Throughput-Efficient Channel Allocation Algorithms in Multi-Channel Cognitive Vehicular Networks

You Han, *Student Member, IEEE*, Eylem Ekici, *Senior Member, IEEE*, Haris Kremo, *Member, IEEE*, and Onur Altintas, *Member, IEEE*

***Abstract*— Many studies show that the dedicated short range communication band allocated to vehicular communications is insufficient to carry the wireless traffic generated by emerging vehicular applications. A promising bandwidth expansion possibility presents itself through the release of large TV band spectra (i.e., the TV white space spectrum) by the Federal Communications Commission for cognitive access. One primary challenge of the so-called TV white space (TVWS) spectrum access in vehicular networks is the design of efficient channel allocation mechanisms in face of spatial-temporal variations of TVWS channels. In this paper, we address the channel allocation problem for multi-channel cognitive vehicular networks with the objective of system-wide throughput maximization. We show that the problem is an NP-hard non-linear integer programming problem, to which we present three efficient algorithms. We first propose a probabilistic polynomial-time $(1-1/e)$-approximation algorithm based on linear programming. Next, we prove that the objective function can be written as a submodular set function, based on which we develop a deterministic constant-factor approximation algorithm with a more favorable time complexity. Then, we further modify the second algorithm to improve its approximation ratio without increasing its time complexity. Finally, we show the efficacy of our algorithms through numerical examples.**

***Index Terms*— Cognitive vehicular networks, channel allocation, linear programming, submodular set function.**

I. Introduction

THE 5.9 GHz Dedicated Short Range Communication (DSRC) band was allocated to vehicular communications by FCC in the United States more than a decade ago. Then, the IEEE Wireless Access in Vehicular Environment standard stack (e.g., IEEE 802.11p and IEEE 1609.4) was proposed to support vehicular communications in the DSRC band. However, both theoretical analysis [1], [2] and simulation results [1]–[3] demonstrate that the DSRC band is insufficient to provide reliable safety message transmissions. Moreover, it is shown that non-safety use of the DSRC band has to be severely restricted during peak hours of traffic to guarantee reliable transmission of safety messages. For example, [4] demonstrates that merely 10% of the DSRC bandwidth is left for non-safety applications at medium traffic density if 95% reliability of transmissions must be guaranteed for safety applications. More importantly, the spectrum scarcity problem is becoming severer due to growing vehicles as well as increasing wireless vehicular applications, such as collision avoidance, safety warning, remote vehicle diagnostic, file downloading, web browsing, and video streaming [5], [6].

A potential method to alleviate the spectrum scarcity problem is to unload some non-safety data traffic from the DSRC band to the unlicensed TV White Space (TVWS) band, which has been released by FCC for cognitive access [7]–[9]. One primary challenge of using the TVWS band in vehicular networks is the design of efficient channel allocation mechanisms that can cope with spatial-temporal variations of TVWS channels [7], [10]. However, the contention-based IEEE 802.11p Medium Access Control (MAC) scheme is not suitable for Cognitive Vehicular Networks (CVNs) due to the lack of cognitive radio capability.

In this paper, we propose a throughput-efficient channel allocation framework for multi-channel cognitive vehicular networks with the objective of maximizing network-wide throughput. In particular, we consider special characteristics of the TVWS channels such as spatial-temporal variations of channel availability and FCC's regulations on the protection of primary users of the band. Moreover, to cope with high mobility of vehicles, we leverage the existing IEEE 1609.4 standard for the design of upper MAC layer. Specifically, in IEEE 1609.4, each Universal Time Coordinated (UTC) second is split into 10 MAC cycles, and thus each MAC cycle is 100 milliseconds long. The MAC cycle is composed of a CCH interval for the exchange of control information and an SCH interval for data delivery. Both the CCH interval and the SCH interval are 50 milliseconds long. In particular, distributed channel contention is also performed in the CCH interval. Similar to this framework, our channel allocation algorithm is also performed at the beginning of each MAC cycle. One difference, however, is that we consider a centralized channel allocation model as our first step due to the great difficulty of TVWS access in vehicular networks and

Manuscript received February 26, 2016; revised June 19, 2016 and September 6, 2016; accepted November 3, 2016. Date of publication November 16, 2016; date of current version February 9, 2017. The associate editor coordinating the review of this paper and approving it for publication was Q. Li.

Y. Han and E. Ekici are with the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210 USA (e-mail: han.639@osu.edu; ekici.2@osu.edu).

H. Kremo was with Toyota InfoTechnology Center, Co., Ltd., 6-6-20 Akasaka, Minato-ku, Tokyo, Japan. He is now with CONNECT Centre, Trinity College Dublin, Dublin 2, Ireland (e-mail: kremoh@tcd.ie).

O. Altintas is with Toyota InfoTechnology Center, USA, Inc., Mountain View, CA 94043 USA (e-mail: onur@us.toyota-itc.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TWC.2016.2629484

identify the development of distributed algorithms as one of our future works. Another significant difference is that, instead of using contention-based channel allocation schemes, we optimize the channel usage based on transmission demands of vehicles as well as the aforementioned characteristics of the TVWS channels.

The theoretical contributions of our work are threefold. First, we formulate the channel allocation problem in CVNs, which is proved to be NP-hard. Second, we propose a probabilistic polynomial time $(1 - 1/e)$-approximation algorithm based on linear programming. Third, we show that our objective function can be written as a submodular set function, based on which we develop a deterministic polynomial-time constant-factor approximation algorithm with a more favorable time complexity. Since the second algorithm has a low approximation ratio, we further modify the algorithm to obtain a new constant-factor approximation algorithm. The improved algorithm is proved to achieve higher approximation ratio without increasing time complexity.

The rest of the paper is organized as follows. Related works are investigated in Section II. The system model and problem formulation are introduced in Section III. In Section IV, we propose our method based on linear programming and analyze its performance. Then, our objective function is proved to be a submodular set function and two deterministic polynomial time algorithms are developed in Section V. We present simulation results in Section VI, followed by a conclusion of our work in Section VII.

II. Related Works

Some recent papers have been published that address the channel allocation problem in cognitive radio networks. For example, a centralized quality-of-experience driven channel allocation framework is proposed in [11] by considering statistical properties of primary channels. Although both analytic and simulation results are provided to demonstrate performance of the proposed framework, no optimality properties are guaranteed. In contrast, all the algorithms developed in our paper have provable performance guarantees. In addition, a channel cooperation scheme is proposed in [12] to address the resource allocation problem in an OFDMA-based cooperative cognitive radio network. This work is different from ours because we assume no SU-PU cooperation. Moreover, [13] studies the joint relay scheduling, channel access and power allocation problem, which particularly considers the imperfect sensing scenario and the tradeoff between system capacity and energy consumption. Since neither spectrum sensing nor energy consumption is critical to our problem formulation, the joint scheduling algorithm in [13] cannot accommodate the channel allocation problem in our work. The channel allocation problem is also studied in [14], in which a receiver-based channel allocation model is proposed. Although the model is proved to be more superior than conventional models, only a heuristic algorithm is developed to solve the formulated NP-hard mixed integer linear program. In contrast, we devise three algorithms to solve the formulated NP-hard channel allocation problem, which all have provable performance guarantees.

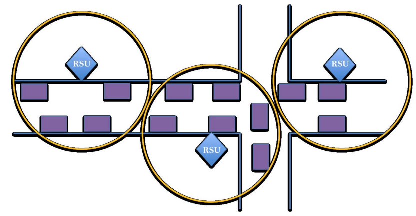


Fig. 1. An example of roadside units

The channel allocation problem is usually formulated as an NP-hard non-linear integer programming (NIP) problem. For example, [15] maximizes the total network throughput while guaranteeing quality-of-service (QoS) requirements of both video and data services through channel allocation. The channel allocation problem is formulated as an NIP problem, to which two heuristic algorithms are proposed. Reference [16] also studies the problem of supporting multiple services over CR networks using channel allocation. The channel allocation problem is formulated as an NIP problem, to which two greedy algorithms are proposed. Similarly, [17] proposes a heuristic algorithm to the formulated channel allocation problem for smooth video delivery over CR networks, which is also an NIP problem. As we can see from these works, due to the difficulty of solving the NIP problems, they only develop heuristic algorithms without theoretical guarantees. Different from these works, although the channel allocation problem is also formulated as an NIP problem in this paper, we devise three efficient algorithms with both provable performance guarantees and polynomial time complexity. Details of the system model and the proposed algorithms are presented in the following sections.

III. System Model and Problem Formulation

We consider a cognitive vehicular network consisting of N vehicles and M available TVWS channels. In IEEE 802.11p, packets are classified into four Access Categories with decreasing priority: $AC[0] \cdots AC[3]$. Therefore, in our model, we associate each priority class $AC[i]$ with a weight A_i subject to $A_i > A_j, \forall i < j$. Similar to existing works on cognitive vehicular networks, in our model, the high mobility of vehicles is handled using Road Side Units (RSUs) [3], [4], [18]–[20]. As shown in Fig. 1, a road is divided into segments with an RSU in each segment. Whenever a vehicle moves from a segment to another segment, it must register with the RSU in the new segment to send channel allocation requests and obtain channel allocation results. For example, if the average speed of a vehicle is 20 m/s and average transmission range of RSUs is 500 meters, then the vehicle needs to register with RSUs every 50 seconds. Similar to IEEE 1609.4, in our model time is partitioned into equal scheduling cycles with length T and RSUs allocate channels to vehicles at the beginning of every scheduling cycle. As discussed in Section I, T is set to 100 milliseconds to cope with high mobility of vehicles.

The occupancy of a free TVWS channel j is modeled through a continuous random variable t_j, which is the residual time until the return of PUs to channel j. We assume that the cumulative distribution function (CDF) $F_j(t_j)$ of t_j is known to RSUs. Note that since spectrum sensing is believed to be not sufficiently accurate to protect PUs in the TVWS band, it has been regulated as only an "optional" functionality for TVWS devices by FCC, while the more accurate geo-location/database access technology has been regulated as a "mandatory" functionality for TVWS devices [9]. More importantly, it is shown in [19] that neither spectrum sharing nor database access along can provide sufficient protection of TVWS PUs in a vehicular environment. Therefore, we assume that RSUs estimate usage patterns of available TVWS channels using both appropriate spectrum sensing techniques [21] and geo-location/database access [8], [9]. Moreover, although the channel allocation algorithms in our paper were initially designed for TVWS channels, they also apply to the joint allocation of TVWS and DSRC channels. The reason is that, in our model a free TVWS channel is modeled by the return time of PUs on this channel. Similarly, a DSRC channel can also be modeled by the return time of PUs on this DSRC channel, which is always equal to infinity because there are no PUs on DSRC channels. Moreover, since our proposed methods do not require specific distributions of PUs' return time, they also apply to DSRC channels.

Since exact behaviors of PUs are unknown to RSUs, vehicular transmissions scheduled to a channel can conflict with PU transmissions with non-zero probabilities. Similar to existing works on CR networks [22], [23], we assume that PUs can tolerate a certain collision probability, i.e., the collision probability caused by scheduled vehicles on a channel j must not exceed a predetermined bound γ_j, which is a system parameter determined by the PU network [24]. With this collision probability constraint, RSUs can compute the maximum allowed scheduling time on each channel which is denoted by T_j^r. Then, the collision probability P_{coll} can be computed as

$$P_{coll} = Pr\{t_j \leq T_j^r\} = \int_0^{T_j^r} f_j(t_j)dt_j = F_j(T_j^r) \leq \gamma_j, \quad (1)$$

and thus the maximum T_j^r can be computed by solving Equation (1). Since the channel allocation is performed at the beginning of each scheduling cycle, the maximum allowed scheduling duration on channel j is $c_j = \min\{T_j^r, T\}$.

In our model, we assume that the transmissions of a vehicle before a PU returns are successful and all remaining packets are thought to be lost (one possible reason is that high transmit power of PUs starves low-power vehicular transmitters). The vehicle will try to retransmit the lost packets in following cycles. Notice that our system model applies to both packet-based (ON-OFF) primary systems and non-packet based primary systems (e.g., TV broadcast). The reason is that we make no assumptions on the specific type of PU systems. Instead, we only consider the residual time that PUs return to a channel given the channel is idle at the beginning of a scheduling cycle. Therefore, how the PU system behaves on the channel doesn't affect our model. For example, in a TV system, we only consider the residual time that next TV broadcast starts on a channel given that the TV broadcast is idle at the beginning of a scheduling cycle. Furthermore, let L_i be the number of packets vehicle i tends to transmit in the current cycle, i.e., transmission demand of the vehicle in the current scheduling cycle. Note that L_i is not necessarily the real packet queue backlog of vehicle i. For example, L_i can be the amount of a particular type of packets vehicle i wants to transmit in the current cycle. Instead of tracking the packet queue length of each vehicle, we assume that L_i value is included in the vehicle's requesting messages to the RSU. Then, the RSU determines required transmission time of vehicle i on channel j as

$$t_{ij} = \min\{L_i/R_j, T_j^r, T\}, \quad (3)$$

where R_j denotes the transmission rate of vehicles on channel j depending on the bandwidth of the channel. Notice that we assume appropriate power control methods will be adopted by vehicles to achieve the transmission rate of R_j on channel j [25], [26]. Moreover, in our model, the transmission rates are assumed to be constants instead of decision variables. Hence, the proposed channel allocation schemes also apply to scenarios where vehicles have different transmission rates on the same channel. Note that it is possible for each vehicle to know the amount of data to be transmitted in a MAC frame. For example, in the polling-based MAC design of IEEE 802.11ad (an amendment of Wi-Fi for millimeter wave communications), a station must specify the amount of time it requests to the access point (AP) such that the AP can allocate sufficient time for its transmissions. The amount of data to be transmitted is computed based on the current queue length, channel measurement results and quality-of-service requirements of the running application [27], [28]. Moreover, constant transmission rates can also be guaranteed using proper modulation and coding schemes and power control policies [27]. For example, similar to IEEE 802.11ad [27] and IEEE 802.22 [29], a channel measurement field can be added to the header of PHY frames, which can be used to measure the background noise. In this case, power control schemes can be utilized to overcome the varying background noise level, e.g., increasing the transmit power level when background noise increases.

Let x_{ij} be our channel assignment variable, i.e., $x_{ij} = 1$ if vehicle i is scheduled to channel j and $x_{ij} = 0$ otherwise. As shown in Equation (2), as shown at the bottom of the next page, we use U_{ij} to denote the expected weighted throughput achieved by scheduling vehicle i to transmit on channel j. Note that, in Equation (2) we let $\sum_{k=1}^{0} t_{kj}x_{kj} = 0$ when $i = 1$, and thus $U_{1j} = \frac{1}{T} A_1 R_j \left(t_{1j}x_{1j} - \int_0^{t_{1j}x_{1j}} F_j(t_j)dt_j \right)$.

An example of our channel allocation model is shown in Fig. 2, where vehicle 1 and 3 are scheduled to TVWS channel 1 and vehicle 2 is scheduled to TVWS channel 2. Our goal is to maximize the total weighted throughput of all vehicles (i.e., $\sum_{i,j} U_{ij}$) by scheduling disjoint sets of vehicles to available TVWS channels. Since more than one vehicle can be scheduled to a channel, the expected throughput of a vehicle is affected by vehicles scheduled ahead of it according to Equation (2). Therefore, the transmission order of vehicles on a channel will affect the total weighted throughput of these vehicles. Lemma 1 states that priority-ordered transmission maximizes the total weighted throughput on a channel, which is proved in Appendix A.

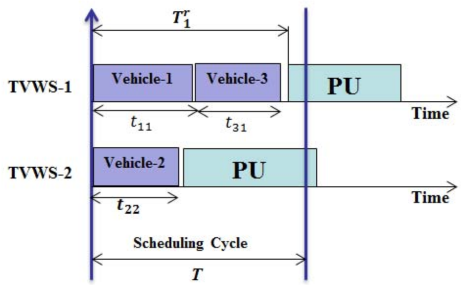


Fig. 2. Example of the channel allocation model

Lemma 1: Given a set of vehicles to be scheduled to a channel, higher priority vehicles must be scheduled earlier than lower priority vehicles in order to maximize the total weighted throughput, while the transmission order of vehicles of the same priority does not affect the total weighted throughput.

In our model, vehicles are sorted and renamed as $\{v_1, v_2, \cdots v_N\}$ such that $A_1 \geq A_2 \cdots \geq A_N$ and $L_i \geq L_j, \forall i \leq j$ when $A_i = A_j$. It means that vehicles are sorted with non-increasing priorities and vehicles with the same priority are ordered with the decreasing order of L_i values.

Given the above definitions, our channel allocation problem can be formulated as

$$\begin{aligned} \max_{x_{ij}\in\{0,1\}} \quad & \sum_{i=1}^{N}\sum_{j=1}^{M}\frac{1}{T}A_iR_j\left(t_{ij}x_{ij} - \int_{\sum_{k=1}^{i-1}t_{kj}x_{kj}}^{\sum_{k=1}^{i}t_{kj}x_{kj}} F_j(t_j)dt_j\right) \\ s.t. \quad & C1: \sum_{i=1}^{N} t_{ij}x_{ij} \leq c_j, \forall j \in \{1, 2\cdots M\} \\ & C2: \sum_{j=1}^{M} x_{ij} \leq 1, \forall i \in \{1, 2\cdots N\} \\ & t_{ij} = \min\{L_i/R_j, T_j^r, T\}, c_j = \min\{T_j^r, T\}, \forall i, j \end{aligned}$$

where $C1$ means that the total transmission time of vehicles scheduled to channel j must be less than the maximum allowed transmission time of the channel, $C2$ means that every vehicle i can be scheduled to at most one channel. As we can see from Problem (4), we schedule vehicles based on their particular transmission demands in the current scheduling cycle, and thus our approach is suitable for highly mobile vehicular networks where message lifetimes are short and messages are expunged from queues if they are not transmitted by their deadlines. Moreover, our approach only requires A_i and L_i values from vehicles in each scheduling cycle, which can be easily obtained even under high mobility scenarios. Next, we show the difficulty of Problem (4) by proving Theorem 1, which guides us to develop the algorithm in Section IV.

Theorem 1: The channel allocation problem (4) is NP-hard.

Proof: This theorem is proved by reducing Problem (4) to the Generalized Assignment Problem (GAP). GAP is to assign different sets of items to bins subject to bin capacity constraints to maximize the overall profit, where both the size s_{ij} and profit p_{ij} are constant metrics. GAP has been proved to be NP-hard [30]. In Problem (4), we consider the case where $Pr\{t_j < T\} = 0$. This is a simplified version of our problem where PUs always return at least T time after the beginning of a cycle, i.e., the TVWS channels are always available. In this case, Problem (4) is reduced to the following problem:

$$\begin{aligned} \max_{x_{ij}\in\{0,1\}} \quad & \sum_{i=1}^{N}\sum_{j=1}^{M}\frac{1}{T}A_iR_jt_{ij}x_{ij} \\ s.t. \quad & \sum_{i=1}^{N} t_{ij}x_{ij} \leq T, \forall j \in \{1, 2\cdots M\} \\ & \sum_{j=1}^{M} x_{ij} \leq 1, \forall i \in \{1, 2\cdots N\} \\ & t_{ij} = \min\{L_i/R_j, T\}. \end{aligned} \tag{4}$$

$$\begin{aligned} U_{ij} &= \frac{1}{T}A_iR_j\int_0^{\infty}\left(t_{ij}x_{ij}\cdot 1_{\{t_j\geq\sum_{k=1}^{i}t_{kj}x_{kj}\}} + \left(t_j - \sum_{k=1}^{i-1}t_{kj}x_{kj}\right)\cdot 1_{\{\sum_{k=1}^{i-1}t_{kj}x_{kj}\leq t_j\leq\sum_{k=1}^{i}t_{kj}x_{kj}\}}\right)f_j(t_j)dt_j \\ &= \frac{1}{T}A_iR_j\left(t_{ij}x_{ij}\int_{\sum_{k=1}^{i}t_{kj}x_{kj}}^{\infty} f_j(t_j)dt_j + \int_{\sum_{k=1}^{i-1}t_{kj}x_{kj}}^{\sum_{k=1}^{i}t_{kj}x_{kj}}\left(t_j - \sum_{k=1}^{i-1}t_{kj}x_{kj}\right)f_j(t_j)dt_j\right) \\ &= \frac{1}{T}A_iR_j\left(t_{ij}x_{ij}\left(1 - F_j(\sum_{k=1}^{i}t_{kj}x_{kj})\right) + \int_{\sum_{k=1}^{i-1}t_{kj}x_{kj}}^{\sum_{k=1}^{i}t_{kj}x_{kj}}\left(t_j - \sum_{k=1}^{i-1}t_{kj}x_{kj}\right)dF_j(t_j)\right) \\ &= \frac{1}{T}A_iR_j\left(t_{ij}x_{ij} - t_{ij}x_{ij}F_j(\sum_{k=1}^{i}t_{kj}x_{kj})\right) \\ &\quad + \frac{1}{T}A_iR_j\left((\sum_{k=1}^{i}t_{kj}x_{kj} - \sum_{k=1}^{i-1}t_{kj}x_{kj})F_j(\sum_{k=1}^{i}t_{kj}x_{kj}) - \int_{\sum_{k=1}^{i-1}t_{kj}x_{kj}}^{\sum_{k=1}^{i}t_{kj}x_{kj}} F_j(t_j)dt_j\right) \\ &= \frac{1}{T}A_iR_j\left(t_{ij}x_{ij} - t_{ij}x_{ij}F_j(\sum_{k=1}^{i}t_{kj}x_{kj})\right) + \frac{1}{T}A_iR_j\left(t_{ij}x_{ij}F_j(\sum_{k=1}^{i}t_{kj}x_{kj}) - \int_{\sum_{k=1}^{i-1}t_{kj}x_{kj}}^{\sum_{k=1}^{i}t_{kj}x_{kj}} F_j(t_j)dt_j\right) \\ &= \frac{1}{T}A_iR_j\left(t_{ij}x_{ij} - \int_{\sum_{k=1}^{i-1}t_{kj}x_{kj}}^{\sum_{k=1}^{i}t_{kj}x_{kj}} F_j(t_j)dt_j\right). \end{aligned} \tag{2}$$

This reduced problem is a GAP, where T is the capacity of all bins, t_{ij} is the size of item i in bin j, and $\frac{1}{T}A_i R_j t_{ij}$ is the profit of assigning item i to bin j. This mapping proves that Problem (4) is NP-hard. ∎

IV. Solution 1: Linear Programming-Based Algorithm

Notice that Problem (4) is much harder than GAP since the profit values p_{ij} in GAP are fixed. In contrast, the utility function of each vehicle in Problem (4) is coupled with other vehicles. Here, "coupling" means that the expected throughput of a vehicle is affected by the vehicles scheduled ahead of it. For a given set of vehicles assigned to a channel, however, the utility of each vehicle on the channel (i.e., the "profit" in GAP) is fixed due to the ordering rule in Lemma 1.

With this observation, our linear programming based (LP) algorithm works as follows. We first formulate an equivalent integer programming (IP) problem for the initial channel allocation problem, i.e., choosing exactly one set of vehicles S_j for each channel j such that the total utility of all channels is maximized. Let $X_j(S_j) \in \{0, 1\}$ be the decision variable indicating assigning set S_j to channel j. By relaxing this constraint to be $X_j(S_j) \in [0, 1]$, we obtain an LP problem. However, this LP problem has an exponential number of variables due to the exponential number of possible sets of vehicles to be assigned to a channel. Although this LP problem cannot be solved in polynomial time, dealing with its dual problem helps to reduce the number of its variables with no performance loss. Although the dual problem has an exponential number of constraints, it is proved in [31] that such problems can be solved in polynomial time using ellipsoid algorithm associated with a separation oracle (to be defined later).

When solving the dual problem with the ellipsoid algorithm iteratively, one is guaranteed to obtain at most one feasible variable for the primal LP problem in each iteration. Since the ellipsoid algorithm terminates in polynomial iterations, one can obtain at most polynomial number of variables. These variables are sufficient to solve the primal problem with no performance loss [32]. In this way, the primal LP problem becomes a new LP problem with a polynomial number of variables, which can be solved in strongly polynomial time [33]. After solving the LP problem, we obtain a fractional solution $\{X_j(S_j)\}$. Then, we propose a rounding algorithm to round the fractional solution into an integer solution with the approximation ratio greater than or equal to $(1 - 1/e)$. Since the optimal solution of the LP problem is an upper bound of the optimal solution of the IP problem, we are guaranteed to achieve at least $(1 - 1/e)$ of the IP optimum. The detailed procedure of the LP algorithm is as follows.

A. Formulating the LP Problem

Let $f_j(S_j)$ be the utility of assigning the set S_j to channel j, and I_j be the set of all feasible assignments of vehicles to channel j obtained from the constraints in Problem (4). Problem (4) can be reformulated as an LP problem as

$$\begin{aligned}
\max_{X_j(S_j)\in[0,1]} \quad & \sum_{j=1}^{M}\sum_{S_j\in I_j} f_j(S_j)X_j(S_j) \\
s.t. \quad & \sum_{j=1}^{M}\sum_{S_j\in I_j: i\in S_j} X_j(S_j) \le 1, \forall i \in \{1, 2, \cdots N\} \\
& \sum_{S_j\in I_j} X_j(S_j) = 1, \forall j \in \{1, 2, \cdots M\},
\end{aligned}$$

where $f_j(S_j)$ can be computed as

$$f_j(S_j) = \sum_{i\in S_j} \frac{1}{T} A_i R_j \left(t_{ij}x_{ij} - \int_{\sum_{k=1}^{i-1} t_{kj}x_{kj}}^{\sum_{k=1}^{i} t_{kj}x_{kj}} F_j(t_j)dt_j \right), \quad (5)$$

the first constraint means that each vehicle can be scheduled to at most one channel and the second constraint means only one set of vehicles can be scheduled to a channel.

B. Reducing the Number of Primal Variables Using the Dual Problem

The Lagrangian equation and dual problem of Problem (5) are as follows:

$$\begin{aligned}
L(\boldsymbol{X}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = & \sum_{j=1}^{M}\sum_{S_j\in I_j} f_j(S_j)X_j(S_j) \\
& + \sum_{i=1}^{N} \mu_i \left(1 - \sum_{j=1}^{M}\sum_{S_j\in I_j: i\in S_j} X_j(S_j) \right) \\
& + \sum_{j=1}^{M} \lambda_j \left(1 - \sum_{S_j\in I_j} X_j(S_j) \right), \quad (6)
\end{aligned}$$

$$\begin{aligned}
\min_{\boldsymbol{\mu},\boldsymbol{\lambda}} \quad & \sum_{i=1}^{N} \mu_i + \sum_{j=1}^{M} \lambda_j \\
s.t. \quad & \sum_{i\in S_j} \mu_i + \lambda_j \ge f_j(S_j), \forall i, S_j \in I_j \\
& \mu_i \ge 0, \forall i \in \{1, 2, \cdots N\}.
\end{aligned}$$

Although the dual Problem (7) has an exponential number of constraints, it is guaranteed to be solved in polynomial time using the ellipsoid algorithm. Before we solve this LP problem, we give some definitions and theorems on the ellipsoid algorithm. Here, we only introduce the general case of the ellipsoid algorithm in solving existence problems, i.e., whether there exists an $\boldsymbol{x}$ in a convex set defined by $P \in \{\boldsymbol{x}|A\boldsymbol{x} \le \boldsymbol{b}\}$ where $A \in R^{m\times n}, \boldsymbol{b} \in R^m$. For the optimization problem $\{\max \boldsymbol{c}^T\boldsymbol{x}|A\boldsymbol{x} \le \boldsymbol{b}\}$, the idea is very similar to the existence problem and a detailed procedure of the ellipsoid algorithm can be found in [31].

The ellipsoid algorithm works as follows to solve the existence problem. It starts with a big ellipsoid E that is guaranteed to contain P. It then checks whether the center of the ellipsoid $\boldsymbol{a}$ is in P. If it is, the algorithm terminates. Otherwise, it finds an inequality $A_i\boldsymbol{x} \le b_i$ violated by the center, where A_i is the i-th row of matrix A and b_i is the

i-th element of $\boldsymbol{b}$. The ellipsoid is split into two parts by the hyperplane defined by $A_i\boldsymbol{x} \leq A_i\boldsymbol{a}$. By choosing the feasible part, the new set becomes $E \bigcap \{\boldsymbol{x}|A_i\boldsymbol{x} \leq A_i\boldsymbol{a}\}$. The procedure iterates in polynomial time until a feasible $\boldsymbol{x}$ is found or it determines that there is no such $\boldsymbol{x} \in P$.

Definition 1 (Separation Oracle): A separation oracle for a linear programming problem with n variables is an algorithm, such that for a given solution $\boldsymbol{x} \in R^n$, the algorithm is able to decide whether $\boldsymbol{x}$ is a feasible solution of the linear programming problem, and finds a constraint that is violated by $\boldsymbol{x}$ if it is not feasible [31].

Theorem 2: For the linear programming problem $\{\max \boldsymbol{c}^T\boldsymbol{x}|A\boldsymbol{x} \leq \boldsymbol{b}\}$ where $A \in R^{m\times n}$ and $\boldsymbol{b} \in R^m, \boldsymbol{c} \in R^n$, let Q be the maximal bit-encoding length of any value in A and $\boldsymbol{b}$, and S be the maximal bit-encoding length of any value in $\boldsymbol{c}$, then given a separation oracle of constraints $A\boldsymbol{x} \leq \boldsymbol{b}$ running in polynomial time, the ellipsoid method can optimize the the linear programming problem in time polynomial with respect to n, Q and S [31].

Proof: A detailed proof is given in [31]. ■

According to Theorem 2, we can solve Problem (7) once we find a polynomial time separation oracle of the constraints, i.e., given a set of solutions $(\lambda_j, \boldsymbol{\mu})$, decide whether there exists $S_j \in I_j$ such that $\sum_{i\in S_j} \mu_i + \lambda_j \geq f_j(S_j)$ and return the violated constraint if the solution is not feasible. Note that U_{ij} is the utility of vehicle i on channel j such that $f_j(S_j) = \sum_{i\in S_j} U_{ij}$, and the constraint can be written as follows:

$$\lambda_j \geq \sum_{i\in S_j}(U_{ij} - \mu_i), \forall j \in \{1, 2, \cdots M\}. \tag{7}$$

The separation oracle now becomes equivalent to finding a feasible set S_j such that $\sum_{i\in S_j}(U_{ij} - \mu_i)$ is maximized for a given $\boldsymbol{\mu}$. This problem is equivalent to a knapsack problem: assigning a set of items into a knapsack to maximize the overall profit subject to a capacity constraint. Each item has a size s_i and profit p_i, and the knapsack has a capacity of W.

There are three general algorithms to solve the knapsack problem in polynomial time. The first one is an exact pseudo-polynomial time algorithm which is applicable to knapsack problems with integer item size and knapsack capacity [34]. Another (1/2)-approximation algorithm works by sorting items according to their profit-to-size ratio p_i/s_i [34]. The third one is a β-approximation algorithm which is used in [35]. The algorithm works by scaling down and rounding the profit p_i into small integers and further using the pseudo-polynomial algorithm mentioned at the beginning of this paragraph to solve the reduced knapsack problem. Both the second and third algorithms require the profits of each item to be fixed.

Note that the profit functions in our problem are coupled and thus the second and third algorithms cannot be used. Therefore, the first pseudo-polynomial time algorithm becomes our only choice. It can be shown that the algorithm applies to knapsack problems with coupling profit functions. A detailed description of the algorithm is given in [34], and thus we only show why the algorithm applies to our problem.

The pseudo-polynomial time algorithm is a dynamic programming algorithm and the core idea is to solve the subproblems recursively to finally solve the initial knapsack problem. Let N and W be defined as the total number of items and the knapsack capacity of a knapsack problem, respectively. Let $V[i, w]$ be the maximum profit achieved by choosing a subset of items with at most i elements, $1 \leq i \leq N$, where the sum of their size is smaller than or equal to w, where $0 \leq w \leq W$. The sub-problem is to determine the values of $V[i, w]$ recursively and in the last step determine $V[N, W]$. The values of $V[i, w]$ can be computed using the following lemma.

Lemma 2: For $1 \leq i \leq N$ and $0 \leq w \leq W$

$$V[i, w] = \max\{V[i-1, w], p_i + V[i-1, w-s_i]\} \tag{8}$$

Proof: A detailed proof can be found in [34]. ■

The values $V[i, w]$ can be obtained with time complexity $O(NW)$. The optimal value of the initial knapsack problem is $V[N, W]$ and its corresponding set can be found by tracking back the procedure of computing $V[N, W]$ with time complexity $O(NW)$. Therefore, the total time complexity of this algorithm is also $O(NW)$.

Next we show that the algorithm can be used to solve our knapsack problem with coupling profit functions. The key observation is that, although the profit function of each vehicle is coupled with other vehicles, the profit of a vehicle becomes fixed when the set of vehicles scheduled ahead of it is known. It is because their transmission order is determined by Lemma 1. In Lemma 2, $V[i-1, w-s_i]$ denotes the maximum total profit achieved by choosing vehicles from $\{v_1, v_2 \cdots v_{i-1}\}$ with total size at most $w - s_i$. Let S_{i-1} be the optimal set attaining $V[i-1, w-s_i]$. Thus S_{i-1} must be a subset of $\{v_1, v_2 \cdots v_{i-1}\}$ and must be scheduled ahead of v_i. Since S_{i-1} is known in the current iteration, the profit of v_i becomes fixed. Therefore, we can further compute $V[i, w]$ using Lemma 2. Using Lemma 2 for $O(NW)$ iterations, we are guaranteed to solve our knapsack problem [34].

One issue of using the dynamic programming algorithm as our separation oracle is that t_{ij} and c_j in Problem (4) are not necessarily integers. This problem can be solved by requiring that the system is time slotted and thus each vehicle can only request integer number of slots in each scheduling cycle. Slotted time approach is widely used in many scheduling problems in the literature such as [22] and [23]. Another issue with the algorithm is that it is pseudo-polynomial instead of polynomial and its time complexity is $O(NW)$. However, this algorithm can be bounded to be polynomial in our case. First, the algorithm is fully polynomial with respect to N, i.e., the number of vehicles. Second, although the algorithm is pseudo-polynomial with respect to W (i.e., c_j in our case), we can bound the value of W. Let W_j be the number of maximum scheduling slots on channel j. Given the length of a time slot t_0, we can bound W_j as follows:

$$W_j = \frac{c_j}{t_0} = \frac{\min\{T, T_j^r\}}{t_0} \leq \frac{T}{t_0}. \tag{9}$$

By adjusting T and t_0 properly, we can make sure our separation oracle is a polynomial-time oracle (see Equation (7)). According to Theorem 2, we can solve Problem (7) in polynomial time.

C. Solving the Primal LP Problem

Using the ellipsoid algorithm, we can reduce the number of variables in the primal LP without performance loss [32]. Since the ellipsoid algorithm always runs for a polynomial number of iterations and only one set S_j is returned in each iteration, the total number of sets for the primal Problem (5) is also polynomial. Therefore, Problem (5) becomes a typical linear programming problem with a polynomial number of variables and constraints, and thus can be solved much more easily in polynomial time.

D. Rounding the Fractional Solution

Since we can obtain a fractional solution $\{X_j(S_j)\}$ by solving Problem (5), we continue to round it to an integer solution with certain performance guarantees. The rounding method works as follows. First, for a given solution to the LP problem, we schedule one set of vehicles S_j to its corresponding channel j with probability $P_j(S_j)$ which is computed as follows:

$$P_j(S_j) = \frac{X_j(S_j)}{\sum_{S_j \in I_j} X_j(S_j)}. \tag{10}$$

It is possible that some vehicles are scheduled to more than one channel since they may be contained in multiple sets. In this case, let g_{ij} be the expected throughput of vehicle i given it is scheduled to channel j which is computed in Equation (12). If a vehicle is scheduled to more than one channel, it is assigned to the channel with the maximum g_{ij} value. Before computing g_{ij}, we define $f_{ij}(S_j)$ as the throughput of υ_i on channel j when it is in the set S_j. The utility of υ_i is affected by the vehicles scheduled ahead of it. However, when the set of vehicles ahead of it is known, its utility is also fixed and can be computed in $O(N)$ as follows:

$$f_{ij}(S_j) = \frac{1}{T} A_i R_j \left(t_{ij} x_{ij} - \int_{\sum_{k=1}^{i-1} t_{kj} x_{kj}}^{\sum_{k=1}^{i} t_{kj} x_{kj}} F_j(t_j) dt_j \right). \tag{11}$$

Given υ_i is scheduled to channel j, its expected throughput can be computed as follows:

$$g_{ij} = \sum_{S_j \in I_j : i \in S_j} f_{ij}(S_j) \frac{X_j(S_j)}{\sum_{S_j \in I_j : i \in S_j} X_j(S_j)}. \tag{12}$$

Since the number of sets obtained by solving Problem (7) is always polynomial with respect to both the number of vehicles and number of channels, and the time complexity of computing $f_{ij}(S_j)$ for a given S_j is $O(N)$, the time complexity of computing g_{ij} is polynomial.

Before we proceed to prove the performance guarantee of our rounding algorithm, we recall the following lemma:

Lemma 3: For any $Y_j \geq 0, l > 1, \sum_j Y_j \leq 1$ and $g_{i1} \geq g_{i2} \geq \cdots \geq g_{il} \geq 0$, the following inequality holds:

$$\begin{aligned} Y_1 g_{i1} + (1 - Y_1) Y_2 g_{i2} + \cdots \prod_{j=1}^{l-1} (1 - Y_j) Y_l g_{il} \\ \geq \left(1 - \left(1 - \frac{1}{l} \right)^l \right) \sum_{j=1}^{l} g_{ij} Y_j. \end{aligned} \tag{13}$$

Proof: The proof can be found in [35]. ■

Define OPT_{LP} as the maximum total weighted throughput obtained by solving Problem (5), and we have the following theorem.

Theorem 3: The total weighted throughput of our rounded solution is at least $(1 - 1/e) OPT_{LP}$.

Proof: For any vehicle υ_i, let Y_j be the probability that it is scheduled to channel j computed as: $Y_j = \sum_{S_j \in I_j : i \in S_j} X_j(S_j)$. Let l_i be the number of channels including υ_i. Then, sort and rename all the channels including υ_i in the non-increasing order of g_{ij} such that $g_{i1} \geq g_{i2} \geq \cdots \geq g_{il_i}$. According to our rounding method, vehicle i is scheduled to the channel with the maximum g_{ij} value, hence the final throughput of vehicle i is g_{i1} with probability Y_1. Similarly, the final throughput of vehicle i is g_{i2} with probability $(1 - Y_1) Y_2$. In this way, we can compute the expected throughput of vehicle i as follows:

$$\begin{aligned} E[g_{ij}] &= Y_1 g_{i1} + (1 - Y_1) Y_2 g_{i2} + \cdots + \prod_{j=1}^{l_i - 1} (1 - Y_j) Y_{l_i} g_{il_i} \\ &\geq \left(1 - \left(1 - \frac{1}{l_i} \right)^{l_i} \right) \sum_{j=1}^{l_i} g_{ij} Y_j \\ &= \left(1 - \left(1 - \frac{1}{l_i} \right)^{l_i} \right) \sum_{j=1}^{l_i} \sum_{S_j \in I_j : i \in S_j} f_{ij}(S_j) X_j(S_j) \\ &\geq \left(1 - \frac{1}{e} + \frac{1}{32 l_i^2} \right) \sum_{j=1}^{l_i} \sum_{S_j \in I_j : i \in S_j} f_{ij}(S_j) X_j(S_j) \\ &\geq \left(1 - \frac{1}{e} \right) \sum_{j=1}^{l_i} \sum_{S_j \in I_j : i \in S_j} f_{ij}(S_j) X_j(S_j) \end{aligned} \tag{14}$$

The first inequality is due to Lemma 3 and the second inequality is proved in [35]. Note that the contribution of vehicle i in the objective function of Problem (5) is

$$\sum_{j=1}^{l_i} \sum_{S_j \in I_j : i \in S_j} f_{ij}(S_j) X_j(S_j). \tag{15}$$

Therefore, each vehicle is guaranteed to achieve at least $(1 - 1/e)$ of the throughput achieved in the LP problem and thus the total throughput is also guaranteed to have an approximation ratio of $(1 - 1/e)$. ■

E. Time Complexity

The time complexity of the LP algorithm is largely determined by the ellipsoid algorithm. The ellipsoid algorithm requires $O(n^2 L)$ iterations [36], where $n = M + N$ is the dimension of our problem and $L = NT$ is the length of the input data in bits in our case. Each iteration requires $O(MN + 2(M + N)^2)$ operations [36]. Therefore, the number of variables in the primal LP problem is $O(NT(M + N)^4)$. As discussed earlier (see Equation (9) and the following discussions), the value of T is bounded, and thus Problem (5) becomes an LP problem with $O(N(M + N)^4)$ number of variables that can be solved using common LP methods.

V. Solution 2: Submodular Set Function-Based Algorithm

Although the LP algorithm can be guaranteed to run in polynomial time, the time complexity is still high for large problem instances. Thus, more time efficient algorithms are needed. Submodular set functions have been used to solve many NP-hard combinatorial optimization problems as well as resource allocation problems at low time complexity [37]. In this section, we first prove that our objective function of Problem (4) can be written as a non-decreasing submodular set function. Then, we propose two deterministic constant-factor approximation algorithms based on the work in [38]. Below are some basic definitions and properties of submodular set functions from [39].

Definition 2 (Submodularity): Let S be a non-empty finite set and f be a mapping from the power set of S to non-negative real numbers. f is said to be submodular if it satisfies $f(A \cup \{v\}) - f(A) \geq f(B \cup \{v\}) - f(B)$ for all $v \in S \setminus B$ and all $A \subseteq B \subseteq S$.

Definition 3 (Monotonicity): Let S be a non-empty finite set and f be a mapping from the power set of S to non-negative real numbers. Then f is non-decreasing if it satisfies $f(A) \leq f(B)$ for all $A \subseteq B \subseteq S$.

Lemma 4: A positive linear combination of submodular functions is submodular.

A. Submodularity and Monotonicity of the Objective Function

In this part, we proceed to show the submodularity and monotonicity of our objective function by presenting the following theorem, and the proof is provided in Appendix B.

Theorem 4: The objective function in Problem (4) can be written as a non-decreasing submodular set function.

Note that since there are multiple channels in our system, once a vehicle is selected to transmit, it is scheduled to the channel which maximizes its utility.

B. Submodular Set Function-Based Algorithm

Our problem becomes the maximization of a monotone submodular set function under linear packing constraints. We first show that the algorithm proposed in [38] applies to our problem. In [38], the authors study the problem of maximizing a monotone submodular set function $f(S)$ subject to linear packing constraints $A\boldsymbol{x} \leq \boldsymbol{b}$, where $f : 2^{[n]} \rightarrow \mathbf{R}^+, A \in [0,1]^{m\times n}, \boldsymbol{b} \in [1,\infty)^m$. Since we have proved the objective function of Problem (4) is a monotone submodular set function, we now proceed to show that the constraints of Problem (4) can be written as $A\boldsymbol{x} \leq \boldsymbol{b}$, where $A \in [0,1]^{(M+N)\times MN}, \boldsymbol{b} \in [1,\infty)^{M+N}$. Recall that the M packing constraints of the initial problem are $\sum_{i=1}^{N} t_{ij} x_{ij} \leq c_j, \forall j \in \{1, 2 \cdots M\}$. We define the new constraint matrix as follows: $A_{ij} = \frac{t_{ij}}{\max_k\{t_{kj}\}}, b_j = \frac{c_j}{\max_k\{t_{kj}\}}, \forall j \in \{1, 2 \cdots M\}$. According to the network model, we have $c_j = \min\{T, T_j^r\} \geq t_{ij} = \min\{T, T_j^r, L_i/R_j\}$ and thus $A \in [0,1]^{(M+N)\times MN}, \boldsymbol{b} \in [1,\infty)^{M+N}$. In contrast, the format of the N assignment constraints don't need to be changed since $A_{ij} \in \{0,1\}$, $b_j = 1, \forall j \in \{M+1, M+2 \cdots M+N\}$. Therefore, Problem (4) becomes a standard problem of maximizing a monotone submodular set function subject to linear packing constraints as studied in [38].

Algorithm 1 Submodular Set Function Method

Require: The total weighted throughput function $f(\boldsymbol{x})_{\boldsymbol{x}\in S} = \sum_{j=1}^{M} f_j(S_j)$ and its marginal value function $f_S(\{i\}) = f(S\cup\{i\}) - f(S)$, knapsack constraints $A\boldsymbol{x} \leq \boldsymbol{b}$, an update factor $\lambda = e^P(M+N)$, where $P = \min\{b_i/A_{ij} : A_{ij} > 0\}$

$S \leftarrow \emptyset$
for $j = 1$ **to** $M+N$ **do**
  $w_j = 1/b_j$
end for
while $\sum_{j=1}^{M+N} b_j w_j \leq \lambda$ and $|S| < N$ **do**
  Let $i \in [N] \setminus S$ be the element with minimal $\sum_{j=1}^{M+N} A_{ij} w_j / f_S(\{i\})$ and j_i be its corresponding channel
  $S \leftarrow S \cup \{i\}$ (i.e., $x_{ij_i} = 1$)
  for $j = 1$ **to** $M+N$ **do**
    $w_j = w_j \lambda^{A_{ij}/b_j}$
  end for
end while
if $A\boldsymbol{x} \leq \boldsymbol{b}$ **then**
  STOP
else
  if $f(S \setminus \{i\}) \geq f(\{i\})$ **then**
    $S \leftarrow S \setminus \{i\}$
  else
    $S = \{i\}$
  end if
end if
return A subset S of $[N]$ which is denoted by $\boldsymbol{x} = \{x_{11}, x_{21}, \cdots x_{N1}, x_{12}, x_{22}, \cdots x_{N2}, x_{1M}, x_{2M}, \cdots x_{NM}\}$ where $x_{ij} \in \{0,1\}$.

The algorithm in [38] is based on a multiplicative update method that works as follows. The algorithm maintains a set of weight factors $\{w_j\}$ that are updated in a multiplicative way. The factors are designed to indicate the extent to which each constraint is close to being violated by the current solution. The algorithm has only one loop and it extends the current solution with a non-selected element that minimizes a normalized sum of weights in each iteration. After the loop terminates, the final solution is returned as long as it is feasible, otherwise, either the last selected element or the resulting solution without the last solution is returned. Let $[N]$ be the set of all vehicles, and the detailed scheduling algorithm is given in Algorithm 1.

Theorem 5: Algorithm 1 is a deterministic polynomial-time algorithm that achieves an approximation ratio larger than $1/(2e \times (M+N)^{(1/P)} + 2)$ for maximizing a monotone submodular function subject to linear packing constraints, where $A \in [0,1]^{(M+N)\times MN}, \boldsymbol{b} \in [1,\infty)^{M+N}$ and $P = \min(b_j/A_{ij} : A_{ij} > 0)$.

Proof: See the proof of [38, Lemma 2.4]. ■

The time complexity of the submodular set function-based algorithm can be computed as follows. Algorithm 1 will terminate after at most N iterations since the algorithm selects at least one element from the N elements in each iteration. Moreover, in each iteration, there are $O(NM)$ operations since the algorithm computes the $\sum_{j=1}^{M} A_{ij}\omega_j/f_S(\{i\})$ value of each element. Therefore, the time complexity of Algorithm 1 is $O(MN^2)$.

C. Improved Submodular Set Function-Based Algorithm

In Algorithm 1, we can see that the approximation ratio $1/(2e \times (M+N)^{(1/P)}+2)$ increases with decreasing P (which is also called the "width" of the linear packing constraints in [38]). In other words, the algorithm achieves higher approximation ratios for larger P values. In particular, the approximation ratio approaches $1/(2e+2)$ for sufficiently large P values. However, in Problem (4), there are not only M linear packing constraints, but also N assignment constraints (which are special cases of packing constraints). Given this observation, we can directly compute P as $P = \min\{b_j/A_{ij} : A_{ij} > 0\} = 1$. Hence, the provable approximation ratio of Algorithm 1 is merely $1/(2e \times (M+N)+2)$, which is small compared with the preceding $(1-1/e)$-approximation LP algorithm.

The reason why Algorithm 1 achieves this low approximation ratio is that the condition for the termination of the main loop (i.e., $\sum_{j=1}^{M+N} b_j\omega_j \le \lambda$) is rather conservative (see Line 5 of Algorithm 1). More specifically, the algorithm terminates far before the linear constraints are violated. Actually, violation of $\sum_{j=1}^{M+N} b_j\omega_j \le \lambda$ is merely a necessary condition for the violation of linear constraints $A\boldsymbol{x} \le \boldsymbol{b}$, but not a sufficient condition as will be shown in Lemma 5. In other words, the linear constraints may not be violated even though the condition $\sum_{j=1}^{M+N} b_j\omega_j \le \lambda$ is violated. Therefore, it is reasonable to replace the loop termination condition $\sum_{j=1}^{M+N} b_j\omega_j \le \lambda$ with the exact linear constraints $A\boldsymbol{x} \le \boldsymbol{b}$, and prove that the new algorithm achieves a higher approximation ratio than the initial Algorithm 1 without increasing its time complexity.

Before we present the proof, we first introduce some notations consistent with those used in [38] as well as a new lemma.

1) First, let $m = M + N$ be the dimension of the linear constraints of Problem (4).
2) Let S_t be the selected set of vehicles at iteration t after the termination of the main loop of Algorithm 1 (i.e., from Line 5 to Line 11 in Algorithm 1), and $S_0 = \emptyset$. Notice that S_t may not be feasible since the main loop itself doesn't guarantee the feasibility of selected sets. Actually, the feasibility of the selected set is considered in Line 12, and the final returned set is guaranteed to be feasible at the end of each iteration.
3) Let S^* be an optimal set of vehicles which maximizes the submodular set function subject to the linear packing and assignment constraints with the value of $f(S^*)$. Note that it is possible that $f(S_t) \ge f(S^*)$ since S_t may not be feasible.
4) Let ω_{jt} be the value of ω_j at iteration t, and $\{\omega_{j0} = 1/b_j, \forall j\}$ be initial values.

Lemma 5: Violation of $\sum_{j=1}^{m} b_j\omega_j \le \lambda$ is a necessary condition for the violation of linear constraints $A\boldsymbol{x} \le \boldsymbol{b}$, but not a sufficient condition.

Proof: First, we show that $\sum_{i=1}^{MN} A_{ij}x_{ij} > b_j$ (i.e., violation of $A\boldsymbol{x} \le \boldsymbol{b}$) implies $\sum_{j=1}^{m} b_j\omega_j > \lambda$. Suppose at iteration t, $\sum_{i=1}^{MN} A_{ij^*}x_{ij^*} = \sum_{i\in S_t} A_{ij^*} > b_{j^*}$ for some j^*, then we have

$$\sum_{j=1}^{m} b_j\omega_j = \sum_{j=1}^{m} b_j\omega_{jt} > b_{j^*}\omega_{j^*t} = b_{j^*}\omega_{j^*0}\prod_{i\in S_t}\lambda^{A_{ij^*}/b_{j^*}} = \lambda^{\sum_{i\in S_t} A_{ij^*}/b_{j^*}} > \lambda, \tag{16}$$

which proves the first part of this lemma. In Equation (16), the first equality comes from the definition of ω_{jt}, and the third equality is due to $\omega_{j0} = 1/b_j, \forall j$ (see Line 3 of Algorithm 1). The first inequality is due to $b_j > 0, \omega_{jt} > 0, \forall j$, and the second inequality is due to $\sum_{i=1}^{MN} A_{ij^*}x_{ij^*} = \sum_{i\in S_t} A_{ij^*} > b_{j^*}$.

Furthermore, we show that $\sum_{j=1}^{m} b_j\omega_j > \lambda$ doesn't necessarily imply $\sum_{i=1}^{MN} A_{ij}x_{ij} > b_j$ for some j. We prove this by presenting a counterexample in which $\sum_{j=1}^{m} b_j\omega_j > \lambda$ while $\sum_{i=1}^{MN} A_{ij}x_{ij} \le b_j, \forall j$. Suppose that $\sum_{i=1}^{MN} A_{ij_0}x_{ij_0} = b_{j_0}$ for some j_0 and $\sum_{i=1}^{MN} A_{ij}x_{ij} < b_j, \forall j \ne j_0$. Then we have

$$\begin{aligned}\sum_{j=1}^{m} b_j\omega_j &= \sum_{j\ne j_0} b_j\omega_{jt} + b_{j_0}\omega_{j_0t} > b_{j_0}\omega_{j_0t}\\ &= b_{j_0}\omega_{j_00}\prod_{i\in S_t}\lambda^{A_{ij_0}/b_{j_0}}\\ &= \lambda^{\sum_{i\in S_t} A_{ij_0}/b_{j_0}} = \lambda,\end{aligned} \tag{17}$$

which shows that $\sum_{j=1}^{m} b_j\omega_j > \lambda$ is not a sufficient condition for violation of linear constraints $A\boldsymbol{x} \le \boldsymbol{b}$. In Equation (17), the inequality is due to $b_j > 0, \omega_{jt} > 0, \forall j$, and the last equality is due to $\sum_{i=1}^{MN} A_{ij_0}x_{ij_0} = \sum_{i\in S_t} A_{ij_0} = b_{j_0}$. ■

Next, we proceed to develop a new constant-factor algorithm based on Algorithm 1 by proving the following theorem.

Theorem 6: Algorithm 1 is still a deterministic polynomial-time algorithm after replacing the loop termination condition $\sum_{j=1}^{m} b_j\omega_j \le \lambda$ (in Line 5 of Algorithm 1) with the exact linear constraints $A\boldsymbol{x} \le \boldsymbol{b}$. The new algorithm achieves a higher approximation ratio than the initial Algorithm 1 for maximizing a monotone submodular function, subject to linear packing and assignment constraints. In particular, the new algorithm becomes a deterministic (1/2)-approximation algorithm when $f(S_t) \ge f(S^*)$, where t denotes the number of iterations after which Algorithm 1 terminates.

Proof: Suppose the main loop terminates after t iterations, and $t \le N$ since the algorithm schedules at least one vehicle at each iteration. Furthermore, notice that the main loop terminates when either $S_t = [N]$ or $\sum_{i=1}^{MN} A_{ij}x_{ij} > b_j$ for some $j \in \{1, 2\cdots m\}$. For the first case, it can be shown that the returned solution is (1/2)-approximation to the optimal solution. More specifically, if S_t is returned by the algorithm then the outcome is clearly optimal since S_t

contains all vehicles without violating any constraint, and if either $S_t \setminus \{j\}$ or $\{j\}$ is returned (where $\{j\}$ is the vehicle selected at iteration t), then the value of the solution is a (1/2)-approximation since

$$\max\{f(S_t \setminus \{j\}), f(\{j\})\} \geq \frac{1}{2}(f(S_t \setminus \{j\}) + f(\{j\}) \geq \frac{1}{2} f(S_t), \quad (18)$$

where the last inequality is due to the submodularity of f. Hence, in the remainder of this proof, we assume that $S_t < [N]$ and the algorithm terminates only when $\sum_{i=1}^{MN} A_{ij} x_{ij} > b_j$ for some j.

First, we prove that new algorithm becomes a deterministic (1/2)-approximation algorithm when $f(S_t) \geq f(S^*)$, which is possible since S_t may not be feasible under the new loop termination condition $\sum_{i=1}^{MN} A_{ij} x_{ij} > b_j$ for some j. However, notice that S_{t-1} (i.e., $S_t \setminus \{j\}$) must be feasible, since otherwise the loop would have terminated at iteration $(t-1)$. Hence, either S_{t-1} or $\{j\}$ will be returned by Algorithm 1. Therefore, Equation (18) implies

$$\max\{f(S_{t-1}), f(\{j\})\} \geq \frac{1}{2} f(S_t) \geq \frac{1}{2} f(S^*), \quad (19)$$

which shows that the new Algorithm 1 achieves an approximation ratio of 1/2 when $f(S_t) \geq f(S^*)$. Furthermore, we show that the new algorithm achieves a higher approximation ratio than the initial Algorithm 1. This claim can be directly obtained from Lemma 5. More specifically, Lemma 5 shows that the initial Algorithm 1 terminates before the new algorithm, and the selected vehicles by the two algorithms are the same at each iteration before the termination of the initial Algorithm 1. Hence, the final returned set of vehicles by the initial Algorithm 1 is merely a subset of those returned by the new algorithm. Moreover, notice that both algorithms guarantee the feasibility of returned set at the end of each iteration. Therefore, the new algorithm achieves a higher utility than the initial one due to the monotonicity of the submodular set function, i.e., the new algorithm achieves a higher approximation ratio than the initial Algorithm 1. ■

In addition, notice that the time complexity of the improved algorithm is still $O(MN^2)$, since at least one vehicle is selected at each iteration and the number of operations at each iteration does not change.

VI. Numerical Results

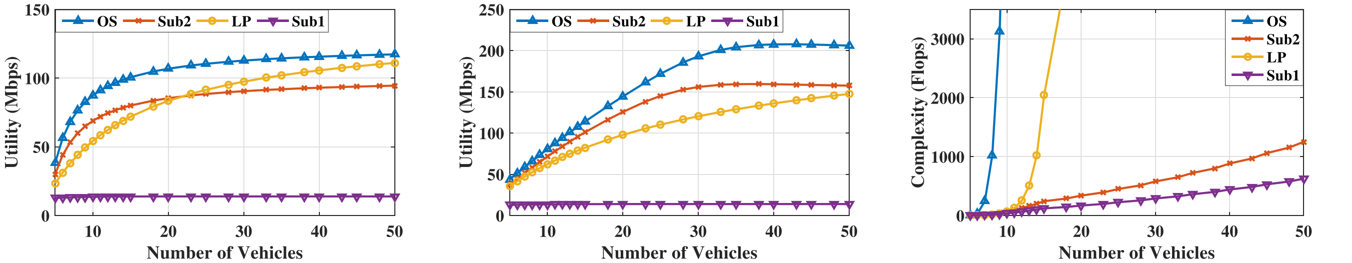
In this section, we evaluate the performance of the proposed algorithms by simulating a cognitive vehicular network with N vehicles and M TVWS channels, where $N \in [5, 50]$ and $M \in \{5, 10\}$. For each given pair of (N, M), we run the three channel allocation algorithms for 100 iterations, each iteration consisting of 100 scheduling cycles. The scheduling cycle T is 100 milliseconds and the time slot is 4 milliseconds long in the first algorithm. The transmission rate of vehicles on all channels is set to 500 $Kbps$. The arrival rates of packets in four access categories follow Poisson processes with different λ values in $\{100, 150, 200, 150\}$ (packets per second) for 1280-byte packets. Every TVWS channel is free at the beginning of a scheduling cycle with the probability of 0.9.

Residual idle time of the channels are all assumed to follow Gamma distributions with $\alpha_j = 2, \forall j \in \{1, 2 \cdots 10\}$, and $\vec{\beta} = \{10, 10, 6, 19, 22, 27, 28, 27, 24, 25\}$ for ten TVWS channels in default. Since the mean of a Gamma distributed variable is $\frac{\alpha}{\beta}$, the mean residual idle times of the TVWS channels are $\{200, 200, 333, 105, 90, 74, 71, 74, 83, 80\}$ milliseconds. We will also change the $\vec{\beta}$ to evaluate the performance of three algorithms under different amount of PU activities. Given the above parameters, the CDF of the residual idle time of channel j is $F_j(x) = 1 - e^{-\beta_j x} - \beta_j x e^{-\beta_j x}, \forall j$. The collision probability constraints of ten channels are set to $\vec{\gamma} = \{0.04, 0.02, 0.03, 0.02, 0.1, 0.03, 0.1, 0.05, 0.05, 0.08\}$ in Equation (1). Note that the α_j, β_j values are chosen such that the mean return times of PUs on these channels (i.e., between 80 and 200 milliseconds) are comparable with the scheduling cycle (i.e., 100 milliseconds). Moreover, small γ_j values are chosen such that PUs are protected sufficiently. Priority weights of four access categories are set to $\{8, 4, 2, 1\}$. At each scheduling cycle, a vehicle transmits one of the four types of packets with equal probabilities. Each vehicle can only transmit packets of single priority in each scheduling cycle.

In the first simulation, the proposed three algorithms are compared with an optimal solver of nonlinear integer programming problems [40] in terms of achieved utility and time complexity. Let "OS", "LP", "Sub1" and "Sub2" denote the optimal solver, the first algorithm based on linear programming, the second algorithm based on submodular set function and the final improved algorithm based on submodular set function, respectively. Then, we compare the three algorithms for a different number of TVWS channels, i.e., $M \in \{5, 10\}$.

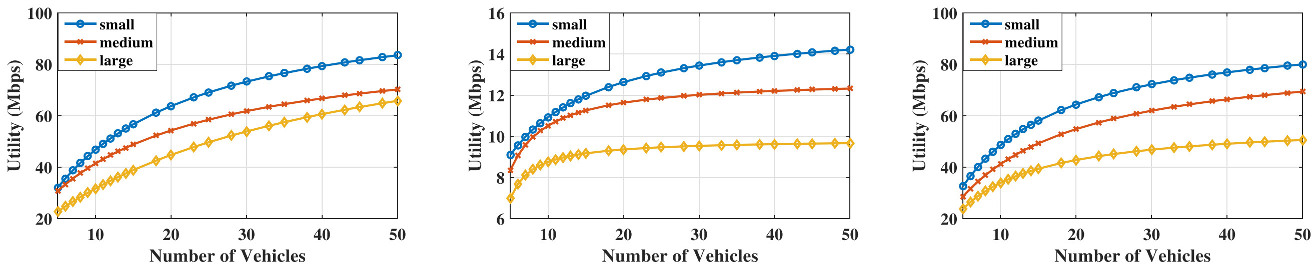
In Fig. (3a) and (3b), we can see that "OS", "LP" and "Sub2" achieve higher utility with increasing vehicle density while the utility change of "Sub1" is substantially small. The reason is that the loop termination condition of "Sub1" is so conservative that the algorithm is only able to schedule the most profitable vehicles. More specifically, in Line 6 of Algorithm 1, "Sub1" selects the vehicle with the minimal $\sum_{j=1}^{M+N} A_{ij} w_j / f_S(\{i\})$ value. Since w_j indicates the extent to which linear constraint j is close to being violated, "Sub1" tends to schedule the vehicle with the minimal $\sum_{j=1}^{M+N} A_{ij} w_j$ value and the maximal marginal utility $f_S(\{i\})$, which can be defined to be the most "profitable" vehicles. However, since "Sub1" terminates far before the constraints are violated, it only achieves a small fraction of optimal utility.

Different from "Sub1", the main loop in "Sub2" terminates until at least one of the real linear constraints is violated, and thus "Sub2" is able to schedule more vehicles than "Sub1", which is shown in Fig. (3a) and (3b). In addition, Fig. (3a) and (3b) also verify that the approximation ratio of "LP" is larger than $(1 - 1/e)$, which is approximately 0.6321. Moreover, it is shown in Fig 3a an 3b that "Sub2" achieves an approximation ratio larger than 1/2. However, it may not be true for other simulation settings because we have not theoretically derived an approximation ratio larger than 1/2.



(a) Utility (M = 5) (b) Utility (M = 10) (c) Time Complexity

Fig. 3. Comparison of LP and two Submodular algorithms in terms of average system utility per cycle and time complexity.



(a) LP Algorithm (b) Sub1 Algorithm (c) Sub2 Algorithm

Fig. 4. Evaluation of the effect of PU activities on the performance of three algorithms.

In Fig. (3c), we can see that "OS" has the highest time complexity because its complexity is exponential while the others are polynomial. In addition, although "Sub2" is proved to have the same time complexity as "Sub1", Fig. (3c) shows that it has higher complexity than "Sub1". This is because "Sub2" will keep scheduling vehicles until some real constraints are violated while "Sub1" stops scheduling vehicles long before the constraints are violated.

In the second simulation, we evaluate the performance of three algorithms with respect to the amount of PU activities for the case $M = 5$. Recall that the residual idle time t_j on channel j is assumed to follow a Gamma distribution. Since $\alpha_j = 2, \forall j$, we can compute the CDF of t_j as $F(t_j) = 1 - e^{-\beta_j t_j} - \beta_j t_j e^{-\beta_j t_j}$, which is a monotonically increasing function over β_j. It means that there are more PU activities on channel j when β_j increases. In addition, we keep the collision probabilities constant in our simulations. Hence, according to Equation (1), we have $P_{coll} = F(T_j^r) = 1 - e^{-\beta_j T_j^r} - \beta_j T_j^r e^{-\beta_j T_j^r} = \gamma_j$ which is constant. It shows that the maximum allowed transmission time T_j^r also decreases with increasing β_j. We evaluate the performance of three algorithms under three different $\{\beta_j\}$ value sets: small $(1.5 \cdot \vec{\beta})$, medium $(3.0 \cdot \vec{\beta})$ and large $(4.5 \cdot \vec{\beta})$, where $\vec{\beta}$ is the setting used in the first simulation. It is verified in Fig. (4) that the system utility decreases with increasing $\vec{\beta}$ due to more PU activities.

In particular, notice that performance of the proposed three algorithms is not affected by the value of scheduling cycle T, i.e., the approximation ratios are independent of T. Instead, the value of T determines the amount of computation load at the RSU. Even though T is set to 100 milliseconds to be compatible with IEEE 1609.4, Figure (3c) shows that the two algorithms based on submodular set function still have superior scalability property. Therefore, the RSU can accommodate high vehicle density scenarios using these two algorithms, and use the linear programming-based algorithm in low vehicle density scenarios.

VII. Conclusion

In this paper, we study the throughput-efficient channel allocation problem in cognitive vehicular networks. First, we formulate the channel allocation problem by considering transmission demands of vehicles and special characteristics of the TVWS channels. Then, we show that the channel allocation problem is an NP-hard non-linear integer programming problem. Furthermore, three constant-factor approximation algorithms are proposed, all with theoretical performance guarantees and polynomial time complexity. Finally, the theoretical analysis is verified by simulation results. The design of algorithms with both good provable approximation ratio and low time complexity is an important topic for our future work. Further research topics also include the study of the channel allocation problem in multi-hop cognitive vehicular networks and design of distributed channel allocation algorithms.

Appendix A
Proof of Lemma 1

First, we prove that for any two vehicles v_1, v_2 scheduled to the same channel, scheduling the vehicle with higher priority ahead of the other vehicle is more rewarding in terms of total weighted throughput of the two vehicles. Let A_1 and A_2 be their priority weights with $A_1 > A_2$ and t_1, t_2 be their scheduled transmission durations. Suppose time duration

$(0, t_0)$ has been assigned to other vehicles, therefore, the first transmitter of the two vehicles starts transmitting at t_0. Let U_{1-2} be their total utility when scheduling v_1 ahead of v_2 and U_{2-1} be the total utility for the reverse order. Let U^1_{1j} and U^1_{2j} be the expected throughput of the two vehicles when v_1 is scheduled ahead of v_2, and U^2_{1j} and U^2_{2j} be the expected throughput of the two vehicles when v_2 is scheduled ahead of v_1. According to Equation (2), we compute U_{1-2} and U_{2-1} in Equation (20) and (21), respectively:

$$\begin{aligned} U_{1-2} &\triangleq U^1_{1j} + U^1_{2j} \\ &= \frac{1}{T} R_j \bigg(A_1 t_1 + A_2 t_2 - A_1 \int_{t_0}^{t_0+t_1} F_j(t_j) dt_j \\ &\quad - A_2 \int_{t_0+t_1}^{t_0+t_1+t_2} F_j(t_j) dt_j \bigg), \end{aligned} \tag{20}$$

$$\begin{aligned} U_{2-1} &\triangleq U^2_{1j} + U^2_{2j} \\ &= \frac{1}{T} R_j \bigg(A_1 t_1 + A_2 t_2 - A_2 \int_{t_0}^{t_0+t_2} F_j(t_j) dt_j \\ &\quad - A_1 \int_{t_0+t_2}^{t_0+t_2+t_1} F_j(t_j) dt_j \bigg). \end{aligned} \tag{21}$$

Then we compare the values of U_{1-2} and U_{2-1} by subtracting U_{2-1} from U_{1-2}:

$$\begin{aligned} U_{1-2} - U_{2-1} = \frac{1}{T} R_j (A_1 - A_2) \bigg(\int_{t_0+t_2}^{t_0+t_2+t_1} F_j(t_j) dt_j \\ - \int_{t_0}^{t_0+t_1} F_j(t_j) dt_j \bigg). \end{aligned} \tag{22}$$

Since $A_1 > A_2$, we now compare the two integral parts which have the same integral function and same interval length t_1. Because $F_j(t_j)$ is a CDF, it is non-decreasing. Since the starting point of the first integral's interval is larger, i.e. $t_0 + t_2 > t_0$, we have $\int_{t_0+t_2}^{t_0+t_2+t_1} F_j(t_j)dt_j \geq \int_{t_0}^{t_0+t_1} F_j(t_j)dt_j$. Therefore, we have $U_{1-2} \geq U_{2-1}$, which proves the first part of Lemma 1. Furthermore, if the two vehicles have the same priority, i.e., $A_1 = A_2$, we can immediately conclude from Equation (22) that $U_{1-2} = U_{2-1}$, which proves the second part of Lemma 1.

Appendix B
Proof of Theorem 4

Recall that S_j has been defined as the set of vehicles scheduled to channel j. We define $f_j(S_j) \triangleq \sum_{i \in S_j} \frac{1}{T} A_i R_j \left(t_{ij} x_{ij} - \int_{\sum_{k=1}^{i-1} t_{kj} x_{kj}}^{\sum_{k=1}^{i} t_{kj} x_{kj}} F_j(t_j) dt_j \right)$ and write the objective function as:

$$\begin{aligned} f(\boldsymbol{x}) &= \sum_{i=1}^{N} \sum_{j=1}^{M} \frac{1}{T} A_i R_j \left(t_{ij} x_{ij} - \int_{\sum_{k=1}^{i-1} t_{kj} x_{kj}}^{\sum_{k=1}^{i} t_{kj} x_{kj}} F_j(t_j) dt_j \right) \\ &= \sum_{j=1}^{M} f_j(S_j). \end{aligned} \tag{23}$$

Since the objective function is a positive linear combination of $f_j(S_j)$, we only need to prove $f_j(S_j)$ is submodular for all j to prove the submodularity of the initial objective function according to Lemma 4. Similarly, if we prove $f_j(S_j)$ is non-decreasing for all j, then the initial objective function is also non-decreasing. First, we prove $f_j(S_j)$ is non-decreasing. Suppose A and B to be two sets of vehicles scheduled to channel j with $A \subseteq B \subseteq S_j$. Set B can be further partitioned into two parts: A and $C = B \setminus A$. When scheduling vehicles in set B, assume vehicles in part A are scheduled ahead of vehicles in part C and let $f_j(C|A)$ be the overall weighted throughput of vehicles in C after A has been scheduled. Hence, according to Lemma 1, $f_j(B) \geq f_j(A) + f_j(C)$ because there may be some vehicles in C whose priorities are higher than some vehicles in A. Therefore, $f_j(B) \geq f_j(A) + f_j(C) \geq f_j(A)$, which proves the monotonicity of $f_j(S_j)$.

Now we continue to prove the submodularity of $f_j(S_j)$. Let A and B be two sets of vehicles scheduled to channel j with $A \subseteq B \subseteq S_j$. Both vehicles in A and B can be partitioned into four parts based on the priorities of their packets and higher priority vehicles are always scheduled ahead of lower priority vehicles. Let t^A_l and t^B_l be the total duration of packets with priority l in A and B respectively where $l \in \{0, 1, 2, 3\}$. Since $A \subseteq B$, it is easy to see that $t^A_l \leq t^B_l, \forall l \in \{0, 1, 2, 3\}$. Therefore, we can rewrite $f_j(A)$ and $f_j(B)$ as follows:

$$\begin{aligned} f_j(A) &= \sum_{i \in A} \frac{1}{T} A_i R_j \left(t_{ij} x_{ij} - \int_{\sum_{k=1}^{i-1} t_{kj} x_{kj}}^{\sum_{k=1}^{i} t_{kj} x_{kj}} F_j(t_j) dt_j \right) \\ &= \frac{1}{T} R_j \sum_{l=0}^{3} A_l \left(t^A_l - \int_{\sum_{k=0}^{l-1} t^A_k}^{\sum_{k=0}^{l} t^A_k} F_j(t_j) dt_j \right), \\ f_j(B) &= \sum_{i \in B} \frac{1}{T} A_i R_j \left(t_{ij} x_{ij} - \int_{\sum_{k=1}^{i-1} t_{kj} x_{kj}}^{\sum_{k=1}^{i} t_{kj} x_{kj}} F_j(t_j) dt_j \right) \\ &= \frac{1}{T} R_j \sum_{l=0}^{3} A_l \left(t^B_l - \int_{\sum_{k=0}^{l-1} t^B_k}^{\sum_{k=0}^{l} t^B_k} F_j(t_j) dt_j \right). \end{aligned} \tag{24}$$

Similarly, for $l = 0$, define $\sum_{k=0}^{l-1} t^A_k = \sum_{k=0}^{l-1} t^B_k = 0$, and $\sum_{k=0}^{l} t^A_k = t^A_0, \sum_{k=0}^{l} t^B_k = t^B_0$. Now we schedule a new vehicle v on channel j whose duration and priority are t_v and l_v, respectively. First, suppose the new vehicle is added to set A. According to Lemma 1, we can always schedule v at the end of vehicles of priority l_v. Thus the remaining scheduled time interval $(\sum_0^{l_v} t^A_l, \sum_0^3 t^A_l)$ is shifted backwards by t_v and thus the new time interval becomes $(\sum_0^{l_v} t^A_l + t_v, \sum_0^3 t^A_l + t_v)$. We first consider the case where $t_v \leq t^A_l, \forall l > l_v$ and we will show our theorem also holds when $t_v > t^A_l$ for some $l > l_v$. In the first case, the marginal utility of v consists $(4 - l_v)$ parts and in each part the corresponding interval is replaced with higher priority packets. Since there are no packets of priority "4", we let $A_4 = 0$. Therefore, we can evaluate the marginal utility of v for both A and B as follows:

$$\begin{aligned} &f_j(A \cup \{v\}) - f_j(A) \\ &\quad = \frac{1}{T} R_j \sum_{l=l_v}^{3} (A_l - A_{l+1}) \left(t_v - \int_{\sum_{k=0}^{l} t^A_k}^{\sum_{k=0}^{l} t^A_k + t_v} F_j(t_j) dt_j \right), \\ &f_j(B \cup \{v\}) - f_j(B) \\ &\quad = \frac{1}{T} R_j \sum_{l=l_v}^{3} (A_l - A_{l+1}) \left(t_v - \int_{\sum_{k=0}^{l} t^B_k}^{\sum_{k=0}^{l} t^B_k + t_v} F_j(t_j) dt_j \right). \end{aligned} \tag{25}$$

Hence, we can obtain the following result:

$$\begin{aligned} &\left[f_j(A\cup\{v\})-f_j(A)\right]-\left[f_j(B\cup\{v\})-f_j(B)\right] \\ &= \frac{1}{T}R_j\sum_{l=l_v}^{3}(A_l-A_{l+1})\Bigg(\int_{\sum_{k=0}^{l}t_k^B}^{\sum_{k=0}^{l}t_k^B+t_v}F_j(t_j)dt_j \\ &\qquad -\int_{\sum_{k=0}^{l}t_k^A}^{\sum_{k=0}^{l}t_k^A+t_v}F_j(t_j)dt_j\Bigg)\geq 0. \end{aligned} \tag{26}$$

The last inequality is due to the facts that (1) $A_i > A_j, \forall i < j$, (2) $t_l^A \leq t_l^B, \forall l \in \{0,1,2,3\}$ and (3) the CDF $F_j(t_j)$ is non-decreasing. Equation (26) proves the theorem when $t_v \leq t_l^A, \forall l > l_v$. When $t_v > t_l^A$ for some $l > l_v$, one can simply partition t_v into sufficiently small sub-intervals $t_{vi}, \forall i \in \{1,2\cdots N_i\}$ such that $t_{vi} \leq t_l^A, \forall l > l_v$, and schedule one interval each time. Since all intervals t_{vi} satisfy Equation (26), we can see that Equation (26) still holds after scheduling N_i sub-intervals. Therefore, the theorem still holds when $t_v > t_l^A$ for some $l > l_v$.

REFERENCES

[1] X. Ma, X. Chen, and H. H. Refai, "Performance and reliability of DSRC vehicular safety communication: A formal analysis," *EURASIP J. Wireless Commun. Netw.*, vol. 1, Jan. 2009, Art. no. 3.

[2] M. Hassan, H. L. Vu, and T. Sakurai, "Performance analysis of the IEEE 802.11 MAC protocol for DSRC safety applications," *IEEE Trans. Veh. Technol.*, vol. 60, no. 8, pp. 3882–3896, Oct. 2011.

[3] K. Fawaz, A. Ghandour, M. Olleik, and H. Artail, "Improving reliability of safety applications in vehicle ad hoc networks through the implementation of a cognitive network," in *Proc. IEEE 17th Int. Conf. Telecommun. (ICT)*, Apr. 2010, pp. 798–805.

[4] Z. Wang and M. Hassan, "How much of DSRC is available for non-safety use?" in *Proc. 5th ACM Int. Workshop Veh. Inter-Netw. (VANET)*, New York, NY, USA, 2008, pp. 23–29.

[5] Y. Han, E. Ekici, H. Kremo, and O. Altintas, "Optimal spectrum utilization in joint automotive radar and communication networks," in *Proc. 14th Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, May 2016, pp. 1–8.

[6] G. Karagiannis *et al.*, "Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions," *IEEE Commun. Surveys Tut.*, vol. 13, no. 4, pp. 584–616, 4th Quart., 2011.

[7] Y. Han, E. Ekici, H. Kremo, and O. Altintas, "A survey of MAC issues for TV white space access," *Ad Hoc Netw.*, vol. 27, pp. 195–218, Apr. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570870514002431

[8] Federal Communication Commission. *Second Report and Order and Memorandum Opinion and Order In the Matter of Unlicensed Operation in the TV Broadcast Bands Additional Spectrum for Unlicensed Devices Below 900 MHz and in the 3 GHz Band*, Nov. 2008, pp. 208–260.

[9] Federal Communication Commission. *Second Memorandum Opinion and Order In the Matter of Unlicensed Operation in the TV Broadcast Bands Additional Spectrum for Unlicensed Devices Below 900 MHz and in the 3 GHz Band*, Sep. 2010, pp. 110–174.

[10] Y. Han, E. Ekici, H. Kremo, and O. Altintas, "Spectrum sharing methods for the coexistence of multiple RF systems: A survey," *Ad Hoc Netw.*, vol. 53, pp. 53–78, Dec. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570870516302153

[11] T. Jiang, H. Wang, and A. V. Vasilakos, "QoE-driven channel allocation schemes for multimedia transmission of priority-based secondary users over cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 7, pp. 1215–1224, Aug. 2012.

[12] H. Xu and B. Li, "Resource allocation with flexible channel cooperation in cognitive radio networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 5, pp. 957–970, May 2013.

[13] C. Luo, G. Min, F. R. Yu, Y. Zhang, L. T. Yang, and V. C. M. Leung, "Joint relay scheduling, channel access, and power allocation for green cognitive radio communications," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 5, pp. 922–932, May 2015.

[14] H. M. Almasaeid and A. E. Kamal, "Receiver-based channel allocation in cognitive radio wireless mesh networks," *IEEE/ACM Trans. Netw.*, vol. 23, no. 4, pp. 1286–1299, Aug. 2015.

[15] X. Zhang, L. Guo, T. Song, W. Xu, Y. Li, and J. Lin, "Dynamic channel allocation supporting multi-service over cognitive radio networks," in *Proc. 16th Int. Symp. Wireless Pers. Multimedia Commun. (WPMC)*, Jun. 2013, pp. 1–5.

[16] L. He, B. Liu, Y. Yao, N. Yu, and C. W. Chen, "MOS-based channel allocation schemes for mixed services over cognitive radio networks," in *Proc. 7th Int. Conf. Image Graph. (ICIG)*, Jul. 2013, pp. 832–837.

[17] S. Li, T. H. Luan, and X. Shen, "Channel allocation for smooth video delivery over cognitive radio networks," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Dec. 2010, pp. 1–5.

[18] Y. Han, E. Ekici, H. Kremo, and O. Altintas, "Throughput-efficient channel allocation in multi-channel cognitive vehicular networks," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr./May 2014, pp. 2724–2732.

[19] H. Kremo and O. Altintas, "On detecting spectrum opportunities for cognitive vehicular networks in the TV white space," *J. Signal Process. Syst.*, vol. 73, no. 3, pp. 243–254, Dec. 2013.

[20] Y. Han, E. Ekici, H. Kremo, and O. Altintas, "Enabling coexistence of cognitive vehicular networks and IEEE 802.22 networks via optimal resource allocation," in *Proc. 13th Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, May 2015, pp. 451–458.

[21] D. Cabric, S. M. Mishra, and R. W. Brodersen, "Implementation issues in spectrum sensing for cognitive radios," in *Proc. Conf. Rec. 38th Asilomar Conf. Signals, Syst. Comput.*, vol. 1. Nov. 2004, pp. 772–776.

[22] R. Urgaonkar and M. J. Neely, "Opportunistic scheduling with reliability guarantees in cognitive radio networks," *IEEE Trans. Mobile Comput.*, vol. 8, no. 6, pp. 766–777, Jun. 2009.

[23] D. Xue and E. Ekici, "Guaranteed opportunistic scheduling in multi-hop cognitive radio networks," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2011, pp. 2984–2992.

[24] S. Zhang, A. S. Hafid, H. Zhao, and S. Wang, "Cross-layer rethink on sensing-throughput tradeoff for multi-channel cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 10, pp. 6883–6897, Oct. 2016.

[25] Y. Alayev *et al.*, "Throughput maximization in mobile WSN scheduling with power control and rate selection," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 4066–4079, Jul. 2014.

[26] Y. M. Cho, J. Yang, S. G. Hwang, D. W. Byun, D. K. Kim, and Y.-I. Kim, "OFDM-TDD based distributed mobile relay power control scheme for downlink throughput enhancement," in *Proc. ICCIT*, Nov. 2009, pp. 897–902.

[27] T. Nitsche, C. Cordeiro, A. B. Flores, E. W. Knightly, E. Perahia, and J. C. Widmer, "IEEE 802.11ad: Directional 60 GHz communication for multi-Gigabit-per-second Wi-Fi [Invited Paper]," *IEEE Commun. Mag.*, vol. 52, no. 12, pp. 132–141, Dec. 2014.

[28] E. Perahia, C. Cordeiro, M. Park, and L. L. Yang, "IEEE 802.11ad: Defining the next generation multi-Gbps Wi-Fi," in *Proc. 7th IEEE Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2010, pp. 1–5.

[29] Y. Chen, G. Yu, Z. Zhang, H.-H. Chen, and P. Qiu, "On cognitive radio networks with opportunistic power control strategies in fading channels," *IEEE Trans. Wireless Commun.*, vol. 7, no. 7, pp. 2752–2761, Jul. 2008.

[30] M. L. Fisher, R. Jaikumar, and L. N. Van Wassenhove, "A multiplier adjustment method for the generalized assignment problem," *Manage. Sci.*, vol. 32, no. 9, pp. 1095–1103, Sep. 1986.

[31] M. Grötschel, L. Lovász, and A. Schrijver, "The ellipsoid method and its consequences in combinatorial optimization," *Combinatorica*, vol. 1, no. 2, pp. 169–197, 1981.

[32] D. Chakrabarty and G. Goel, "On the approximability of budgeted allocations and improved lower bounds for submodular welfare maximization and GAP," in *Proc. 49th Annu. IEEE Symp. Found. Comput. Sci.*, Oct. 2008, pp. 687–696.

[33] É. Tardos, "A strongly polynomial algorithm to solve combinatorial linear programs," *Oper. Res.*, vol. 34, no. 2, pp. 250–256, Mar. 1986.

[34] G. L. Nemhauser and L. A. Wolsey, *Integer and Combinatorial Optimization*. New York, NY, USA: Wiley, 1988.

[35] L. Fleischer, M. X. Goemans, V. S. Mirrokni, and M. Sviridenko, "Tight approximation algorithms for maximum general assignment problems," in *Proc. ACM-SIAM SODA*, 2006, pp. 611–620.

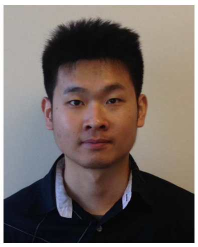
[36] D. Goldfarb and M. J. Todd, "Modifications and implementation of the ellipsoid algorithm for linear programming," *Math. Program.*, vol. 23, no. 1, pp. 1–19, Dec. 1982.

[37] N. Golrezaei, K. Shanmugam, A. G. Dimakis, A. F. Molisch, and G. Caire, "FemtoCaching: Wireless video content delivery through distributed caching helpers," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 1107–1115.

[38] Y. Azar and I. Gamzu, "Efficient submodular function maximization under linear packing constraints," in *Proc. 39th Int. Colloq. Conf. Automata, Lang., Program. (ICALP)*, vol. 1. Berlin, Germany, 2012, pp. 38–50.

[39] G. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions—I," *Math. Program.*, vol. 14, no. 1, pp. 265–294, Dec. 1978.

[40] *Opti Toolbox v2.16*, accessed on Jul. 25, 2015. [Online]. Available: http://www.i2c2.aut.ac.nz/Wiki/OPTI/index.php/Main/HomePage



You Han (S'14) received the B.E. degree in electrical engineering and automation from Zhejiang University, Hangzhou, China, in 2012. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH, USA. He interned with Google [x] Lab (now X: The Moonshot Factory) as a Software Engineer in 2014 and 2015, respectively. He also interned with Google Access and Energy as a Software Engineer in 2016. He was involved in network simulation, embedded system, routing algorithm design, and frontier wireless technologies during the three internships. His research interests include connected vehicles, cognitive radio, network optimization theory, and resource management in wireless networks.

Eylem Ekici (S'99–M'02–SM'11) received the B.S. and M.S. degrees in computer engineering from Boğaziçi University, Istanbul, Turkey, in 1997 and 1998, respectively, and the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2002. He is currently a Professor with the Department of Electrical and Computer Engineering, The Ohio State University. His current research interests include cognitive radio networks, vehicular communication systems, and next generation wireless systems, with a focus on algorithm design, medium access control protocols, resource management, and analysis of network architectures and protocols. He is also the TPC Co-Chair of the IEEE INFOCOM 2017. He is an Associate Editor of the IEEE TRANSACTIONS ON MOBILE COMPUTING and the *Computer Networks Journal* (Elsevier). He was an Associate Editor of the IEEE/ACM TRANSACTIONS ON NETWORKING.

Haris Kremo (M'12) received the Dipl.Ing. degree from the School of Electrical Engineering, University of Sarajevo, Bosnia and Herzegovina, in 2000, and the M.S. and Ph.D. degrees from Rutgers, The State University of New Jersey, in 2005 and 2010, respectively. From 2000 to 2002, he was a Research Engineer with ENERGOINVEST d.d., Sarajevo. From 2003 to 2010, he was a Graduate Assistant with the Wireless Information Networks Laboratory, where he was one of the engineers responsible for design and maintenance of the ORBIT wireless testbed. In 2010, he was with EDGE Lab, Princeton University, as a Post-Doctoral Research Associate. He was a Post-Doctoral Researcher with Toyota InfoTechnology Center Co., Ltd, Tokyo. He is currently a Research Fellow with the CONNECT Centre, Trinity College Dublin, Dublin, Ireland.

Onur Altintas (S'92–M'95) received the B.S. and M.S. degrees from Orta Dogu Teknik Universitesi, Ankara, Turkey, and the Ph.D. degree from the University of Tokyo, Japan, all in electrical engineering. He was a Research Scientist with Ultra High Speed Network and Computer Technology Labs, Tokyo. From 1999 to 2001, he was with Toyota Motor Corporation and a Visiting Researcher with Telcordia Technologies. From 2001 to 2004, he was with Toyota InfoTechnology Center, USA, Inc., Mountain View, CA, USA, where he is currently a Fellow with the Network Division. He is also an IEEE VTS Distinguished Lecturer. He has been the Co-Founder and the General Co-Chair of the IEEE Vehicular Networking Conference since 2009. He serves as an Associate Editor of the IEEE *ITS Magazine* and in the Editorial Board of Connected Vehicles Series of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY.