the end-to-end key establishment obtained in the above protocol are integrated with the key tree technique as well as the blockchain technology.

As stated earlier, our scheme has a distributed feature. We adopt the one proposed in (33) to be employed with a group of users. The main feature of this scheme is its low communication and computation overheads which are important for the deployment of the SMs. The scheme is based on the one-way function tree technique (23) in which meters are organized in a binary tree as depicted in Fig. ??? where each node has two codes as follows.

1. A binary code is used to discover the position of the member. A decimal code is used to compute the intermediate nodes’ key(s). Any meter can calculate all decimal codes which it belongs to by removing the last digit from the right. For instance, if $SM\_{1}$ has a decimal code ($045$) as its decimal parent code, the meter can remove the last digit to get ($04$) as the decimal code for ($00$) node.
2. Symmetric intermediate keys are assigned to the intermediate nodes which can be used to secure communication within the multicast group when updating the group key.
The intermediate node key is calculated by the following formula.

$$K\_{Intermediate−node}=f(K\_{G}⊕Code\_{Intermediate−node})$$

1. where the decimal code of a binary node can be calculated by concatenating the decimal code of its parent code with a random digit as illustrated by following expression,

$$Code\_{Child−node}=(Code\_{Parent−node}∥Random−digit).$$

* As an example of the above design, consider that in Fig.???, $SM\_{1}$ is assigned to a parent code $(000)$ associated with $045$ as a decimal code. It can determine its intermediate node keys as follows.
* $K\_{1,2}=f(K\_{G}⊕045)$ and $K\_{1,4}=f(K\_{G}⊕04)$
* Where: $K\_{1,2}$ can be used to encrypt group key communication between $SM\_{1}$ and $SM\_{2}$. Also, $K\_{1,4}$ can be used to encrypt data between $SM\_{1}$, $M\_{2}$, $M\_{3}$ and $SM\_{4}$.
1. A decimal code is used to compute the intermediate nodes’ key(s) which are used to encrypt data to specific meters in the group. Any meter could calculate all decimal codes which it belongs to by removing the last digit from the right. For instance, if $SM\_{1}$ has a decimal code ($045$) as its decimal parent code, the meter can remove the last digit to get ($04$) as the decimal code for ($00$) node.
2. Each meter is equipped with specific software to support the generation of a public key and private key.
3. The Public Key Table (PKT) plays a critical rule in secure group communication. It is used to store the public key of SMs associated with their parent binary codes. Meters can use PKT is share a secret session key between group members. The contents within the PKT is updated once a meter joins or leaves a multi-cast group. To ensure integrity of keys within PKT, the contents of the PKT is stored with in a blockchain.

### Proposed Blockchain

To guarantee the integrity of the PKT, and since there is no central server to manage the key management process, we propose the use of blockchain technology to allow smart meters in a decentralized way to keep records of the PKT so that once any meter joins or leaves the group, the updated PKT is stored within the blockchain. Any change to the PKT data is eventually stored in a ledger formed by chronologically connected blocks which exist in each meter’s memory. The following procedures are necessary to assure data accuracy before storing any data in meters memory: data broadcast and verification via a voting mechanism. Then, the updated PKT is added in a block. Finally, the mining process and verification of the mining result via the voting mechanism are done to ensure the integrity of the PKT in all meters. In the following, we show in detail the one purpose proposed blockchain, which consists of the following steps:

1. *Data Broadcast and Verification* :
In this phase, once a smart meter joins or leaves the group, the PKT table should be updated and this change should be stored on the blockchain. The joined smart meter broadcasted its newly public key signed with its private key to all nodes. All meter nodes which receive the broadcast information need to verify the received message signature. To achieve consensus on the received message before a verified data are packed as a block to be connected to the previous ledger, we adopt an address-based distributed voting mechanism (34). The basic idea is that each meter has one chance to verify the authenticity of the received message. The voting scheme works as follows, assume $K$ meter in a NAN where each one votes on the verification result for the received message. The data is accepted only when the following condition matches:

$$\frac{N}{K}>T$$

* Where: $N$ represents the number of most votes and $T$ is a threshold whose value should be greater that 50% so that nodes can make sure that the voting result on certain message has been agreed on majority of nodes.
1. *Mining and Generation of Blocks* The stored data in the blockchain is stored through chain of linked blocks. In the blockchain network, each block has the following contents: block number, data content, timestamp, previous block hash, block hash result, and nonce solution. The meanings of the contents are shown in Table. ???. As in Bitcoin (35), SHA-256 is used to mine the block obtain a hash output with certain criterion. If certain PKT is updated and all nodes has verified that change. Then, based on the data content on the $(i−1)$-th block, the current timestamp, some nodes can solve a puzzle to find a certain nonce value.

Block contents.

|  |  |
| --- | --- |
| Item | Description |
| Block number | Current block sequence number. |
| Data content | Current block PKT encapsulated data. |
| Timestamp | The time when the last verified PKT data is |
|  | added into the current block |
| Previous block hash | The hash value of the previous block. |
| Block hash | A hash value of the current block. |
| Nonce solution | Puzzle solution for the current block. |

Let $C$ is the overall block content as follows:

$$C=B||D||t\_{s}||HP||nonce$$

where: $B$ is block number, $D$ is the updated PKT data, $t\_{s}$, $HP$ the hash of previous block and *nonce* which is a random number.

SHA-256 is applied to the previous block content so the puzzle problem could be defined as follows:

$$Digest=SHA256(C)$$

The solution of this puzzle problem is to find a certain *nonce* value such that $Digest$ is less than a certain target value, $T$ as:

$$Digest\leq T$$

To solve the puzzle, there is no way other than to try as many as nonce values till the required $T$ is obtained. The difficulty of solving the puzzle depends on the value of $T$, the lower the target value; the higher the difficulty of finding solution for the puzzle.

Any node in the network can work as a miner to find the puzzle’s solution. Up on finding the nonce, it will be broadcasted to the remaining nodes so they can check if (???) applies. Then, again, the address-based voting scheme in (???) is used to vote on the puzzle’s solution. Once, the two conditions are satisfied, the block content can be added to the ledger.



 Distributed ledger structure.

### Smart Meter Joining

When a new SM asks to join the group, the joining protocol is employed to update the group key and the intermediate node codes. We discuss the SM joining protocol from Fig. ???. First, when the new meter $SM\_{8}$ asking to join the group selects a random number $x8\in Z\_{p}^{\*}$ and calculates $g^{x8}$. Then, he/she computes a signature on $g^{x8}$ using its private key $σ(g^{x8})$. Afterthat, he/she broadcasts $g^{x8}||σ(g^{x8})$ to other meters. All meter nodes which receive the broadcast information need to verify the received message signature. Then, The result is tested by the the address-based distributed voting mechanism in Eq. ???, which ensures if there are enough nodes agreeing on the received message. Then, before adding it to the PKT in the blockchain. Some nodes can operate as miners by attempting to solve the puzzle problem in Eq. ???. Once the first meter gets the nonce value, it broadcasts the result nonce to other nodes so that they can verify whether the solution is valid by checking whether it satisfies the condition in ??? or not. The address-based distributed voting mechanism is utilized again to vote on the verification result before the current PKT block to be allowed to cryptographically connect to the previous ledger.

Second, the meter, $SM\_{7}$, which has no sibling, updates its position from (01) to (011), and then computes a decimal code associated with its parent code (011) as in (eq. ???). After that, it uses the PKT to share a shared session key ($K\_{s}$) with $SM\_{8}$ as in ???. Then, it updates the group key by applying a one-way function to the previous group key as in Eq. ??? and sends them to $SM\_{8}$ encrypted by the shared session key. Also, we employ a hash-based Message Authentication Code (MAC) generation algorithm by using $K\_{s}$ on the sent message to pursue integrity as in eq. (4).

$$code\_{node\_{(011)}}=(08∥9)=089$$

$$SM\_{7}⇒K\_{s}=H((g^{x8})^{x7}),SM\_{8}⇒K\_{s}=H((g^{x7})^{x8})$$

$$K\_{N}G=f(K\_{G})$$

$$\begin{matrix}SM\_{7}\rightarrow SM\_{8}:\{K\_{NG},089,HMAC\_{K\_{s}}\}\_{\{encr\}K\_{s}}\end{matrix}$$

The other remaining group members update their group key by just applying the one-way function as in eq. (3). Also, SMs in the affected path renew the intermediate node codes as illustrated in eqs. (5) and (6).

$$SM\_{7},SM\_{8}\leftrightarrow K\_{7,8}=f(K\_{NG}⊕089)$$

$$\begin{matrix}SM\_{5},…,SM\_{8}\leftrightarrow K\_{5,8}=f(K\_{NG}⊕08)\end{matrix}$$



 Smart meter joining and eviction.



 PKT is updated when $SM\textsubscript{8}$ joins.

### Smart Meter Eviction

When a smart meter requests to leave the group (e.g., when it is defected/encounters malfunctions, or requires maintenance), it needs to ensure the forward secrecy. In this vein, the group key needs to be renewed using the following SM eviction protocol. If $SM\_{8}$ is to leave the group, its sibling $(SM\_{7})$ upgrades its position in the tree to $(01)$ and updates the PKT by deleting the departing member as shown in Fig. ???. Also, it informs the other SMs regarding the change by sending a broadcast message. Note that: the same procedures regarding updating the PKT in the blockchain is followed as in Sec. ???. This is mandatory to ensure the integrity of the PKT in a distributed network. Then, $SM\_{7}$ computes a new group key ($K\_{NG}$) by employing the Kgen$(1^{b})$ algorithm and uses his PKT to share a key with any SM in each branch of the tree. Then, it sends the new group key to them as illustrated in eqs. (7) and (8).

$$\begin{matrix}SM\_{7}\rightarrow SM\_{1}:\{K\_{NG},HMAC\_{K\_{1}}\}\_{\{encr\}K\_{1}},\end{matrix}$$

$$\begin{matrix}SM\_{7}\rightarrow SM\_{5}:\{K\_{NG},HMAC\_{K\_{2}}\}\_{\{encr\}K\_{2}},\end{matrix}$$

where $K\_{1}=H((g^{x1})^{x7})$ and $K\_{2}=H((g^{x5})^{x7})$ are shared secret keys between ($SM\_{7}$ and $SM\_{1}$) and ($SM\_{7}$ and $SM\_{5}$) respectively.

On the other hand, $SM\_{1}$ and $SM\_{5}$ can make use of the intermediate keys to send the new group key encrypted to other SMs in the group as in eqs. (9) and (10).

$$\begin{matrix}SM\_{1}\rightarrow SM\_{2},…,SM\_{4}:\{K\_{NG},HMAC\_{K\_{1,4}}\}\_{\{encr\}K\_{1,4}},\end{matrix}$$

$$\begin{matrix}SM\_{5}\rightarrow SM\_{6}:\{K\_{NG},HMAC\_{K\_{5,6}}\}\_{\{encr\}K\_{5,6}}.\end{matrix}$$

In order to assure the forward secrecy feature, the SMs in the affected branch should update their intermediate node codes as in eq. (11).

$$SM\_{5},SM\_{6},SM\_{8}\leftrightarrow K\_{5,7}=f(K\_{NG}⊕08)$$



 PKT is updated when $SM\textsubscript{8}$ departs.

### Secure Unicast and Multicast Communication

Our scheme adopts the following message transmission methods in order to achieve confidentiality and integrity.

**Secure Unicast Communications.** When the AHE wants to send a message ($mi$) to $SM\textsubscript{$i$}$, they both can use the key agreement to generate a shared secret session key ($K\_{s}$) and send the following message,

where $Ti$ is the recorded time instance of sending the message in order to prevent replay attacks. Also, $HMAC$ is based on the sent message, and is used to ensure the integrity of the message.

**Secure Multicast Communications.** The shared group key ($K\_{G}$) is used to secure the multicast communication. As discussed earlier, our multicast management approach is used to maintain $K\_{G}$ in a self-organized manner so that the group key can be sent by any group member to the AHE. To illustrate the multicast communication, the AHE or a meter $SM\_{1}$ can send a multicast message ($mi$) to other meters in the group as follows.

Here, $HMAC$ and $Ti$ are used to ensure the integrity and prevent replay attacks, respectively.

# Security Analysis and Performance Analysis

In this section, we discuss our proposed scheme in terms of key management efficiency. First, the security features of our proposal are illustrated. Then, we conduct a performance analysis on real-life devices.

## Security Analysis

We analyze the security features the end-to-end key establishment protocol first,then the analysis of the proposed blockchain, followed by an analysis of the multicast key management protocol.

### Analysis of the proposed blockchain

The proposed blockchain network serve as a firewall against any manipulation attacks. It is required to ensure the integrity of the PKT allowing meters in a distributed way to collaboratively update their group keys whenever a meter joins or leaves the group. However, the major challenge is the selection of meter-nodes so that they can solve the puzzle problem in the mining process. One possible way is that some nodes are pre-determined to act as miners, and are responsible for solving the puzzle problem. Another solution is to randomly select some nodes as miners. At each join/leave operation, certain meters can be selected on a random basis among the meters. The later solution is more complex since it requires upgrading hardware of current meters than the predetermined solution.

### Security Analysis of the Proposed Key Agreement Protocol

Our proposed key agreement protocol is able to achieve the following security features.

*Achieves resistance to the MITM attack.* In the MITM attack, the adversary secretly relays and probably changes the communication between two participants so that they believe they are talking directly, while they are talking through the attacker. By verifying if $C\_{AHE}=C\_{SM\_{i}}$, the meter can verify the authenticity of the received message. Also, the AHE is doing the same by checking the message ($M\_{3}$). In this fashion, the MITM attack scenarios can be thwarted by adopting our proposal.

*Achieves data integrity.* We implement a hash function ($H1$) that allows our proposal to achieve data integrity. This is made possible because the implemented hash function includes a number, which is exploited as shown in the following simple example. Consider a situation when $SM\_{i}$ sends its identity $ID\_{i}$ and the public key ($PK\_{i}$) concatenated with the hash value of both. Upon receiving the message, the AHE can verify the integrity of the received message by recalculating its hash value.

*Provides mutual authentication.* In our proposed scheme, the system parameters ($Z\_{p}^{\*}$, $g$, $p$, $e$), which are preloaded to SMs, can provide a means of authentication with the AHE.

### The Multicast Key Management Protocol

The proposed multicast key management protocol uses the shared session keys generated by the key agreement protocol to send the PKT to SMs which is used to allow the SMs to collaborate with one another to effectively manage the group key. As a group key management protocol, our proposal has the following security properties.

*Distributed Feature:* Our proposed multicast key management removes the need for a centralized entity to generate or update security keys while facilitating the re-keying process. The latter appears as a good feature for the microgrid architecture that may be isolated from the main SG system. Moreover, SMs during the joining phase do not have to exchange keys. Also, at the departing phase, the new group key is forwarded to the remaining SMs by employing intermediate node codes.

*Backward Secrecy:* Our proposal uses backward secrecy to prevent a new member from decoding exchanged messages prior to the actual joining event. As illustrated in the joining process, when $SM\_{8}$ asks to join the group, the group key is renewed and the intermediate node codes in the affected path should be updated. By doing so, the SM is not able to decipher even the previous messages.

*Forward Secrecy:* Our proposal uses forward secrecy to prevent a departing or removed group member from continuing to access the communication within the group. This can be achieved as illustrated in the eviction process described earlier. When $SM\_{8}$ leaves the group, the tree, and the PKT are updated so that it is not able to further decipher group messages encrypted with the new group key after it has left the group.

## Performance Analysis

In order to analyze the performance of our proposed scheme, we consider two aspects, namely, computation and communication cost. Then, we compare these results with two relevant, existing schemes, which were proposed by Liu et al. (20) and Wan et al. (10), respectively. Since no comparable distributed key management protocol does not exist in the literature, we compare our proposal with these two methods to demonstrate the effectiveness of our scheme, particularly when dealing with SMs.

### Computation cost

Computation cost is the processing overhead needed by the nodes in the key management process. The computation costs evaluation due to the end-to-end key establishment and the multicast key management are shown in Table ???. The results For both Liu et al.’s and Wan et al.’s schemes are readily obtained from (10).

For the end-to-end key agreement, in our scheme, each SM needs to agree on the authenticated shared key to compute 2 exponentiation operations, 1 signature verification operation and 4 hashes. Meanwhile, the head end needs 2 exponentiation operations, 1 digital signature operation, and 3 hashes.

For the multicast key management, we analyze the computation cost required by SMs. Three criteria are used to estimate the overall computation cost needed by group members which are cost due to group key update, encryption overhead, and intermediate node keys update (33; 10) as illustrated below.

* For the join operation:
* Each SM needs to compute $1C\_{f}$ to get the new group key that totally costs $nC\_{f}$. $2C\_{E}$ encryption overhead. Also, the computation cost due to the update of intermediate node codes can be approximated to $(n−2)C\_{f}$. Hence, the total computation overhead = $nC\_{f}$ + $(n−2)C\_{f}$+$2C\_{E}$=$2nC\_{f}$.
* For the departure operation:
* Only $1C\_{f}$ is required for the new group key generation. Then, each SM should perform at least 1 encryption operation to get the group key which costs $nC\_{E}$. Also, the computation cost due to the update of the intermediate node codes can be approximated to $(n−5)C\_{f}$. Therefore, the total computation overhead= $1C\_{f}$ + $nC\_{E}$ +$(n−5)C\_{f}$=$2nC\_{E}−4C\_{f}$.

Computation time for cryptographic operations.

|  |  |  |
| --- | --- | --- |
| Cryptographic operation | MICAZ | Pentium IV 3GHz |
| Paring Operation | 5.32s | 3.88ms |
| Point Multiplication | 2.45s | 1.82ms |
| Hash Function | 0.023ms | $≈$0ms |
| AES Encryption/Decryption | 0.023ms | $≈$0ms |
| Public Encryption/Verify\_Signature(.) | 0.79s | 0.57ms |
| Public Decryption/Sign(.) | 21s | 16ms |

In order to simulate our scheme on a real life environment, cryptographic operations are carried out in wireless sensor nodes such as MICAZ sensors, which emulate SMs in nature. Each emulated SM comprises 4KB Random Access Memory (RAM), 128KB Read Only Memory (ROM), and equipped with a low-power ATmega128L micro-controller working at 7.3MHz. Also, a 3GHz Pentium IV PC is set up as the head end. We assume that the encryption is conducted by employing the Advanced Encryption Standard (AES) with a 128-bit symmetric key. Table ??? lists the estimated time for performing the various cryptographic operations (adopted from (36; 32)).



 Simulation result for the proposed key establishment protocol with Wan et al’s scheme. AHE-Wan and SM-Wan refers to the computation costs of the AHE and SM in Wan et al.’s scheme, respectively.



Computation cost comparison on SMs for joining or departing events of a meter.

For this particular performance evaluation of our proposed scheme, we conduct simulations based on MATLAB (37). The simulation results are depicted in Figs. ???, ???, and ???. First, for the end-to-end key agreement protocol, from Fig. ???, it can be noticed that the computation overhead on the SM in our proposed scheme is less than that in Wan et al.’s scheme. Wan et al.’s scheme has less overhead on the head end which are huge servers with high processing capabilities. However, it performs rather poorly on the SMs, which need to avoid the significant overhead incurred upon them when Wan et al.’s scheme is applied. For the multicast key management protocol as shown in Fig. ??? and Fig. ???, we can conclude that our scheme exhibits a much lower computation overhead at the SMs in the rekeying process than Liu et al.’s scheme. Even though Wan et al.’s scheme shows a better efficiency, our scheme has no overhead on the centralized node due to the contributory feature of our proposed scheme. As a result, our scheme can achieve scalability and efficiency despite its distributed nature.

### Communication cost

Communication cost is used to determine the number of messages required to be transmitted in the key agreement protocol or during the group re-keying process when a member joins/leaves the group. Communication cost results are given in Table ???. As shown in Table ???, our scheme costs only three unicast messages to run the end-to-end key establishment which is the same as that achieved by Wan et al.’s scheme. Moreover, for the rekeying process, in our distributed multicast key management approach, it only requires a single message to be delivered to the new member at the joining phase that contains the new key as a unicast message and one broadcast message to update the PKT by the public value of the newly added meter. Also, at the eviction phase, it needs a number of messages of the order $log\_{2}n$ as the unicast messages to renew the PKT are required. These results demonstrate that our proposed scheme incurs a low communication overhead compared to that achieved by Liu et al.’s scheme. In addition, our proposal is as efficient as Wan et al.’s scheme (while improving the computation cost). The results indicate that our proposal fits the microgrid scenario from the communication overhead perspective.



 Transactions cycle in the Blockchain

# Application in Blockchain

Recently, blockchain has attracted more and more attention from both academy and industry because of its decentralization. In this section, we will give a overview of the blockchain technology and how our proposed distributed key management scheme will be helpful in the blockchain era.

Blockchains open up the world of possibilities for solving problems that have afflicted us for many years. It enables us to create distributed consensus between mutually distrustful parties. In blockchain, all peers are form a distributed network. Each transaction record is encapsulated as a block and added onto the existing block chains. The recorded block contents are collectively referred to as the ledger. All information is then updated synchronously to the entire network so that each peer keeps a record of the same ledger. From Fig. ???, we can examine how a blockchain works. Firstly, users can interact with the blockchain by using a pair of private/public keys. The public key is used as the address for the client. Also, the private key is to sign their own transactions. Asymmetric cryptography is used for achieving authentication, integrity, and non repudiation in the network. Secondly, Every signed transaction is broadcasted by a user’s node to its one-hop peers. Thirdly, The neighboring peers make sure this incoming transaction is valid before relaying it any further, invalid transactions are discarded. Fourthly, the transactions that have been collected and validated by the network using the process above during an agreed-upon time interval, are ordered and packaged into a timestamped candidate block. This is a process called mining. The mining node broadcasts this block back to the network. Depending on the consensus mechanism used by the network, that will determine which nodes are responsible for mining and also the contents or the size of the block. Fifthly, The nodes validate whether the suggested block $(i)$ contains a valid transactions, and $(ii)$ references via hash the correct previous block on their chain. If the verification is valid, they add the block to their chain, and apply the transactions it contains to update their world view. If that is not the case, the proposed block is discarded. At the end, by the adding the block to the blockchain, the transaction is considered complete.

The blockchain has been commonly used in financial applications, and also other research areas like Internet of Things (IoT) and smart grid. Recently, (34) proposed distributed blockchain-based distributed framework for SG. The author first points out the effects of external cyber-attacks like False Data Injection Attack (FDIA) which may lead to wrong results by the control center. Then, the authors proposed a distributed blockchain-based data protection framework for enhancing the data security of the modern power system against cyber-attack by increasing the self-defensive capabilities of modern power systems against data manipulation by attackers. The authors pointed out by simulation that using the blockchain in SG can efficiently decreases the attacker capabilities of launching a successful data manipulation attacks.

The usefulness of blockchains in smart grid does not stop there. Instead of using regular schemes for achieving protection and privacy of users, (38) proposes a privacy-preserving and efficient data aggregation scheme based on the blockchain to preserve user’s privacy in smart grid. Instead of usual aggregation schemes which need a third-party to collect data, in (38), the authors suggested in each time slot, they select the mining node based on the difference between the user’s electricity consumption data and the average value to guarantee the fairness of the selection. Then, the mining node records all user’s data into the blockchain and publishes the blockchain within the group to ensure the message integrity. To defend against the snoop from other users in the same group, the author adopt pseudonyms to protect user’s identity and each user associate his data with multiple pseudonyms for further obfuscation.

However, as Blockchain still in its early development stages, there is a little focus on the key management area. The security of the above schemes can be well enhanced with our scheme. As mentioned early in sec. ???, our scheme has the distributivity feature as meters can communicate with each other independently without involving a third party. Hence, our scheme could be used in conjunction with blockchain to secure the peer-to-peer communications. In future, we will propose a unified blockchain-based communication to the smart grid.

# Conclusion

In this paper, we introduced a key management scheme to secure communication for the SG. First, a key agreement protocol between the AHE and smart meters was proposed. Then, to secure group communication, a blockchain-based multicast key management was proposed to secure group communications in which no central server needs to be used to distribute keys or update them when a member’s status changes. Also, we conducted the security and performance analysis of our proposal which indicates that it has low computation and communication overhead on the smart meters. Through security analysis, we demonstrated that our proposed scheme achieves forward and backward secrecy. In Addition, we propose how our distributed nature of our scheme can be used in with the blockchain.

# References

1.Fadlullah, Z. M. *et al.*. Toward intelligent machine-to-machine communications in smart grid. *IEEE Communications Magazine* **49,** (2011).

2.Coll-Mayor, D., Paget, M. & Lightner, E. Future intelligent power grids: Analysis of the vision in the European Union and the United States. *Energy Policy* **35,** 2453–2465 (2007).

3.Sæle, H. & Grande, O. S. Demand response from household customers: Experiences from a pilot study in Norway. *IEEE Transactions on Smart Grid* **2,** 102–109 (2011).

4.Sauter, T. & Lobashov, M. End-to-end communication architecture for smart grids. *IEEE Transactions on Industrial Electronics* **58,** 1218–1228 (2011).

5.Metke, A. R. & Ekl, R. L. Security technology for smart grid networks. *IEEE Transactions on Smart Grid* **1,** 99–107 (2010).

7.Fadlullah, Z. M., Fouda, M. M., Kato, N., Shen, X. & Nozaki, Y. An early warning system against malicious activities for smart grid communications. *IEEE Network* **25,** (2011).

9.Fang, X., Misra, S., Xue, G. & Yang, D. Smart grid—The new and improved power grid: A survey. *IEEE communications surveys & tutorials* **14,** 944–980 (2012).

10.Wan, Z., Wang, G., Yang, Y. & Shi, S. SKM: Scalable key management for advanced metering infrastructure in smart grids. *IEEE Transactions on Industrial Electronics* **61,** 7055–7066 (2014).

11.Baza, M. *et al.*. Blockchain-based Privacy-Preserving Charging Coordination Mechanism for Energy Storage Units. *arXiv preprint arXiv:1811.02001* (2018).

13.Diffie, W. & Hellman, M. New directions in cryptography. *IEEE transactions on Information Theory* **22,** 644–654 (1976).

14.Nicanfar, H., Jokar, P. & Leung, V. C. M. Smart grid authentication and key management for unicast and multicast communications. in *Innovative Smart Grid Technologies Asia (ISGT), 2011 IEEE PES* 1–8 (IEEE, 2011).

15.Shamir, A. Identity-based cryptosystems and signature schemes. in *Workshop on the theory and application of cryptographic techniques* 47–53 (Springer, 1984).

16.Mohammadali, A., Haghighi, M. S., Tadayon, M. H. & Nodooshan, A. M. A novel identity-based key establishment method for advanced metering infrastructure in smart grid. *IEEE Transactions on Smart Grid* (2016).

17.Kamto, J., Qian, L., Fuller, J. & Attia, J. Light-weight key distribution and management for advanced metering infrastructure. in *GLOBECOM Workshops (GC Wkshps), 2011 IEEE* 1216–1220 (IEEE, 2011).

19.Nicanfar, H., Jokar, P., Beznosov, K. & Leung, V. C. M. Efficient authentication and key management mechanisms for smart grid communications. *IEEE systems journal* **8,** 629–640 (2014).

20.Liu, N., Chen, J., Zhu, L., Zhang, J. & He, Y. A key management scheme for secure communications of advanced metering infrastructure in smart grid. *IEEE Transactions on Industrial electronics* **60,** 4746–4756 (2013).

21.Wong, C. K., Gouda, M. & Lam, S. S. Secure group communications using key graphs. *IEEE/ACM Transactions on Networking (TON)* **8,** 16–30 (2000).

22.Chen, L. & Kudla, C. Identity based authenticated key agreement protocols from pairings. in *Computer Security Foundations Workshop, 2003. Proceedings. 16th IEEE* 219–233 (IEEE, 2003).

23.Sherman, A. T. & McGrew, D. A. Key establishment in large dynamic groups using one-way function trees. *IEEE transactions on Software Engineering* **29,** 444–458 (2003).

24.Benmalek, M. & Challal, Y. eSKAMI: efficient and scalable multi-group key management for advanced metering infrastructure in smart grid. in *Trustcom/BigDataSE/ISPA, 2015 IEEE* **1,** 782–789 (IEEE, 2015).

26.Fouda, M. M., Fadlullah, Z. M., Kato, N., Takeuchi, A. & Nozaki, Y. A novel demand control policy for improving quality of power usage in smart grid. in *Global Communications Conference (GLOBECOM), 2012 IEEE* 5154–5159 (IEEE, 2012).

27.Sauter, T. & Lobashov, M. End-to-end communication architecture for smart grids. *IEEE Transactions on Industrial Electronics* **58,** 1218–1228 (2011).

28.Lasseter, R. H. & Paigi, P. Microgrid: A conceptual solution. in *Power Electronics Specialists Conference, 2004. PESC 04. 2004 IEEE 35th Annual* **6,** 4285–4290 (IEEE, 2004).

29.Rengaraju, P., Lung, C.-H. & Srinivasan, A. Communication requirements and analysis of distribution networks using WiMAX technology for smart grids. in *Wireless Communications and Mobile Computing Conference (IWCMC), 2012 8th International* 666–670 (IEEE, 2012).

30.Veitch, C. K., Henry, J. M., Richardson, B. T. & Hart, D. H. Microgrid cyber security reference architecture. *Sandia National Laboratories: Albuquerque, NM, USA* (2013).

31.Rafaeli, S. & Hutchison, D. A survey of key management for secure group communication. *ACM Computing Surveys (CSUR)* **35,** 309–329 (2003).

32.Wang, H. & Li, Q. Efficient implementation of public key cryptosystems on mote sensors (short paper). in *International Conference on Information and Communications Security* 519–528 (Springer, 2006).

33.Mortazavi, S. A., Pour, A. N. & Kato, T. An efficient distributed group key management using hierarchical approach with Diffie-Hellman and Symmetric Algorithm: DHSA. in *Computer Networks and Distributed Systems (CNDS), 2011 International Symposium on* 49–54 (IEEE, 2011).

34.Liang, G., Weller, S. R., Luo, F., Zhao, J. & Dong, Z. Y. Distributed Blockchain-Based Data Protection Framework for Modern Power Systems against Cyber Attacks. *IEEE Transactions on Smart Grid* (2018).

35.Nakamoto, S. Bitcoin: A peer-to-peer electronic cash system. (2008).

36.Lee, J., Kapitanova, K. & Son, S. H. The price of security in wireless sensor networks. *Computer Networks* **54,** 2967–2978 (2010).

37.MATLAB. *version 1.7.0 (R2017a)*. (The MathWorks Inc., 2017).

38.Guan, Z. *et al.*. Privacy-preserving and efficient aggregation based on blockchain for power grid communications in smart communities. *arXiv preprint arXiv:1806.01056* (2018).

39.Aitzhan, N. Z. & Svetinovic, D. Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams. *IEEE Transactions on Dependable and Secure Computing* (2016).

40.Christidis, K. & Devetsikiotis, M. Blockchains and smart contracts for the internet of things. *Ieee Access* **4,** 2292–2303 (2016).

41.Baza, M., Fouda, M. M., Eldien, A. S. T. & Mansour, H. A. K. An efficient distributed approach for key management in microgrids. in *Computer Engineering Conference (ICENCO), 2015 11th International* 19–24 (IEEE, 2015).

42.Xia, J. & Wang, Y. Secure key distribution for the smart grid. *IEEE Transactions on Smart Grid* **3,** 1437–1443 (2012).

43.Berthier, R., Sanders, W. H. & Khurana, H. Intrusion detection for advanced metering infrastructures: Requirements and architectural directions. in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on* 350–355 (IEEE, 2010).

44.Nabil, M., Alsharif, A., Sherif, A., Mahmoud, M. & Younis, M. Efficient Multi-Keyword Ranked Search over Encrypted Data for Multi-Data-Owner Settings. in *2018 IEEE International Conference on Communications (ICC)* 1–6 (IEEE, 2018).

45.Curtis, E. B. & Morrow, J. A. *Inverse problems for electrical networks*. (World Scientific, USA, 2000).

46.Banfield, J. D. & Raftery, A. E. Model-Based Gaussian and Non-Gaussian Clustering. *Biometrics* **49,** (1993).

47.Arabie, P. & Carroll, J. D. A mathematical programming approach to fitting the ADCLUS models. *Psychometrika* (1980).

48.Beale, E. M. L. Euclidean cluster analysis. *Bulletin of the International Statistical Institute* **43,** 92–94 (1969).

49.Bezdek, J. C. Numerical taxonomy with fuzzy sets. *Journal of Methematical Biology* **1,** 57–71 (1974).

50.Heard, N. A., Holmes, C. C. & Stephens, D. A. A Quantitative Study of Gene Regulation Involved in the Immune Response of Anopheline Mosquitoes: An Application of Bayesian Hierarchical Clustering of Curves. *J. Am. Statist. Assoc.* **101,** 18–29 (2006).

51.Fan, J. & Peng, H. Nonconcave penalized likelihood with a diverging number of parameters. *Ann. Statist.* **32,** 928–61 (2004).