ParkBid: An Incentive Based Crowdsourced Bidding Service for Parking Reservation

Shahid Noor, Ragib Hasan, and Arsh Arora
{shaahid, ragib, ararora}@uab.edu
Department of Computer and Information Sciences, UAB, AL 35294-1170

***Abstract*—Finding parking at a desired spot and time is tough, especially in an urban area. Users may also want to pay more to get a parking spot during urgent need. Although the existing infrastructure-based approaches can solve this problem partially, they require a high initial investment and maintenance cost. As a consequence, deploying such approaches on a large scale in the real world is infeasible. A more economically feasible solution is using crowdsourcing-based approaches where a user near a free parking spot informs the interested users about the available parking spot in exchange for some forms of incentives. However, most of the crowdsourced approaches suffer due to the lack of proper models for price negotiation, information verification, and assurance. In this paper, we propose ParkBid, a crowdsourcing-based parking service for automobiles where the information of free parking is circulated among the interested users following a bidding process. ParkBid determines the incentives for providing any parking information based on several primitives, such as time, location, reputation, urgency, etc. Along with a detailed discussion about the challenges in ParkBid for both the users and information providers, we provide a set of policies as countermeasures. Also, we present an extensive simulation to evaluate the impact of ParkBid for users under different circumstances. Our experimental results show that users can save a significant amount of time and can have more success rate during searching for a free parking spot using ParkBid.**

***Keywords*-Crowdsourcing, Parking, Mobile Sensing**

I. INTRODUCTION

In any urban environment, finding a parking spot in the desired location and time is a major challenge for commuters. Traffic congestion combined with the lack of enough parking spaces can create problems, especially during peak hours. Moreover, lack of public transport in many cities and the increase in the usage of private automobiles have made the situation worse. Research on parking shows that around 40% of the total traffic intensity in the major cities is due to the congestion caused by the vehicles trying to find a parking spot [1]. In addition to the increase in environmental pollution [2] and fuel cost [3], the low-speed cruising of vehicles also leads to road rage and accidents in some cases [4]. To resolve the ever-increasing parking problem, researchers have proposed infrastructure-based approaches [5, 6] where the potential parking places are monitored by previously installed smart devices, such as parking occupancy sensors, camera, radar, etc. The devices continuously collect data and upload the data to a central system for processing. When a user requests for a parking spot, the central system provides the users information of available parking spot based on the processed data and the users' preference. On the other hand, some researchers proposed crowdsourcing-based approaches [7–11] where people close to any parking lot are requested to provide information about any available parking spot in exchange of receiving some forms of incentives. The user seeking a parking spot can select a parking spot from the parking pool information provided by the crowd.

Some of the ongoing research projects, such as OpenSpot [12], PrimoSpot [13], SFPark [14], Smart Parking Solution [15], and Parking Spotter [16] recently released their products for assisting users regarding available parking spots. All of these projects are either infrastructure or crowdsourced based. So far, none of these projects have achieved large-scale commercial deployment [17]. The infrastructure-based approaches are unrealistic to implement at a large scale because of the high cost of installing and maintaining sensor devices. Moreover, none of the infrastructure-based approaches can predict which parking spots are going to be free in future. On the other hand, we identified several key requirements of users or information providers that were not addressed in any of the existing crowdsourcing-based parking solution research works. *First*, in all the existing crowdsourced approaches, the incentive is provided to the information provider as soon as a user receives the information. Most of the users hesitate to buy the information because there is no way to get a refund in case the information provider is dishonest or some other vehicle takes the parking spot before the commuter reaches the location. *Second*, none of the existing crowdsourced approaches provide any on-site verification mechanism to authenticate the information provided on the parking spot. We need to provide users with a certain degree of assurance about the authenticity of the provider and the information. *Third*, since all the crowdsourced information providers are rewarded equally regardless of their previous reputation as an information provider, it is hard to motivate them for providing authentic information. *Last*, current solutions also do not handle users with urgent parking needs and they lack a centralized management system for price-detection and monitoring. In addition, they do not have any alternative policies for users who do not get parking after their first attempt.

To resolve these issues, we propose *ParkBid*, a bidding-based approach for finding the desired automobile parking spot. Initially, all the users will register themselves and will only be allowed to participate in a bidding process after they provide their vehicle and contact information for authentication purposes. The registered users are considered as bidders and can take part in the bidding process. Each bidder receives an initial reputation point (RP) that specifies how trustworthy that bidder is as an information provider. A bidder initiates his bidding by providing the information of a currently available parking spot or a parking spot that is going to be available shortly and sends the information to ParkBid central system (PCS). The PCS verifies the users' provided information, determines the parking price, and verifies the reliability of the information.

When a user requests the PCS for parking, the PCS selects an appropriate subset of the parking information based on the user's specifications and sends them to the user in an opaque form. The PCS also creates a link between users and bidders so that any user with an urgent parking need can negotiate directly with a bidder. The PCS monitors both the user and the bidder once a user reserves a place, determines any malicious activities, and penalizes them accordingly.

Contributions:

1) ParkBid is the first approach that considers both centralized pricing scheme for regular users and negotiation based pricing scheme for users with urgent parking needs.
2) ParkBid enables users to choose an alternative parking spot without charging them for those spots in case a user cannot park his car in the reserved spot.
3) We provide a method for deriving a confidence value for each parking spot that specifies the probability that a user will be able to get the reserved spot once the user reaches the parking location.

The rest of the sections are organized as follows: In section II, we discuss the motivation of our work. Section III illustrates some of the related work. We describe the underlying architecture of ParkBid in section IV followed by a detailed operational model in section V. We discuss the challenges in our system along with countermeasures for addressing those challenges in section VI and VII respectively. We provide our simulated experimental result in section VIII. Finally, we conclude our work in section IX.

II. Motivation

Over the past decade, there has been a tremendous increase in the number of automobiles [18]. This increase has resulted in creating numerous problems in our urban ecosystem. It has given rise to parking trouble, pollution, traffic, travel time, and user frustration. All these issues are interlinked to each other. Researchers have studied how long it takes for users to find a parking spot [19]. The study found that in 35% cases, users spend 10 to 15 minutes to find a spot, while in 30% cases, the time is more than 20 minutes. Only in 2.7% cases, the wait time is less than 5 minutes. Also, the parking spot may not always be the perfect spot for the commuter. The commuter might have to walk some extra distance to reach their destination. Therefore, the duration of the trip will be a bit longer than expected.

Another motivating factor for our study is the lack of consideration of the urgency factor in similar studies. Sometimes, users need parking urgently and would be willing to pay other users to relinquish their spots in exchange of a monetary incentive. In our study, we give prime focus to the urgency factor. We try to predict the urgency and try to give an appropriate solution according to the need. We also focus on preventing any exploitation of the users. The urgency factor will lead to slightly higher prices but there will not be a massive difference, which will be classified as exploitation of customers. This motivation is unique to our study.

Moreover, we have also taken inspiration from the Priceline.com bidding model [20]. The Priceline bidding model is based on how much a consumer is willing to pay and if the specific amount is feasible to other entity. There is an element of surprise attached to the Priceline bidding model as the exact location is not visible to the user. If the bid price is accepted, the transaction is complete, and the location is made visible to the bidder. In ParkBid, we have used a similar approach. More research will be carried on this particular topic, in order to derive a fair and efficient cost model.

III. Related Work

Hoh et al. proposed TruCentive [9], whereas Yan et al. proposed CrowdPark [7] for securely providing parking information for users. In both approaches, the providers collect the information on any currently available parking or parking that can be free at some future time point from the crowdsourced users. However, both the models have several shortcomings. First, they did not address how a user can verify that a contributor is providing them with the correct information about their parking spot release time. Second, a user has to pay money as soon as they reserve a parking spot regardless of they are getting the free spot or not. Third, several consumers may want to park their vehicle for a shorter term. Therefore, reselling approach as a motivation, to tell the truth, might not work. Lan et al. proposed an automated system for notifying when a parking spot becomes free [21]. They used sensors, such as a GPS, accelerometer, gyroscope, and digital compass of the contributor's smartphone to determine if a driver in any parking spot is planning to leave the place or not. However, the system is unable to provide the information of free parking in the near future. Additionally, the automated calculation based on sensors collected data can generate some false positive about the free parking.

Nandugudi et al. proposed PocketParker, a crowdsourcing based automated system for informing the parking availability [11]. The users install the app in their smartphones that automatically monitors the traffic flow in a parking lot. Based on the analysis of the inbound/outbound traffic following its activity recognition algorithm, it determines the probability of a free parking spot. However, the major flaw in PocketParker is that it just shows the possibility of getting a free parking. Besides, the approach does not work when a user is searching for a street parking spot. Liao et al. proposed CroPark that collects information from mobile sensors and used an advanced machine learning algorithm to detect real-time knowledge about the roadside parking occupancy [8]. However, both PocketParker and CroPark fail to inform which parking slot in a lot or in the roadside will be free in near future.

Researchers proposed several infrastructure-based approaches for efficiently detecting free parking spots. Smart Parking Solution, led by Siemens [15] and Parking Spotter led by Ford and Georgia Tech [16] both use radar for detecting free parking spots in the street. Mathur et al. proposed ParkNet that used a GPS receiver and an ultrasonic rangefinder for collecting parking occupancy information [5]. Nawaz et al. proposed ParkSense that used Wifi beacon signal for detecting when a driver enters or leave a parking spot [22]. However, all these approaches can be implemented only in a small scale because of the high initial investment and maintenance cost of the sensors.

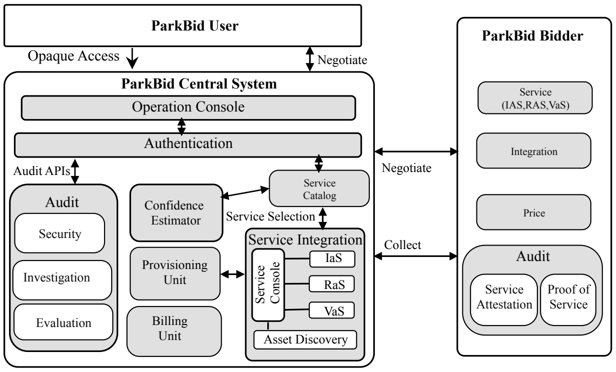

Fig. 1: ParkBid Architecture

For finding an appropriate parking based on user specifications, such as price, distance, etc., Grazioli et al. proposed a web based parking assisting model [23]. Google Maps and FourSquare are integrated for drawing parking areas along with their detailed information, such as the cost of parking, occupancy, etc. Kopecky et al. proposed ParkJam that incorporates the collected information from multiple users and shows users multiple parking spots along with the possible parking cost [24]. If a user misses a parking spot once, he can quickly pick another place from the updated information. However, they did not mention the pricing scheme for the alternative parking spots. While charging for an alternative parking spot will cost the user more for getting a free parking, users might misuse the service if the provider does not charge anything for the alternative spots.

IV. Conceptual Architecture

To resolve the shortcomings of existing approaches, we propose ParkBid, a crowdsourcing based service where the information providers receive incentives following a bidding strategy. The architecture is shown in fig. 1.

Overview: At first, the ParkBid central system (PCS) initiates bidding to know the users that are interested in joining in their system. The PCS also requests users to submit some of their personal information that is required for the participation. After a thorough verification of those provided information, the PCS accepts them for participating in ParkBid and tags them as bidders. Each bidder holds a reputation point that indicates how much trustworthy the bidder is for providing information. The users interested in parking their vehicle submit their requirements, such as location, price, reliability, etc. The PCS accepts the requirements and selects some of the accumulated information from the bidder. The PCS evaluates the assurance of that information known as confidence and sends that confidence tagged information as an obfuscated form to the user. The PCS evaluates the assurance of those information known as confidence and sends those confidence tagged information in an obfuscated form to the user. Additionally, the user can take a further protection plan from ParkBid for ensuring more trustworthiness about the bidder's provided parking information just like adding a protection plan during car rental. The user needs to pay some additional amount of money if he would like to add further protection depending upon the protection scheme he selects. A user can book multiple places at a time and later release the rest of the places once he parks its vehicle at one of his reserved spots.

A. ParkBid Central System (PCS)

The PCS is composed of the following units:

1) Service Catalog: The Service catalog unit first collects the ParkBid user's specifications from the operational console. It picks a service from the Service Integration unit and verifies the bidders' provided information. If a match is found then it creates an opaque information object from the original information and sends that to the user.

2) Confidence Estimator: We provide a level of confidence for each provided information that indicates the probability that a user would be able to park its vehicle into its reserved spot. The level of confidence varies from 0 to 1. The higher confidence value indicates that there is more chance that user will be successful in getting a free parking after confirming a reserved spot. Since the parking demand in the majority of big cities is extremely high, the probability that a user's reserved parking spot will be occupied by some other user is also very high. Therefore, the confidence of the provided information will be low in such circumstances. Based on the confidence of the provided information, users can decide whether they should reserve the parking spot or not.

3) Service Integration: This unit collects the information of a parking spot from the ParkBid bidders and stores the information into its database according to the service. It maintains records for all the different types of services. When the service catalog requests for a service, it retrieves the services sequentially and delivers that to the service catalog. Additionally, when a service is taken, it informs the provisioning unit about that reservation.

4) Provisioning Unit: The provisioning unit is responsible for mapping a specific service to a user so that the same parking place cannot be assigned to a different user. As soon as a user, reserves a spot, the provisioning unit updates its information table. When the service integration engine receives a service information from a bidder, it queries the provisioning unit to check whether that service is available or not. The provisioning unit replies about the status of the reservation.

5) Audit: The first responsibility of the audit unit is the verification of the service offered by the ParkBid bidders. A bidder needs to send some additional information as a proof of the authenticity of their provided service. The audit unit analyzes those proofs (described in section VII) and accepts the service if it can verify the information following the proof. It notifies the service integration module about the authenticity of the service. A ParkBid user might still claim an incongruity in the ParkBid bidder provided information and therefore, refuse to pay. The audit unit investigates and generates a report that describes the reason of a service failure. After the completion of an event, the audit unit evaluates the performance of both the bidder and the user, and updates their reputation value.

6) Billing: The billing unit determines the cost of a service offered by the bidder. A bidder might unfairly increase the price for a service. Therefore, it determines the price for a service so that it becomes fair for both the user and bidder. In addition, the billing unit also determines the service charge for the ParkBid provider from the negotiated money between the user and bidder.

B. ParkBid User

A ParkBid user is a person who is planning to take any parking service from the ParkBid. First, the user needs to register himself by providing a unique user id, password, and account information. The user can use either his bank account or Paypal information. Once the PCS successfully verifies that information, the user can search or reserve any service. Each user receives reputation point that indicates his reputation in his previous task. For searching a service, a user enters the address of the place they are planning to visit and the maximum distance they would like to walk from their reserved parking place. The user can also provide the time when he needs the parking. As soon as the user submits that information, he receives a list of possible available parking places inside his requested region from where he can reserve the spots.

C. ParkBid Bidder

1) Offered Service: A ParkBid bidder is a person that provides information about a parking spot that is currently free or is going to be available shortly. First, a bidder needs to register as a normal user. In addition, a bidder needs to provide some other information, such as his car VIN number, Car Tag number, and driving license number. The PCS verifies all the bidder's provided information and confirms the bidder as an active bidder after verifying that information successfully. A bidder can provide three types of service; Information as a Service (IAS), Reservation as a Service (RAS), and Verification as a Service (VAS).

IAS: In this case bidder just provides the information of an avaialble parking spot. Bidder stands near the parking spot takes a snap of the parking spot, and uploads that information as a proof of free parking. However, the bidder does not have any control of the parking spot. It is possible that the parking spot is occupied by some other vehicles before a user reaches to the spot after the reservation.

RAS: A bidder who has already parked his car in a parking spot can sell his parking spot. The bidder provides a range of time as a waiting time. The bidder uploads the snap of his parked car from where the PCS and a user can see both the car and the parking spot. When a user reaches to that place within the bidder's provided time interval, he contacts the bidder for releasing the place for him so that he can park in the bidder's current parking spot.

VAS: Bidder can also act as a verifier for any IAS or RAS bidders. He just needs to visit the place, take a snap of that place, and upload that image along with the coordinate location of where he has taken the snap. The VAS bidder does not have any responsibility regarding the parking availability, instead just need to upload his verification proof.

V. Operational Model

A. Process of Determining Reputation Point

We follow a very similar scheme for the rating point proposed by Noor et al. [25] for providing the reputation point (RP) to each bidder. The RP of each bidder lies between 0 to 1. When a bidder first joins, he receives an initial RP of 0.5. Based on bidder's authenticity in his previously provided information, we reward or penalize him. An IAS bidder is penalized 0.1 if his uploaded information is proved fake by the PCS. However, an IAS bidder does not receive any penalty as long as he provides correct information about a parking spot even the user does not find that available anymore after reaching there. On the other hand, we penalize an RAS bidder more for providing wrong information. A user can be confident that he will receive parking from an RAS bidder as he will leave that spot until the user reaches there. Since the user is entirely relying on the RAS bidder's provided information, any fake or misleading information can have a significant impact on the user's daily schedule. Therefore, we penalize 0.2 for a misleading information given by an RAS bidder. We compute the updated RP as follows:

$$R_p = R_{p(old)} - P$$

(P=0.1 for dishonest IAS and 0.2 for dishonest RAS.)

On the other hand, a bidder can gain a reward of 0.1 only if a user successfully parks his car following the bidder's provided information. The updated RP is computed as follows:

$$R_p = Min(R_{p(old)} + Reward, 1)$$

B. Process of Determining Confidence

The confidence of each information is based on the following criteria; i) Reputation Point (RP) ii) Type of Information iii) Place i) Time v) Delay

RP: A bidder RP increases when he provides authentic information during his previous task. Therefore, the probability of authenticity of an information of avaialble parking provided by a higher reputed bidder is greater than a lower reputed bidder.

Type of Information (I): We can trust an RAS bidder's information more than an IAS bidder's information since the RAS bidder has full control of the parking spot. Unless an RAS bidder leaves his holding parking spot, no other user can take that place.

Place (P): Place plays an important part for the parking. For example, big cities have fewer parking spots as compared to the total number of cars searching for parking. Similarly, the places where people frequently visits, such as shopping malls, getting a free parking spot might be difficult. Therefore, even a bidder provides information of free parking, there is a high chance that another user will occupy that spot.

Time (T): Besides the place, time is another important factor for getting a free parking. Even in a small town, getting a free parking spot during the peak hour is hard. Besides, if an event occurs in a place at a particular time regularly, such as a Saturday football match, the probability that the parking spot will be occupied is higher during that period.

Delay:(D) The likelihood that a user will be able to see his reserved parking spot free is inversely proportional to the time he takes to reach that spot.

In the case of RAS, regardless of the time or place, an honest bidder can always wait until a user reaches to his reserved spot. Therefore, the confidence value for RAS solely depends on the bidder reputation and expressed as $C = RP$.

However, for IAS we consider all the factors to compute the confidence value. From our observations of some of the major cities in USA, we can divide day into 7 different time intervals: i) t0(7 am- 9 am) ii) t1(9 am - 10 am) iii) t2(10 am -12 pm) iv) t3(12 am-3 pm) v) t4(3 pm -6 pm) vi) t5(6 pm -9 pm) vii) t6(9 pm - 7 am). We compute the weight w_i of a time interval t_i as, $tw_i = 1 - i/7$. For determining the effect of delay on getting a parking, we surveyed 50 people and asked them their personal experience about their favorite parking spots when they reach those spots after a certain delay. Around 50% of the time the users located 5-10 minutes distance from their preferred spots can park their cars in those spots. On the other hand, around 25% of the time users located 10 to 20 minutes distance and 10% of the time a user located above 20 minutes distance can park their cars in their favorite spots. Therefore, we compute the weight for the user delay as follows:

$dw = 1$ when $d <= 10$, $dw = 0.5$ when $10 < d <= 20$, and $dw = 0.2$ when $d > 20$

From our observation of parking demand in different places of Birmingham City, we divide it into ten categories $p_0, p_1, .., p_9$ in descending order of demand. We compute the weight for a category p_i as follows: $pw_i = 1 - i/10$.

For calculating the confidence value, we surveyed 50 people and asked them to submit their opinion about the place, time, and delays effect on parking. From the study, we computed a cumulative score of each of those individual parameters. We found that time is more correlated with getting free parking, followed by place. On the other hand, users consider that delay does not have a significant impact on parking. From the test score, we assign the weight of time, place, and delay to 0.5, 0.4, and 0.1 respectively. We determine an intermediate confidence value as, $0.5 * tw_i + 0.4 * pw_i + 0.1 * dw$

However, in ParkBid we need to consider bidder's reputation as well. If the RP of a bidder is R then the final confidence value, $C = (1 - (0.5 * tw_i + 0.4 * pw_i + 0.1 * dw)) * R$

C. Cost Model

Defining a cost model is a difficult task in any form of business. It is a work-in-progress during the entire course of business, one has to be pro-active and keep a close watch on its competitors. While designing our price-model, we take care of the following aspects: distance of the parking spot (D), the number of available parking spots (S), prospective time duration (T) in which the parking spot is desired. These factors lay down the basis for calculating the urgency factor (Z- factor) that is the needed to calculate appropriate price or points. Due to the cost model being in developing phase, we still have to decide on the best monetary scheme for the bidder. To calculate the Z-factor, we give slight preference to the time duration as it carry more weight towards the urgency of the customer. We derive the following equation based on the above assumption:

$$Z = \frac{[(S*0.4)+(T*0.6)]}{D}$$

We multiply the Z-factor by 10 in order to depict it with grace on the graph shown in fig 2 .

Fig. 2 shows an example to illustrate the calculation. It depicts the three variables in various colors and the Z-factor is calculated with our equation in Grey. The difference is significantly visible as a change in any of the variables results in a subsequent shift in the Z-factor. Let us take number 1 and number 2 for explanation purposes. In both the cases, distances and time are same as five units (distance being 50 meters and time being 5 minutes), but the availability of spots is different. Thus, in this case, calculation of Z-factor is directly proportional to the number of spots. As the number of spots increases, the Z-factor also increases. The concept we are following is similar to

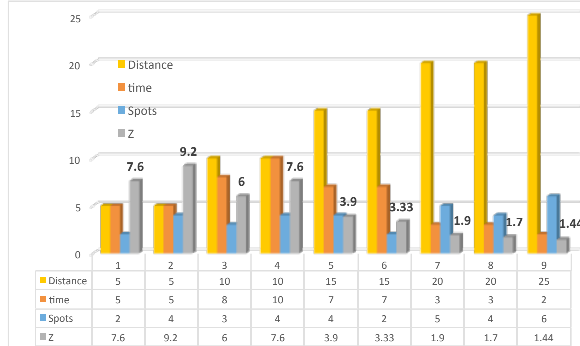

Fig. 2: A sample Z-factor evaluation process

the Gaussian Distribution function (bell curve)[26] to maintain consistency and not to exploit users if the supply of the number of available spots is less. We encourage the bidder to charge fair price to the users rather than charging them higher prices when fewer spots are available.

The final result of the Z-factor will help determining the proportion of cost that will be charged to the users who are bidding for the parking spot. Few cost model logistics are still to be finalized which are to be performed as future work.

VI. Challenges

A. Bidder is dishonest

A bidder can provide misleading information intentionally for earning money. Without having knowledge of any available parking spot, he can misinform the availability of parking. Also, he might claim that he is planning to leave his currently holding parking spot without having any intention to leave that spot. Besides, a bidder can fake a location or time. For example, a bidder might send a free parking location of one place and claims that it is the inside the area where the user is looking for free parking. Moreover, the user can collect the free parking information at one time point, such as during off-peak hour and send that information during peak hour.

B. User is dishonest

The user can be dishonest in several ways. First, the user may be aware of the free parking spot without reserving the spot. He can then just park his vehicle in that spot. The bidder will not receive any money even though the user parks his vehicle by collecting information from the bidder. A user can reserve multiple locations, and choose to park in one of those places and provide the other free parking information to some other users. In that case, the user can unfairly receive incentives from other users, and the bidder who provided the information originally will not receive any incentives for his rest of the provided information.

C. Both the bidder and user are honnest

It is possible that a bidder observes a free parking spot at time t_1 and informs about that to a user. The user reaches the parking spot at time t_2 after reserving the location. However, at time t where $t_1 < t < t_2$ another regular user can take the place. In such cases, we cannot blame any of the parties. Also, for various reason bidder might need to leave the place earlier or later than he was initially informed to the user. A user can also make some delay on reaching the place because of several environmental factors, such as traffic, accident, etc. In such scenarios, there will be a mismatch of timing between the user arrival and receiving the free parking spot.

D. Urgency

A major factor that may cause hindrance is the urgency of time to the commuter. It is likely that the commuter might be getting late for a job interview or lecture or any medical emergency. The above scenario was not taken into consideration in any of the previous approaches proposed. In some cases, one has to be biased and give preference to those users, but that would be unfair to the other users.

VII. Countermeasure

In ParkBid, we propose the following set of rules for addressing all the challenges as discussed in the previous section.

A. Authenticity of bidder

First, we would ensure the authenticity of the IAS provided by any bidder. A bidder needs to take a picture of the parking spot using the ParkBid provided app and upload that image along with his location coordinate information using the ParkBid app image upload feature (fig. 3). This information is sent to the PCS. There are several image analysis tools for detecting an event in a parking spot. We use the approach proposed by Liu et al. for detecting a free parking spot from the images captured by the parking integrated cameras [27]. This method is very efficient for free parking spot detection and can detect approximately 97% of the free parking spots. Since in ParkBid the images are taken and uploaded by the user instead of the parking integrated camera, we can expect that the picture quality will be better and transparent than the pictures uploaded by an integrated camera.

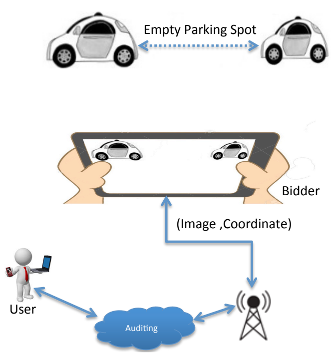

Fig. 3: Proof

Besides, a user can pay some additional money for confirming whether the location exists or not. In such cases, we use a witness based strategy (similar to WORAL [28]) where a third party is used as a witness and is responsible for notarizing the user provided information. In our system, some bidders who would like to give verification as a service. We use a random bidder from those bidders as a witness and inform them to visit the free parking space for attesting the location information. Once the bidder is at this place, he has to take a picture of that area using the ParkBid provided by the app.

B. Authenticity of user

A user can release his reservations only in two ways. First, if the user parks the car at a reserved spot, in that case, all the other reserved slot except the one he has parked his car will be considered free. Second, if the user proves that the parking place is occupied by a different vehicle. To determine that, the user needs to visit the parking place and take a picture positioning his camera to the similar coordinates as positioned by the bidder that sent the parking information. The picture is matched by the auditor in ParkBid. The Auditor matches the coordinate, analyzes the image using the same strategy proposed by Liu et al. [27], and allows the user to release his reservation if he finds that the parking spot is not empty from the image analysis.

C. Provide notification for easier communication

When a user reserves a place, he also specifies the time he is planning to reach the location. Similarly, a bidder mentions the time he is planning to release the location for the user. However, it is very unlikely that the actual parking release time or user arrival time will be the same as the information provided during the reservation. Therefore, we define a threshold time δ_t for each negotiation. A fake report from the user or bidder will be investigated only if any incident happens after the threshold time. Additionally, for making the communication simple, we define a threshold distance δ_d and as soon a user reaches to a location that is δ_d far from the bidder's parked location, the bidder will be notified. Thus instead of waiting beside his vehicle, the bidder can wait until he receives the notification.

D. Monetary negotiation

A user will be charged money as soon as he reserves a location. However, the bidder will not receive the incentive immediately after the reservation. Once the threshold time is over, the corresponding bidder will receive the incentive that was held by ParkBid during the reservation. Therefore, even if a user does not get a parking spot, he needs to pay for the spot. However, a user can avoid this situation by reserving multiple spots up to P number of spots. If a user reserves n spots, he will be initially charged for all the n number of spots. As soon as a user parks his vehicle to one of his reserved spots, he needs to notify the ParkBid. The user will be charged only for the parking spot where he has parked his vehicle and only the bidder that informed the user about that spot will receive the incentive. The rest of the parking spot will be released, and the user would be credited back for releasing those parking spots. On the other hand, if a user finds that the reserved spot is occupied, he needs to notify that to ParkBid along with image proof. The ParkBid verifies immediately and direct the user to his another reserved parking spot. If all the reserved spot becomes non-empty, the user will be charged only for one spot and the corresponding money will be distributed among all the bidders. Therefore, even a user reserves multiple spots at a single time point, he will be charged only for one spot.

E. Information obfuscation

In ParkBid, an exact free parking spot from the bidder will not be shown to a user until he reserves the place. The information will be represented in an obfuscated form, from where the user will be able to know whether the ParkBid has any information that matches user specification or not. Besides, the user can also know the confidence of the provided information. The exact parking spot will be revealed only after the user reserves that spot. The process will prevent the user from using that parking place without buying it from the ParkBid. Besides, for increasing the probability of getting free parking, a user can reserve multiple spots. Since user knows that he just has to pay for one place, he might try to sell the other spots or leak the information of other spots to other users. This act will harm the other bidders that provided a user with the information of available parking because in that case, they will not receive any money. Therefore, the user will be able to see the exact parking location one at a time. To obtain the information of other obfuscated parking spots, the user needs to prove that the current parking spot is non-empty, which in essence prevents the user from reusing the information.

F. Z Factor (Urgency)

A user may try to fake the urgency to get the desired parking, so we have to careful about such events in the ParkBid model. This challenge led us to give extreme emphasis on the urgency factor in our approach. The calculation of price is based on the Z-factor. If a user shows urgency, then they have to pay a higher price for the particular desired spot. The payment of high prices acts as a blessing in disguise for the business as well as the person providing the information.

VIII. Experimental Evaluation

A. Simulation Setup

We created our simulation model in Java. We considered a random number of bidders and users. We assumed two types of users; regular user and ParkBid user. We considered a 50x50 Km^2 area where bidders are users are randomly distributed. We assumed that users and bidders are moving randomly inside that area and parking spots are distributed randomly within the area. We considered each simulation unit equivalent to one minute, and the probability that a bidder send a parking information or a user would request for parking is 0.5. However, the request could be initiated only if the bidder is within 50m interval from the parking spot. A bidder can send a maximum of 5 parking information at a time and a user can request for a maximum of 5 parking spots at a time. A user always selects the lowest price parking spot first from his reserved spots. We considered two scenarios; in the first scenario, we assumed that users do not use ParkBid. They moved randomly near their place of interest, searched for free parking, and parked the vehicle as soon as he received one. In the second scenario, we assumed that the user used the ParkBid, reserved places, and visited those places based on the ascending order of price until he found a parking spot or all the reserved spots were visited. Besides, we assumed that some random regular users could occupy the place. We defined that the probability that such random users would hold a place is between 0 to 1. We ran our simulation for 1000 simulation units and measured the average traversing distance for getting a free parking and success-to-failure ratio for different circumstances for the users using the ParkBid and users not using the ParkBid.

B. Simulation Results

In our first experiment, we varied the number of users that are planning to search a parking place. For the user who used ParkBid, we gradually increased the number of interested bidders and measured the performance. The corresponding figures are shown in fig. 4 and fig. 5 respectively.

From fig. 4, we see that the regular users need to traverse more distance before getting a free parking than the ParkBid user. We also see from fig. 5 that the success rate for the ParkBid user is much higher than the regular user. From fig. 4, we see that there is no significant relation between the traversed distance and the bidder size. The reason is that a user can

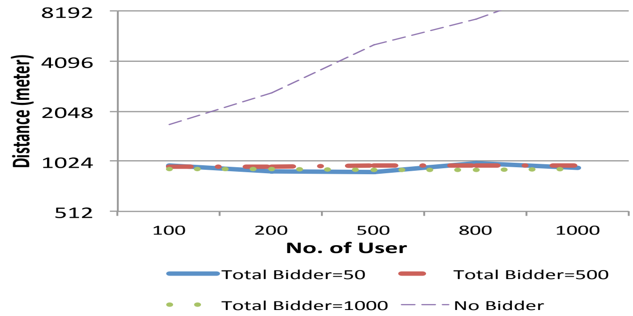

Fig. 4: Traversing distance with different size of user

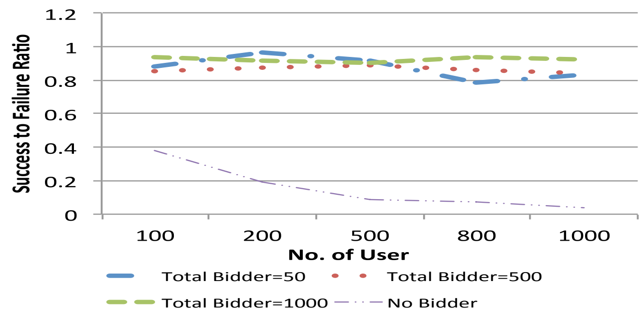

Fig. 5: Success to failure ratio with different size of user

only visit a parking place once a bidder circulates it. Therefore, when a small number of bidders participate, a user does not receive information of all the free parking spot. Consequently, the bidder visits only those circulated parking spots. Hence, the average traversing distance remains the same. However, the bidder size has effect on success to failure ratio especially for large user size. We can see from fig. 5 that the success rate is little bit more for larger bidder size. The reason is that with increasing bidder size the number of explored free parking spots also increases. Therefore, each user can have an option to request for multiple spots. Hence, the probability of finding a free parking spot becomes higher for larger bidder size.

In our next experiment, we changed the total number of parking spots and measured the performance. The corresponding figures are shown in fig. 6 and fig. 7 respectively. From fig. 6, we see that when there is more demand for the parking spots than supply, the average traversing distance for the ParkBid user is much smaller than the regular user. However, for a very large number of free parking spots, ParkBid does not provide any benefits for its users. The primary reason is that a user can easily get a free parking spot just next to his place of interest. On the other hand, in ParkBid, we prefer a low-cost parking place even if it is located a little bit far from a user's place of interest. We can also see that for a large number of bidders, a user needs to travel a smaller distance due to the exploration of more number of parking spots. From fig. 7, we see that the success rate increases for larger number of parking spots and with increasing number of bidders. We can also see that for the smaller number of parking spots, the probability of getting a free parking for the regular user is very low.

In our next experiment, we gradually increased the probability that a parking spot is free in each unit of time from 0.5 to 0.95 and measured the performance. The corresponding results are shown in fig. 8 and fig. 9 respectively.

From fig. 8, we see that the user needs to traverse less

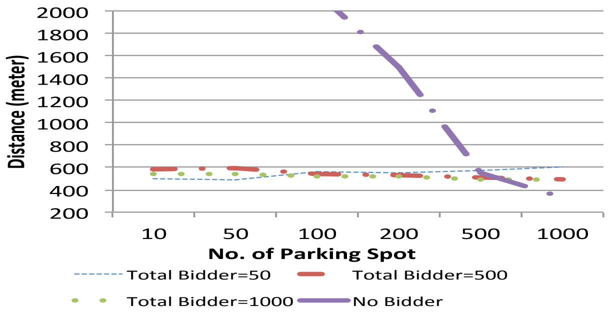

Fig. 6: Traversing distance with different size of parking spot

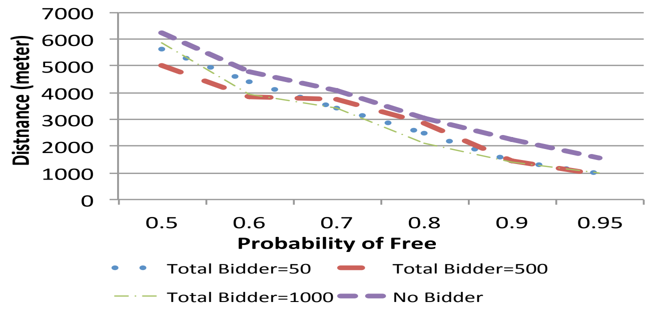

Fig. 8: Traversing distance for different values of probability

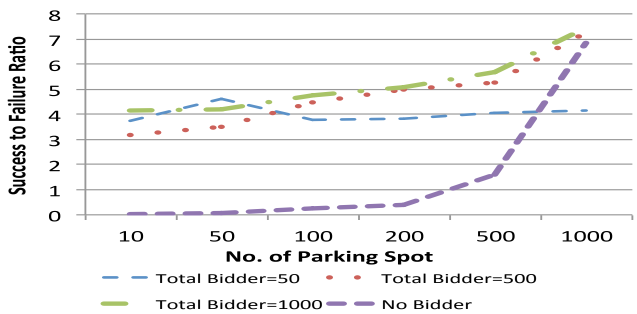

Fig. 7: Success to failure ratio with different size of parking spot

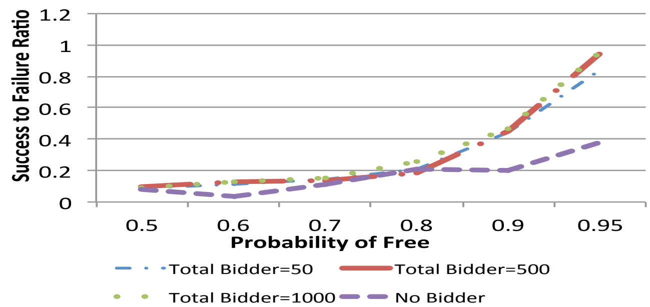

Fig. 9: Success to failure ratio for different values of probability

distance to get a free parking for larger probability value. The distance is much smaller for the ParkBid user, especially for larger bidder size. However, from fig. 9, we notice that for lower probability value, the success rates for the ParkBid users and regular users are almost the same. The reason is that the parking place is occupied immediately by some other random users before the ParkBid user reaches there. However, for intermediate or high probability, ParkBid provides a better success rate.

IX. Conclusion

Search for a desired parking spot at the correct time has always been a challenge in the big cities with impact on traffic, pollution level, wastage of fuel, and user frustration. Our solution to this problem, ParkBid, uses a crowdsourcing-based parking service where a bidding process is provided for users searching for the desired parking. ParkBid provides various incentives for the information about parking spots. Our experimental results demonstrate that users save a significant amount of time while finding the desired parking spot and the incentives make the users provide correct information about the parking spots. In future work, we will explore a real life deployment of the scheme involving a large number of users and parking locations in an urban environment.

X. Acknowledgements

This research was supported by the National Science Foundation through the NSF CAREER Award CNS-1351038 and ACI-1642078.

References

[1] D. C. Shoup, "Cruising for parking," *Transport Policy*, vol. 13, no. 6, 2006.
[2] E. I. Richard Arnott, "An integrated model of downtown parking and traffic congestion," *Urban Economics*, vol. 60, no. 3, 2006.
[3] M. Caliskan, D. Graupner, and M. Mauve, "Decentralized discovery of free parking places," in *Vehicular Ad Hoc Networks*, 2006.
[4] J. V. Derbeken, "Fatal stabbing over parking," http://www.sfgate.com/bayarea/article/SAN-FRANCISCO-Fatal-stabbing-over-parking-2488076.php, 2006.
[5] S. Mathur, T. Jin, N. Kasturirangan, J. Chandrasekaran, W. Xue, M. Gruteser, and W. Trappe, "Parknet: Drive-by sensing of road-side parking statistics," in *MobiSys*, 2010.
[6] D. S. GREGORY PIERCE, "Sfpark: Pricing parking by demand," https://escholarship.org/uc/item/0j41t7rz, 2013.
[7] T. Yan, B. Hoh, D. Ganesan, K. Tracton, T. Iwuchukwu, and J. sik Lee, "Crowdpark: A crowdsourcing-based parking reservation system for mobile phones," UMASS Technical Report, UM-CS-2011-001, 2011.
[8] X. Chen, E. Santos-Neto, and M. Ripeanu, "Crowdsourcing for on-street smart parking," in *Design and Analysis of Intelligent Vehicular Networks and Applications*, 2012.
[9] B. Hoh, T. Yan, D. Ganesan, K. Tracton, T. Iwuchukwu, and J. S. Lee, "Trucentive: A game-theoretic incentive platform for trustworthy mobile crowdsourcing parking services," in *Intelligent Transportation Systems*, Sept 2012.
[10] K.-C. Lan and W.-Y. Shih, "An intelligent driver location system for smart parking," *Expert Syst. Appl.*, vol. 41, no. 5, 2014.
[11] A. Nandugudi, T. Ki, C. Nuessle, and G. Challen, "Pocketparker: Pocketsourcing parking lot availability," in *UbiComp*, ser. UbiComp '14, 2014.
[12] K. J., "Googles open spot makes parking a breeze, assuming everyone turns into a good samaritan." http://techcrunch.com/2010/07/09/google- parking- open- spot/.
[13] B. N., "Finding that prime parking spot with primospot," http://bits.blogs.nytimes.com/2009/12/03/finding-that-prime-parking-spot-with-primospot/.
[14] "Sfpark - about the project." http://sfpark.org/about- the- project/.
[15] "Integrated smart parking solution," www.mobility.siemens.com/mobility/global/en/urban-mobility/road-solutions/integrated-smart-parking-solution/Pages/integrated-smart-parking-solution.aspx, 2015.
[16] "Parking spotter," www.media.ford.com/content/fordmedia/fna/us/en/news/2015/01/06/mobility-experiment-parking-spotter-atlanta.html, 2015.
[17] I. Sherwin, "Google labs' open spot: A useful application that no one uses," http://www.androidauthority.com/google-labs-open-spot-a-useful-application-that-no-one-uses-15186/.
[18] J. Voelcker, "1.2 billion vehicles on world's roads now, 2 billion by 2035: Report," http://www.greencarreports.com/news/1093560_1-2-billion-vehicles-on-worlds-roads-now-2-billion-by-2035-report, 2014.
[19] C. Snyder, "Solving parking problems in metropolitan areas," http://www.pages.drexel.edu/ asc38/FrshDsnPrj/.
[20] www.priceline.com.
[21] K.-C. Lan and W.-Y. Shih, "An intelligent driver location system for smart parking," *Expert Systems with Applications*, 2014.
[22] S. Nawaz, C. Efstratiou, and C. Mascolo, "Parksense: A smartphone based sensing system for on-street parking," in *MobiCom*, 2013.
[23] A. Grazioli, M. Picone, F. Zanichelli, and M. Amoretti, "Collaborative mobile application and advanced services for smart parking," in *Mobile Data Management*, June 2013.
[24] J. Kopecký and J. Domingue, "Parkjamjam: Crowdsourcing parking availability information with linked data," in *The Semantic Web: ESWC 2012*, 2015.
[25] S. A. Noor, R. Hasan, and M. M. Haque, "Cellcloud: A novel cost effective formation of mobile cloud based on bidding incentives," in *IEEE Cloud*, 2014.
[26] Andale, "Bell curve (normal curve)," December 2014. [Online]. Available: http://www.statisticshowto.com/bell-curve/
[27] J. Liu, M. Mohandes, and M. Deriche, "A multi-classifier image based vacant parking detection system," in *ICECS*, 2013.
[28] R. Hasan, R. Khan, S. Zawoad, and M. Haque, "WORAL: A witness oriented secure location provenance framework for mobile devices," *IEEE TETC*, 2015.