PADaaV: Blockchain-based Parking Price Prediction Scheme for Sustainable Traffic Management

RIYA KAKKAR1, JAFAR ALZUBI2, AMIT DUA3, SMITA AGRAWAL1, SUDEEP TANWAR1 (Senior Member, IEEE), RAJAT AGRAWAL3, GULSHAN SHARMA4, PITSHOU N BOKORO5, RAVI SHARMA6

1Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, Gujarat, India (e-mails: 21ftphde56@nirmauni.ac.in, smita.agrawal@nirmauni.ac.in, sudeep.tanwar@nirmauni.ac.in)
2Faculty of Engineering, Al-Balqa Applied University, Jordan e-mail: (j.zubi@bau.edu.jo)
3Department of Computer Science and Information Systems, BITS Pilani, Pilani, Rajasthan, 333031, India, e-mail: (amit.dua@pilani.bits-pilani.ac.in, f20180182@pilani.bits-pilani.ac.in)
4Department of Electrical Engineering Technology, University of Johannesburg, Johannesburg, South Africa. (email: gulshans@uj.ac.za)
5Department of Electrical and Electronics Engineering, University of Johannesburg, Johannesburg 2006, South Africa. (email: pitshoub@uj.ac.za)
6Centre for Inter-Disciplinary Research and Innovation, University of Petroleum and Energy Studies, Dehradun, 248001, India. e-mail: (ravisharmacri@gmail.com)

Corresponding authors: Sudeep Tanwar (sudeep.tanwar@nirmauni.ac.in), Smita Agrawal (smita.agrawal@nirmauni.ac.in), and Gulshan Sharma (gulshans@uj.ac.za).

ABSTRACT In most countries, traffic congestion has reached a level where managing traffic is tedious for regulatory bodies. The traffic management faced many issues such as route routing based on congestion, delivery of messages/emails to end-users, and real-time allocation of parking slots. There have been many works on predicting parking prices for traffic management, but most favor users or owners and are not secure. To address these issues, a blockchain and Interplanetary File System (IPFS)-based parking price prediction scheme (PADaaV) is proposed to facilitate the users to reserve a parking slot securely and efficiently. It mainly focuses on ensuring security, privacy, and transparency for parking slot owners and users. Furthermore, we employ a second price auction model to optimize the parking price for users, and parking slot owners can also get benefit from it. The performance of the PADaaV has been simulated for 100 users with 40 parking slots based on different auction models. The various performance parameters considered are profit for users, profit for parking slot owners, overall revenue of the system, scalability, computation time, and data storage cost. The performance results show that the PADaaV is secure and beneficial for users and parking slot owners.

INDEX TERMS Blockchain, Smart contracts, Second price auction model, Traffic management

I. INTRODUCTION

TRAFFIC management is one of the challenging tasks that need to be controlled efficiently. It is necessary to regulate the flow of traffic for sustainable traffic management. Traffic congestion is the main reason that it is getting difficult day by day for the drivers to find an empty parking slot, especially during peak hours (1). There are many research studies conducted (2) (3), which state that vehicles looking for free parking slots include approximately 8% of traffic. Drivers spend approximately 3.5 to 14 minutes to find an empty parking slot (4) (5). Even in developing countries, many types of fatal accidents are caused due to traffic congestion (6) (7). There are various studies on vehicle parking prices, and they have shown that a vehicle pays approximately 5 dollars or more, which is a huge amount of total travel cost (8) (9). The survey data of 2013 and 2015 states that on-street parking price on weekdays has increased from 2.00 USD to 4.40 USD per hour (10). Many researchers worldwide discussed the various parking price prediction schemes. Still, with the help of a centralized authority, which is vulnerable to various security issues such as data modification, spoofing, Man-in-the-middle (MitM) attack, etc. (11). Centralized authority can charge high parking prices to allocate a parking slot for users for their benefit (12) (13). These issues can deviate users from utilizing the parking slots leading to the loss of parking slot owners.
It is necessary to ensure security, privacy, and transparency while predicting the parking price so that users can book the parking slot at the optimum price. To mitigate the aforementioned issues, a blockchain-based trusted system is needed to ensure parking price security, confidentiality, and privacy (21) (22). Many researchers discussed the blockchain-based system to mitigate the security and privacy issues of the centralized authority. Some of the works are: Chai et al. (16) proposed a consortium-based blockchain model for resource sharing on the internet of vehicles using a consensus mechanism to ensure security in the system. Later, Syed et al. (17) presented a blockchain-based framework for vehicle tracking integrated with IoT devices.

Authors in (15) presented a blockchain-based layered architecture for a smart parking system to establish trust between users. Then, Hassija et al. (12) proposed a distributed parking slot allocation framework based on virtual voting and an adaptive pricing algorithm to allocate the parking slot for users in an optimal way. Later, Jog et al. (23) discussed a smart parking technology and automated parking to allocate parking space for users using different technologies such as wireless sensors networks, RFID technology, etc. Then, Simhon et al. (14) proposed a smart parking pricing system to predict the occupancy rate of a parking area using a machine learning approach.

Peyal et al. in (19) proposed a smart car parking system in urban areas with the help of IoT. The main focus is to reduce traffic congestion and make parking easy for users. Table 1 shows the comparative analysis of various state-of-the-art parking price prediction schemes with the PADaaS PADaaS with or without blockchain. Some of the solutions in (15) (16) (17) (19) given by the researchers are only focusing on enhancing the security of the system. Still, there is no emphasis on optimizing the parking prices for users, profit for vehicle owners, cost-efficiency, scalability, and computation time. Although there are some research works in (12) (14), which emphasize optimizing the parking price for users, they are not focusing on enhancing the security of the system.

Motivated by this, we propose a secure blockchain-based parking price prediction scheme using a second price auction model for sustainable traffic management. The second price auction model optimizes the price for users so that users who booked parking slots with the highest price get the parking slot, but they have to pay the second-highest price. Furthermore, the integrated IPFS with the proposed scheme mitigates the cost storage issues of the blockchain, which further enhances the scalability and reduces computation time and data storage cost of the system. So, users can book the parking slot at the optimized price, reducing traffic congestion, and parking slot owners can also benefit from it.

A. RESEARCH CONTRIBUTIONS

The research contributions of this paper are as follows:

• We employ a second price auction model to optimize the parking price for users so that parking slot owners can also get benefit from it.

• Finally, we estimate the performance of the PADaaS in terms of profit for users, profit for parking slot owners, revenue of the system, scalability, computation time, and data storage cost of the system.

B. ORGANIZATION OF PAPER

The rest of the paper is organized as follows. Section II discusses the system model and problem formulation. Then, section III shows the proposed model, and Section IV shows the performance evaluation based on the different auction models and blockchain-based results. Finally, Section V concludes the paper. TABLE 2 shows the various symbols and their description used in the blockchain-based proposed scheme using the second price auction model to predict optimum and cost-efficient parking prices for users.

II. SYSTEM MODEL AND PROBLEM FORMULATION

A. SYSTEM MODEL

FIGURE 1 shows a blockchain-based parking price prediction scheme (PADaaS) for sustainable traffic management. It consists of a set of users (Su), the set of parking slots (Sp), and the set of parking slot owners (Sp) communicating through distributed Ethereum network. Su can communicate with any parking slot owners to book a parking slot by spending an amount from their wallet. Sp can set the reserve price (rSp) so that users can book the parking slots with the bidding price of bp with the condition that the bidding price of user should be greater than the reserve price of the parking slot.

Users can book the parking slots assigned by the parking slot owners, but they should fulfill some conditions priority. However, before that, the transaction between users and parking slot owners about the parking allocation can be performed efficiently with the help of blockchain through the usage of IPFS. But, data storage in the blockchain tends to be costly due to its feature of storing a whole block of data. Therefore, an IPFS data storage protocol can be incorporated with the blockchain after executing the smart contract for authentication purposes. A smart contract can validate the user’s and parking slot owner’s data that needs to be stored in the IPFS. If a smart contract finds malicious data in the network, then data access can be denied, ensuring the system’s security and confidentiality. Now, certain conditions need to be discussed to allocate the parking slot to the user with a fair parking price. Firstly, users can bid for a parking slot in a network to check whether the current time (ζj) is greater than the bidding period (βp) or not. If ζj is greater than βp, then the second price auction model can be implemented directly. Then, after implementing the second price auction model, there is a condition to check, i.e., whether bp is greater than rp or not. If the bidding price(bp) is greater than the reserve price(rp), a smart contract can be executed to allocate the parking slot to the user. Implementing the smart contract
ensures the security in the system to allocate the parking slot to the users efficiently.

There can also be a condition that if $\zeta_t$ is less than $\beta_p$, then it has to check for the available parking slots, i.e., whether the number of parking requests ($N_{pr}$) is greater than a number of available parking slots ($N_{as}$) or not. If $(N_{pr}) > (N_{as})$, then a second price auction model can be implemented to allocate available parking slots to the user. We have to check one more scenario in which if any additional parking slots need to be added; then a smart contract can be executed to add the additional parking slots with its reserve price and count, i.e., $(r_p, c)$. Finally, different conditions can be considered to allocate the available parking slot to the user and add any available parking slot in the blockchain network through the intermediary IPFS for improved cost-efficiency.

### TABLE 1: A relative comparative analysis of various state-of-the-art parking price schemes with the PADaaV

| Ref  | Year | Objective | Blockchain Adoption | Pros | Cons |
|------|------|-----------|---------------------|------|------|
| (14) | 2017 | Proposed a smart parking pricing system using machine learning approach | No | Improved accuracy, optimized price | No emphasis on security against malicious attacks |
| (15) | 2018 | Presented a blockchain-based layered architecture for a smart parking system | Yes | Security against DDOS attacks | Security issues against cyber attacks, No focus to optimize parking price |
| (16) | 2019 | Proposed a consortium-based blockchain model in IoT | Yes | Security against double spending attacks and malicious attacks | No efforts to optimize the parking price |
| (17) | 2020 | Investigated a blockchain-based framework for vehicle tracking with IoT devices | Yes | Improved efficiency, security against single point of failure | No focus to optimize the parking price, security issues against DDOS attacks |
| (12) | 2020 | Proposed a distributed parking slot allocation framework to allocate the parking slot for users | No | Optimized parking price, secure against DDOS attacks | No emphasis on security issues against cyber attacks |
| (18) | 2020 | Designed a vehicle parking management integrated with the blockchain and computer vision | Yes | Improved accuracy, secure against bottleneck attack | high data storage cost, no focus on scalability and optimized parking price |
| (19) | 2021 | Proposed a smart car parking system in urban areas with IoT to make the parking easy for users | No | Reduced traffic congestion, communication in real time | No focus to optimize the parking price, no user friendly environment, security issues against malicious attacks |
| (20) | 2021 | Investigated a blockchain and AI-enabled dynamic parking pricing scheme for an efficient parking | Yes | Fair parking price, reduced traffic congestion | Need to focus on optimizing the parking price |
| PADaaV | 2022 | Proposed a secure blockchain-based parking price prediction scheme | Yes | Optimized parking price, highly secure, and reliable | - |

### TABLE 2: Symbols and their description

| Symbols | Description | Symbols | Description |
|---------|-------------|---------|-------------|
| $N_{as}$ | Set of users | $S_{u}$ | Number of available parking slots |
| $P_{price_u}$ | Set of parking slots | $\Psi$ | Wallet |
| $P_{price_u}$ | Set of parking slot owners | $\beta_p$ | Price for reserving a parking slot |
| $b_p$ | Reserve price | $\zeta_t$ | amt Bidding amount of user |
| $\delta_u$ | Bidding price | $\psi$ | Residual amount |
| $\zeta_t$ | Count of additional parking slots | $\gamma$ | Space Additional parking slot |
| $V_{c}$ | Current time | $\gamma$ | Valuation of the parking slot |
| $\beta_p$ | Bidding period | $\gamma$ | Profit for parking slot owner |
| $N_{pr}$ | Number of parking requests | $P_{price_u}$ | Profit for the user |
| $G_{KB}$ | Gas price for 1KB of data | $E_{2^{P^{}}}$ | Ethereum price |
| $G_{S_{u}}$ | Standard Gas price | $C_{os}\gamma^{USD}$ | Cost in USD |

where \( \alpha \) signifies the relationship between \( p \) number of users and \( t \) number of parking slot owners.

Users \( u_k \) can bid to reserve a parking slot in a distributed network. For that, there is a condition to check if \( \zeta_t \) is greater than the \( \beta_p \) or not. A smart contract can be implemented to allocate the parking slot based on the second price auction model if the condition is satisfied. In the second-price auction

\[
\Psi_{u_k} \xrightarrow{\rho_h} \Psi_{u_o},
\]

\[
u_k(b_{p_u}) \xrightarrow{\alpha} \sum_{o=1}^{t'} q_{o}, \quad t' < t,
\]

\[
q_{o}(r_{p}) \xrightarrow{\alpha} \sum_{k=1}^{p'} u_{k}, \quad p' < p.
\]

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3173162, IEEE Access

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
model, if \( b_p > r_p \), a parking slot can be assigned to the user by executing a smart contract.

But there can also be a possibility that if \( \zeta_t \) is less than the \( \beta_p \), then we have to verify another condition, i.e., if \( N_{pr} > N_{as} \), then second-price auction model can be executed to allocate the parking slot to the user. These associations can be represented as follows:

\[
\eta = \begin{cases} 
\pi, & \text{If current time greater than bidding period} \\
\chi, & \text{If current time less than bidding period} 
\end{cases}
\]

\[
\chi = \{(\zeta_t, \beta_p) > (N_{pr}, N_{as}) > (b_p, r_p)\}
\]

\[
u_k(b_{pu}, \pi) \xrightarrow{\gamma} \sum_{h=1}^{r''} \rho_h(r_{pu}), \quad r'' < r,
\]

\[
u_k(b_{pu}, \chi) \xrightarrow{\gamma'} \sum_{h=1}^{s'} \rho_h(r_{pu}), \quad o' < r,
\]

where \( \eta \) and \( \chi \) denotes the \( r'' \) and \( o' \) number of parking slots assigned to the users based on the current time and bidding period, so users can get the parking slot based on the different conditions by executing the smart contract using the second price auction model. We have introduced the second price auction model so that users with the highest \( b_p \) can get the parking slot, but they have to pay the second-highest price (\( Price_{u_k} \)), which can be represented as follows:

\[
\sum_{k=1}^{p''} Price_{u_k} \xrightarrow{\omega} \sum_{u=1}^{s'} b_{pu+1}.amt, \quad p'' < p, \quad s' < s
\]

where \( amt \) denotes bidding amount of \( u + 1^{st} \) user. \( \omega \) signifies the \( p'' \) prices users have to pay to book the parking slots associated with \( s' \) number of bidding prices. In \( PADaaV \), the allocation scheme should be implemented in such a way that the prediction of parking price should be in favor of users so that it also encourages them to bid high. After allocating the parking slot to a user, all other users participating in reserving the parking slot can get their residual amounts (\( \delta_{uk} \)) back. It
can be represented as follows:

$$
\sum_{k=1}^{n} b_{uk} \xi \rightarrow \sum_{u=1}^{n'} b_{pu}.amt - \sum_{k=1}^{n''} Price_{uk}
$$

where $\xi$ denotes the association between $v$ number of users to get their residual amount back. But, if the user is not allocated any parking slot due to the unavailability of the parking slot of the low bid, then the amount can be returned to the user. If there is any available parking slot that has been made unreserved by the user, then a smart contract can be executed to add the available parking slot with a reserve price and count, i.e., $\rho_h(r_{pu}, c)$.

$PA\text{DAaV}$ focuses on allocating the parking slot to the users efficiently and maximizing the revenue of the parking slot owners. Choosing an appropriate allocation scheme is required to satisfy the users to get the parking slot at an optimized price. So, implementing an allocation scheme with a second price auction model is advantageous for users to reserve a parking slot with the second-highest price, i.e., less than their bidding price. The allocation has been executed properly so that if users don’t get the parking slot, then their residual amount can be returned. So, $PA\text{DAaV}$ is beneficial for users and parking slot owners.

### III. $PA\text{DAaV}$: THE PROPOSED MODEL

The price to reserve a parking slot needs to be predicted for the users, parking slot owners, and parking slots in a distributed network. Users can reserve the parking slots available by the parking slot owners in the blockchain network. Blockchain and IPFS protocol implemented with the second price auction model plays an important role in storing the transactions of users and parking slot owners in a secure and cost-efficient way. However, there is a major difference between blockchain and IPFS in data storage cost. As IPFS incorporated with the blockchain seems beneficial for users due to its feature to store hash data instead of storing the whole block of data. Furthermore, this section proposes algorithms to efficiently allocate the parking slots to the users.

#### A. DYNAMIC PRICING ALLOCATION SCHEME

A dynamic parking pricing scheme is elaborated to allocate the parking slot to the users efficiently. FIGURE 2 shows a blockchain-based parking price prediction scheme, $PA\text{DAaV}$, in which a second price auction model is implemented to allocate the price to the users efficiently in a distributed network. The proposed second-price auction model works based on the concept of a sealed-bid auction in which users do not know about the other users’ bidding price, which maintains the privacy and security of the system. Furthermore, a smart contract is used to run the dynamic pricing allocating scheme based on the second pricing model so that users can get the parking slots at the optimized price according to their bid. Parking slot owners can add the parking slot back to the parking slots if they have been made unreserved by the user by submitting the request to the smart contract.

FIGURE 2 depicts that the multiple parking slots and multiple users are connected in the network. There are $r$ parking slots $\rho_h$ for which $p$ users can bid $b_{pu}$ in the distributed network. We have considered an example in which, initially, two users are bidding with bidding prices $\{b_{p1}, b_{p2}\}$ to reserve a parking slot. Firstly, user bids have been collected so that these bids can be compared for allocating the parking slot.

#### Algorithm 1: Dynamic Pricing Algorithm

**Data:** $n$ users, $m$ parking slots.

**Result:** Output: Notify the allotment of parking slot to a user.

1. $B = \{b_{p1}, b_{p2}, \ldots, b_{pn}\}$;
2. $L = \{\rho_1, \rho_2, \ldots, \rho_m\}$;
3. $\beta_p$ is the duration for which bids are accepted;
4. $b_{pi}$ record contains two fields $b_{pi}.amt$ and $b_{pi}.bidder$, which are the bids and address of a bidder respectively;
5. $\rho_j$ record contains three fields $\rho_j.owner$, $\rho_j.rp$, $\rho_j.num$ which are the address of the owner, reserve price and number of available parking slots;
6. **Procedure** $ALLOCATE\; (p, b_{pi}, n, m, \text{Auction\_end\_time}, \beta_p)\; if\; (Current\_time < \text{Auction\_end\_time}\; and\; n < m)$ then

   7. “The auction has not ended yet and exit” Return
   8. end
   9. count $\leftarrow 0$
   10. $j \leftarrow 1$
   11. for $i = 1, 2, \ldots, n$ do
       12. if ($i < n$) then
           13. Price$_{uk} \leftarrow b_{p_{i+1}}.amt$
           14. else
           15. Price$_{uk} \leftarrow b_{p_i}.amt$
       end
       16. while ($j <= m$ and $\rho_j.rp > Price_{uk}$)
       17. or($\rho_j.num == 0$)) do
           18. $j \leftarrow j + 1$
       end
       19. if ($j > m$) then
           20. $b_{pi}.bidder.transfer\_the\_money(b_{pi}.amt)$
           21. else
           22. $\rho_j.num --$
           23. $c.owner.transfer\_the\_money(Price)$
           24. $b_{pi}.bidder.transfer\_the\_money(b_{pi}.amt - Price_{uk})$
           25. “ith User allocated jth Parking Location”
       end
       26. if($\rho_j.num == 0$) count $\leftarrow 1$
   end
   29. end
30. $n \leftarrow 0$
end

$\text{Auction\_end\_time} \leftarrow (\text{current\_time} + \beta_p)$

end Procedure
In the next step, the second price auction model has been implemented as an allocation scheme with the help of a smart contract based on the various conditions.

In this section, various algorithms are proposed, which decide that users bidding for multiple parking slots pay the second-highest price according to the second price auction model, which is less than their bidding price. We have compared it to other approaches like the first-price auction model, in which the user has to pay more price than the second price auction model. The second price auction model helps maximize parking slot owners’ revenue and ensures that users get the parking slot at an optimized price conveniently. Moreover, the parking slot owner can set the reserve price to be added to the blockchain network can be represented as

\[ \text{Result: Output: m number of parking slots available} \]

\[ \text{owner is the address of Owner of parking slot;} \]

\[ \rho \text{ is the sequence of records of m parking slots in descending order of } r_p; \]

\[ \text{L = \{ } \rho_1, \rho_2, \ldots, \rho_m \}; \]

\[ \rho_j \text{ record contains three fields } \rho_j.\text{owner}, \rho_j.r_p, \rho_j.num \text{ which are the address of the owner, reserve price and number of available parking spaces;} \]

\[ \text{Procedure PARKING_Lot(m, } \rho, r_p, c, \text{ owner }) \]

\[ s \leftarrow \text{Space(owner, } r_p, c \text{)} \]

\[ \text{if } (j <= m \text{ and } s.r_p <= \rho_j.r_p) \text{ do} \]

\[ j = j + 1; \]

\[ \text{end while} \]

\[ \text{end procedure} \]

### B. EFFECT ON THE PRICE DUE TO BIDS

The number of vehicles is increasing on the road. However, limited infrastructure, uncontrolled externalities, and varying traffic conditions make predicting parking demand at a particular time slot challenging based on the past condition. The use of an auction-based model helps accurately reveal the demand at the particular instance. In contrast to a single price auction model, the highest bidder wins. The highest bidder wins in the second price auction model but pays the second-highest bid amount, as shown in equation 8.

Therefore, to win, the bidders are motivated to bid the highest amount while still ensuring that they are not paying an amount much higher than others. If a user is not allotted a parking slot due to the low bid’s unavailability of the parking slot, then the amount can be returned to the user.

Algorithm 2: Adding additional parking slots

Data: \( r_p, c \)

Result: Output: m number of parking slots available

1. \( \text{owner is the address of Owner of parking slot;} \)
2. \( \rho \text{ is the sequence of records of m parking slots in descending order of } r_p; \)
3. \( L = \{ \rho_1, \rho_2, \ldots, \rho_m \}; \)
4. \( \rho_j \text{ record contains three fields } \rho_j.\text{owner}, \rho_j.r_p, \rho_j.num \text{ which are the address of the owner, reserve price and number of available parking spaces;} \)
5. Procedure PARKING_Lot(m, \( \rho \), \( r_p \), \( c \), \text{ owner }) \)
6. \( s \leftarrow \text{Space(owner, } r_p, c \text{)} \)
7. \( j = 1 \)
8. \( \text{while } (j <= m \text{ and } s.r_p <= \rho_j.r_p) \text{ do} \)
9. \( j = j + 1; \)
10. \( \text{end while} \)
11. \( \text{end procedure} \)

Algorithm 2 explains how we can add a parking slot with its \( r_p \) and \( c \) in the blockchain network. It defines a utility function for entering new parking slots on the blockchain network. The utility function adds the additional parking slots corresponding to a parking slot owner with a count and reserve price accordingly. It is also used to add the parking slots back once freed from a user. The time complexity of algorithm 2 turns out to be \( O(m) \) where \( m \) is the number of parking slots present in the sequence, and It has \( O(1) \) space complexity, but the array to store \( m \) parking slots require \( O(m) \) space. The additional parking slots that needs to be added to the blockchain network can be represented as follows:

\[ b_{pu_k} = b_{pu}.\text{bidder.transfer_the_money}(b_{pu}.amt), \quad (10) \]

\[ \delta_{uk} = b_{pu}.\text{bidder.transfer_the_money}(b_{pu}.amt-Price_{uk}) \quad (11) \]


\[ \text{Space}(\text{owner}, r_p, c) \leq \sum_{h=1}^{x} \rho_h, \ x < r \] (12)

Algorithm 3: Submitting bids for parking slots

**Data:** \( u_k \), amount bid by user  
**Result:** \( n \) number of bidders

1. **Procedure** ENTER_BID(b, n, u_k, map, amount)  
2. if (map[u_k].isPresent and amount <= map[u_k].bid) then  
   3. “LOW BID and EXIT”  
   4. return n  
3. if (map[u_k].isPresent) then  
   4. \( u_k \).transfer_the_money(map[u_k].bid)  
   5. loc = map[u_k].pos  
   6. map[u_k].bid = amt  
   7. \( b_{ploc}.amt \leftarrow \text{amount} \)  
   8. for (i = loc + 1, \ldots, n) do  
   9. if (\( b_{p_{i-1}}.amt < b_{pi}.amt \)) then  
      10. break  
   11. end  
   12. swap(b_{pi - 1}, b_{pi})  
   13. end  
   14. map[u_k].pos = i - 1  
15. else  
16. \( b \leftarrow \text{Bid}(u_k, \text{amount}) \)  
17. i = i + 1  
18. while (i <= n and b.amt <= b_{pi}.amt) do  
   19. \( i = i + 1 \)  
20. end  
21. for (i = 1, \ldots, m) do  
   22. swap(b, b_{pi})  
23. end  
24. \( b_{pi}.push(b) \)  
25. \( n = n + 1 \)  
26. end  
27. Return n  
end procedure

Algorithm 3 explains the procedure of accepting bids in which the users are bidding to reserve a parking slot. It defines a utility function for accepting bids of users within the given \( \zeta_p \). Users can resubmit their bids within the \( \zeta_p \); for that, the newly submitted bid must be greater than the previous ones. The function, i.e., \( \text{map[user]} \), helps to track the bids already made by the user in the distributed blockchain network. The use of maps helps reduce the gas consumption for searching the presence of users and checking the validity of bids, which further reduces the gas costs. Now, in case the user resubmits the bid, then bid \( b_{ploc}.amt = \text{amount} \) is updated and the bids are reordered in the \( b_p \) sequence, which takes linear time \( O(n) \). In the case of a new bid, the bid is inserted at an appropriate position as shown in Algorithm 3, which leads to the linear time complexity of \( O(n) \) for accepting the bids.

In a smart contract, the use of bidding price \( b_p \) in descending order and parking slots \( \rho_h \) in descending order helps avoid extra gas costs due to additional memory for sorting methods like merge sort. It also helps avoid the time complexity of \( O(n\log n) \) or \( O(n^2) \) for the allocation process in Algorithm 1. It gives a better time complexity of \( O(n) \) in the allocation of parking slots and it has \( O(1) \) space complexity for the number of bids. To avoid costly gas loops, we need to maintain a map to track the already submitted bids.

![FIGURE 3: Net Profit for users from different models.](image)

**IV. PERFORMANCE EVALUATION BASED ON THE DIFFERENT AUCTION MODELS AND BLOCKCHAIN**

In this section, the performance evaluation of the dynamic pricing allocation scheme is analyzed based on the different auction models such as fixed price, first-price auction model with the proposed second-price auction model. We have also analyzed the performance of the blockchain-based proposed scheme in terms of scalability, computation time, and data storage cost of the system. The evaluation of the results has been analyzed in Remix integrated development environment (IDE) to show the improved and cost-efficient parking slot...
booking in favor of users and parking slot owners. The results based on the different auction models and blockchain, along with the simulation, can be represented as follows:

**A. SIMULATION**

1) Dataset Generation and Initial Experiment Setup

The analysis of the proposed scheme has been simulated over the Remix IDE (25). The smart contracts of the proposed scheme are developed, compiled, run, and tested over the Remix IDE using solidity source code (26). The experiment has been simulated for 100 users with 40 parking slots. We have randomly generated a valuation of the parking slot for a user corresponding to his purchasing power using a Gaussian distribution with a mean of 50 and a variance of 15 (30% of the mean). In addition, the reserve prices of the parking slots have been randomly generated with a mean of 40 and a variance of 10 (25% of the mean). The second price auction model simulation has been performed in python 3.2 with the help of generated dataset for 100 users and 40 parking slots, along with the generated reserve prices of the parking slot.

2) Bids and Prices in Different Models

The second price auction model is a revealing demand model. Therefore, bids of a parking slot by a user in a second price auction model have been assumed to be equal to the valuation of the parking slots. In a first-price auction model, users do not reveal their true value. So, a bid of a user, which is less than valuation, has been randomly selected. The simulation results in the first pricing model have been averaged over multiple iterations.

A profit rate of 15% over the reserve price has been assumed for determining the price in simulating a fixed price model. The prices for a parking slot in a first-price and second-price auction model have been determined based on the auction allocation scheme. A user with the highest bid is allocated a parking slot with the highest price in the auction allocation scheme. In a second price auction model, the price of a parking space is the second-highest bid. In the first-price auction model, the price of a parking space is the bid’s value.

3) Evaluation of profit based on different auction models

Profit for a user can be defined as follows:

- Profit for the user \( P_{u_k} = 0 \), if a user is not allocated a parking slot.
- Profit for the user \( P_{u_k} = V_{p_k} - Price_r \), if user is allocated a parking slot.

where \( Price_r \) is the price of the \( r^{th} \) parking slot allocated based on the different models. Profit for users is relatively less when they are quoted a price closer to their valuations. As auction-based models help reflect consumer demand at a time, thus it helps in driving the prices up, which are closer to the users’ valuation. As the second price auction model is revealing demand mode, the bids in a second price auction model are closer to the users’ valuation. Therefore, driving the prices of the parking slots high, the net profit for all users is the least in the second price auction model.

**Profit for parking slot owners:** Profit for a \( r^{th} \) parking slot can be defined as follows:

- Profit for \( r^{th} \) parking slot \( P_{r_p} = 0 \), if the parking slot is not allocated to a user.
- Profit for \( r^{th} \) parking slot \( P_{r_p} = Price_r - r_p \), if user is allocated a parking slot

where \( Price_r \) is the price of the \( r^{th} \) parking slot allocated based on the different models. Higher prices help increase the net profit for parking slot owners. As the prices go high in a second-price auction model, parking slot owners have the highest net profit. As discussed in the previous scenario, the auction-based models help drive the prices up, thus reducing the profit for users. It helps transfer the profit from the users to the owners of the parking slots. As the profit for users is least in the second price auction model, the net profit for parking owners is highest according to the second price auction model.

4) Revenue

Revenue can be defined as follows:

- Revenue = \( Price_r \times x \)
- \( x = 0 \), if the \( r^{th} \) parking slot is not allocated.
- \( x = 1 \), if the \( r^{th} \) parking slot is allocated to a user.

where \( Price_r \) is the price of the \( r^{th} \) parking slot allocated based on the different models. As the prices of the allocated parking slots increase, the system’s total revenue increases. As previously mentioned, the prices are higher in a second price model. Thus, it has the highest revenue among all the auction models.

5) Confidence Interval for First Price

The first price auction model has calculated the parameters, i.e., revenue and profit. As in a first-price auction model, the bids submitted by the users are not equal to true bids, i.e., true valuations \( V_{p_k} \). So, we have randomly generated bids for the users corresponding to the valuations. The different parameters have been calculated for 25 samples and their corresponding 95% confidence intervals. The red horizontal line represents the mean, i.e., values for different parameters.
Kakkar et al.: PADaaV: Blockchain-based Parking Price Prediction Scheme for Sustainable Traffic Management

FIGURE 6: Confidence interval for the first price

(a) Confidence Interval for various samples

(b) Confidence Interval for profit for users for various samples

(c) Confidence interval for profit for owners from various samples

FIGURE 7: Scalability analysis.

FIGURE 8: Security analysis of the PADaaV over smartcheck tool

between the proposed scheme with IPFS and blockchain data storage in terms of scalability. It can be perceived from the graph that the scalability of the proposed scheme with IPFS tends to provide more efficient data storage than the usage of blockchain as the number of transactions increases. However, the graph shows that the scalability seems to be in the same alignment due to the involvement of a lesser number of transactions between users and parking slot owners for both the data storage.

7) Computation time
This section discusses the computation time of the proposed scheme to store and retrieve parking allocation data of users and parking slot owners from the blockchain. Foremost, the computation time of uploading the data in the blockchain directly after authentication with a smart contract can be determined in 80ms (27). However, the computation time for users and parking slot owners involved to upload data to the blockchain with the usage of IPFS can be calculated in 110ms (28). Now, it can be observed that IPFS consumes slightly higher computation time to store data than blockchain due to the involvement of an intermediary storage protocol. However, usage of IPFS protocol with its feature of hash data storage reflects a quite low data retrieval time than blockchain, which is being discussed further. As data retrieval time of blockchain incorporated with IPFS or without IPFS should be calculated to show the benefits of the low data storage cost protocol. Therefore, the computation time involved in the data retrieval from the blockchain for users and parking slot owners can be determined as $6 \times 10^5$ms (29), with the help of the SHA-256 algorithm. On the other hand, the computation time of the data retrieval from blockchain through an intermediary IPFS is computed as 75ms (27), which seems to be quite improved and low than the traditional schemes with blockchain.

8) Data storage cost
In this section, we have considered the IPFS with the Ethereum blockchain to obtain the efficient data storage cost calculated by the first price auction model. As shown in FIGURE 6, means lie in most of the confidence intervals of different samples. Thus, the values calculated by the first price auction model help predict the simulation results accurately.

6) Scalability
The usage of IPFS with the blockchain-based proposed scheme proves to be cost-efficient for the users and parking slot owners involved in the system. FIGURE 7 distinguishes

VOLUME 4, 2016
so that users and parking slot owners can access the IPFS for data storage after validation with the smart contract. The cost associated with the blockchain to store data is somewhat high, making the system unfavorable for the users and parking slot owners. We can find the cost of blockchain with the help of the gas price of a single word of 256 bits. The gas price for the single word and 1KB of data can be computed as:

\[
1_{kw} = 20,000\ Gas(G_s = 20K) \\
G_{KB} = (2^{10}/256) \times 20,000\ Gas
\]

Furthermore, cost \((Cos^P)\) for data storage of \(P\) number of words in blockchain can be determined with the parameters standard gas price \((G_{s^{PR}})\) and Ethereum price \((Et^{PR})\) which are 23.186 \(gwei\) and 232.96 USD. The cost for \(P\) words can be calculated considering the two:

\[
1E^f = 10^g\ gwei\ and\ Exp = (p * G_s)/10^9
\]

Therefore, the computation of cost \((data\ storage\ of\ P\ words\ in\ USD)\) can be written as follows:

\[
Cos^{P/us} = (G_{s^{PR}} * Cos_{P}) * Et^{PR}
\]

Finally, it can be concluded that blockchain incurs high data storage costs than incorporated IPFS with the blockchain. Therefore, the blockchain-based proposed scheme with the IPFS tends to cost optimization for users and parking slot owners’ data storage (30).

**B. SECURITY VERIFICATION OF THE PADAAV**

In this section, the smartcheck security tool has been used for security analysis of the smart contracts designed for the PADaaV. We have implemented the source code from the Ethereum platform to detect the security-related issues in the PADaaV. FIGURE 8b depicts how the smartcheck tool verifies that the source code of the PADaaV does not contain any threat or vulnerability. But, smartcheck tool generates one severity in the source code by default as depicted in the FIGURE 8a (31). We have removed that identified severity from the source code as depicted in FIGURE 8b.

**V. CONCLUSION**

In this paper, a blockchain-based parking price prediction scheme, PADaaV, is proposed for sustainable traffic management. First, we explored the traditional systems with or without blockchain to get insights into their security, privacy, and optimized parking price issues. Then, we employ the second price auction model to predict the optimized price for users to reserve a parking slot conveniently and parking slot owners can also benefit from it. We have performed the security analysis of the PADaaV by implementing source code on Remix IDE using a smart contract. Finally, the performance of the PADaaV has been evaluated in terms of profit for users, profit for parking slot owners, and overall revenue of the system considering different auction models and blockchain-based results in terms of scalability, computation time, and data storage cost. Results show that the proposed scheme is more secure, cost-efficient, and beneficial for users and parking slot owners than the traditional schemes.

In the future, we will explore the scenario in which multiple bidders bid the same amount simultaneously to reserve the parking slot with the optimized price using game-theoretical aspects.

**REFERENCES**

[1] D. Reebadiya, R. Gupta, A. Kumari, and S. Tanwar, “Blockchain and ai-integrated vehicle-based dynamic parking pricing scheme,” in 2021 IEEE International Conference on Communications Workshops (ICC Workshops), pp. 1–6, IEEE, 2021.

[2] G. Dalla Chiara, K. F. Krutein, A. Ranjbari, and A. Goodchild, “Understanding urban commercial vehicle driver behaviors and decision making,” Transportation Research Record, p. 03611981211003575, 2021.

[3] A. H. Schrotenboer, M. A. Broek, P. Buijs, and M. W. Ulmer, “Fighting the e-commerce giants: Efficient routing and effective consolidation for local delivery platforms,” arXiv preprint arXiv:2108.12608, 2021.

[4] C. Dowling, T. Fiez, L. Ratliff, and B. Zhang, “How much urban traffic is searching for parking,” arXiv preprint arXiv:1702.06156, 2017.

[5] B. Li, F. Hou, H. Ding, and H. Wu, “Community based parking: Finding and predicting available parking spaces based on the internet of things and crowdsensing,” Computers & Industrial Engineering, vol. 162, p. 107755, 2021.

[6] S. Rajendran and B. Ayyasamy, “Short-term traffic prediction model for urban transportation using structure pattern and regression: an indian context,”

[7] V. Kebande, F. Awaysheh, I. Adeyemi, S. Alawadi, and M. Alshehri, “A blockchain-based multi-factor authentication model for cloud-enabled internet of vehicles,” Sensors, 09 2021.

[8] Z. S. Qian and R. Rajagopal, “Optimal parking pricing in general networks with provision of occupancy information,” Procedia-Social and Behavioral Sciences, vol. 80, pp. 779–805, 2013.

[9] R. Kakkar, R. Gupta, S. Tanwar, and J. J. P. C. Rodrigues, “Coalition game and blockchain-based optimal data pricing scheme for ride sharing beyond 5g,” IEEE Systems Journal, pp. 1–10, 2021.

[10] J. Lee, D. Agdas, and D. Baker, “Cruising for parking: New empirical evidence and influential factors on cruising time,” Journal of Transport and Land Use, vol. 10, no. 1, pp. 931–943, 2017.

[11] A. Kumari, R. Gupta, S. Tanwar, and N. Kumar, “A taxonomy of blockchain-enabled softwarization for secure uav network,” Computer Communications, vol. 161, pp. 304–323, 2020.

[12] V. Hassija, V. Saxena, V. Chamola, and R. Yu, “A parking slot allocation framework based on virtual voting and adaptive pricing algorithm,” IEEE Transactions on
Kakkar et al.: PADaaV: Blockchain-based Parking Price Prediction Scheme for Sustainable Traffic Management

Vehicular Technology, 2020.

[13] M. Chen, T. Malook, A. Rehman, Y. Muhammad, M. Alshehri, A. Akbar, M. Bilal, and M. Khan, “Blockchain-enabled healthcare system for detection of diabetes,” Journal of Information Security and Applications, vol. 58, p. 102771, 02 2021.

[14] E. Simhon, C. Liao, and D. Starobinski, “Smart parking pricing: A machine learning approach,” in 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 641–646, IEEE, 2017.

[15] O. Abdulkader, A. M. Bamhdi, V. Thayananthan, K. Jambi, and M. Alrasheedi, “A novel and secure smart parking management system (spms) based on integration of wsn, rfid, and iot,” in 2018 15th Learning and Technology Conference (L&T), pp. 102–106, IEEE, 2018.

[16] H. Chai, S. Leng, K. Zhang, and S. Mao, “Proof-of-reputation based-consortium blockchain for trust resource sharing in internet of vehicles,” IEEE Access, vol. 7, pp. 175744–175757, 2019.

[17] T. A. Syed, M. S. Siddique, A. Nadeem, A. Alzahrani, S. Jan, and M. A. K. Khattak, “A novel blockchain-based framework for vehicle life cycle tracking: An end-to-end solution,” IEEE Access, vol. 8, pp. 111042–111063, 2020.

[18] E. Sammut, F. Inguanez, and A. Cortis, “Automating car park management with blockchain and computer vision,” in 2020 IEEE 10th International Conference on Consumer Electronics (ICCE-Berlin), pp. 1–6, 2020.

[19] M. M. K. Peyal, A. Barman, T. Tahiat, Q. M. A. U. Haque, A. Bal, and S. Ahmed, “Iot based cost effective car parking management for urban area,” in 2021 4th International Symposium on Agents, Multi-Agent Systems and Robotics (ISAMSR), pp. 70–75, IEEE, 2021.

[20] D. Reebadiya, R. Gupta, A. Kumari, and S. Tanwar, “Blockchain and ai-integrated vehicle-based dynamic parking pricing scheme,” in 2021 IEEE International Conference on Communications Workshops (ICC Workshops), pp. 1–6, 2021.

[21] R. Gupta, A. Kumari, and S. Tanwar, “A taxonomy of blockchain envisioned edge-as-a-connected autonomous vehicles,” Transactions on Emerging Telecommunications Technologies, vol. 32, no. 6, p. e4009, 2021.

[22] P. Zhang, Y. Hong, N. Kumar, M. Alazab, M. Alshehri, and C. Jiang, “Be-edgefl: Defensive transmission model based on blockchain assisted reinforced federated learning in iot environment,” IEEE Transactions on Industrial Informatics, vol. PP, pp. 1–1, 09 2021.

[23] Y. Jog, A. Sajeev, S. Vidwans, and C. Mallick, “Understanding smart and automated parking technology,” International Journal of u-and e-Service, Science and Technology, vol. 8, no. 2, pp. 251–262, 2015.

[24] J. Kim, H. Park, G. Ho Lee, J. Kyun Choi, and Y. Heo, “Seal-bid renewable energy certification trading in power system using blockchain technology,” in 2020 International Conference on Information and Communication Technology Convergence (ICTC), pp. 1752–1756, 2020.

[25] I. Remix-Solidity, “https://remix.ethereum.org/,” Accessed: 7 Dec, 2021.

[26] R. Gupta, A. Shukla, and S. Tanwar, “Bats: A blockchain and ai-empowered drone-assisted telesurgery system towards 6g,” IEEE Transactions on Network Science and Engineering, vol. 8, no. 4, pp. 2958–2967, 2021.

[27] L. Eisenring, “Performance analysis of blockchain off-chain data storage tools,” Bachelor Thesis, University of Zurich Department of Informatics (IFI), 2018.

[28] E. Nyaletey, R. M. Parizi, Q. Zhang, and K.-K. R. Choo, “Blockipfs - blockchain-enabled interplanetary file system for forensic and trusted data traceability,” in 2019 IEEE International Conference on Blockchain (Blockchain), pp. 18–25, 2019.

[29] B. Shrimali and H. B. Patel, “Blockchain state-of-the-art: Architecture, use cases, consensus, challenges and opportunities,” Journal of King Saud University-Computer and Information Sciences, 2021.

[30] R. Gupta, A. Nair, S. Tanwar, and N. Kumar, “Blockchain-assisted secure uav communication in 6g environment: Architecture, opportunities, and challenges,” IET Communications, 2021.

[31] A. López Vivar, A. T. Castedo, A. L. Sandoval Orozco, and L. J. García Villalba, “An analysis of smart contracts security threats alongside existing solutions,” Entropy, vol. 22, no. 2, p. 203, 2020.