TruCentive: A Game-theoretic Incentive Platform for Trustworthy Mobile Crowdsourcing Parking Services

Baik Hoh[2], Tingxin Yan[1], Deepak Ganesan[1], Kenneth Tracton[2], Toch Iwuchukwu[2], and Juong-Sik Lee[2]

***Abstract*— The shortage of parking in crowded urban areas causes severe societal problems such as traffic congestion, environmental pollution, and many others. Recently, crowdsourced parking, where smartphone users are exploited to collect real-time parking availability information, has attracted significant attention. However, existing crowdsourced parking information systems suffer from low user participation rate and data quality due to the lack of carefully designed incentive schemes.**

In this paper, we address the incentive problem of trustworthy crowdsourced parking information systems by presenting an incentive platform named *TruCentive*, where high utility parking data can be obtained from unreliable crowds of mobile users. Our contribution is three-fold. First, we provide hierarchical incentives to stimulate the participation of mobile users for contributing parking information. Second, by introducing utility-related incentives, our platform encourages participants to contribute high utility data and thereby enhances the quality of collected data. Third, our active confirmation scheme validates the parking information utility by game-theoretically formulated incentive protocols. The active confirming not only validates the utility of contributed data but re-sells the high utility data as well. Our evaluation through user study on Amazon Mechanical Turk and simulation study demonstrate the feasibility and stability of *TruCentive* incentive platform.

I. Introduction

Parking shortage in urban streets imposes a spectrum of societal problems. It is observed that over 40% of total traffic volume in urban areas are vehicles cruising for parking [15]. The long queue of cruising vehicles on the road can cause traffic congestion with blocking of a few streets. In addition, low speed cruising incurs significant amount of automobile emission [2] that aggravates air pollution. Long time circling also adds to drivers' stress levels, and has been reported to increase road rage and accidents [4].

The lack of parking availability information worsens the parking problem. According to studies by the US Department of Transportation, parking patrons "often do not know where the best parking locations are", and "most importantly, whether a parking place will be available when they arrive" [10]. These concerns have led to significant efforts to design online real-time parking information systems that provide up-to-date information about parking availability. For example, SFPark is a pilot project to monitor real-time parking availability in San Francisco by deploying a massive network of sensors [14]. While such parking information systems can help direct drivers to available parking locations, they are in their pilot stages and face daunting scaling and budgetary challenges given the vast volume of street parking in U.S. cities. Continuous monitoring of street parking requires installation of occupancy sensors on hundreds of thousands of parking spots or parking meters, and a vast wireless infrastructure to obtain and transmit sensing data in a reliable manner. Such an investment cannot be expected in emerging countries.

While infrastructure-based parking availability systems suffer from their limitations, crowdsourcing-based parking availability information has attracted significant attention from the latest intelligent transportation and navigation systems, for the sake of agility, large scale, and low cost. Several pilot applications, such as OpenSpot [6] from Google and PrimoSpot [3], exploit smartphone users to provide empty parking spots they witnessed, despite inaccurate crowdsourced reports. The recently released Apple iOS 6 also includes crowdsourced parking as one of the key features of the Apple map service. Mathur at al. [8] has also shown that using distance sensors attached to taxis for obtaining parking availability is promising, although it is expensive and less scalable than using mobile users. Yan et al. [17] has claimed that a combination of crowdsourcing and mobile sensing techniques enables sharing available-soon spots as well as available-now spots with high accuracy. Compared with infrastructure-based approaches such as SFPark [14], crowdsourced parking availability systems have the potential to be deployed at large scale with low cost. For instance, PrimoSpot has already covered several thousands of street parking spots from seven cities in US without extra investment on parking infrastructure [3].

Although crowdsourcing-based parking availability information systems are intriguing, trustworthy crowdsourcing [5], [18] presents several challenges — how can we obtain high quality data from unreliable crowds of mobile users who can be selfish, erroneous or even malicious? Incentives are generally introduced for the purpose of user participation as well as high quality data [9][18][13]. However, without a carefully designed incentive platform, existing crowdsourcing parking services suffer from unreliable user participation and low data quality. Users often face outdated, misplaced, or even bogus parking information in existing crowdsourced parking systems such as OpenSpot, where incentives are as simple as fixed virtual credits. Monetary incentives could potentially attract user attention and encourage high quality data from crowdsourcing participants, but they are always vulnerable to malicious users whose goal is solely maximizing rewards by providing large amount of data without regard for data quality.

In this paper, we explore the incentive problem for trustwor-

*This work was done while the second author was an intern at Nokia

[1]Dept. of Computer Science, University of Massachusetts, Amherst MA 01003 yan, dganesan at cs.umass.edu

[2]Nokia Research Center, Palo Alto CA 94304 baik.hoh, ken.tracton, toch.iwuchukwu, juong-sik.lee at nokia.com

thy crowdsourcing and propose an incentive platform named *TruCentive* with two design consideration in mind. First, we want to encourage mobile users to contribute parking availability information with high data quality; second, we want to prevent malicious participants from spamming the parking service with high volume of useless data. The novelty of *TruCentive* is three-fold. First, it provides hierarchical incentives for people contributing parking information with static and dynamic incentives. Second, it encourages participants to contribute data with high quality by dynamically determining incentives according to the utility of contributed data. Third, *TruCentive* validates the data quality of parking information from the confirmation of data consumers and the confirmation process is completed through a game-theoretically formulated incentive protocol. The confirmation not only validates the utility of contributed data but initiates the resell of high utility parking availability information as well.

The rest of the paper is organized as follows. §II provides a brief survey of closely related work. §III provides an overview of our *TruCentive* incentive protocol. §IV describes the theoretical framework of *TruCentive* in detail. In §V, we evaluate the performance of *TruCentive* and in §VI we conclude the paper.

II. Backgrounds

Before presenting our design of the *TruCentive* incentive platform, we briefly review existing crowdsourced parking availability systems and related incentive schemes.

Intelligent parking has been studied for years. Existing work focuses on modeling the user behavior in parking lots such as the choices of parking spots [11] and the distribution of vacant parking spots [16]. Our work takes a different approach by studying the problem of how to share crowdsourced parking spots with other drivers and the incentive schemes that lead to trustworthy crowdsourcing.

In recent years we have seen the rise of crowdsourcing-based parking sharing applications. Mobile users who participate in the system provide information about available parking spots in exchange for certain incentives. Google's OpenSpot [6] and PrimoSpot [3] are typical examples of pilot systems where participants, either pedestrians or drivers, are asked to manually report open spot information via smartphone client applications for virtual reputation points for users. This manual reporting system may be cumbersome and may incur high error in parking reports. The reputation-based incentives are also not attractive enough for users for stable and long-term participations.

Yan et al. [17] greatly enhanced the crowdsourcing parking sharing systems by automating the parking reports using mobile sensing approaches. By exploiting activity recognition and geofencing-based trajectory analysis, they claim that parking availability can be automatically obtained and shared with other users with high accuracy. In addition, the mobile sensing approach can be used to detect malicious behavior thus preventing malicious participants from masquerading as fake participants for incentives. However, they still use fixed incentives for all parking information, without any notion of the utility of the shared parking information, or "which parking reports are more useful".

On the other hand, incentive schemes have been extensively studied in the pervasive computing community for various applications. One category of incentive schemes sets the incentives dynamically with the utility of the information or products provided by participants via bidding. One example of bidding-based incentive schemes [7] introduced reversed auction to dynamically decide the incentive for a particular crowdsourcing task. In their scheme, the incentives to participants are dynamically determined according to the bid prices claimed by participants. Participants express their different expectations by bids and the system can tradeoff total cost of the incentives with the utility of collected data. Although the true value of a crowdsourcing task can be revealed through the bidding process, the requirement of multiple rounds of bidding renders inadequate for crowdsourcing parking since it slowly adapts to unknown distributions of contributors' expectations.

Our incentive scheme addresses the limitation of static and bidding-based dynamic incentives by introducing a hierarchical incentive protocol where the incentive for parking availability information is linked to its utility. While a fixed amount of base incentive makes the incentive system simple enough for practical uses, an additional reward (based on the data utility) fulfills higher expectations from more loyal participants. Further our incentive scheme is orthogonal to the concrete approaches of obtaining the parking availability information, thus can easily work with existing crowdsourced parking systems. Throughout the paper, we consider the SenPark system presented by Yan et al. [17] as a baseline system for our incentive platform.

III. *TruCentive* Overview

In this section, we present an overview of our *TruCentive* incentive platform. We will first introduce the work flow of a typical crowdsourced parking availability system and then introduce our incentive protocol that fits into these systems.

Crowdsourced Parking Availability Systems. A typical crowdsourced parking availability system gathers parking availability information from mobile participants and delivers it to drivers who might be searching for parking spots. There are two types of users in the system: *contributors* are the participants who provide the parking availability information in exchange of incentives and *consumers* are the customers of the service who utilize the crowdsourced parking information for finding a parking spot.

In crowdsourced parking availability systems, contributors provide the information of when and where a parking spot is available or soon-to-be available (we will call such parking available information PA messages hereafter). Crowdsoured parking availability systems (hereafter noted as service providers) gather PA messages from contributors and distribute them to drivers adjacent to the location of the PA message to help them find an empty parking spot. The service provider pre-selects the areas spatially and temporally on their prior

User Credentials	GPS	Parking Spot Identification	Released Time of Parking Spot
	Timestamp		

Fig. 1. Contents of a PA message sent by a contributor. “Parking spot identification” and “Released Time of Parking Spot” fields are provided by a contributor’s smartphone application automatically to provide the location and time information of an available parking spot.

knowledge (e.g., geographical distribution of parking tickets issued over a year) where parking demand is higher than supply, thereby ensuring the usefulness of the PA messages to nearby drivers searching for parking near the corresponding areas. For the sake of simplicity, we assume that contributors’ mobile client applications automatically report PA messages to the service provider with reasonably accurate location and available time information, as described in SenPark [17].

***TruCentive* Idea.** The core functionality of *TruCentive* is to provide a platform for contributors and consumers to “trade” PA information. *TruCentive* uses system credits as the incentive for each PA information transacted between contributors and consumers. System credits are another form of monetary rewards and several web service providers including Facebook support virtual credits which can be exchanged for monetary rewards or products in a store [1].

A PA message contains location and time information of the parking spot to allow a consumer to accurately locate a spot and arrive at the spot in a timely manner. Location information is specified by a combination of inputs: 1) GPS location at the parking spot, 2) identifier of the parking spot, and 3) an identification of the contributor’s vehicle such as the color, make, and license plate number of the contributor’s car. Note that the contributors vehicle identification is collected only once upon registering with *TruCentive* and does not have to be re-sent each time. We assume that the client application automatically determines location of the spot and predicts when to leave. [17]

A. *TruCentive Trade Protocol*

The trading process under *TruCentive* is described in Figure 2. The entire protocol consists of three major steps: 1) a contributor submits a PA message to *TruCentive*, 2) a consumer “buys” a PA message for available parking spot information, and 3) the consumer drives to the parking spot and sends back an active confirmation of whether the parking spot information led to a successful parking or not. We explain these three steps in more detail.

Contributor submits PA to *TruCentive*. In order to incentivize users to share PA information, *TruCentive* pays contributors two types of rewards - a static reward of D points and a bonus of X points that is dynamically determined. The static reward is granted immediately after a PA message is accepted by *TruCentive*. This incentive is provided irrespective of whether the spot is purchased by a consumer. Such a participation incentive can help bootstrap our system, and ensure a steady stream of available spot information to make

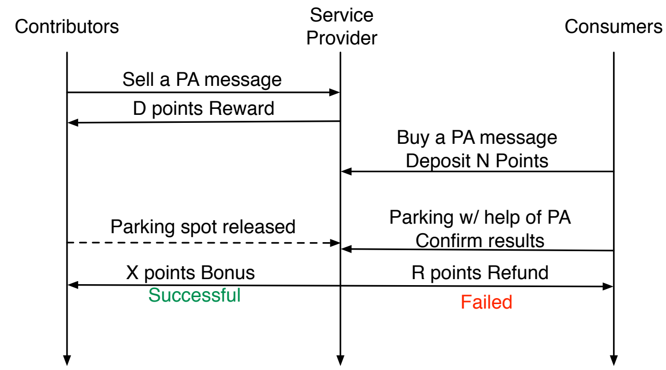

Fig. 2. The transaction process of a PA message under *TruCentive* protocol. It is a full refund model when $R = N$.

the marketplace attractive to consumers. This fixed reward is expected to be a small amount, say a quarter, although this may be adjusted based on the importance of the location and time of the sale. For example, the reward for downtown San Francisco may be higher than suburban areas, and can vary over time depending on parking demand.

The bonus reward is granted right after the parking is confirmed a success, that is, after a consumer has successfully parked according to the PA message she bought. The amount of bonus can be significantly larger than the fixed reward, since the PA message led to a successful parking. *TruCentive* collects the statistics of successful parking rate of PA messages, and for the locations with successful parking rate, the bonus is dynamically increased by *TruCentive* to encourage contributors for high utility data. This is similar to the lottery principle — higher payouts can be won with less probability. Similar to static reward, the baseline of bonuses can be also determined by the importance of locations and time scale.

Consumers buy PA for available parking spots. A consumer who wishes to use our crowdsourced parking service provides a destination and arrival time to *TruCentive* and receives a list of matching PA messages. The returned PA messages are first presented to consumers in obfuscated form, where the precise information in a PA is obfuscated. For example, the actual time is replaced by a time window of 10 minutes and the location is replaced by the distance to consumers’ destination. The obfuscated mode allows consumers to identify the appropriate PA messages while not revealing the PA information in its entirety before consumers pay for it. After a consumer pays for a PA message, the actual content of the PA including the location and leaving time information is made visible to the consumer. The trade is complete at this point. The payment from consumers, N points per PA message, is deposited to *TruCentive*. Note here that the payment from consumers will be refunded. The deposit itself is only for fairness purpose — preventing one consumer from taking arbitrary number of parking availability information.

Consumers confirm parking results. The final step is a confirmation from the consumer about the success or failure of the parking. If the PA led to a successful parking of the

Fixed incentive for Contributor	D
Contributor bonus for successful parking	X
consumer deposit for transaction	N
consumer refund for unsuccessful parking	R
Probability of successful parking	p
Probability of unsuccessful parking	q
Alternate revenue (e.g. Ads)	C

TABLE I
TruCentive PARAMETERS.

consumer, the contributor receives a bonus of X; otherwise a refund R is returned to the consumer. Note that refund R can be different or same from the deposit paid by the consumer, depending on the business model of concrete service providers. Also note that the base reward D granted by the service provider depends on the *TruCentive* dynamics as well as the desired profit margin for the service provider. We will discuss the setting of parameters in detail in § IV.

IV. A GAME-THEORETIC INCENTIVE PLATFORM

The main challenge for *TruCentive* is to determine the parameters of deposit, refund, and bonus for encouraging high utility data from contributors. In this section, we present a game-theoretical design of incentive protocol to achieve this goal.

Recall that we give additional rewards to contributors when the transaction leads to a successful parking. This raises two concerns: 1) how to verify the consumer's unreliable confirmation and 2) how to choose incentive parameters while ensuring the profitability of the service provider. More specifically, there are two key challenges that we address in this section:

- *How to ensure that consumers are honest?* A malicious consumer can buy a PA and park successfully but deny the fact for a (full) refund. We introduce a game-theoretical design that ensures rational consumers realize that telling the truth could maximize their gain, thereby discouraging consumers from telling a lie in confirmation.
- *How to ensure profitability for the service provider?* The parameters of the trade protocol influence the margins for the parking service provider. In §IV-B, we provide a cost-benefit analysis from the provider's perspective and describe how it can be used to determine protocol parameters.

Under the constraints of "ensuring honest consumer" and "profitable service provider", we carefully set the different reward and payment variables such as D, R, and X in the system.

A. Incentivizing Honest Consumers

The first challenge that we address is dishonest consumers. In the protocol described earlier, a consumer can deny that a parking was successful even if it was indeed successful, and thereby receive a full refund. Consumer's dishonesty also has the negative consequence since the contributor who provided the PA message is not rewarded with bonus, which can discourage contributors' participation. We now present a game theoretical design of the parking trade protocols which ensures that consumers maximize their gain by telling the truth. With such a design, rational consumers would choose to be honest in our system.

	Honest	Dishonest
Successful	$D + pX$	R
Unsuccessful	R	D

TABLE II
THE GAIN FOR CONSUMERS BY BEING HONEST AND DISHONEST WHEN PARKING IS SUCCESSFUL AND UNSUCCESSFUL.

The key idea behind our approach is that a consumer who successfully parks at a traded spot can re-sell that spot through the *TruCentive* system if they tell the truth. If they deny a successful parking, obviously they cannot re-sell this spot. We use this idea to set reward parameters to ensure that the average gain of re-selling is higher than the gain by lying and receiving a refund. The approach has two benefits: a) it achieves honest reporting from consumers, and b) it encourages consumers to keep re-selling a spot thereby ensuring that a steady pool of spots are available on *TruCentive*. Thus, the presented incentive platform plays a key role in retaining and promoting user participation to crowdsourcing parking availability information. We now describe the approach more formally.

As mentioned in the previous section, the gain of selling a PA message contains two parts: a constant reward D and a bonus reward X. If the probability of selling a PA and confirmed successfully is p, the reward for selling a PA message is: $D + pX$. To ensure that a consumer is better off being truthful, we only need to ensure that the average reward of reselling a traded parking spot is higher than R (equals to N in case of full refund).

Table II shows the gain for a consumer if they tell the truth vs lie, and when parking was successful vs unsuccessful. To ensure consumers maximize their gain by being honest, we need the following constraint to hold:

$$D + pX \geq R \geq D \tag{1}$$

$$\text{or,}\quad X \geq \frac{1}{p}(R - D) \tag{2}$$

The above analysis assumes that we know p, the probability that a PA is sold and confirmed successful. In a live system, p is simply a measured system parameter, and the current measured value of p can be used in the inequality.

While Equation 2 provides a lower bound for X, the bonus has to be upper bounded as well since a service provider has profit considerations.

B. Service Provider Cost-Benefit Analysis

Under what conditions is *TruCentive* a profitable business enterprise? As we will show, the analysis for this question helps further narrow down the region of the values for bonus X and refund R.

Let's consider a single PA transaction. For each PA message, the cost for the service provider is as follows: 1) reward to the Contributor is $D + pX$, where p is the probability that a PA is sold and confirmed successful, and 2) refund to the consumer is qR, where q is the probability that a PA is sold and confirmed unsuccessful. Note that the sum of p and q indicates the probability that a PA is sold, which is less than 1. The benefit for the service provider is the income that can be earned from each PA, which is $(p + q)N$.

Mobile businesses typically have multiple sources of revenue such as targeted advertisements (e.g. for parking garages), so they may not be reliant solely on income from parking trades. Let us assume that service provide has a per-PA revenue of C from these alternate sources. Our goal is to ensure that the service provider can break even. In other words, we have the following constraint:

$$\begin{aligned} C + (p+q)N &\geq D + pX + qR \\ \text{or} \quad X &\leq \frac{1}{p}(C + (p+q)N - D - qR) \end{aligned} \tag{3}$$

Together with (2), we have upper and lower bounds for bonus X as follows:

$$\frac{1}{p}(R - D) \leq X \leq \frac{1}{p}(C + (p+q)N - D - qR) \tag{4}$$

The above inequality implies that:

$$\begin{aligned} R - D &\leq C + (p+q)N - D - qR \\ \text{or,} \quad R &\leq \frac{1}{(1+q)}(C + (p+q)N) \end{aligned}$$

Together with (1), we can derive the constraints for refund R as follows:

$$\begin{aligned} D \leq R &\leq \frac{1}{1+q}(C + (p+q)N) \\ \text{or,} \quad D &\leq \frac{1}{1+q}((p+q)N + C) \end{aligned} \tag{5}$$

This is the constraint for base reward D and PA price N given a budget C from service provider.

Example We now instantiate the above analysis with a simple example. Let $C = 0$, which means that there is no additional source of income, and *TruCentive* needs to be profitable on its own. Then, we can derive the relation between D and N based on (5) as follows:

$$\begin{aligned} D &\leq \frac{1}{1+q}((p+q)N) \\ \text{or,} \quad \frac{D}{N} &\leq \frac{p+q}{1+q} \end{aligned} \tag{6}$$

This is the condition that *TruCentive* is profitable. We can change the ratio of D and N based on the trade probability, p that we learn over time. For example, assume that $N = \$2$, and that $p = q = 0.1$. We can derive other system parameters as follows:

- From (6), we have $D \leq \$0.36$.
- From (1), we have $\$0.20 \leq R \leq \$0.36$.
- From (4), we have $\$1.5 \leq X \leq \$1.65$.

Thus, we have derived system parameters D, R, and X that encourage honesty from consumers and ensure that the service provider is profitable from *TruCentive*.

C. Special Case: a Full Refund

Now we consider giving a full refund to the consumer who fails parking. Given that $R = N$, we need to derive C, D, and X. To meet the equation 1, let us set $D = N$. Then, the honest consumer always has larger gain than lying since $N + pX$ is always greater than N, which is a full refund for lying. In case of unsuccessful parking, telling the truth receives a refund of N, but falsely confirming (even though failed parking indeed) will be filtered out by malicious contributor detection [17]. Thus, having set $R = D = N$ successfully ensures honest consumer constraint as long as we assign $X \geq 0$ for any given p.

Next, we replace D by N in equation 6, then we have the following condition for C:

$$(1 - p)N \leq C \tag{7}$$

To meet the profit consideration of the service provider, C should meet the condition in equation 7. Thus, the service provider should figure out the expected C based on their alternate revenue sources and compare with $(1 - p)N$. Once C meets the condition, equation 4 determines the range of X. When we start considering the situation where the PA can be bought by multiple consumers (e.g., multicast), then the condition for C (to have X exist) gets relaxed since the service provider's gain becomes larger like $C + m(p+q)N$, where m is the number of potential consumers.

In real world scenarios, the service provider can make revenue from merchants who want to display their advertisements or coupons to targeted customers around the location of specific PA. Also, navigation software/hardware manufacturers can integrate the presented parking sharing system into their navigation service to better serve drivers. Besides commercial uses, the historical collection of these parking transactions becomes valuable sources of data for urban and smart city planning for municipals once the chosen PA is correlated with driver's origin and destination.

V. Evaluation

In this section, we evaluate the performance of *TruCentive*. We evaluate the consumer behavior under our incentive protocol through a user study and further explore how to set incentive parameters under practical settings. More specifically, we evaluate the fraction of rational users via a user study and demonstrate the relation between reward and truth-telling. We also evaluate how a service provider can set the bonus despite the presence of dishonest consumers.

A. User study: consumer behavior under *TruCentive* Incentives

The goal of this user study is to understand whether users in *TruCentive* choose to play honestly in the system given different probabilities of reselling a parking spot. Our first hypothesis is that participants can quickly perceive the chance of reselling and play rationally according to the chance. In other words, when the chance of resell is high, they would choose to resell to earn extra reward, otherwise they would choose to not resell. Our second hypothesis is that a consumer can quickly determine the strategy that leads to the best outcome and we want to study how quickly can a consumer make a decision.

The experiment setting is as follows. We utilized Amazon Mechanical Turk (AMT) [12] as the experimental platform for this experiment where we recruited 131 participants from AMT for this study. We designed a web-based questionnaire that replicates the actual scenario of consumer confirmation as follows: participants assume that they have already paid \$2 for trading a parking spot. Depending on whether they successfully park or not, they can either claim a partial refund of \$1 or re-sell the traded parking spot. Re-selling gives them a base reward of \$0.20 and a bonus of \$2 if re-selling leads to another successful parking. For each participant, the probability that they can re-sell their parking spot successfully, denoted as p, is set a priori but hidden from participants. The re-selling probability is randomly chosen from a set of 0.1, 0.3, 0.5, 0.7, and 0.9 uniformly. Note that changing the probability p effectively changes the expected bonus pX. Also note that when the probability is 0.4, the gain of re-selling becomes the same as partial refund. Rational users would choose to re-sell against refund for probability greater than 0.4. In order to observe how participants converge to a final decision, we asked each participant to answer this survey question 15 times. Each time, the re-selling result is randomly generated based on the preset probability p. Since one of our goals is to test how quickly users can converge to the best strategy, we ask participants to maximize their gain during 15 rounds and give them a \$5 bonus for their achievements. We divide the 15 round responses into three groups: 1-5 rounds, 6-10 rounds, and 11-15 rounds in the following evaluation.

Figure 3 illustrates how users behave in terms of rationality when p varies. From this figure, we first find that users can roughly perceive the resell probability, and that is good enough for them to make rational decisions. When the resell probability is as high as 90%, over 90% of the participants chose to resell their parking spots after 10 rounds, and all of them chose to resell after 15 rounds. When the resell probability is as low as 10%, over 75% of participants chose to not resell parking spots after 15 rounds.

From Figure 3, we also find that people can learn the 0.4 threshold quickly. We observe that the fraction of honest confirmation remains almost constant for the first five rounds regardless of the reward amount because users are trying to infer the underlying hidden probabilities. But after 15 rounds,

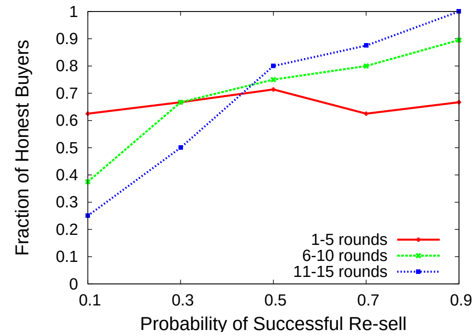

Fig. 3. User Study of rational behavior of consumers under *TruCentive* incentive protocol using Amazon Mechanical Turk.

people learn the best strategy: when p is 0.1, around 80% participants choose lying, which is the best strategy for a rational user in this case. However, when p approaches 0.9, almost all participants choose honest confirmation, which is again the best strategy for a rational user in this case.

Unlike X and p both preset in the above study, p is entirely determined by the market of consumers and contributors in a real deployment and X is the system operational variable that a service provider can control. Hence, the service provider changes the value of X dynamically such that it would lead to quick convergence to the honest strategy as indicated in the user study.

B. Simulation study: practical incentive protocol parameters

We now study how to set bonuses when the fraction of rational users (or honest confirmation) varies. In our simulation, we assume that the budget from service provider is \$0.10 per PA message, and the price for a PA message is \$2. Following the steps presented in the example in §IV, we choose $D = \$0.20, R = \$0.30$, and $X = \$0.30 + \frac{\$0.20}{\hat{p}}$, where the value of $\hat{p}$ denotes the estimated value of p and X is chosen to make the bonus meet the truth-telling constraint and budget constraint. In our simulator, the value of $\hat{p}$ is predicted by a service provider by counting the number of successful parking events over the previous 10,000 PA messages.

Figure 4 shows the lower and upper bounds for bonus and the actual bonus setting in our simulation. We vary the fraction of rational users from 1% to 100% with a step 1%, and for each fraction setting, we simulate 100,000 PA transactions. This figure shows that bonus is correctly set within its bounds despite the presence of dishonest users. Meanwhile, we can also conclude from this figure that when the fraction of honest consumers is over 50%, the value of bonus is relatively stable, implying that a fixed bonus is robust to variation in the fraction of honest consumers.

VI. Conclusion

Crowdsourcing has seen considerable interest in recent years and has been seen as a potential solution for a wide range of technical problems including parking in urban areas. However, realizing the promise of crowdsourcing necessitates that we

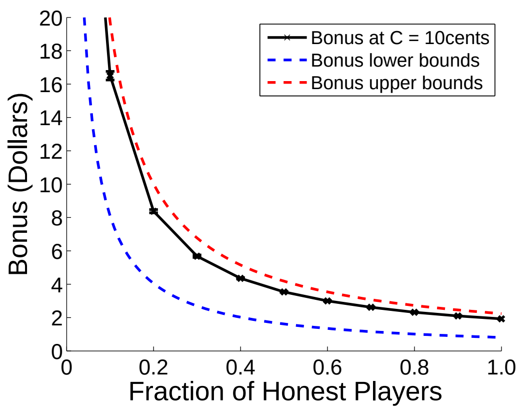

Fig. 4. Relation between bonus setting and the fraction of rational consumers in *TruCentive*.

tackle thorny technical problems including incentive design, authenticity of information, precise localization, and a plethora of other real-world issues. This paper specifically highlights the design of incentive platform to promote participants to contribute more valuable parking information. We address the challenges using a combination of incentive protocol design, game-theoretical and cost-benefit analysis. Specifically, we use game-theoretic framework for ensuring users to tell the truth about parking events, thereby achieving a feedback system that identifies the utility of parking information. While this paper presents a specific combination of approaches to address the parking problem, we believe that the methodology that we follow is more widely applicable to other mobile crowdsourcing applications.

The presented game-theoretic framework can be relatively simple to reflect the dominant factors affecting the behavior of drivers in real world parking scenarios. Addressing the modification of the framework suitable for real-world scenarios by conducting a user study remains future work.

REFERENCES

[1] Facebook credits. http://www.facebook.com/credits/.

[2] R. Arnott and E. Inci. An integrated model of downtown parking and traffic congestion. Journal of Urban Economics, 60:418–442, November 2006.

[3] N. Bilton. Finding That Prime Parking Spot With Primospot. http://bits.blogs.nytimes.com/2009/12/03/finding-that-prime-parking-spot-with-primospot/.

[4] J. V. Derbeken. Fatal stabbing over parking. http://articles.sfgate.com/2006-09-19/bay-area/17312921_1_parking-space-parking-spot-stabbed.

[5] C. for Research on Computation and Society. Trustworthy crowdsourcing. http://crcs.seas.harvard.edu/research-areas/.

[6] J. Kincaid. Googles Open Spot Makes Parking A Breeze, Assuming Everyone Turns Into A Good Samaritan. http://techcrunch.com/2010/07/09/google-parking-open-spot/.

[7] J. Lee and B. Hoh. Sell your experiences: A market mechanism based incentive for participatory sensing. In Percom, pages 60–68. IEEE Computer Society, 2010.

[8] 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 ACM MobiSys, 2010.

[9] M. Musthag, A. Raij, D. Ganesan, S. Kumar, and S. Shiffman. Exploring micro-incentive strategies for participant compensation in high-burden studies. In Proceedings of the 13th international conference on Ubiquitous computing, UbiComp '11, pages 435–444, New York, NY, USA, 2011. ACM.

[10] U. D. of Transportation. Advanced parking management systems: A cross-cutting study. http://ntl.bts.gov/lib/jpodocs/repts_te/14318.htm.

[11] S. D. OTTOMANELLI M., DELL'ORCO M. Parking choice behaviour using possibility theory. Transportation Planning and Technology, 34:247–267, 2011.

[12] G. Paolacci, J. Chandler, and P. Ipeirotis. Running experiments on amazon mechanical turk. Judgment and Decision Making, 5, August 2010.

[13] S. Reddy, D. Estrin, M. Hansen, and M. Srivastava. Examining micro-payments for participatory sensing data collections. In Proceedings of the 12th ACM international conference on Ubiquitous computing, Ubicomp '10, pages 33–36, New York, NY, USA, 2010. ACM.

[14] SFMTA. SFPark - About the Project. http://sfpark.org/about-the-project/.

[15] D. Shoup. Cruising for parking. Transport Policy, 13(6):479–486, November 2006.

[16] D. Teodorovic and P. Lucic. Intelligent parking systems. European Journal of Operational Research, 175(3):1666–1681, 2006.

[17] T. Yan, B. Hoh, and D. Ganesan. Exploiting mobile sensing to enable automated crowdsourced parking availability sharing. Technical Report UM-CS-2012-014, University of Massachusetts, Amherst, 2012.

[18] Z. Zhai, D. Hachen, T. Kijewski-Correa, F. Shen, and G. Madey. Citizen engineering: Methods for "crowdsourcing" highly trustworthy results. Hawaii International Conference on System Sciences, 0:3406–3415, 2012.