A Privacy-Preserving and Accountable Billing Protocol for Peer-to-Peer Energy Trading Markets

Kamil Erdayandi*, Lucas C. Cordeiro* and Mustafa A. Mustafa*†
*Department of Computer Science, The University of Manchester, UK
†imec-COSIC, KU Leuven, Belgium
Email: {kamil.erdayandi, lucas.cordeiro, mustafa.mustafa}@manchester.ac.uk

***Abstract*—This paper proposes a privacy-preserving and accountable billing (PA-Bill) protocol for trading in peer-to-peer energy markets, addressing situations where there may be discrepancies between the volume of energy committed and delivered. Such discrepancies can lead to challenges in providing both privacy and accountability while maintaining accurate billing. To overcome these challenges, a universal cost splitting mechanism is proposed that prioritises privacy and accountability. It leverages a homomorphic encryption cryptosystem to provide privacy and employs blockchain technology to establish accountability. A dispute resolution mechanism is also introduced to minimise the occurrence of erroneous bill calculations while ensuring accountability and non-repudiation throughout the billing process. Our evaluation demonstrates that PA-Bill offers an effective billing mechanism that maintains privacy and accountability in peer-to-peer energy markets utilising a semi-decentralised approach.**

***Index Terms*—Billing, Privacy, Accountability, Peer-to-peer Energy Market, Homomorphic Encryption, Blockchain**

NOMENCLATURE

c_i, p_j, u_k	i-th consumer , j-th prosumer, k-th user
N_C , N_P, N_U	Number of consumers, prosumers, users
V^{P2P}	P2P market's traded electricity volume array
V^{Real}	Real electricity consumption array
$\pi_{P2P}, \pi_{FiT}, \pi_{RT}$	P2P, FiT, Retail price
$Stat$	Array of the statements of the users
Bal_{sup}	Balances of the supplier
$inDev$	Array of the individual deviations of the users
Dev^{Tot}	Total deviations of the users
$KGen_{pe}(k)$	Paillier key generation method
PK_{sup} , SK_{sup}	Public, Private (Secret) key pair of Supplier
$\{.\}_{\mathcal{E}}$	Data homomorphically encrypted with PK_{sup}.
$H(.)$	Hash Function

I. INTRODUCTION

A. Motivation and Background

Peer-to-peer (P2P) energy trading enables users to obtain clean energy at more reasonable prices than traditional suppliers, making it accessible to a wider society [1]. It facilitates direct energy exchange between households that harness renewable energy sources (RES) [2]. This approach empowers individuals to become active participants in the energy system [3], allowing RES owners to optimise their profits and reduce their bills through trading with other users [4].

Although P2P energy trading markets offer various benefits, some challenges hinder their widespread adoption. Firstly, the vast amount of data exchanged can reveal sensitive information about users [5], such as their energy usage habits and lifestyle patterns. Access to this data poses significant privacy risks [6] and could potentially violate privacy protection regulations, e.g., GDPR [7]. Thus, it is crucial to ensure privacy-preserving data processing and protect data from unauthorised access [8]. Secondly, such markets require secure and accountable solutions. However, it is challenging to audit transactions without a tamper-proof system [9]. To ensure fair and accurate energy trading, it is also essential to guarantee integrity and verifiability of any data used. Thirdly, often what users commit at P2P markets deviates from what they deliver due to intermittent RES output. Hence, any billing models will need mechanisms to deal with such deviations.

B. Relevant Literature

Within P2P energy trading, two crucial phases are market clearance and billing & settlement [10]. Since privacy-preserving market clearing mechanisms have already been explored [4], [11], [12], this paper focuses on the billing phase.

Madhusudan et al. [13] propose four billing models for P2P energy markets which account for deviations in energy volumes from the users' bids and incorporate individual, social, or universal cost-sharing mechanisms to ensure cost-effectiveness for both consumers and prosumers. Nonetheless, they do not explore user privacy. A privacy-preserving billing protocol that incorporates an individual cost-sharing mechanism has been proposed in [14]. However, it relies on a remote server for bill calculations, which poses a risk of a single point of failure.

Singh et al. [15] propose a method that uses blockchain and homomorphic schemes to protect the confidentiality of user data while enabling efficient data analysis. They do not explore any billing mechanisms. Gür et al. [16] propose a system based on blockchain technology and IoT devices to facilitate billing. To ensure data confidentiality, the system employs session keys and stores the encrypted data on the blockchain. However, this is still vulnerable to breaches as unauthorised parties can gain access to these keys, enabling them to access sensitive data.

This work was supported by EPSRC through EnnCore [EP/T026995/1] and by the Flemish Government through FWO-SBO SNIPPET project [S007619]. K.E is funded by The Ministry of National Education, Republic of Turkey.

In summary, no prior study on P2P market billing fully satisfies the three essential requirements: protecting user privacy, maintaining strong system accountability, and accommodating variations in user consumption. Neglecting any of these elements undermines the market trust, transparency and fairness, which are essential to their success and sustainability. Furthermore, integrating these three features within a single platform efficiently poses considerable challenges.

C. Contributions and Organization

To address the issues raised in the existing literature, we propose a novel privacy-preserving and accountable billing (PA-Bill) protocol, which effectively mitigate the challenges surrounding security, privacy, accountability, and user consumption variations prevalent in current studies. PA-Bill utilises a universal cost-splitting billing model that mitigates the risk of sensitive information leakage due to individual deviations. It also avoids a single point of failure by performing most calculations locally in a semi-decentralised manner. To preserve privacy, the mechanism employs homomorphic encryption in bill calculations. Moreover, PA-Bill utilises blockchain technology to integrate accountability mechanisms that addresses possible conflicts during the billing calculation process. To minimise privacy leakage, only the hashed version of the data is stored on the blockchain. Finally, PA-Bill can support large communities of 500 households.

Unlike other solutions, PA-Bill integrates privacy protection, accountability, and accommodating user consumption variations into a single solution in an efficient way. To the best of our knowledge, no previous work has successfully implemented an efficient billing model that simultaneously preserves privacy, ensures accountability, and effectively handles discrepancies between committed and delivered volume.

The rest of the paper is structured as follows: Section II outlines the preliminaries. The proposed PA-Bill is presented in Section III. The security analysis of PA-Bill is presented in Section IV, while its performance is evaluated in Section V. Finally, Section VI concludes the paper.

II. Preliminaries

A. System Model

Our proposed billing protocol, illustrated in Fig. 1, involves prosumers, consumers, a trading platform (TP), a distributed ledger/Blockchain (DLT), a referee, and a supplier. Prosumers generate energy through renewables, consume the volume they require, and sell any surplus energy. Consumers solely consume energy. Households have home energy management systems (HEMs) and smart meters (SMs) that measure electricity flows, provide real-time measurements, and facilitate P2P trading for the user. Prosumers and consumers can trade electricity through a P2P market using a trading platform (TP). If necessary, they can also buy or sell electricity from/to a supplier as a backup option. However, P2P trading is more beneficial than relying on the supplier due to pricing considerations [4]. Financial reconciliation occurs during settlement cycles (SCs) for users involved in trading. Within each SC,

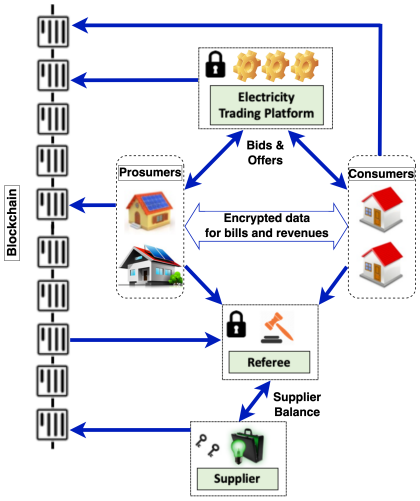

Fig. 1: System model.

data regarding the actual electricity usage of households and their commitments to trade in the market are stored on DLT. Households calculate their bills locally in a decentralised manner. If a dispute arises, a referee intervenes to resolve it by requesting data from households and retrieving it from DLT.

B. Threat Model and Assumptions

Our threat model comprises untrustworthy and semi-honest entities. Prosumers and consumers who may attempt to violate the protocol specifications and obtain sensitive data of other users are considered to be untrustworthy. Prosumers may try to maximise their revenue, while consumers may aim to minimise their expenses. Semi-honest entities include the TP, referee, and supplier. They adhere to the protocol specifications, but they may still be curious to learn sensitive data of users.

SMs are tamper-proof and sealed. Anyone, including their users, can not tamper with them without being detected. Users act rationally by seeking the most cost-effective electricity to buy or sell [17]. We assume that the entities communicate over secure and authentic communication channels.

C. Design Requirements

- No single point of failure (SPF): To avoid SPF, calculations and data storage should be distributed [18].
- Privacy: Confidentiality of individual users' volumes of energy traded and consumed as well as individual deviation and deviation sign should be provided.
- Accountability: Disputes arising from erroneous bill calculations must be addressed in an accountable way to prevent any party from denying responsibility.
- Fair deviation cost distribution: cost of P2P market deviation should be split fairly among market participants.

D. Building Blocks

Homomorphic encryption (HE) enables computations to be performed on encrypted data, resulting in encrypted outputs that produce the same results as if the operations were conducted on unencrypted data [19]. Specifically, we deploy the

Paillier cryptosystem which supports homomorphic addition and scalar multiplication on ciphertexts [20]. Our solution ensures the privacy of households by encrypting sensitive information such as energy consumption data per SC. Billing calculations are performed on this encrypted data, thereby preserving the confidentiality of the information. We use blockchain technology to provide accountability by ensuring that transactions are permanently recorded in a decentralised and immutable system with append-only storage. Transactions recorded on a blockchain cannot be altered by design, ensuring that they are accurate and trustworthy [15].

III. Privacy Preserving and Accountable Billing (PA-Bill) Protocol

In this section, we propose a privacy-preserving and accountable billing protocol for P2P energy market where users' actual energy consumption may differ from the volumes they committed. It protects sensitive household information and enables system entities to verify accurate billing calculations.

A. PA-Bill Overview

The process of PA-Bill protocol is illustrated in Fig. 1, which includes interactions between the entities. The system utilises the public-private key pair of the supplier for all homomorphically encrypted calculations. A distinct set of HE keys, namely PK_{sup} and SK_{sup} are generated for each billing month. Additionally, each month the consumers and prosumers are paired together to perform accountable calculations.

In the energy trading model, users send homomorphically encrypted bid-offer data to the TP, which calculates the final trading price π_{P2P} and the amount of energy $\left\{V^{P2P}[u_k]\right\}_{\mathcal{E}}$ that each user u_k will trade via the P2P market, as in [4].

During each SC, π_{P2P} is publicly released. $\left\{V^{P2P}[u_k]\right\}_{\mathcal{E}}$ is shared with related paired users for future calculations, and its hash is stored on the DLT for future verification. SMs measure their users' actual imported/exported electricity and transmit the encrypted version ($\left\{V^{Real}[u_k]\right\}_{\mathcal{E}}$) to relevant users. The hash of this encrypted version is also stored on the DLT.

After sending and storing related data for billing, the calculation of bills among prosumers and consumers is performed in three stages in a privacy-preserving way. Firstly, individual deviations of users are calculated. Consumers calculate the individual deviations of prosumers and vice versa. Secondly, the total deviations of consumers and prosumers are calculated by six user selected from consumers and prosumers. Thirdly, statements (bills/revenues) of users are calculated.

To protect sensitive data such as energy consumed/traded, and individual energy deviations of households, our work utilises HE scheme to process data while preserving privacy. However, it is crucial to design the billing algorithm in such a way that it avoids any indirect leakage of private information despite the use of encryption. Traditional billing methods [13], [14] have the potential to expose confidential information by using individual deviations between actual and committed energy volumes to determine the "conditions" in calculating bills. This enables inferences to be made about whether the actual electricity consumption volume is lower or higher than the committed data. To address this issue, we propose a privacy-preserving and accountable cost-splitting billing that uses total deviations of consumers and prosumers rather than individual deviations to determine billing conditions.

In the event of a dispute, the referee requests the necessary data from households, as well as it retrieves the hash of the previously stored data from DLT (to ensure the accuracy of the data requested from households) to settle the dispute. In this case, the referee corrects erroneous computations of the pair of customer and prosumer whose calculations do not match each other and identifies the responsible party in the pair. The responsible party is penalised, incentivising them to act truthfully, which would otherwise result in penalties. Besides, the referee can directly calculate the supplier's balance since the calculations do not involve any confidential information.

Finally, at the end of the month, final bills and revenues, and the balance of the supplier are released with the help of the referee and the private homomorphic key of the supplier.

B. Technical Details of PA-Bill

At the start of each billing period (e.g., a month), the following two steps (1-2) are carried out.

1) Generation of Keys: The supplier generates a public-private HE (Paillier) key pair: $KGen_{pe}(k) \rightarrow PK_{sup}, SK_{sup}$.

2) Matching customers and prosumers: The referee conducts a random matching process in which each consumer is paired with a list of prosumers and vice versa. The number of users in the lists may exceed one or be zero in cases where $N_C > N_P$ or $N_C < N_P$, while the lists contain only one user if $N_C = N_P$. Here, N_C and N_P denote the respective number of customers and prosumers. The function $M(u_k)$ returns the list of users that have been matched to the user u_k.

At each SC, the following six steps (3–8) are carried out.

3) Transfer and Storage of P2P Traded Data: TP makes the P2P trading price public by storing it at DLT in plaintext. For each u_k, TP transmits homomorphically encrypted value of traded volume $\left\{V^{P2P}[u_k]\right\}_{\mathcal{E}}$ to user u_k and to users in $M(u_k)$. The privacy-preserving calculation of the encrypted traded values by user u_k ($\left\{V^{P2P}[u_k]\right\}_{\mathcal{E}}$) can be performed after the transmission of bids-offers in a homomorphically encrypted format. It is assumed the TP has already calculated $\left\{V^{P2P}[u_k]\right\}_{\mathcal{E}}$. Once the data has been transmitted to relevant parties, the TP also hashes the homomorphically encrypted traded volume of user u_k, i.e., $H(\left\{V^{P2P}[u_k]\right\}_{\mathcal{E}})$, and stores the result at the DLT, together with a timestamp and ID of u_k.

4) Collection, Transfer and Storage of SM Data: At the end of each SC, each SM measures the real volume of energy imported from (or exported to) the grid by their user, i.e., $V^{Real}[u_k]$, encrypts it with PK_{sup} and hashes it, i.e., $H(\left\{V^{Real}[u_k]\right\}_{\mathcal{E}})$. It then stores the hash value to DLT with timestamp and ID of u_k. The user SM also stores $\left\{V^{Real}[u_k]\right\}_{\mathcal{E}}$ as well as sends it to the users in $M(u_k)$.

5) Calculation of Individual Deviations: in this step, each user u_k calculates the individual deviations ($inDev$) from the volume of energy they committed for themselves and their

Algorithm 1: Calculating Individual Deviations

Input: N_U
Output: $\{inDev\}_{\mathcal{E}}, \{inDev_M\}_{\mathcal{E}}$

for *each* u_k **do**
  $\{inDev[u_k]\}_{\mathcal{E}} \leftarrow \{V^{Real}[u_k]\}_{\mathcal{E}} - \{V^{P2P}[u_k]\}_{\mathcal{E}}$;
  for *each* m_l **in** $M(u_k)$ **do**
    $\{inDev_M[M(m_l)]\}_{\mathcal{E}} \leftarrow \{V^{Real}[M(m_l)]\}_{\mathcal{E}} - \{V^{P2P}[M(m_l)]\}_{\mathcal{E}}$;
  end
end

Algorithm 2: Calculating Bills and Revenues

Input: $N_U, \{V^{P2P}\}_{\mathcal{E}}, \{V^{Real}\}_{\mathcal{E}}, Dev_C^{Tot}, Dev_P^{Tot}, \pi_{P2P}, \pi_{RT}$
Output: $\{Stat\}_{\mathcal{E}}, \{Stat_M\}_{\mathcal{E}}$

for *each* u_k **do**
  if $Dev_P^{Tot} = Dev_C^{Tot}$ **then**
    $\{Stat[u_k]\}_{\mathcal{E}} \leftarrow \{V^{P2P}[u_k]\}_{\mathcal{E}} \cdot \pi_{P2P} + \{inDev[u_k]\}_{\mathcal{E}} \cdot \pi_{P2P}$
    for *each* m_l **in** $M(u_k)$ **do**
      $\{Stat[m_l]\}_{\mathcal{E}} \leftarrow \{V^{P2P}[m_l]\}_{\mathcal{E}} \cdot \pi_{P2P} + \{inDev[m_l]\}_{\mathcal{E}} \cdot \pi_{P2P}$
    end
  end
  if $Dev_P^{Tot} < Dev_C^{Tot}$ **then**
    $\{Stat[u_k]\}_{\mathcal{E}} \leftarrow \{V^{P2P}[u_k]\}_{\mathcal{E}} \cdot \pi_{P2P} + \{inDev[u_k]\}_{\mathcal{E}} \cdot \pi_{RT}$
    for *each* m_l **in** $M(u_k)$ **do**
      $\{Stat[m_l]\}_{\mathcal{E}} \leftarrow \{V^{P2P}[m_l]\}_{\mathcal{E}} \cdot \pi_{P2P} + \{inDev[m_l]\}_{\mathcal{E}} \cdot \pi_{RT}$
    end
  end
  if $Dev_P^{Tot} > Dev_C^{Tot} c$ **then**
    if u_k *is a consumer* **then**
      $\{Stat[u_k]\}_{\mathcal{E}} \leftarrow \{V^{P2P}[u_k]\}_{\mathcal{E}} \cdot \pi_{P2P} + \{inDev[u_k]\}_{\mathcal{E}} \cdot \pi_{P2P}$
      for *each* m_l **in** $M(u_k)$ **do**
        $\{Stat[m_l]\}_{\mathcal{E}} \leftarrow \{V^{P2P}[m_l]\}_{\mathcal{E}} \cdot \pi_{P2P} + \{inDev[m_l]\}_{\mathcal{E}} / Dev_P^{Tot} \cdot TotRev_P$
      end
    else
      $\{Stat[u_k]\}_{\mathcal{E}} \leftarrow \{V^{P2P}[u_k]\}_{\mathcal{E}} \cdot \pi_{P2P} + \{inDev[u_k]\}_{\mathcal{E}} / Dev_P^{Tot} \cdot TotRev_P$
      for *each* m_l **in** $M(u_k)$ **do**
        $\{Stat[m_l]\}_{\mathcal{E}} \leftarrow \{V^{P2P}[m_l]\}_{\mathcal{E}} \cdot \pi_{P2P} + \{inDev[m_l]\}_{\mathcal{E}} \cdot \pi_{P2P}$
      end
    end
  end
  $\{stat^{Tot}[u_k]\}_{\mathcal{E}} \leftarrow \{stat^{Tot}[u_k]\}_{\mathcal{E}} + \{stat[u_k]\}_{\mathcal{E}}$
  for *each* m_l **in** $M(u_k)$ **do**
    $\{stat_M^{Tot}[m_l]\}_{\mathcal{E}} \leftarrow \{stat^{Tot}[m_l]\}_{\mathcal{E}} + \{stat[m_l]\}_{\mathcal{E}}$
  end
end

corresponding matched users in $M(u_k)$ (see Alg. 1). To calculate $inDev$, each user u_k subtracts their committed volume from the volume measured by their SM for themselves (u_k) and the users m_l in $M(u_k)$. The calculations are carried out in homomorphically encrypted format. The espective encrypted results $\{inDev\}_{\mathcal{E}}$ and $\{inDev_M\}_{\mathcal{E}}$ are sent to the referee.

After the referee receives the encrypted individual deviations from users, it checks whether the computations have been done correctly. For each user and its matched user, the referee receives four encrypted results. The user u_k provides its own encrypted result, $\{inDev[u_k]\}_{\mathcal{E}}$, as well as that of its matched user. For the matched consumer c_i and prosumer p_j, the referee checks if the calculated values are the same. In order to achieve this, the referee subtracts these two calculated values from each other in a homomorphically encrypted format. The result of this subtraction is then sent to the supplier who has the private key to perform homomorphic encryption operations. The supplier decrypts the result of subtraction and sends it back to referee. The referee checks whether the received value from the supplier is zero or not. If it is zero, it considers the calculations to be accurate and proceeds to store the hash of the resulting computation of user u_k (not that of the matched user) in DLT along with the corresponding ID and timestamp of u_k, to facilitate future verification. Otherwise (if the received result is not zero), the referee intervenes to correct any erroneous calculations and identify the responsible party. To do so, the referee requests $\{V^{Real}\}_{\mathcal{E}}$ and $\{V^{P2P}\}_{\mathcal{E}}$ from the users, checks their correctness by hashing and comparing them with the previously stored hashes in blockchain by TP and SMs. If the encrypted data received from the users is accurate, the referee recalculates the inDev in encrypted format for c_i and p_j, whose results were incorrect. Next, the referee follows the same process of subtracting the calculated values and having the result decrypted by the supplier to compare the recalculated outcome with the values obtained from c_i and p_j. The referee then identifies the party that is accountable for the mismatch.

6) Calculation of Total Deviations: To calculate total demand and supply deviations, the referee selects three consumers and three prosumers. Each consumer c_i sends their respective $\{inDev[c_i]\}_{\mathcal{E}}$ to the selected prosumers and vice versa. Selected prosumers and consumers verify the received encrypted deviations by hashing and comparing them with stored hashes in DLT. Then, selected prosumers sum up $\{inDev[c_i]\}_{\mathcal{E}}$ for each c_i to calculate $\{Dev_C^{Tot}\}_{\mathcal{E}}$ (eq. 1) and selected consumers do the same for each p_i, (eq. 2).

$$\{Dev_C^{Tot}\}_{\mathcal{E}} \leftarrow \sum_{i=0}^{N_C-1} \{inDev_C[c_i]\}_{\mathcal{E}} \tag{1}$$

$$\{Dev_P^{Tot}\}_{\mathcal{E}} \leftarrow \sum_{j=0}^{N_C-1} \{inDev_P[p_j]\}_{\mathcal{E}} \tag{2}$$

After calculating $\{Dev_C^{Tot}\}_{\mathcal{E}}$ and $\{Dev_P^{Tot}\}_{\mathcal{E}}$, selected prosumers and consumers send them to a referee for verification. If the results match, the referee sends them to the supplier. The supplier then decrypts the results and makes them publicly available by storing Dev_C^{Tot} and Dev_P^{Tot} into DLT. If the results do not match, the referee corrects any erroneous calculations and identifies the responsible party. This is done by recalculating (eq. 1) and (eq. 2) in encrypted format after requesting and verifying the necessary data via DLT.

7) Calculation of Bills and Rewards: we present our proposed privacy-preserving and accountable universal cost-splitting billing model that employs total deviations instead

of individual deviations to establish billing conditions. The proposed billing model is presented in Alg. 2. The algorithm takes as input $\{V^{P2P}\}_{\mathcal{E}}$, $\{V^{Real}\}_{\mathcal{E}}$, π_{P2P}, π_{RT} and π_{FiT} and calculates the bills/revenues of consumers/prosumers. The algorithm outputs Statements $Stat[u_k]$, $Stat_M[u_k]$ for user u_k and its matched users in $M(u_k)$, respectively. $Stat[u_k]$ indicates the bill of u_k when u_k is a consumer and it stands for the revenue of u_k if u_k is a prosumer. We have devised universal formulas such as $Stat[u_k]$ which is applicable to both consumers and prosumers. The algorithm works in three modes based on the difference between total deviations of consumers and prosumers, and proceeds as follows.

If $Dev_P^{Tot} = Dev_C^{Tot}$, prosumers have generated enough electricity to meet the demand of customers, resulting in a balanced P2P market. In this case, individuals can purchase the required energy from other households and sell their excess energy to other households at π_{P2P} in addition to their commitments in the P2P market rather than relying on suppliers. Energy sharing between households to compensate for deviations is advantageous for both consumers and prosumers, as they can exchange energy at a price of π_{P2P}, which is higher than π_{FiT} and lower than π_{RT}, compared to relying on suppliers to buy electricity at π_{RT} and sell electricity at π_{FiT}. The statements for each user u_k and for paired users in $M(u_k)$ are calculated between ln. 3-6 in the algorithm.

If $Dev_P^{Tot} < Dev_C^{Tot}$, there is a shortage of electricity in the P2P market as prosumers have not generated enough electricity to meet customer demand. If there is a shortage of electricity that cannot be compensated by other users, the only option is to purchase it from the supplier at π_{RT}. Users with a shortage of electricity can buy it at this price, while households with a surplus can sell it at π_{RT} instead of selling it to the supplier for π_{FiT}, which is advantageous for prosumers. In accordance with this, the statements for each user u_k and for paired users in $M(u_k)$ are calculated between ln. 9-11 in the algorithm.

If $Dev_P^{Tot} > Dev_C^{Tot}$, there is excess electricity in the P2P market as prosumers have generated more electricity than is needed to meet customer demand. In this case, consumers can purchase energy from prosumers at π_{P2P} to compensate for their energy shortage due to deviation. The total revenue of the prosumers is distributed among them in proportion to the excess energy they provided. To calculate this, the total revenue generated by prosumers due to excess energy is first determined. Some of the excess energy is sold to consumers with a shortage of electricity at π_{P2P}, while the remainder is sold to the supplier at π_{FiT}. Therefore, the total revenue of prosumers, $TotRev_P$, can be calculated as

$$TotRev_P = (Dev_C^{Tot} \cdot \pi_{P2P} + (Dev_P^{Tot} - Dev_C^{Tot}) \cdot \pi_{FiT}) \quad (3)$$

The total revenue $TotRev_P$ is distributed among the prosumers in proportion to $inDev_P[u_k]/Dev_P^{Tot}$. In accordance with this, Alg. 2 calculates statements for each user u_k and for paired users in $M(u_k)$ between ln. 16-19, if u_k is a consumer. Otherwise, the statements are calculated between ln. 21-24.

At the end of the algorithm, statements are accumulated on $stat^{Tot}$ in encrypted format for u_k and user in $M(u_k)$ assuming that $stat^{Tot}$ was set to zero before the first SC.

After each pair calculates their statements bilaterally, they send the results to the referee for verification. If the results do not match, the referee intervenes to correct any erroneous calculations and identify the responsible party. This is done by running Alg. 2 for the unmatched pairs after requesting and verifying the required data for computation via DLT.

8) Calculating the of Balance of the Supplier: The referee calculates the supplier's balance using only public information, and does so in a non-encrypted format. In the case where $Dev_P^{Tot} = Dev_C^{Tot}$, Bal_{sup} is set to zero ($Bal_{sup} \leftarrow 0$) since there is no excess or shortage of electricity in the P2P market to compansate from the supplier. If ($Dev_P^{Tot} > Dev_C^{Tot}$), there is excess energy in P2P market and the supplier purchases it at FiT price π_{FiT}, resulting in a negative balance for the supplier to pay. Bal_{sup} is calculated as the negative product of the total excess energy $(Dev_P^{Tot} - Dev_C^{Tot})$ and π_{FiT}, i.e.

$$Bal_{sup} \leftarrow -(Dev_P^{Tot} - Dev_C^{Tot}) \cdot \pi_{FiT} \quad (4)$$

If ($Dev_P^{Tot} < Dev_C^{Tot}$), there is a shortage of energy in P2P market that needs to be compensated by the supplier at retail price π_{RT}. Bal_{sup} is calculated as the product of supplied energy $(Dev_P^{Tot} - Dev_C^{Tot})$ and π_{RT}, i.e.

$$Bal_{sup} \leftarrow (Dev_C^{Tot} - Dev_P^{Tot}) \cdot \pi_{RT}. \quad (5)$$

At each SC, the resulting Bal_{sup} is accumulated to the total supplier balance except when the SC is equal to zero where Bal_{sup}^{Tot} is set to Bal_{sup}.

The next step is carried out at the end of each billing period.

9) Transfer and Announcement of Bills, Revenues and Supplier Balance: The final accumulated monthly statements of households are not protected from the supplier, as payments must be made, the referee sends encrypted statements consisting of bills and revenues to the supplier. The supplier then decrypts these statements using their HE private key and hashes and stores the decrypted version on the DLT system for future verification during the payment process. The supplier's balance is also hashed and stored on the DLT.

IV. Security, Privacy and Accountability Analysis

The PA-Bill protocol addresses the security concern of avoiding SPF by distributing the majority of calculations and data storage locally. It addresses privacy concerns by utilising HE to encrypt sensitive user data such as V^{Real} and V^{P2P}, ensuring that sensitive information remains confidential during billing computations. In addition, the PA-Bill protocol employs a cost-splitting mechanism that utilises the total deviations of users rather than individual deviations to calculate billing modes. This method avoids indirect privacy leakage of individual deviations. It employs Blockchain technology to create an unalterable record of the hashes of essential data necessary for billing computations. This ensures the verification and integrity of critical data, thereby enabling all parties to be held accountable for their actions during the billing process.

V. Performance Evaluation

In this section, we demonstrate that PA-Bill achieves computational efficiency without compromising privacy, accountability, or the ability to accommodate user consumption variations. PA-Bill effectively addresses these critical aspects while maintaining a level of computational efficiency. We prove our claims through both theoretical analysis and experiments.

A. Theoretical Analysis

The time complexity of the method is mainly determined by the input parameters of Alg. 1 and Alg. 2, which include the number of users (N_U). The time required to perform the algorithm grows depending on the input size. Specifically, the nested double loops in Alg. 1 and Alg. 2 lead to a quadratic time complexity of $\mathcal{O}(n^2)$ for cases where in cases where $N_C > N_P$ or $N_C < N_P$, the time complexity is reduced to $\mathcal{O}(n)$ with a single iteration in the inner loop when $N_C = N_P$ where each user has only one matched user. The time complexity of the calculations in eq. 1 and eq. 2 is $\mathcal{O}(n)$, where n depends on the inputs N_C and N_P, respectively.

B. Experimental Results

We evaluate the performance of PA-Bill by running simulations on a PC with Intel Core i5 CPU @ 2GHz CPU and 16GB of RAM to demonstrate its efficiency. We utilise the SHA3-256 algorithm for hashing and the Paillier cryptosystem for homomorphic encryption with 2048-bit keys. These operations were implemented using the Python libraries hashlib and phe, respectively. We utilised Ethereum network to prototype the blockchain platform. To deploy and test Ethereum for our project, we used Ganache[1], wrote smart contracts in Solidity[2], and compiled them on Remix[3]. To connect our project with the Ethereum network, we utilised the Python Web3[4] library. As we utilised existing tools to design the blockchain platform, we did not conduct a separate performance assessment of the platform itself. Our previous work [4] is deployed as electricity trading platform, so we do not reevaluate it in this context either. Instead, our primary focus lies in evaluating the performance of the privacy and accountable billing model.

The billing model simulations were conducted on a sample of 500 users, consisting of 250 consumers and 250 prosumers. We measured PA-Bill's execution time (ET) for computationally intensive components in two scenarios: worst-case (every household makes an incorrect bill calculation (unintentionally or maliciously), thus requiring an intervention from the referee) and best-case (all households make correct calculations, hence no referee intervention is deployed). The SC is set to be one hour. Table I demonstrates the average execution time per SC for PA-Bill components, computed over a one-month billing period comprising 720 SCs (24 SCs per day). The execution time which results in milliseconds for both worst-case and best-case scenarios, tested with a large group of 500 users, indicate that our proposed billing protocol offers a computationally efficient solution for PA-Bill.

[1]https://www.trufflesuite.com/ganache

[2]https://solidity.readthedocs.io/en/v0.8.7/

[3]https://remix.ethereum.org/

[4]https://web3py.readthedocs.io/en/stable/

TABLE I: Execution time results per settlement cycle.

Calculation step	Worst-case ET	Best-case ET
Individual Deviations	23.84 ms	48.64 ms
Total Deviations	69.25 ms	246.23 ms
Bills and Rewards	25.76 ms	50.15 ms

VI. Conclusion

In this work, we proposed PA-Bill, a privacy-preserving and accountable billing protocol that addresses security, privacy, and accountability issues in P2P markets at the billing and settlements stage. PA-Bill utilises a universal cost-splitting billing model, local semi-decentralised calculation, and Homomorphic Encryption for privacy protection. Blockchain technology is deployed for accountability mechanisms that resolve conflicts during billing calculation. PA-Bill is evaluated on a community of 500 households. In our future work, we plan to investigate network constraints.

References

[1] E. A. Soto, L. B. Bosman, E. Wollega, and W. D. Leon-Salas, "Peer-to-peer energy trading: A review of the literature," *Applied Energy*, 2021.

[2] W. Tushar, T. K. Saha, C. Yuen, D. Smith, and H. V. Poor, "Peer-to-peer trading in electricity networks: an overview," *IEEE Transactions on Smart Grid*, 2020.

[3] M. Gržanić, T. Capuder, N. Zhang, and W. Huang, "Prosumers as active market participants: A systematic review of evolution of opportunities, models and challenges," *Renewable and Sustainable Energy Reviews*, 2022.

[4] K. Erdayandi, A. Paudel, L. Cordeiro, and M. A. Mustafa, "Privacy-friendly peer-to-peer energy trading: A game theoretical approach," in *Power & Energy Society General Meeting (PESGM)*, 2022, pp. 1–5.

[5] Y. Ding, B. Wang, Y. Wang, K. Zhang, and H. Wang, "Secure metering data aggregation with batch verification in industrial smart grid," *IEEE Tran. on Industrial Informatics*, vol. 16, no. 10, pp. 6607–6616, 2020.

[6] K. M. Ramokapane, C. Bird, A. Rashid, and R. Chitchyan, "Privacy design strategies for home energy management systems (hems)," in *CHI Conf. on Human Factors in Computing Systems*, 2022, pp. 1–15.

[7] "Regulation (EU) 2016 - general data protection regulation," (Accessed on 02/07/2023). [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32016R0679

[8] M. Montakhabi, A. Madhusudan, S. Van Der Graaf, A. Abidin, P. Ballon, and M. A. Mustafa, "Sharing economy in future peer-to-peer electricity trading markets: Security and privacy analysis," in *Proc. of Workshop on Decentralized IoT Systems*, 2020, pp. 1–6.

[9] M. Moniruzzaman, A. Yassine, and R. Benlamri, "Blockchain and cooperative game theory for peer-to-peer energy trading in smart grids," *International Journal of Electrical Power & Energy Systems*, 2023.

[10] A. Abidin, R. Callaerts, G. Deconinck, S. Van Der Graaf, A. Madhusudan, M. Montakhabi, M. A. Mustafa, S. Nikova, D. Orlando, J. Schroers *et al.*, "Poster: Snippet—secure and privacy-friendly peer-to-peer electricity trading," in *NDSS 2020, San Diego, CA, USA*.

[11] F. Funk, F. Teske, J. Franke, C. Heider, M. König, and O. Soukup, "A privacy-preserving, sealed double-auction smart contract for local energy markets," in *Workshop on Blockchain for Renewables Integration (BLORIN)*. IEEE, 2022.

[12] A. Abidin, A. Aly, S. Cleemput, and M. A. Mustafa, "An MPC-based privacy-preserving protocol for a local electricity trading market," in *Int. Conf. on Cryptology and Network Security*, 2016, pp. 615–625.

[13] A. Madhusudan, F. Zobiri, and M. A. Mustafa, "Billing models for peer-to-peer electricity trading markets with imperfect bid-offer fulfillment," in *2022 IEEE Int. Smart Cities Conf. (ISC2)*, pp. 1–7.

[14] R. Thandi and M. A. Mustafa, "Privacy-enhancing settlements protocol in peer-to-peer energy trading markets," in *ISGT-NA*. IEEE, 2022, pp. 1–5.

[15] P. Singh, M. Masud, M. S. Hossain, and A. Kaur, "Blockchain and homomorphic encryption-based privacy-preserving data aggregation model in smart grid," *Computers & Electrical Eng.*, vol. 93, p. 107209, 2021.

[16] A. Ö. Gür, Ş. Öksüzer, and E. Karaarslan, "Blockchain based metering and billing system proposal with privacy protection for the electric network," in *Istanbul smart grids and cities congress and fair (ICSG)*. IEEE, 2019, pp. 204–208.

[17] A. Abidin, A. Aly, S. Cleemput, and M. A. Mustafa, "Towards a local electricity trading market based on secure multiparty computation."

[18] K. Erdayandi, L. C. Cordeiro, and M. A. Mustafa, "Towards privacy preserving local energy markets," in *CADE 2022*, 2022, pp. 1–8.

[19] X. Yi, R. Paulet, E. Bertino, X. Yi, R. Paulet, and E. Bertino, *Homomorphic encryption*. Springer, 2014.

[20] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *EUROCRYPT*. Springer, 1999, pp. 223–238.