A New Motivation Framework for Cellular Traffic Offloading

NERELLA PRASHANTH KUMAR¹, M. SUDARSHAN²

¹PG Scholar, Dept of CSE, Alfa College of Engineering & Technology, Allagadda, AP, India, E-mail: nerellaprashanth999@gmail.com.
²Assistant Professor, Dept of CSE, Alfa College Of Engineering & Technology, Allagadda, AP, India, E-mail: medasare@gmail.com.

Abstract: Cellular networks (e.g., 3G) are currently facing severe traffic overload problems caused by excessive traffic demands. Offloading part of the cellular traffic through other forms of networks, such as Delay Tolerant Networks (DTNs) and WiFi hotspots, is a promising solution. However, since these networks can only provide intermittent connectivity to mobile users, utilizing them for cellular traffic offloading may result in a nonnegligible delay. As the delay increases, the users’ satisfaction decreases. In this paper, we investigate the tradeoff between the amount of traffic being offloaded and the users’ satisfaction. We provide a novel incentive framework to motivate users to leverage their delay tolerance for cellular traffic offloading. To minimize the incentive cost given an offloading target, users with high delay tolerance and large offloading potential should be prioritized for traffic offloading. To effectively capture the dynamic characteristics of users’ delay tolerance, our incentive framework is based on reverse auction to let users proactively express their delay tolerance by submitting bids. We further illustrate how to predict the offloading potential of the users by using stochastic analysis for both DTN and WiFi cases. Extensive trace-driven simulations verify the efficiency of our incentive framework for cellular traffic offloading.

Keywords: Cellular Traffic Offloading, Auction, Delay Tolerant Networks, Wifi Hotspots.

I. INTRODUCTION

The recent popularization of cellular networks (e.g., 3G) provide mobile users with ubiquitous Internet access. However, the explosive growth of user population and their demands for bandwidth-eager multimedia content raise big challenges to the cellular networks. A huge amount of cellular data traffic has been generated by mobile users, which exceeds the capacity of cellular network and, hence, deteriorates the network quality [1]. To address such challenges, the most straightforward solution is to increase the capacity of cellular networks, which however is expensive and inefficient. Some researchers studied on how to select a small part of key locations to realize capacity upgrade, and shift traffic to them by exploiting user delay tolerance [2]. Remaining the capacity of cellular networks unchanged, offloading part of cellular traffic to other coexisting networks would be another desirable and promising approach to solve the overload problem. Some recent research efforts have been focusing on offloading cellular traffic to other forms of networks, such as DTNs and WiFi hotspots [3], [4], [5], and they generally focus on maximizing the amount of cellular traffic that can be offloaded. In most cases, due to user mobility, these networks available for cellular traffic offloading only provide intermittent and opportunistic network connectivity to the users, and the traffic offloading hence results in non negligible data downloading delay. In general, more offloading opportunities may appear by requesting the mobile users to wait for a longer time before actually downloading the data from the cellular networks, but this will also make the users become more impatient and, hence, reduce their satisfaction.

In this paper, we focus on investigating the tradeoff between the amount of traffic being offloaded and the users’ satisfaction, and propose a novel incentive framework to motivate users to leverage their delay tolerance for traffic offloading. Users are provided with incentives; i.e., receiving discount for their service charge if they are willing to wait longer for data downloading. During the delay, part of the cellular data traffic may be opportunistically offloaded to other networks mentioned above, and the user is assured to receive the remaining part of the data via cellular network when the delay period ends. The major challenge of designing such an incentive framework is to minimize the incentive cost of cellular network operator, which includes the total discount provided to the mobile users, subject to an expected amount of traffic being offloaded. To achieve this goal, two important factors should be taken into account, i.e., the delay tolerance and offloading potential of the users. The users with high delay tolerance and large offloading potential should be prioritized in cellular traffic offloading. First, with the same period of delay, the users with higher delay tolerance require fewer discounts to compensate their satisfaction loss. To effectively capture the dynamic characteristics of the users’ delay tolerance, we propose an incentive mechanism based on reverse auction, which is proved to conduct a justified pricing. In our mechanism, the users act as sellers to send bids, which include the delay that they are willing to experience and the discount that they want.
to obtain for this delay. Such discount requested by users is called “coupon” in the rest of the paper.

The network operator then acts as the buyer to buy the delay tolerance from the users. Second, with the same period of delay, users with larger offloading potential are able to offload more data traffic. For example, the offloading potential of a user who requests popular data is large because it can easily retrieve the data pieces from other contacted peer users during the delay period. Also, if a user has high probability to pass by some WiFi hotspots, its offloading potential is large. To effectively capture the offloading potential of the users, we propose two accurate prediction models for DTN and WiFi case, respectively. The optimal auction outcome is determined by considering both the delay tolerance and offloading potential of the users to achieve the minimum incentive cost, given an offloading target. The auction winners set up contracts with the network operator for the delay they wait and the coupon they earn, and other users directly download data via cellular network at the original price. More specifically, the contribution of the paper is threefold:

- We propose a novel incentive framework, Win-Coupon, based on reverse auction, to motivate users leveraging their delay tolerance for cellular traffic offloading, which have three desirable properties:
  - truthfulness, 
  - individual rationality, and 
  - low computational complexity.

- We provide an accurate model using stochastic analysis to predict users’ offloading potential based on their data access and mobility patterns in the DTN case.

- We provide an accurate Semi Markov-based prediction model to predict users’ offloading potential based on their mobility patterns and the geographical distribution of WiFi hotspots in the WiFi case.

The rest of the paper is organized as follows: In Section II, we briefly review the existing work. Section III provides an overview of our approach and the related background. Section IV describes the details of our incentive framework, and proves its desirable properties. Section V evaluates the performance of Win-Coupon through the driven simulations and Section VI discusses further research issues. Section VII concludes the paper.

II. RELATED WORK

To deal with the problem of cellular traffic overload, some studies propose to utilize DTNs to conduct offloading. Ristanovic et al. [6] propose a simple algorithm, MixZones, to let the operator notify users to switch their interfaces for data fetching from other peers when the opportunistic DTN connections occur. Whitbeck et al. [7] design a framework, called Push-and-Track, which includes multiple strategies to determine how many copies should be injected by cellular network and to whom, and then leverages DTNs to offload the traffic. Han et al. [3] provide three simple algorithms to exploit DTNs to facilitate data dissemination among mobile users, to reduce the overall cellular traffic. Many research efforts have focused on how to improve the performance of data access in DTNs. In [8], the authors provide theoretical analysis to the stationary and transient regimes of data dissemination. Some later works [9], [10] disseminate data among mobile users by exploiting their social relations.

The authors of [5] measure the availability and the offloading performance of public WiFi based on vehicular traces. Lee et al. [4] consider a more general mobile scenario, and present a quantitative study on delayed and on-the-spot offloading by using WiFi. The prediction of future WiFi availability is important to the offloading scheme design, and has been studied in [11], [12]. The authors of [11] propose to enable mobile users to schedule their data transfers when higher WiFi transmission rate can be achieved based on the prediction. In [12], a Lyapunov framework-based algorithm, called SALSA, is proposed to optimize the energy-delay tradeoff of the mobile devices with both cellular network and WiFi interfaces. Different from the existing work, in this paper, we propose an accurate model to predict how much traffic can be offloaded via WiFi hotspots if a mobile user is willing to wait for certain delay time. All the existing offloading studies have not considered the satisfaction loss of the users when a longer delay is caused by traffic offloading. To motivate users to leverage their delay tolerance for cellular traffic offloading, we propose an auction-based incentive framework. Auction has been widely used in network design. Applying auction in the spectrum leasing is one of the most practical applications. Federal Communications Commission (FCC) has already auctioned the unused spectrum in the past decade [13], and there are a large amount of works on wireless spectrum auctions [14], [15]. Moreover, auction has also been applied for designing incentive mechanism to motivate selfish nodes to forward data for others [16], [17]. However, none of them has applied auction techniques to cellular traffic offloading.

The authors of [5] measure the availability and the offloading performance of public WiFi based on vehicular traces. Lee et al. [4] consider a more general mobile scenario, and present a quantitative study on delayed and on-the-spot offloading by using WiFi. The prediction of future WiFi availability is important to the offloading scheme design, and has been studied in [11], [12]. The authors of [11] propose to enable mobile users to schedule their data transfers when higher WiFi transmission rate can be achieved based on the prediction. In [12], a Lyapunov framework-based algorithm, called SALSA, is proposed to optimize the energy-delay tradeoff of the mobile devices with both cellular network and WiFi interfaces. Different from the existing work, in this paper, we propose an accurate model to predict how much traffic can be offloaded via WiFi hotspots if a mobile user is willing to wait for certain delay time. All the existing offloading studies have not considered the satisfaction loss of the users when a longer delay is caused by traffic offloading. To motivate users to leverage their delay tolerance for cellular traffic offloading, we propose an auction-based incentive framework. Auction has been widely used in network design. Applying auction in the spectrum leasing is one of the most practical applications. Federal Communications Commission (FCC) has already auctioned the unused spectrum in the past decade [13], and there are a large amount of works on wireless spectrum auctions [14], [15]. Moreover, auction has also been applied for designing incentive mechanism to motivate selfish nodes to forward data for others [16], [17]. However, none of them has applied auction techniques to cellular traffic offloading.

The authors of [5] measure the availability and the offloading performance of public WiFi based on vehicular traces. Lee et al. [4] consider a more general mobile scenario, and present a quantitative study on delayed and on-the-spot offloading by using WiFi. The prediction of future WiFi availability is important to the offloading scheme design, and has been studied in [11], [12]. The authors of [11] propose to enable mobile users to schedule their data transfers when higher WiFi transmission rate can be achieved based on the prediction. In [12], a Lyapunov framework-based algorithm, called SALSA, is proposed to optimize the energy-delay tradeoff of the mobile devices with both cellular network and WiFi interfaces. Different from the existing work, in this paper, we propose an accurate model to predict how much traffic can be offloaded via WiFi hotspots if a mobile user is willing to wait for certain delay time. All the existing offloading studies have not considered the satisfaction loss of the users when a longer delay is caused by traffic offloading. To motivate users to leverage their delay tolerance for cellular traffic offloading, we propose an auction-based incentive framework. Auction has been widely used in network design. Applying auction in the spectrum leasing is one of the most practical applications. Federal Communications Commission (FCC) has already auctioned the unused spectrum in the past decade [13], and there are a large amount of works on wireless spectrum auctions [14], [15]. Moreover, auction has also been applied for designing incentive mechanism to motivate selfish nodes to forward data for others [16], [17]. However, none of them has applied auction techniques to cellular traffic offloading.

![Fig. 1. The main idea of Win-Coupon.](image_url)
III. OVERVIEW

A. The Big Picture

In this section, we give an overview of the Win-Coupon framework. By considering the users’ delay tolerance and offloading potential, Win-Coupon uses a reverse auction based incentive mechanism to motivate users to help cellular traffic offloading. Fig 1 illustrates the main idea. The network operator acts as the buyer, who offers coupons to users in exchange for them to wait for some time and opportunistically offload the traffic. When users request data, they are motivated to send bids along with their request messages to the network operator. Each bid includes the information of how long the user is willing to wait and how much coupon he wants to obtain as a return for the extra delay. Then, the network operator infers users’ delay tolerance. In addition, users’ offloading potential should also be considered when deciding the auction outcome. Based on the historical system parameters collected, such as users’ data access and mobility patterns, their future value can be predicted by conducting network modeling, and then based on the information, users’ offloading potential can be predicted. The optimal auction outcome is to minimize the network operator’s incentive cost subject to a given offloading target according to the bidders’ delay tolerance and offloading potential. The auction contains two main steps: allocation and pricing. The winning bidders (e.g., user u1 and u2 shown in Fig. 1) obtain the coupon, and are assured to receive the data via cellular network when their promised delay is reached. For example, suppose p is the original data service charge, if user u1 obtains the coupon with value c in return for delay t, it only needs to pay p - c for the data service. During the delay period, u1 may retrieve some data pieces from other intermittently available networks, for example, by contacting other peers that cache the data or moves into the wireless range of APs. Once delay t passes, the cellular network pushes the remaining data pieces to u1 to assure the promised delay. The losing bidders (e.g., user u3 shown in Fig. 1) immediately download data via cellular network at the original price.

B. User Delay Tolerance

With the increase of downloading delay, the user’s satisfaction decreases accordingly, the rate of which reflects the user’s delay tolerance. To flexibly model users’ delay tolerance, we introduce a satisfaction function $S(t)$, which is a monotonically decreasing function of delay t, and represents the price that the user is willing to pay for the data service with the delay. The satisfaction function is determined by the user himself, his requested data, and various environmental factors. We assume that each user has an upper bound of delay tolerance for each data. Once the delay reaches the bound, the user’s satisfaction becomes zero, indicating that the user is not willing to pay for the data service. Fig. 2 shows an example of the satisfaction function $S(t)$ of a specific user for a specific data, where $t_{\text{bound}}$ is the upper bound of the user’s delay tolerance, $p$ is the original charge for the data service, and the satisfaction curve represents the user’s expected price for the data as the delay increases. For example, with delay $t_1$ the user is only willing to pay $p_1$ instead of $p$. $p_1$ is the satisfaction loss caused by delay $t_1$.

C. Auctions

In economics, auction is a typical method to determine the value of a commodity that has an undetermined and variable price. It has been widely applied to many fields. Most auctions are forward auction that involves a single seller and multiple buyers, and the buyers send bids to compete for obtaining the commodities sold by the seller. In this paper, we use reverse auction [19] that involves a single buyer and multiple sellers, and the buyer decides its purchase based on the bids sent by the sellers. To begin with, we introduce some notations:

Bid($b_i$): It is submitted by bidder i to express i’s valuation on the resource for sale, which is not necessarily true. Private value($x_i$): It is the true valuation made by bidder i for the resources, i.e., the true price that i wants to obtain for selling the resource. This value is only known by i. Market-clearing price($p_i$): It is the price actually paid by the buyer to bidder i. This price cannot be less than the bids submitted i.

Utility($u_i$): It is the residual worth of the sold resource for bidder i, namely the difference between i’s market-clearing price $p_i$ and private value $x_i$. The bidders in the auction are assumed to be rational and risk neutral. A common requirement for auction design is the so-called individual rationality.

Definition 1: An auction is with individual rationality if all bidders are guaranteed to obtain nonnegative utility. The rational bidders decide their bidding strategy to maximize their utility. Let N denote the set of all bidders. The concept of weakly dominant strategy is defined as.

Definition 2: $b_i = \tilde{b}_i$ is a weakly dominant strategy for user i if and only if: $u_i(\tilde{\beta}_i, \beta_{-i}) \geq u_i(\beta_i, \beta_{-i}), \forall \beta_{-i} \neq \tilde{\beta}_i$.

Here, $\beta_{-i} = \{\beta_1, \beta_2, \ldots, \beta_{i-1}, \beta_{i+1}, \ldots, \beta_n\}$ denotes the set of strategies of all other bidders except for bidder i. We see a weakly dominant strategy maximizes i’s utility regardless of the strategies chosen by all other bidders. If for every bidder, truthfully setting its bid to its private value is a weakly dominant strategy, the auction is truthful (strategy proof).
**Definition 3:** An auction is truthful if each bidder, say i, has a weakly dominant strategy, in which \( b_i = x_i \).

The truthfulness eliminates the expensive overhead for bidders to strategize against other bidders and prevents the market manipulation. Also, it assures the efficient allocation by encouraging bidders to reveal their true private values. Vickrey-Clarke-Groves (VCG) [20], [21], [22] is the most well-studied auction format, due to its truthful property. However, VCG only ensures truthfulness when the optimal allocation can be found, and it usually cannot assure the truthfulness when applied to the approximation algorithms [23]. Unfortunately, the allocation problem in Win-Coupon is NP-hard. It is known that an allocation algorithm leads to be truthful if and only if it is monotone [24]. To maintain the truthfulness property, we design an approximation algorithm and make it monotone in a deterministic sense. Therefore, our incentive mechanism possesses three important properties: 1) truthfulness, 2) individual rationality, 3) low computational complexity.

**IV. MAIN APPROACH OF WIN-COUPON**

In this section, we illustrate the details of Win-Coupon. In the reverse auction-based Win-Coupon, the buyer is the network operator who pays coupon in exchange for longer delay of the users. The sellers are the cellular users who sell their delay tolerance to win coupon. The right side of Fig. 1 shows the flow chart of Win-Coupon. At first, the network operator collects the bids to derive the delay tolerance of the bidders, and predicts their offloading potential. Then, based on the derived information, a reverse auction is conducted, which includes two main steps: allocation and pricing. Finally, the network operator returns the auction outcome to the bidders. In the rest of this section, we first introduce the bidding. Then, we present auction mechanism and prove its properties. Finally, we illustrate how to predict bidders’ offloading potential for both DTN and WiFi cases.

**A. Bidding**

To obtain coupon, the users attach bids with their data requests to reveal their delay tolerance. For each user, the upper bound \( b_{\text{bound}} \) of its delay tolerance can be viewed as the resources that it wants to sell. The user can divide \( b_{\text{bound}} \) into multiple time units, and submit multiple bids \( b = \{b_1, b_2, \ldots, b^e\} \) to indicate the value of coupon it wants to obtain for each additional time unit of delay, where \( 1 \) equals \( \frac{b_{\text{bound}}}{e} \) and \( e \) is the length of one time unit. By receiving these bids, the network operator knows that the user wants to obtain coupon with value no less than \( \sum_{k=1}^{k} b^e \) by waiting for \( k \) time units. The length of time unit \( e \) can be flexibly determined by the network operator. Shorter time unit results in larger bids with more information, which increases the performance of the auction, but it also induces more communication overhead and higher computational complexity. To simplify the presentation, in the rest of the paper delay \( t \) is normalized by time unit \( e \). As shown in Fig. 2, \( p - S(t) \) is the satisfaction loss of the user due to delay \( t \). Then, \( p - S(t) \) represents the private value of the user to the delay, namely the user wants to obtain the coupon with value no less than \( p - S(t) \) for delay \( t \). Thus, the private value of the user to each additional time unit of delay is \( x = \{x_1, x_2, \ldots, x^e\}, \) where \( x^e (k \in \{1, \ldots, e\}) \) equals \( S(k-1) - S(k) \).

![Fig. 3. Private value.](image)

**B. Auction Algorithms**

Win-Coupon is run periodically in each auction round. Usually, the auction would result in an extra delay for the bidders to wait for the auction outcome. However, different from other long-term auctions, such as the FCC-style spectrum leasing, the auction round in our scenario is very short, since hundreds of users may request cellular data service at the same time. Also, because the bidders who are willing to submit bids are supposed to have a certain degree of delay tolerance, the extra delay caused by auction can be neglected. Next, we describe two main steps of the auction: allocation and pricing.

**Allocation:** In traditional reverse auction, the allocation solution is purely decided by the bids; i.e., the bidders who bid the lowest price win the game. However, in our scenario, besides the bids that express the bidders’ delay tolerance, the offloading potential of the bidders should also be considered. Let \( \{t_1, t_2, \ldots, t_{N_i}\} \) represent the allocation solution, where \( t_i \) denotes the length of delay that network operator wants to buy from bidder \( i \). Note that because each bidder is asked to wait for integer multiples of time unit, \( t_i \) is an integer. If \( t_i \) equals zero, bidder \( i \) loses the game. The allocation problem in Win-Coupon can be formulated as follows.

**Definition 4:** The allocation problem is to determine the optimal solution \( f(t_1, t_2, \ldots; t_{N_i}) \) that minimizes the total incentive cost, subject to a given offloading target:

\[
\min_{\mathbf{t}} \
\sum_{i=1}^{N} \sum_{k=1}^{k} b_i^k
\]

\[
s.t. \sum_{i=1}^{N} V_i^k(t_i) \geq V_0 \tag{1}
\]

\[
\sum_{i=1}^{N} V_i^k(t_i) \geq V_0 \tag{2}
\]

\[
\forall i, \ t_i \in \{0, 1, 2, \ldots, t_{N_i}\}. \tag{3}
\]
In (1) \( \sum_{i=1}^{n} b_i \) denotes the value of the coupon that the network operator needs to pay bidder i in exchange for its delay \( \tau_i \). \( V^b_i(\tau_i) \) in (2) denotes the expected traffic that can be offloaded, if bidder i downloads data d and is willing to wait for delay \( \tau_i \). We will provide the details on how to predict \( V^b_i(\tau_i) \) in Section 4.3 and 4.4 for both DTN and WiFi cases, respectively. We assume that within a short auction round, each bidder only requests one data item, so that each i is mapped to a single d. Thus, this constraint ensures that the total expected offloaded traffic is no less than the offloading target \( v_0 \). Equation (3) ensures that the delay that each bidder i waits does not exceed \( \tau_i \), the maximum number of time units that i is willing to wait. It is easy to prove that our allocation problem can be reduced to the 0-1 knapsack problem, under the assumption that \( \tau_i = 1 \), for all i. The 0-1 knapsack problem is proved to be NP-hard, and thus, our problem is also NP-hard. Next, we transform the original problem, and derive the optimal solution of the new problem by dynamic programming (DP). We replace constraint (2) with \( \sum_{i=1}^{n} V^b_i(\tau_i)M \geq \lceil m \rceil M \) where \( M = 10^a \) is a common scalar, to transform \( V^b_i(\tau_i) \) and \( v_0 \) into integers. In this way, a table for DP can be formed and the values in the table can be resolved gradually. With a larger M, the optimal solution of the new problem becomes closer to that of the original problem, and the former converges to the latter when M increases to infinity.

On the other hand, larger M increases the computational complexity of the algorithm, and when M is infinite, the approximation algorithm has pseudopolynomial complexity. The operator needs to select a proper scalar M to balance the accuracy and the computational complexity of the allocation algorithm. We define \( V^b_i(\tau_i) = \lceil V^b_i(\tau_i)M \rceil \) and \( \tau_i = \lceil \tau_i M \rceil \). Let \( T^b_i \) denote the minimum time units of delay that bidder i needs to wait to offload \( v \) volume of traffic, and \( C^b_i \) denote the corresponding value of coupon that i requests. Note that here and in the rest of this section, traffic volume \( v \) is scaled by M. Then, we have

\[
T^b_i = \arg\min_k \{ V^b_i(\tau_i) \geq v \},
\]

(4)

\[
C^b_i = \sum_{k=1}^{T^b_i} b_k.
\]

(5)

We use \( B = \{b_1, b_2, \ldots, b_{|N|} \} \) to denote the bid set including all the bids sent by the bidders in set N, and use \( B_i = \{b_1, b_2, \ldots, b_k \} \) to denote the bid set including all the bids sent by the first i bidders in N. Assume only the first i bidders join the auction, we define \( C^b_i \) to be the minimal incentive cost incurred to achieve a given offloading target \( v \) with the bid set \( B_i \), and define \( T^b_i = \{\tau_1, \tau_2, \ldots, \tau_i \} \) to be the corresponding optimal allocation solution. Our allocation algorithm is illustrated in Algorithm 1 with \( T^b_B \) giving the optimal allocation solution. In Algorithm 1, lines 4 to 8 update \( T^b_B, C^b_B, B_i \) to include a new bidder at each iteration. Line 6 searches for the optimal allocation solution \( T^b_i \) to obtain minimal \( C^b_i \). The complexity of the algorithm is \( O(|N|v^6) \).

Pricing: The VCG-style pricing is generally used in forward auction, which involves single seller with limited resources for sale, and multiple buyers. The bidders who have the highest bid win the game, and each winning bidder pays the “opportunity cost” that its presence introduces to others. It is proved that this pricing algorithm provides bidders with the incentives to set their bids truthfully. Based on the basic idea, in our pricing algorithm, the network operator also pays bidder i the coupon with value equal to the “opportunity cost” exerted to all the other bidders due to i’s presence. Given the offloading target \( v_0 \), let \( c_1 = C^b_i(v_0) \) denote the total value of coupons requested by all the bidders under the optimal allocation solution without the presence of i. Let \( c_2 = (c_i^0 - \sum_{k=1}^{i-1} b_k) \) denote the total value of coupons requested by all the bidders except for i under the current optimal allocation solution. Then, i’s “opportunity cost” is defined as the difference between c1 and c2. Thus, i’s market-clearing price can be derived as

\[
p_i = c_1 - c_2 = C^b_i(v_0) - \left( C^b_i - \sum_{k=1}^{i} b_k \right)
\]

(6)

The pricing algorithm is illustrated in Algorithm 2, and the computational complexity of the algorithm is \( O(|N|v^2) \) where A is the number of winning bidders.

Algorithm 1: Win-coupon-Allocation (N, B)

1. for \( v = 0 \) to \( v_0 \) do
2. \( T^b_B = \{\} \);
3. \( C^b_B = C^b_B(v_0) \);
4. for i = 2 to \(|N|\) do
5. for \( v = 0 \) to \( v_0 \) do
6. \( s = \arg\max_{\tau_i \in [0,v]} \{ C^b_B(v_0) + C^b_i(\tau_i) \} ;
7. \( T^b_B = T^b_B \cup \{\tau_i\} ;
8. \( C^b_B = C^b_B(v_0) + C^b_i(\tau_i) ;
9. \) return \( T^b_B, C^b_B ) ;

Algorithm 2: Win-coupon-Pricing (N, B, \( T^b_B, C^b_B ) \)

1. for i = 1 to \(|N|\) do
2. if i is the winning bidder then
3. \( C^b_B(N \setminus \{i\}, B \setminus \{b_i\}) ;
4. \( p_i = C^b_B(v_0) - \left( C^b_B(v_0) - \sum_{k=1}^{i-1} b_k \right) ;
5. \) else
6. \( p_i = 0 ;
7. \) return \( p_i \) for all i;

Properties: In Sections 4.2.1 and 4.2.2, we have shown that Win-Coupon can be solved in polynomial time, if a suitable scalar M is selected. Next, we prove that Win-Coupon also has the properties: truthfulness and individual rationality.

Theorem 1: In Win-Coupon, for each bidder, say i, setting its bids truthfully, i.e., \( b_i = x_i \), is a weakly dominant strategy.

Proof: We assume that when bidder i sets its bids truthfully, i.e., \( b_i = x_i \), network operator would buy delay \( ti \) from it, and its market-clearing price is \( p_i = C^b_B(v_0) - \left( C^b_B(v_0) - \sum_{k=1}^{i} b_k \right) \). Then, the utility obtained by i is

\[
u_i = p_i - C^b_B(v_0) - \left( C^b_B(v_0) - \sum_{k=1}^{i} b_k \right).
\]

(7)

Now, suppose that bidder i sets its bids untruthfully, i.e., \( b_i \neq x_i \). Then, the length of delay \( ti \) that network operator would buy from i falls into two cases: 1) \( t_i = t_i \) and 2) \( t_i \neq t_i \). In case 1, the market-clearing price paid to bidder I...
would become \( \rho_i = c_{B_i}(x_i) - (c_{B_i} - \sum_{j=1}^{k} c_{B_j}) \). Due to the sub problem optimality in deriving the incentive cost \( c_{B_i} = c_{B_i}(x_i) + \sum_{j=1}^{k} c_{B_j} \). Then, we have \( \rho_i = c_{B_i}(x_i) - c_{B_i}(x_i) - \sum_{j=1}^{k} c_{B_j} \), where \( c_{B_i}(x_i) \) and \( c_{B_i}(x_i) \) are independent of the bids sent by bidder i. Therefore, if \( \rho'_i = \rho_i \), then \( \rho_i = \rho_{i-1} \), which is unaffected and the utility of bidder i has no change. In case 2, similarly the market-clearing price paid to bidder i would be changed to \( \rho'_i = c_{B_i}(x_i) - c_{B_i}(x_i) - \sum_{j=1}^{k} c_{B_j} \). Then, the new utility obtained by i equals \( u'_i = c_{B_i}(x_i) - c_{B_i}(x_i) - \sum_{j=1}^{k} c_{B_j} - \sum_{j=1}^{k} c_{B_j} \). The utility gain obtained by bidder i by setting \( b_i' \neq b_i \) can be calculated as

\[
\Delta u_i = u_i' - u_i = \left( c_{B_i}(x_i) - c_{B_i}(b_i) - \sum_{j=1}^{k} c_{B_j} \right) - \left( c_{B_i}(x_i) - c_{B_i}(b_i) - \sum_{j=1}^{k} c_{B_j} \right)
\]

\[
= \left( \sum_{j=1}^{k} c_{B_j} \right) + \sum_{j=1}^{k} c_{B_j} - \sum_{j=1}^{k} c_{B_j} - \sum_{j=1}^{k} c_{B_j} \]

\[
= \left( \sum_{j=1}^{k} c_{B_j} \right) + \sum_{j=1}^{k} c_{B_j} - \sum_{j=1}^{k} c_{B_j} - \sum_{j=1}^{k} c_{B_j} \]

(8)

When bidder i sets its bids truthfully as \( b_i = x_i \), Buying delay with length \( t_i \) from it is the optimal solution of the network operator to minimize the incentive cost. Therefore, keeping other settings unchanged, the solution with buying delay \( t_i \) instead of \( t_i \) from bidder i leads to larger incentive cost. Thus, we have

\[
\left( \sum_{j=1}^{k} c_{B_j} \right) + \sum_{j=1}^{k} c_{B_j} - \sum_{j=1}^{k} c_{B_j} - \sum_{j=1}^{k} c_{B_j} \]

(9)

Since \( c_{B_i}(x_i) \) is independent of \( b_i \), and \( b_i = x_i \), we have \( c_{B_i}(x_i) + \sum_{j=1}^{k} c_{B_j} \). Thus, \( \Delta u_i < 0 \). under this case, bidder i also cannot obtain higher utility by setting \( b_i' \neq x_i \).

**Theorem 2**: In Win-Coupon, all bidders are guaranteed to obtain nonnegative utility.

**Proof**: We have proved that for each bidder, say i, if it participates the auction game, setting its bids truthfully as \( b_i = x_i \) is a weakly dominant strategy. The utility that i obtains equals

\[
u_i = c_{B_i}(x_i) - c_{B_i}(b_i) - \sum_{j=1}^{k} c_{B_j} - c_{B_i}(x_i)
\]

(10)

where \( t_i \) is the optimal length of delay that the network operator would buy from i to minimize the incentive cost. Since \( c_{B_i}(x_i) \) is the incentive cost incurred by the solution with network operator buying delay with length of 0 instead of \( t_i \) from bidder i, we have \( c_{B_i}(x_i) \geq c_{B_i}(t) \). Therefore, Win-Coupon guarantees that all bidders would obtain nonnegative utility.

**C. Prediction of Offloading Potential: The DTN Case**

By motivating users to wait for some time, part of the cellular traffic can be offloaded to other intermittently available networks. One such example is DTN that generally coexists with cellular networks, and does not rely on any infrastructure. Mobile users can share data via DTNs by contacting each other. In urban area with higher user density, mobile users have more chances to contact other users who have their requested data. Large data requests such as video clips tend to drain most of the cellular network resource, and such requests can also tolerate some delay. By offloading them via DTNs, the payload of cellular network can be significantly reduced. In this section, we illustrate how to predict the potentials of the users to offload their traffic via DTNs.

**Models**: Due to high node mobility, large data items are hard to be completely transmitted when two nodes contact. In [25], it has been proved that the random linear network coding (RLNC) techniques can significantly improve the data transmission efficiency, especially when the transmission bandwidth is limited. Thus, in our model, RLNC is adopted to encode the original data into a set of coded packets. As long as the requester collects enough number of any linearly independent coded packets of its requested data, the data can be reconstructed. Due to page limit, we omit the details of RLNC and suggest interested readers to refer to [26]. Besides, when the data item is large, multigeneration network coding is usually adopted. To balance the data transmission efficiency, the computational, and the transmission cost, how to decide the generation size and how to schedule their generation transmissions should be carefully considered. Since this is not the focus of this paper, we will not discuss it in the paper.

**The Main Idea of Prediction**: We describe the rationale of prediction in one auction round. The starting time of this round is denoted by \( t_0 \). The objective of the prediction is to calculate the expected volume of traffic \( V(t) \) that can be offloaded to DTNs, if node i requests data item d and is willing to wait for delay t. By using RLNC, data item d has been encoded into a set of coded packets, and any sd linear independent packets can be used to reconstruct d. We say that a node retrieves an innovative coded packet, if the packet is linearly independent to all the coded packets cached in the node. It has been proven that as long as the subspace spanned by the sender’s code vectors does not belong to receivers, the probability is very small. Node i can retrieve one packet by contacting a node that has cached some coded packets of the requested data, the data can be encoded into a set of coded packets. As long as the size of the finite field to generate coding coefficients is large enough, the probability is very small.

In practice, if the size of the finite field to generate the coding coefficients is large enough, the probability is very close to 1. Node i can retrieve one packet by contacting a node that has some coded packets of data item d, until it has collected all sd linear independent coded packets of its requested data. Large data requests such as video clips tend to drain most of the cellular network resource, and such requests can also tolerate some delay. By offloading them via DTNs, the payload of cellular network can be significantly reduced. In this section, we illustrate how to predict the potentials of the users to offload their traffic via DTNs.

\[
V_i(t) = h \int_0^t R(t) (1 - F_{t_0}(t)) dt.
\]

(7)
where \( h \) is the size of one coded packet. \( 1 - F_{T_{d,i}}(t) \) is the probability that node \( i \) has not received all \( sd \) packets at time \( t \). \( R(t) \) represents the receiving rate of node \( i \) at time \( t \). Due to the i.i.d Poisson contact processes with rate \( \lambda \) between node pairs, \( R(t) \) equals \( \lambda N_d(t) \), where \( N_d(t) \) denotes the total number of nodes that has at least one packet of data \( d \) at time \( t \). Next, we describe how to calculate \( N_d(t) \) and \( F_{T_{d,i}}(t) \).

**Calculation of \( N_d(t) \):** Based on nodes’ interests to data \( d \), all the nodes in the network except for node \( i \) can be divided into two classes: \( D \) and \( I \), where \( D \) contains all the noninteresters and \( I \) contains all the interesters. The interesters include both the nodes which are downloading the data, and those which have already downloaded the data. To facilitate our analysis, we further divide class \( I \) into \( s_d + 1 \) subclasses: \( I_0, I_1, ..., I_{sd}, I_{sd+1} \) based on the nodes’ current downloading progress of data \( d \). Specifically, \( I_j \) (\( j \in [0, s_d] \)) includes all the nodes in the network other than node \( i \), which have already downloaded \( j \) packets of data \( d \), and \( IE \) includes all the nodes that have finished data downloading before and already deleted the data from their buffer. To characterize the different waiting delays of the nodes, we further decompose each class \( I_j \) into \( g + 1 \) subclasses \( I_{j1}, I_{j2}, ..., I_{jg}, I_{jg+1} \), where \( g \) denotes the maximal remaining delay of the current downloading nodes. \( I_g \) (\( g \in [0, s_d - 1] \)), \( k \in [1, g] \) includes the nodes in class \( I_j \) whose remaining delay is \( k \) time slots. For the new requesters that transit from class \( D \) to class \( I \) after time \( t_0 \), we assume they prefer waiting a long delay to retrieve the complete data \( d \) via DTNs. Such new requesters in class \( I_j \) are classified into the subclass \( I_{j1} \). Under this assumption, the derived \( V_d(t) \) is a lower bound of the actual value, due to the following reason. If the delays of the new requesters are limited, after \( dt \) seconds, the network operator would directly push the traffic to the node when its promised delay ends. The passive transition processes are marked as the black arrows in Fig. 4. In the following, we use ordinary differential equations (ODEs) to first analyze the active transition process. We assume that there are \( q \) nodes transit from class \( D \) to class \( I_{j1} \) between time \( t \) and \( t + dt \), with \( dt \) being infinitesimal, and \( q \) is the query rate decided by the popularity of data \( d \). As a node in class \( I_j \) (\( j \in [0, s_d - 1] \)) contacts another node in class \( I_{j'} \) (\( j' \in [1, gd] \)), the former node retrieves a packet from the latter and transit into class \( I_{j1} \). Let \( R_{I_{j1}}(t) \) denote the receiving rate of the node in class \( I_{j1} \) at time \( t \), and we have

\[
R_{I_{j1}}(t) = \lambda \left( \sum_{j \in [0, s_d]} N_{I_j}(t) \right).
\]

(8)

(9)

where \( 1 \) in (9) represents the node itself, since the node cannot retrieve new packet from itself. After a node has completely downloaded data \( d \), it may delete it from its local buffer. We assume that there are \( \gamma d \) portion of the nodes in class \( I_{sd} \) that delete data \( d \) and transit to class \( IE \), between time \( t \) and \( t + dt \). Given all the initial value of the number of nodes in each classes at the starting time, \( N_{I_{sk}}(t) (j \in [0, s_d], k \in [1, g] \cup \infty) \) can be computed by solving the following ODEs:

\[
\frac{dN_{I_{sk}}(t)}{dt} = N_d(t)q - N_{I_{sk}}(t)\gamma_{sk},
\]

(10)

\[
\frac{dN_{I_{sk}}(t)}{dt} = -N_{I_{sk}}(t)\gamma_{sk}, k \in [1, g],
\]

(11)

\[
\frac{dN_{I_{sk}}(t)}{dt} = N_{I_{sk}}(t)\gamma_{sk}, k \in [1, g] \cup \infty,
\]

(12)

\[
\frac{dN_{I_{sk}}(t)}{dt} = \sum_{k} N_{I_{sk}}(t)\gamma_{sk} - N_{I_{sk}}(t)\gamma_{sk}.
\]

(13)

Equation (10) characterizes the varying rate of \( N_{I_{sk}}(t) \), which is composed of two parts: 1) \( N_d(t)q \) nodes transit to this class from class \( D \) by generating a request for \( d \), 2) \( N_{I_{sk}}(t)\gamma_{sk} \) nodes transit from the class to class \( I_{sk} \) by retrieving a packet from its contacted node. Equation (11) depicts the varying rate of \( N_{I_{sk}}(t) \). (\( N_{I_{sk}}(t)\gamma_{sk} \) nodes transit from class \( I_{sk} \) to class \( I_{sk} \) by retrieving a packet from others. Equation (12) shows the varying rate of \( N_{I_{sk}}(t) (j \in [1, s_d - 1], k \in [1, g] \cup \infty) \), (13), which also consists of two parts: 1) \( N_{I_{sk}}(t)\gamma_{sk} \) nodes join the class from class \( I_{sk} \), 2) \( N_{I_{sk}}(t)\gamma_{sk} \) nodes leave from

---

**Fig. 4. Class transition processes.**
the class to class $I_{(j+1)k}$. Equation (13) shows the varying rate of $N_{T_{j,k}}(t)$, where the first term denotes the number of nodes that join the class from class $I_{(j+1)k}$ ($k \in [1,g]$), and the second term denotes the number of nodes that delete the data and transit to class $IE$. The passive transition would happen at the end of each time slot. At the end of each time slot, we update the number of nodes in each class as follows:

$$N_{T_{j,k}}(t) = N_{T_{j,k}(t)} + \sum_{i=0}^{\infty} N_{T_{i,j}(t)}.$$  \hfill (15)

The number of nodes in the rest of the classes which are not listed in (14) and (15) remains the same. Also, at the end of each time slot, the maximal delay of the existing downloading nodes would minus 1 (i.e., $g = g - 1$). By combining the active and passive transition processes, the network state at any time $t$ ($t > t_0$) can be derived. Thus, we can calculate $N_d(t)$, the number of nodes which has at least one packet of data $d$ at time $t$, as $N_d(t) = \sum_{i=1}^{d} N_{T_{i,j}(t)}$.

**Calculation of $F_{T_{j,k}}(t)$:** The derivative of $F_{T_{j,k}}(t)$ ($r ∈ [1,∞]$) is represented as follows by using ODEs:

$$\frac{dF_{T_{j,k}}(t)}{dt} = \frac{R(t)dt(Pr(T_{j,k} ≤ t) - Pr(T_{j,k} ≤ t))}{dt}$$  \hfill (16)

$$= \frac{R(t)dt(Pr(T_{j,k} ≤ t) - Pr(T_{j,k} ≤ t))}{dt}$$  \hfill (17)

$$= R(t)(F_{T_{j,k}}(t) - F_{T_{k,j}}(t)), \forall t ∈ [2,s_d].$$  \hfill (18)

We ignore the probability that node $i$ receives more than one packet during a very short time interval $dt$. Thus, the probability that $Tr$, the time for node $i$ receives $r$ packets, is between the range of $[t, t + dt]$ equals the probability that node $i$ exactly receives $r - 1$ packets before time $t$, and receives the $r$th packet during time $t$ to $t + dt$. Thus, we derive (17) from (16). Similarly, we also derive $\frac{dF_{T_{j,k}}(t)}{dt} = R(t)(1 - F_{T_{k,j}}(t))$. Therefore, given the initial values $F_{T_{j,k}}(t_0) = 0$ ($r ∈ [1,∞]$), $F_{T_{j,k}}(t)$ can be derived by solving the following ODEs:

$$\frac{dF_{T_{j,k}}(t)}{dt} = R(t)(1 - F_{T_{k,j}}(t)),$$ \hfill (19)

$$\frac{dF_{T_{j,k}}(t)}{dt} = R(t)(F_{T_{j,k}}(t) - F_{T_{k,j}}(t)), \forall r ∈ [2,s_d].$$ \hfill (20)

**Numerical Results:** To verify the accuracy of our DTN-based prediction model and analyze the impacts of the system parameters, we numerically solve the ODEs and compare the prediction results to the actual values derived from the Monte-Carlo simulations. In the simulations, we generate 300 nodes following i.i.d. Poisson contact process, and one data item with 16 packets and query rate $q = 0.001$. The same set of parameters is imported to the ODEs. The results given by the simulation are averaged over 200 runs. Fig. 5a shows the results with different contact rate $\lambda$. We can see that the prediction results are very close to the values given by the simulations, which verifies the accuracy of our prediction model. The larger the contact rate is, the earlier the node collects all 16 packets. We further compare the results when the query rate $q$ varies, as shown in Fig. 5b. The prediction also achieves results close to that of the simulations. As the query rate increases, the node collects more packets from other peers as time passes. This implies that if a node requests a popular item, its offloading potential is large.

**D. Prediction of Offloading Potential**

The WiFi Case Similar to the DTN case, by motivating mobile users to wait for some time, part of their cellular traffic may be redirected to WiFi networks when they contact some WiFi hotspots. In urban areas with wide deployment of WiFi networks, WiFi offloading can significantly mitigate the cellular network overload problem. In this section, we illustrate how to predict the potential of the users to offload their data traffic via WiFi networks.

**The Main Idea of Prediction:** Similar to Section 4.3, the objective of the prediction is to calculate the expected traffic $\nu_{ij}(t)$ that can be offloaded to WiFi, if node $i$ requests data item $d$ and is willing to wait for delay $t$. We assume that node $i$’s initial state is $j$; i.e., the node is in state $j$ when it submits the bid. To simplify the presentation, we drop the superscript of $X^i_n$ and $T^i_n$ and use node $i$ as the default target node in the following analysis. The associated time homogeneous semi Markov kernel $Q$ is defined as

$$Q_{jk}(t) = \begin{cases} 0 & \text{if } j \neq k, \\ p_{jk} & \text{if } j = k, \end{cases}$$ \hfill (21)

where $p_{jk} = Pr(X_{n+1} = k | X_n = j)$, and $P[p_{jk}]$ is the transition probability matrix of the embedded Markov chain. $S_{jk}(t)$ is the sojourn time distribution at state $j$ when the next state is $k$; i.e., $S_{jk}(t)$ is the probability that the node will move from state $j$ to $k$ within sojourn time $t$, which can be derived as

$$S_{jk}(t) = Pr(T_{n+1} - T_n ≤ t | X_n = j), \text{ and } \lim_{t \to \infty} S_{jk}(t) = 0.$$ \hfill (22)

Let $S_{j}(t) = Pr(T_{n+1} - T_n ≤ t | X_n = j)$ denote the probability that the node will leave the current state $j$ to another state within sojourn time $t$, which represents the probability.
distribution of the sojourn time in state j regardless of the next state. Then, \( S_j(t) = \sum_{k=1}^{n} Q_{jk}(t) \). We assume the time is discrete in our model, and define the homogeneous semi Markov process as \( Z = (Z_t, t \in \mathbb{N}) \), which describes the state of node at time t. The transition probability of \( Z \) is defined by

\[
\phi_k(t) = \Pr(Z_t = k | Z_0 = j)
\]

which can be calculated as

\[
\phi_k(t) = (1 - \delta_{jk})\phi_k + \sum_{l=1}^{n} \hat{Q}_l(t)\delta_{lk} + \sum_{l=1}^{n} \hat{Q}_l(t)\delta_{lk},
\]

(23)

where \( \delta_{jk} \) is the Kronecker delta function, which equals to 1 if and only if \( j = k \); otherwise, it is zero. \( 1 - S_j(t) \) is the probability that the node stays at state j between time 0 and t without any transition. \( \sum_{l=1}^{n} \hat{Q}_l(t)\delta_{lk} \) represents the probability that the node transits at least once between time 0 to t, where \( \hat{Q}_l(t) = Q_{jl}(t) - Q_{dl}(t-1) \) is the probability that the node will transit from state j to state l at time t. Given the transition probability of \( Z \), we can calculate \( V^d(t) \), the expected traffic that can be transmitted to WiFi networks within time t when node i requests data d and moves in the network. The size of data d is denoted as \( s_d \). We define \( D_{jk}(t) \) as the expected traffic that can be transmitted within time t with the initial state j and the final state k. Then, we obtain

\[
V^d_i(t) = \sum_{k=1}^{n} D_{jk}(t)\phi_k(t) = \sum_{k=1}^{n} \min(\tau_{ij}, s_d)(1 - S_j(t))\delta_{jk}
\]

\[
+ \sum_{l=1}^{n} \min(\tau_{ij} + D_{lk}(t - \tau), s_d)\hat{Q}_l(t)\delta_{lk},
\]

(24)

\( (25) \)

where \( \min(\tau_{ij}, s_d) \) represents the traffic that can be offloaded if the node stays at state j with no transition before time t, and the traffic is bounded by the total amount of data requested by the node. \( \min(\tau_{ij} + D_{lk}(t - \tau), s_d) \) is the traffic that can be offloaded if the node transits at least once before time t. Thus, we derive the offloading potential in WiFi case. Next, we describe how to calculate the transition probability matrix \( P \) and the sojourn time probability distribution \( S_j(t) \).

**Calculation of \( P \) and \( S_j(t) \):** To calculate \( P \) and \( S_j(t) \) node’s mobility histories are needed. Mobile users can upload their mobility information to the network operator periodically through WiFi interfaces on their mobile phones or through wired networks by using their PCs as shown in Fig.6. \( P \) is the transition probability matrix of the embedded Markov chain. Each element \( p_{jk} \in P \) represents the probability that node i will transit from state j to k. We define \( p_{jk} \) as the observed transition frequency in the node mobility trace. Then, we obtain \( p_{jk} = \frac{\text{num}_{jk}}{\text{num}_p} \) where numjk is the number of transitions from state j to state k, and numj is the number of transitions from state j without considering the next transition state. \( S_j(t) \) is the sojourn time probability distribution at state j when the next transition state is k. Based on the node mobility history, \( S_j(t) \) can be estimated as

\[
S_{jk}(t) = \Pr(t_{jk} \leq t) = \frac{\text{num}_{jk}(t_{jk} \leq t)}{\text{num}_{jk}},
\]

(26)

where \( t_{jk} \) is the sojourn time at state j when followed by state k, and \( \text{num}_{jk} \) is the number of transitions from state j to state k with the sojourn time less than t. In this way, the cellular network operator can derive the transition probability matrix and the sojourn time probability distribution for each node based on their uploaded mobility history.

**V. PERFORMANCE EVALUATION**

In this section, we evaluate the performance of Win-Coupon through trace-driven simulations for both DTN and WiFi cases. For each case, we first introduce the simulation setup, and then evaluate the performance of Win-Coupon under various system parameters. In the evaluation, the following performance metrics are used:

- **Offloaded traffic.** The total amount of traffic that is actually offloaded.
- **Allocated coupon.** The total incentive cost spent by the network operator for offloading purpose.
- **Average downloading delay.** The average time a bidder spends to get the complete data after sending the request.

**A. The DTN Case**

**Simulation Setup:** Our performance evaluation in the DTN case is conducted on the UCSD trace [29], which records the contact history of 275 HP Jornada PDAs carried by students over 77 days. Based on the trace, we generate 50 data items, and each contains eight packets. The query rate \( q_d \) for each data d is generated following Zipf distribution, and the default skewness parameter \( w \) is set to 1.5. The delete rate \( \gamma_d \) for each data d is randomly generated within the range of \( [1.0 \times 10^{-8}, 0.5 \times 10^{-8}] \) following the uniform distribution. When nodes request data, they can choose to attach bids with the request message based on their satisfaction function. In the simulations, we model the user satisfaction function as:

\[
S(t) = p - at^b,
\]

where \( p \) is the original data service charge, and we assume that all the data items have the same charge \( p = 0.8 \). \( a \) determines the scale of \( S(t) \), and a smaller \( a \) results in higher delay tolerance. \( b \) determines the shape of \( S(t) \). When \( b > 1 \), \( \gamma_d \) follow a concave, linear, and convex function, respectively. In the simulations, we randomly generate parameters \( a \) and \( b \) within the range of \([0.04, 0.08]\) and \([0.8, 1.2]\) respectively, the scale of the trace, in terms of the number of users and their contact frequencies, is rather
small. This results in long auction rounds for the network operator to collect enough bids, as well as long downloading delay experienced by the users as shown in Fig. 7. In a university there would probably be a larger number of users; thus, we further generate a large scale trace by replicating the nodes in the original trace 10 times, which seems like a more reasonable network scale. The evaluation results on the large-scale trace are given in Section 5.1.5.

Fig. 7. Impact of bidder number, reserve price, and delay tolerance—DTN.

B. The WiFi Case
Simulation Setup: To evaluate the performance of Win-Coupon in the WiFi case, we use the UMass DieselNet trace [30], which includes the mobility histories of 32 buses. In the trace, each bus is equipped with a GPS device, and periodically records its GPS location. To apply our prediction model, the map is divided into 10 _ 15 uniform-sized geographical grids. Based on the mobility information provided by the trace, we further add synthetic WiFi information. We assume that some WiFi hotspots are distributed on the map. We preset a WiFi coverage rate, which represents the ratio of the number of grids with some WiFi hotspots to the total number of grids. The downlink data rate for those grids with WiFi hotspots are randomly generated within the range of 50 Kbps and 1 Mbps. To derive the transition probability matrix and the corresponding sojourn time probability distributions for each node, we take two-week traces as the training data as shown in Fig. 8. We pick up one day trace (11-06-2007) which has relatively high network density to perform Win-Coupon. The first auction round begins at 8:30 AM and the auction is performed for 10 consecutive rounds with the interval of one hour. Since the total number of nodes in the trace is quite limited, we assume that each node will participate in the auction to increase the number of participants.
The size of data requested by nodes are randomly generated within the range of 100 and 500 Mb. Similar to the DTN case, we also define the user satisfaction function as \( S(t) = p - at^b \) to model user delay tolerance. We randomly generate parameters \( a \) and \( b \) within the range of \([0.2, 0.3]\) and \([0.8, 1.2]\) respectively, following the uniform distribution for each node to each data unless specified differently. The presented results are averaged over 10 runs.

VI. SIMULATION RESULTS—IMPACT OF WIFI COVERAGE RATE

We set the WiFi coverage rate to 0.4 and 0.6, respectively, to evaluate the impact of WiFi availability on the performance of Win-Coupon. The evaluation results are shown in Fig. 9. Figs. 9a and 9b show the percentage of offloaded traffic and the average delay, respectively, when offloading target increases from 1,000 to 4,500 Mb. When the offloading target is set to be low, for example, less than 2,000 Mb, different WiFi coverage rate results in similar percentage of offloaded traffic. This is because the relatively low offloading target can be easily achieved in both network scenarios. As explained in Section 4.2.4, the virtual bidder is added to ensure the network operator gaining nonnegative profit. In other words, if the actual bidders have low delay tolerance or small offloading potential, the network operator would not trade with them and ask them to directly download data via cellular network (letting the virtual bidders win the game), even if the offloading target cannot be achieved. As can be seen when the WiFi coverage rate reaches 0.6, the offloading target can be almost achieved, due to the large offloading potential of the bidders.

VII. CONCLUSION

In this paper, we proposed a novel incentive framework for cellular traffic offloading. The basic idea is to motivate the mobile users with high delay tolerance and large offloading potential to offload their traffic to other intermittently connected networks such as DTN or WiFi hotspots. To capture the dynamic characteristics of users’ delay tolerance, we design an incentive mechanism based on
reverse auction. Our mechanism has been proved to guarantee truthfulness, individual rationality, and low computational complexity. Moreover, we design two accurate models to predict the offloading potential of the users for both DTN and WiFi cases. Extensive trace driven simulations validate the efficiency and practical use of our incentive framework.

VIII. REFERENCES
