P^3-Coupon: a Probabilistic System for Prompt and Privacy-preserving Electronic Coupon Distribution

Boying Zhang*, Jin Teng*, Xiaole Bai†, Zhimin Yang‡ and Dong Xuan*

*Department of Computer Science & Engineering
The Ohio State University
Columbus, OH, USA
{zhangboy, tengj, xuan}@cse.ohio-state.edu

†Department of Computer & Information Science
University of Massachusetts Dartmouth
North Dartmouth, MA, USA
xbai@umassd.edu

‡Microsoft Corporation
One Microsoft Way
Redmond, WA, USA
yang.1070@gmail.com

Abstract—In this paper, we propose P^3-coupon, a *Prompt* and *Privacy-preserving* electronic coupon distribution system based on a *Probabilistic* one-ownership forwarding algorithm. In this algorithm, only one hop of forwarder (coupon owner) information instead of the complete forwarder list is recorded to keep the coupon short and privacy-preserving. Such information is updated probabilistically following two ownership flipping models, namely, One-Flip and Always-Flip models. We also use a Bluetooth Service Discovery Protocol (SDP) toolkit to enable fast, configuration-free coupon exchange. We have implemented the system in Java ME. Our experiments on real world mobile phones, such as Nokia and Samsung phones, and large scale simulations show that our system is efficient in peer to peer coupon distribution and capable of massive deployment. We believe our key methodology can serve as a general framework for facilitating information propagation on mobile phones, which requires promptness and privacy protection.

Fig. 1. Electronic Coupon Working Flow

I. INTRODUCTION

A. Motivation

Mobile phones have become an inseparable part of our everyday life. This is particularly true for smartphones, which are equipped with powerful operating systems and are capable of almost all functionality that a traditional computer offers. Apart from phone calls and SMS services, we increasingly rely on them for information acquisition, social networking, entertainment, etc.

Nowadays, there is another emerging category of mobile phone applications: electronic coupons [12], [13]. These electronic coupons are the same as their paper counterparts and can be stored on mobile devices. People may download these coupons from Internet websites or they can receive them from peer-to-peer coupon pushing services. The peer-to-peer approach requires no infrastructure and can make use of people's social and spatial colocation for more efficient coupon distribution. The core idea behind this approach is that the people one encounters are likely to show similar interests. For example, one may meet strangers nearby heading for the store where he just received the coupons. In most cases, incentives, like credits or bonuses, may be needed to keep the coupons circulating. Thus some mechanism needs to be employed to record forwarder information along the path.

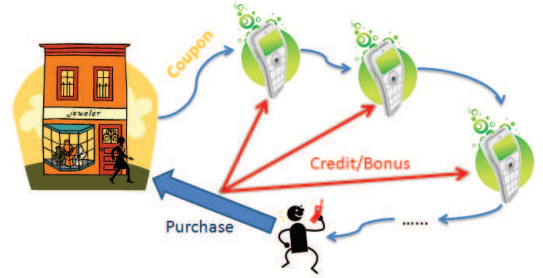A typical peer-to-peer coupon forwarding scenario is shown in Figure 1. A user walks out of a store where she just received a coupon and she stores it on her mobile phone. As she moves along the busy streets, she encounters many people, most of whom she just passes by. Some of those people receive the coupon through established wireless connections with her, such as WiFi, and might decide to continue forwarding. And every forwarder is registered on the coupon. Finally one of the receivers makes a purchase with the coupon. Then rewards are awarded to the initial user and the following forwarders.

This kind of peer-to-peer coupon distribution has recently attracted interest from the research community and several works follow the natural approach described above [6], [7]. They all feature coupon forwarding using established wireless connections among strangers and require recording the entire forwarding path. However, in actual implementation, there are two major problems:

- *Promptness:* Encounters between two strangers are generally very brief. As the majority of such encounters last less than one minute [20], information exchange must be prompt. Existing wireless technologies, such as WiFi and Bluetooth, require manual connection establishment and configuration before communications begin. These are greatly time-consuming as well as annoyingly obtrusive. Moreover, in the coupon forwarding systems, the entire forwarding path may be very long, and its transmission, together with that of the coupon itself, may not fit into an encounter's brief time. Thus a new type of prompt coupon forwarding with as minimal content and no requisite manual connection setup is very desirable.

– *Privacy:* The intuitive way of recording all forwarders on the coupon imposes a significant danger of privacy leakage, as users' trajectories and social contacts can be easily inferred in practice. On the other hand, establishing a wireless connection with strangers poses a risk in itself. Thus we need to design a set of mechanisms to provide privacy-preserving communications without sacrificing forwarder information.

B. Contributions

This paper presents the P^3-Coupon system, where P^3 stands for *Probabilistic*, *Prompt* and *Privacy-preserving*. It is a distributed mobile phone based system that enables prompt and privacy-preserving electronic coupon distribution. We make the following contributions in this paper:

– We have developed the P^3-Coupon system. The core of the system is a probabilistic one-ownership forwarding algorithm. Instead of the entire forwarding chain, the algorithm only keeps the information of one forwarder, who receives the entire reward and hence we term him the owner. This makes the whole coupon exchange process brief and privacy-preserving. When the coupon is relayed, its ownership probabilistically flips such that, statistically, the desired percentage of coupon bonuses goes to each forwarder. Two flipping models, i.e., the One-Flip model and the Always-Flip model, are proposed to satisfy different promotion strategies of individual stores.
– We have implemented the P^3-Coupon system on various types of mobile phones, such as Nokia and Samsung phones. We use our own Bluetooth toolkit to enable communications without connection establishment [18]. In P^3-Coupon, information is exchanged in a connection-less and configuration-free manner. Our entire implementation is based on Java ME, so the system can be easily deployed on most commercial off-the-shelf (COTS) mobile phones that are shipped with Bluetooth and Java.
– We have conducted extensive experiments and simulations to show that our system distributes coupons in a fast and secure way, and the bonuses are rewarded correctly given enough coupon redemptions. Specifically, our evaluations show that in an experimental community of 5,000 people, 50 coupons with a single forwarder information can achieve highly precise bonus distribution, as if all forwarders have been recorded.

Our P^3-Coupon system is the first practical coupon distribution platform capable of large-scale deployment. It incorporates several key elements to enable prompt and privacy-preserving information exchange in a probabilistic way, which is otherwise impossible and has never been realized in the e-commerce community. It is a general platform with many potential extensions. Apart from the above-mentioned e-commerce application, it can also be introduced into distribution of Lost and Found information as well as police bounty information.

The rest of the paper is organized as follows. Section II discusses related work. Section III elaborates on the detailed system design. Section IV presents our probabilistic forwarding algorithms. Section V describes the system implementation. Section VI presents experimental results and evaluations of our system. Section VII discusses the potential extensions and possible security considerations. Finally, we conclude in Section VIII.

II. RELATED WORK

In this section, we mainly focus on literature related to electronic coupons and mobile advertising.

Electronic coupons are popular and increasingly used by merchants such as Coca-Cola [12], Kentucky Fried Chicken [13], etc. Customers can download the coupon codes from store websites and save them in their shopper cards distributed by the stores. During actual purchases, customers can redeem the coupons by scanning their shopper cards.

The distribution of electronic coupons is studied by some existing works such as *eNcentive* [6] and *Coupons* [7]. Both *eNcentive* and *Coupons* present a similar coupon scheme that facilitates peer-to-peer electronic marketing in mobile ad-hoc environments. As a user passes by a retailer store, the coupon or promotion information will be pushed to the user's PDA and saved for future use. A user can also become a coupon distributor, and his PDA ID will be recorded as one of the users that distributed a particular coupon. Once the coupon is redeemed, the store can reward the distributors based on the recorded ID list. However, recording the entire list of the coupon distributors introduces large space overhead and induces privacy leakage. Several techniques have been proposed in different domains to reduce information storage space. The Bloom filter [16] is one space-efficient information coding technique for membership queries. In [15], Savage et al. introduced a probabilistic packet marking technique for IP trackback against DDoS attacks in the Internet. Our work shares the similarity with their work in terms of using the probabilistic approach to save space. However, the problems addressed in [15] and in our paper are different. The work in [15] focuses on IP traceback in the Internet, while our work aims to distribute bonuses according to a pre-determined distribution in a social community. Moreover, the goals of the two systems are different. The approach in [15] tries to recover the whole forwarding path to expose the attacking route. On the contrary, our work endeavors to hide the path to protect user privacy. Finally, different from the fixed marking probability approach used in [15], a dynamic ownership flipping probability mechanism, where every forwarder on the path uses a different flipping probability, is designed in our work to fit the desired reward distribution.

Other papers focus on developing targeted advertising delivery systems. Aalto et al. [1] propose B-MAD, a system for delivering permission-based location-aware advertisements to mobile phones. The system uses Bluetooth positioning and wireless application protocol (WAP) push. Kurkovsky and Harihar present a mobile advertising system named SMMART [2] whose purpose is to deliver targeted promotions to mobile users. Yuan and Tsao propose MALCR [3], a customized mobile advertising system that automatically learns user's preferences

by analyzing his advertisement browsing behavior. De Castro and Shimakawa present the Concierge [4] system that pushes all advertisements related to a user's preferences when he is near the advertised items. Ad-Me [8] and MoMa [5] are similar advertising systems that can deliver advertisements to users in stores. Unlike our work, the aforementioned researchers only focus on local information distribution in the store, their works are not concerned with large-scale distribution afterwards.

In addition to the design of mobile advertising systems, researchers study strategies for effective marketing. Among these strategies, the influence maximization problem [9]–[11], [14] attracts much research interest. The problem is deciding which users to select as initial users in order to influence the largest number of people in a given network. Domingos and Richardson [9], [10] are the first to study influence maximization as an algorithmic problem. Kempe, Kleinberg, and Tardos [11] formulate the problem as a discrete optimization problem and address it using a greedy algorithm. However, their work does not scale to large deployments. To address this issue, [14] proposes a scalable influence maximization scheme for viral marketing in large-scale social networks. Our work differs from the above work in that we do not rely on any underlying social network to maximize coupon influence. Rather, we focus on efficiently distributing coupons in a mobile ad hoc environment.

III. System Design

In this section, we introduce the detailed design of P^3-Coupon. We first discuss the system design rationale, then present the system architecture followed by the P^3-Coupon system workflow.

A. Design Rationale

We have mentioned two major challenges in Section I in realizing the P^3-Coupon system, i.e., promptly exchanging information without manual connection setup and forwarding coupons while preserving privacy. We use the following techniques to address these two challenges.

Our key technique is probabilistic sampling on the forwarding chain. We sample it and record only one forwarder in one coupon, whom we term the owner. The entire reward for a given coupon is given to the owner. Since we only have information about one forwarder, the coupon size is kept short and the privacy implied in the forwarding chain will not be leaked. In order to determine who forwards how many redeemed coupons, the ownership is flipped with pre-calculated probabilities at each hop such that each user is statistically associated with the correct percentage of coupons and, hence, receives the correct reward amount.

In a real human society, there are many chances for interpersonal encounters [21]. Given several dozen coupons circulating in our P^3-Coupon system, there will be enough coupon redemptions to reveal the coupon association distribution, by which stores can fairly mete out rewards. Actually, we will see later that with about 50 coupons circulating in a community with 5,000 people, we can fairly distribute the correct reward amount to each user just as if all forwarders' information
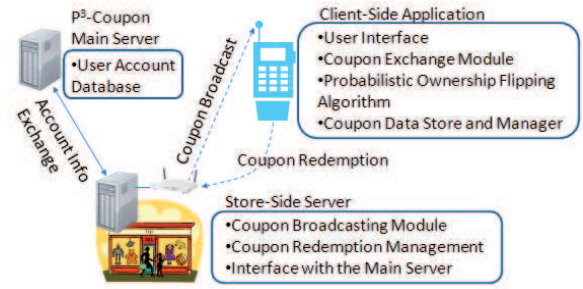

Fig. 2. System Architecture

was recorded. We believe that if the P^3-Coupon platform is deployed, it is very likely that there will be far more than 50 coupons and the population may well exceed 5,000 people, which makes the probabilistic reward approximation very accurate.

It is also worth noting that different stores may have different promotion strategies. Some stores may prefer designating a fixed amount of money to users on different hops, while others may prefer a relative amount decided by the total length of a forwarding chain, e.g., equal share for every forwarder. We need to develop mathematical foundations to support these economic models.

An auxiliary technique we use in this paper is connectionless information exchange enabled by Bluetooth SDP, a technique that we incorporated into a toolkit [18]. We encode the information into the service name of Bluetooth devices. Thus some information can be exchanged via the service names of nearby Bluetooth devices. This kind of communication is configuration free and we can dispense with the obtrusive and time-consuming connection setup phase. Meanwhile, as no connection is actually established, the risk of attack is much lower than if a connection was established.

B. System Architecture

The system architecture of P^3-Coupon is shown in Figure 2. The P^3-Coupon main server stores all users' registration information, including account names and balances. It will exchange information with store-side servers to update reward information for all users.

On the store side, there is a central server governing all coupons related to a specific store. This server is responsible for broadcasting the coupon through a wireless interface to customers. It will also read the redeemed coupon information through an interface in the store and give the designated reward to the coupon owner based on his account in the P^3-Coupon system. Essentially, it is a database recording all related coupons that has access to user accounts on the P^3-Coupon main server.

On the client side, the major components are the coupon data store, the coupon forwarding manager, the coupon exchange and the user interface. We describe the functions of each component as follows:

– *Coupon Data Store:* This component stores coupons collected by users and metadata that configures the system environment. The stored coupon information has two parts. The first part is static and records the product description and discount information. The second part is dynamic and is updated by the Coupon Manager after each forwarding between two users. The coupon information format will be described in Section V.

– *Coupon Forwarding Manager:* This component updates coupon information according to our probabilistic coupon forwarding scheme. The coupon manager will change the ownership according to the flipping probability function. If duplicate coupons are received, the coupon manager will only keep the earliest one. The coupon manager also performs simple verification to check whether the coupon is forged.

– *Coupon Exchange:* This component supports automatic coupon exchange via any short-range communication technology. In our design, Bluetooth is the primary communication tool, and the toolkit developed in [18] is used here. We use a Bluetooth SDP server to create and publish a service record with coupon information as service attribute values. This component can also be extended to WiFi and other short-range communication technologies.

– *User Interface:* This component provides interfaces for users to perform basic coupons operations. Users can search for interesting coupons in physical proximity, search for and delete coupons on their mobile phones, and select coupons to publish. In addition, all these operations can be preset and performed automatically via user interface configuration.

C. Workflow

The basic workflow is as follows. Initially, the store issues a coupon for product P and broadcasts it to nearby customers. When user A stops by the store, his P^3-Coupon system's Coupon Exchange component discovers the coupon and saves it in the Coupon Data Store. Later, user A encounters user B at some place within short-range communication distance. The Coupon Exchange component on both mobile phones will automatically exchange A and B's coupon information. Meanwhile, the Coupon Forwarding Manager changes the related coupon information in A and B's P^3-Coupon system. The coupon relaying process will continue when A or B meets other P^3-Coupon users, until user X buys the product P and redeems the coupon. Based on the owner information recorded in the electronic coupon, the store will give some reward to one of the forwarders. Finally, each user will get the expected reward amount after forwarding enough coupons.

IV. Probabilistic Forwarding Algorithm

In this section, we introduce our probabilistic one-ownership coupon forwarding algorithm. The purpose of this algorithm is to ensure the flipping leads to the same ownership percentage for users at each hop as the store-designated reward percentage. The algorithm has a variation, which we will expound later in this section and compare with the original version.

A. Algorithm Description

The key idea of our algorithm is based on an "Always-Flip model", in which coupon ownership keeps flipping with a certain probability when the coupon is forwarded along a relay path. Each user has a certain probability to flip the current coupon ownership to himself after forwarding the coupon. An ownership flipping probability function is calculated for each hop according to the designated reward schemes. Each user's reward is simply the superposition of his rewards on different coupon forwarding paths.

Algorithm 1 summarizes our method in terms of the P^3-Coupon workflow described in Section III. Initially, the store will broadcast coupons to nearby users and set them as the current coupon owners. When these seed users encounter other users, either friends or strangers, they can exchange coupons following the *Forwarding* part of our algorithm. This algorithm will be executed on every user's device. The algorithm can be applied to two different communication scenarios. In the unicast mode, users can use Bluetooth to forward coupons to nearby users one by one. In the broadcast mode, users can use WiFi or other short-range communication technologies to forward coupons to all nearby users at once.

To illustrate how the ownership flipping probability function is decided, we specify the following store reward scheme: The store fixes a total reward R and requires that the reward be distributed to all users on the forwarding path. In particular, a user at hop i gets k times the bonus given to a user at hop $i+1$. To realize the above scheme, we calculate the ownership flipping probability P_i using the following formula:

$$P_i = \begin{cases} 1, & \text{if } i = 1 \\ \frac{P_{i-1}}{k+P_{i-1}}, & \text{if } i > 1 \end{cases}. \tag{1}$$

The above formula is derived from the following calculation. First, we calculate the probability that a user at hop i would be the final coupon owner, which is given by:

$$S_i = P_i \prod_{j=i+1}^{n} (1 - P_j). \tag{2}$$

where n is the hop count. Then, based on the fact that the probability S_i is proportional to the corresponding user's reward, we have:

$$S_i = kS_{i+1}. \tag{3}$$

Combining Formula (2) and Formula (3), we can easily get Formula (1). We use the following example to show the application of Formula 1.

Example 1: The store sets the base reward R to be \$10 and wants every forwarder to get an equal share ($k = 1$). Suppose there are three forwarders, then User 1 ,2, 3 should each get \$10/3 = \$3.33. According to formula 1, User 1 flips the ownership of all coupons with possibility $P_1 = 100\%$. The coupon ownership can be continuously flipped after this. Then, User 2 flips the ownership with possibility $P_2 = P_1/(k+P_1) = 100\%/(1 + 100\%) = 50\%$. Finally, User 3 flips the ownership with $P_3 = P_2/(k + P_2) = 50\%/(1 + 50\%) = 33.3\%$.

Algorithm 1 Probabilistic Coupon Forwarding

1: *Variables*:
2: ownership flipping probability: p;
3: hop count: n;
4:
5: *Initialization*:
6: let $SetU$ be a group of user near the store S;
7: **for** each user u_i in $SetU$
8: S sets u_i as *owner* of an electronic coupon c;
9: S sends c to u_i;
10: **end for**
11:
12: *Forwarding*:
13: **for** each user u_i
14: let N_i be the set of users in the neighborhood of u_i
15: **if** $|N_i| > 0$ **then**
16:   u_i in *Unicast Mode*:
17:    **for** each user u_j in N_i
18:     u_i forwards c to u_j;
19:     let x be a random number from [0..1);
20:     $n \leftarrow n + 1$;
21:     set p according to formula 1;
22:     **if** $x < p$ **then**
23:      u_i is set as *owner* of c;
24:     **end if**
25:    **end for**
26:   u_i in *Broadcast Mode*:
27:    u_i forwards c to each u_j in N_i;
28:    let x be a random number from [0..1);
29:    $n \leftarrow n + 1$;
30:    set p according to formula 1;
31:    **if** $x < p$ **then**
32:     u_i is set as *owner* of c;
33:    **end if**
34: **end if**
35: **end for**

B. Extensions

We extend the Always-Flip model in Algorithm 1 and propose the One-Flip model. The key idea of this model is that once a coupon's ownership is flipped, it remains the same on a forwarding path. Unlike the first model, this model is good at distributing an absolute amount of bonuses irrelevant of the number of following forwarders, e.g., the first hop user gets 10%, the second hop user gets 5% etc.

To specify this model, we propose another possible store reward scheme as follows: The store fixes a base reward R and assigns a percentage q_i to the forwarder on each hop of the forwarding path. Then, a user at hop i can get a total reward of $R * q_i$. To implement the above scheme, we calculate the ownership flipping probability P_i using the following formula:

$$P_i = \begin{cases} q_i, & \text{if } i = 1 \\ \frac{q_i}{1 - \sum_{j=1}^{i-1} q_j}, & \text{if } i > 1 \end{cases} . \quad (4)$$

This formula means that when a number of coupons flow into the forwarding paths, a user at hop i can only flip the ownership of $1 - \sum_{j=1}^{i-1} q_j$ percent of all coupons with probability P_i. The total reward amount meted out is upper-bounded by R, but not lower-bounded, depending on the series of q_i. The following example illustrates how the flipping probability is set up.

Example 2: The store sets the base reward R to be \$10 and wants every forwarder to get \$1 ($q_i = 10\%$) of this bonus. In particular, the store wants the first hop user to get more, i.e, \$3 ($q_1 = 30\%$). Suppose there are three forwarders. According to Formula (4), User 1 flips the ownerships of all coupons with possibility $P_1 = 30\%$. After the flipping, 30% of the coupons' ownership cannot be flipped again. Then, User 2 flips the remaining coupons with possibility $P_2 = 10\%/(1 - 30\%) = 14.3\%$. Finally, User 3 flips the remaining coupons with $P_3 = 10\%/(1-30\%-10\%) = 16.7\%$.

C. Comparison

The two rewarding schemes are complementary to each other. The Always-Flip model is a scheme for a relative amount of reward. Forwarders on the path cannot know how much reward they will get until the coupon is redeemed. They can only tell the relative reward compared with his 'parent' and 'child' forwarders. On the other hand, the One-Flip model is a scheme for an absolute amount of reward. Forwarders on the path know for sure how much reward they will get if the forwarded coupon is redeemed. The Always-Flip model works better for general commercial products when the purchase probability of each user is unknown or small. The One-Flip model suits the need of popular commodities, where the store can predict with high probability in how many hops the coupon is very likely to be redeemed. In other words, when the forwarding path is relatively short, then the stores can achieve finer control of the reward with the One-Flip model.

There are several potential attacks against our rewarding schemes whose primary goal is to get an unfair share of the rewards. We briefly classify these attacks into two types. The first type is relatively simple as malicious users could only tamper with the coupon information outside of our system. The second type is more sophisticated as malicious users could break into our system and change the coupon ownership from inside. For the simple attack, we implement a standard encryption scheme to provide a certain level of protection for our rewarding schemes. For the sophisticated attack, our schemes exhibit varying degrees of robustness. We quantify it with a representative example in which the store reward R is expected to be evenly distributed to the n forwarders on the coupon forwarding path. Without loss of generality, the attacker can randomly be one of the forwarders with probability $1/n$. According to Formula (1) - (4), the unfair (extra) rewards gained by the attacker is $(n - 1) \cdot R/2n$ in the Always-Flip model and $(n - 1) \cdot R/n$ in the One-Flip model. Since the Always-Flip model gives away less rewards to the attacker, it is more robust than the One-Flip model.

As a potential solution for the One-Flip model, a certain percentage of sample coupons can be set up to detect the tampering. Specifically, the ownership of these coupons is flipped in advance. If any coupon's ownership is flipped again, its current owner will be blacklisted and corresponding actions will be taken. One may argue that the forwarders of these
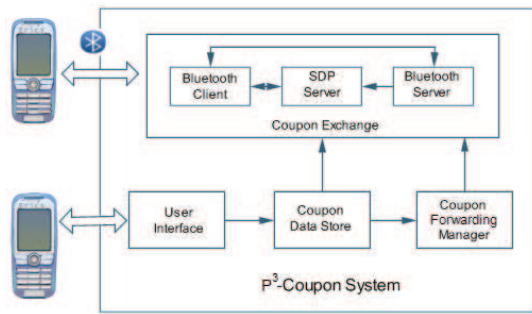
Fig. 3. Detailed Implementation of P^3 Coupon System

Fig. 4. Format of An Electronic Coupon

detection coupons might exert efforts in vain. To address this issue, we can use more tolerant blacklisting mechanisms, such as AREX [17]. AREX is a game theory based blacklisting mechanism that forces attackers to attack less often. Based on AREX, we can compensate the cost on detection coupons by dynamically adjusting the rewards for benign users and malicious users. In our future work, we plan to explore other approaches to protect our rewarding schemes against these simple and sophisticated attacks.

V. IMPLEMENTATION

We develop the prototype of the P^3-Coupon system on real-world mobile phones which are equipped with Bluetooth and the Symbian S60 platform that supports application development. In the following, we describe our prototype's implementation details.

We implement our system with the Eclipse SDK and the Sun Java Wireless Toolkit for Connected Limited Device Configuration (CLDC) based on the Mobile Information Device Profile (MIDP) specification. The size of the MIDlet application JAR file is about 1390KB. It consists of 17 class files, which can be divided into four major components: user interface and system configuration, coupon data store, coupon manager and coupon exchange. The detailed implementation is shown in Figure 3.

For implementing nearby device discovery and the P^3 Coupon services, our system explores the Bluetooth SDP structure and uses the toolkit [18] to publish coupon information such that we do not need a Bluetooth connection for information exchange between two users. More specifically, we generate a service record by using a known UUID to initiate a virtual service. Each service record contains a set of attributes. Each attribute has its own attribute ID and attribute value. We use attribute values to represent coupon information. This service is "virtual" because a client cannot actually consume this service. The Coupon Exchange component in Figure 3 shows an overview of Bluetooth-based communication in our system. The Bluetooth server publishes coupon information through the phone's SDP server. The Bluetooth client acquires the other phone's coupon information by sending a SDP request to its SDP server.

Figure 4 illustrates our electronic coupon data structure. There are two information parts. The first part contains the coupon description, including the product description, the discounts, the coupon's start and end date, the coupon code, and the coupon issuer. Our store-side server generates a digital signature to protect this information part from forgery and abuse. The second part contains the coupon forwarder information, indicating the current coupon owner. Note that coupon ownership is automatically changed according to our flipping models. To prevent the simple attack mentioned in Section IV, our client-side application encrypts and decrypts the entire coupon using a symmetric key P_k. The same key is stored in the program of each client-side application. In our implementation, we use 60 bytes for coupon description, 3 bytes for coupon forwarder information, and 64 bytes for coupon encryption. The total size is 127 bytes which fits into one Bluetooth SDP service record on most mobile phones.

VI. EVALUATIONS

In this section, we report and discuss the performance evaluation results of our prototype system. We use two metrics to evaluate P^3-Coupon's performance: system performance and large-scale distribution performance. System performance includes coupon forwarding time and power consumption. Large-scale distribution performance includes the coupon distribution time with different coupon numbers, population sizes and initial seed sizes. To evaluate the system performance, we conduct experiments on real-world mobile phones. To evaluate large-scale distribution performance, we need to run a massive field test, which is costly and time-consuming. To solve this issue, we perform simulation tests to simulate coupon distribution among a massive population.

A. Experimental Evaluations

In our experiments, we use 6 mobile phones (2 Nokia N82, 2 Nokia 6650, 1 Nokia E71x and 1 Samsung SGH-i550). We repeat each experiment 10 times and average the results for performance stability.

1) Coupon Forwarding Time: We define the coupon forwarding time as the period from starting a search for nearby devices to the time of finishing coupon exchange with a nearby user. To explore how different factors affect the coupon forwarding time, we conduct two sets of experiments.

In the first set of experiments, we only let one coupon circulate in the system, and we examine how the coupon
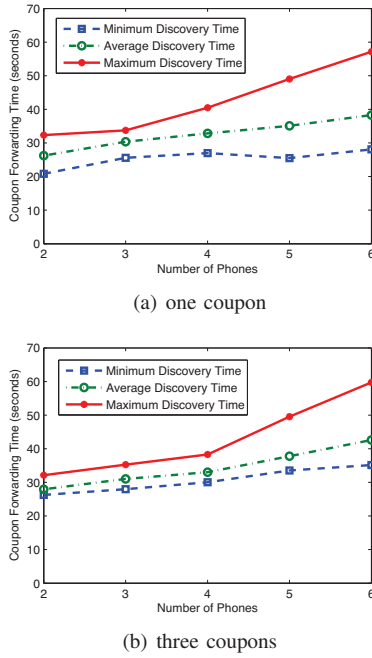
(a) one coupon

(b) three coupons

Fig. 5. Coupon forwarding time

discovery time varies according to the number of nearby P^3-Coupon users. According to our experience in [18], we place two phones about 5 meters apart to simulate the effect of people encountering each other in physical proximity. We set the Bluetooth search intervals to 60 seconds, because the difference on search interval does not significantly impact coupon discovery. We progressively increase the number of phones from 2 to 6, then record the time for coupon discovery between two phones.

Figures 5 (a) shows the minimum, average and maximum coupon discovery time. From the trend on the figure, we can see that the coupon discovery time increases as the number of phones increases. A possible reason to explain the fact is that multiple phones might increase encounter collisions, hence the larger discovery time and number of discovery failures.

In the second set of experiments, we examine how the number of coupons impacts the coupon discovery time. For this purpose, we repeat the previous experiments by changing the number of coupons circulating in the system to 3. The experiment results are shown in Figure 5 (b). Figure 5 (b) shows a similar trend to Figure 5 (a). This implies that the number of coupons does not significantly impact coupon discovery time. A possible explanation is that coupon discovery time is dominated by Bluetooth device discovery time.

In summary, the minimum, average and maximum coupon discovery times in our experiments are 20.82 s, 33.52 s, and 59.37 s, respectively, among all the P^3-Coupon data we recorded. We use the average coupon discovery time as a threshold to determine whether two encountered users have time to exchange a coupon in our simulation experiments.

2) Power Consumption: To study trends regarding power consumption, we place two Nokia N82 phones 3 meters apart,

both fully charged. We record the run time of P^3-Coupon on a fully charged N82 phone (1050 mAh) until the battery is exhausted. In our experiment, we turn on the Bluetooth radios and let the P^3-Coupon run as a background application. When the Bluetooth search interval is set to 60 seconds, the phone runs slightly longer than 25 hours. Using the same settings on the Bluetooth radios, we redo the experiment without running P^3-Coupon. In this case, the phone runs 32 hours until the battery is depleted. This result is encouraging since there is no major difference between turning our application on and off. Thus users can run our system in the background.

B. Simulation Evaluations

It is difficult to conduct real user experiments at large scale. Hence, we conduct extensive simulations to further validate the performance of our scheme over large volume of people. The simulator is written in Java. Each person is instantiated as an object running over one thread. We conduct each simulation 100 times and average the results.

We adopt the human mobility model proposed by Song and Kotz in [19]. Based on the realistic contact traces of a production network with more than $5,000$ users, this model integrates human mobility regularity by associating the probability of future encounter with the encounter history.

Following this model, we divide the encounter history of any two persons A and B in ΔT into a sequence of n periods starting from 0 to $n-1$. If A had any encounter with B during a given period i, we mark 1 on A's contact status S_i; otherwise, we mark 0 on the contact status. The probability p_{AB} that A directly meets B in the next ΔT can be estimated as the average of the contact status in prior intervals, i.e., $p_{AB} = \Sigma_{i=0}^{n-1} S_i / n$.

Our simulation also integrates the contact duration time distribution model. We use the model proposed by Wang et al. in [20]. It shows that the average contact duration is around 370 seconds, and that the overall contact duration time follows a Pareto distribution. Such model is from the real traces that were collected by $12,332$ unique Bluetooth devices. In addition, we apply the dataset of real trace presented by [21] to infer timescales from encounter times. Data in [21] shows on average one person encounters other people 19 times per day.

Simulation results are reported in Fig. 6 (a)$-$(f).

Fig. 6 (a) shows coupons can be delivered to 5,000 people within 32 hours given that there is only one user who initially has the coupons. We conduct the same experiment on 20,000 people. The result in Fig. 6 (b) shows all coupons are delivered within 40 hours. In this figure, we can see that coupon holder number increases exponentially as time elapses. Such a trend indicates our system helps coupons distribute faster when more people are involved.

Fig. 6 (c) presents the saturation time, i.e., the time for 20 coupons to reach 5,000 users. The trend in the figure shows that slightly increasing the seed number at the beginning will dramatically improve the propagation speed. Such phenomenon provides strong motivation for stores to increase the number of initial coupon holder especially at the early stage.
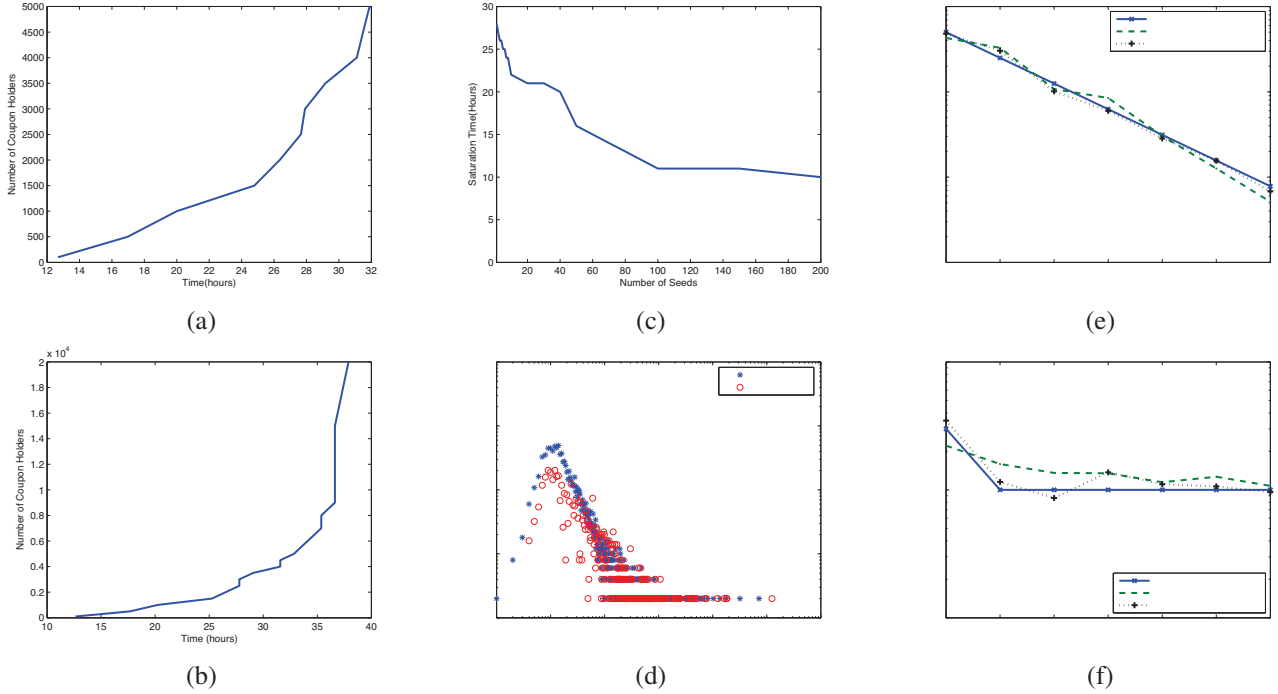
Fig. 6. (a) Number of coupon holders vs. Time (5,000 people). (b) Number of coupon holders vs. Time (20,000 people). (c) Distribution saturation time vs. Number of seeds. (d) Coupon ownership distribution(100 coupons). (e) Deviation between theoretical and actual reward based on different number of coupons in the Always-Flip model. The theoretical value is the designated percentage of reward due to users on each hop. (f) Deviation between theoretical and actual bonus based on different number of coupons in the One-Flip model.

In Fig. 6 (d), each point ("∗" or "◦") represents the percentage (y axis) of the people who obtain the ownership of a certain number of coupons (x axis). We simulate the case when 100 coupons are propagated by our system among 5,000 people starting from a single seed user. Note that in this simulation, we study one propagation structure, i.e., one situation in which people encounter with each other and propagate the coupons. In our simulation, the result shows that the percentage first increases as the ownership number gets large and then drops after a threshold value (10 in this case). For people who have more than 10 coupons, the percentage distribution follows the power law. This phenomenon holds for both the Always-Flip model and the One-Flip model. Our result further shows that for one specific propagation structure, different people may have different percentages of ownership. Thus, the fairness of each person having the same chance to get the reward must be guaranteed by multiple propagation structures. To achieve this, stores could distribute different coupons to different seed users.

Fig. 6 (e) and (f) respectively illustrate for the Always-Flip model and the One-Flip model, the deviation between theoretical reward distribution (that is calculated by Formula 1 and 4 in Section IV) and the reward distribution obtained in large scale simulation. The reward distribution is represented by the probability of a person to be the final owner of a redeemed coupon when he is at a certain position of the coupon's propagation chain. We simulate our coupon propagation system among 5,000 people. We set $k = 2$ for the Always-Flip model such that theoretically, any nodes's reward will be twice the

reward of its direct successor. We set 30% for the first hop and 10% for the remaining hops in the One-Flip model so that the first hop user receives greater reward and the remaining users evenly share the reward along the propagation chain. Simulation results shows that, when there are only 50 coupons circulating in the system, the deviation between theoretical and actual reward is already small. When the coupon number reaches 100, the reward distribution almost converges to the theoretical values. When there are more than 5,000 people to propagate the coupons, the deviation will become even smaller. This reflects the soundness of our system design.

VII. DISCUSSION

In this section, we discuss the potential extensions of our work and possible system security considerations.

A. Potential Extensions

First, we could include progressive rewards for sales as a new incentive to encourage user's forwarding behavior. In our current design, the Always-Flip and One-Flip models still cannot accommodate another widely adopted incentive scheme that relies on a progressive percentage change on bonus distribution. A representative case is Amway's multi-level marketing scheme, which gives higher reward percentages to users who entice more people to purchase products. To incorporate this scheme, we can add several bits of information to a coupon. The correlation of these bits across all coupons can reveal, at least approximately, a certain user's sales amount.

Second, user's decision could also be included in our future design to increase the elasticity of our system. In other words, there should be a criterion by which users can decide whether to relay the coupon or not. For example, one usable criterion could be a (expected) threshold on bonus or device remaining energy. Although it is hard to find a perfect and comprehensive scheme, we could still generate some heuristics to facilitate this. Also we could move the reward distribution procedure from store server to client's devices. In P^3-Coupon, each store should take care of its reward distribution. However, to make the reward distribution more effective and efficient, we should let the users cooperate to negotiate a commonly accepted reward distribution scheme, such as the incentive mechanism in [22].

B. Security Considerations

Our design provides to protect system security to some extent with the proposed probabilistic forwarding algorithm. By keeping only one owner for each coupon, we greatly reduce the possibility for malicious attackers to add or delete nodes on the coupon forwarding path. In our future work, we plan to further enhance our system's security. Currently, the private key P_k mentioned in Section V is hard coded in the client-side application. All clients have the same private key and they do not need to dynamically share the key with other clients. However, there may be a vulnerability through memory analysis: An attacker could just retrieve the key from memory and use it to decrypt and modify the coupon owner. We plan to explore some existing strategies that can effectively encrypt or hide the private key in our program, so that it takes much effort for an attacker to retrieve the key.

VIII. Final Remarks

In this paper, we have proposed P^3-Coupon, a *Prompt* and *Privacy-preserving* electronic coupon forwarding system based on a *Probabilistic* forwarding algorithm. We aimed to enhance the distribution promptness as well as privacy protection in the electronic coupon forwarding system. In order to achieve this, we made use of a key technique, namely a probabilistic ownership flipping algorithm that records only one forwarder. We also used a Bluetooth SDP toolkit that we developed to realize configuration-free coupon forwarding. We have shown with real implementation and large scale simulations that the P^3-Coupon system lives up to all the initial design goals.

Acknowledgment

References

[1] L. Aalto, N. Gohlin, J. Korhonen and T. Ojala. "Bluetooth and WAP Push-Based Location-Aware Mobile Advertising System". In *Proceedings of the 2nd International Conference on Mobile Systems, Application, and Services*. pp. 49–58. ACM, New York, USA, 2004

[2] S. Kurkovsky, K. Harihar. "Using Ubiquitous Computing in Interactive Mobile Marketing". In *Personal and Ubiquitous Computing*, Volume 10, Issue 4, pp. 227–240, 2006

[3] S. T. Yuan and Y. W. Tsao. "A Recommendation Mechanism for Contextualized Mobile Advertising". In *Expert Systems with Applications*. Volume 24, Issue 4, pp. 399–414, 2003

[4] J.E. De Castro and H. Shimakawa. "Mobile advertisement system utilizing user's contextual information". In *Proceedings of the 7th International Conference on Mobile Data Management*. pp. 91. IEEE Computer Society, Washington, DC, USA, 2006

[5] R. Bulander, M. Decker, B. Kolmel, and G. Schiefer. "Enabling personalized and context sensitive mobile advertising while guaranteeing data protection". In *Proceedings of EURO mGOV 2005*. pp. 445C454, Mobile Government Consortium International LLC, Brighton, UK, 2005.

[6] O. Ratsimor, T. Finin, A. Joshi, and Y. Yesha. "encentive: a framework for intelligent marketing in mobile peer-to-peer environments". In *Proceedings of the 5th international conference on Electronic commerce*. pp. 87–94, ACM Press, New York, NY, USA, 2003

[7] A. Garyfalos and K. C. Almeroth, "Coupons: A multilevel incentive scheme for information dissemination in mobile networks". In *IEEE Trans. Mobile Comput.*. vol. 7, no. 6, pp. 792–804, Jun. 2008.

[8] N. Hristova, and G. O'Hare. "Ad-me: Wireless Advertising Adapted to the User Location, Device and Emotions". In *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*, pp. 285–294. IEEE Computer Society, Washington, USA, 2004

[9] P. Domingos and M. Richardson. "Mining the network value of customers". In *Proceedings of the 7th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 57C66, 2001.

[10] M. Richardson and P. Domingos. "Mining knowledgesharing sites for viral marketing". In *Proceedings of the 8th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 61–70, 2002.

[11] D. Kempe, J. M. Kleinberg, and E. Tardos. "Maximizing the spread of influence through a social network". In *Proceedings of the 9th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 137–146, 2003.

[12] http://www.mycokerewards.com/

[13] http://www.kfc.com/coupons/ageverification.asp

[14] W. Chen, C. Wang and Y. Wang. "Scalable influence maximization for prevalent viral marketing in large-scale social networks". In *Proceedings of the 16th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2010.

[15] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, "Practical network support for IP traceback". In *Proc. ACM SIGCOMM*, Aug. 2000, pp.295–306.

[16] B. Bloom, "Space/time tradeoffs in hash coding with allowable errors," In *Communications of ACM*, vol. 13, no. 7, pp. 422–426, 1970.

[17] B. Lagesse, M. Kumar, and M. Wright. "AREX: An adaptive system for secureresource access in mobile P2P systems". In *Eighth International Conference on Peer-to-Peer Computing*, IEEE Computer Society, pages 43–52, 2008.

[18] http://www.cse.ohio-state.edu/~tengj/btsdp

[19] L. Song and D. F. Kotz, "Evaluating opportunistic routing protocols with large realistic contact traces". In *ACM MobiCom workshop on Challenged Networks (CHANTS)*, Sep 2007.

[20] W. Wang, V. Srinivasan, and M. Motani. "Adaptive Contact Probing Mechanisms for Delay Tolerant Applications." In *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking*, pp. 230–241, Montreal, Quebec, Canada, 2007.

[21] A. G. Miklas, K. K. Gollu, K. K. W. Chan, S. Saroiu, P. K. Gummadi, and E. de Lara, "Exploiting social interactions in mobile systems." In *Proc. of Ubicomp*, 2007.

[22] U. Shevade, H. Song, L. Qiu, and Y. Zhang, "Incentive-aware Routing in DTNs," IEEE ICNP, Orlando, FL, USA, October 2008.