A Holistic Approach to Service Delivery in Driver-in-the-Loop Vehicular CPS

Xu Li, *Member, IEEE,* Chunming Qiao, *Fellow, IEEE,* Aditya Wagh, *Student Member, IEEE,*
Raghu Sudhaakar, Sateesh Addepalli, Changxu Wu, *Member, IEEE,* and Adel Sadek

***Abstract*—Vehicular *Cyber-Physical Systems* (*VCPS*) provide human drivers with various services related to road safety, and on-road infotainments. Since a service (message) delivery includes *service transmission*, *service display* and *driver processing*, many challenges arise due to limited network resources, possible pre-emption and contention between services for the display and non-negligible driver processing delay. In this paper, we address a new *Driver-centric Service Delivery Problem* (*DSDP*) from a cross-disciplinary resource allocation standpoint. Our goal is to deliver a number of services to a set of intended drivers in a given time period so as to maximize the system-wide performance in terms of *total utility income* (*TUI*) to drivers. We show that DSDP differs from all existing problems and is NP-Complete. A number of efficient heuristics are proposed to address several issues, including wireless transmission failure as well as distributed implementation of the multi-sender systems. Utilizing real traces collected from taxis in the city of Shanghai, we also present a case study in a more realistic scenario and conduct comprehensive simulations providing numerical results.**

***Index Terms*—Service delivery, driver-in-the-loop, resource allocation, vehicular cyber-physical systems, NP-complete.**

I. INTRODUCTION

VEHICULAR Cyber-Physical Systems (*VCPS*) can significantly improve road safety and enrich driving experience and on-road infotainment. Although a lot of research has been done in this field, it has been divided into two main focus areas. On one hand, traditional Human Factors (HF) based research mainly focuses on how human drivers could be affected by external factors such as road signs, on-board warning systems, weather, or other environmental and psychological aspects [2]-[9]. On the other hand, most existing works on VANETs pay attention to only communication and networking protocols design [10]-[18]. However, given the complex nature of the multiple interactions (and influences) among the cyber system, transportation system and the human elements, VCPS research requires a cross-disciplinary study. Compared to existing research, the novelty of this work comes from our integrated study of all three aspects: cyber, transportation and HF.

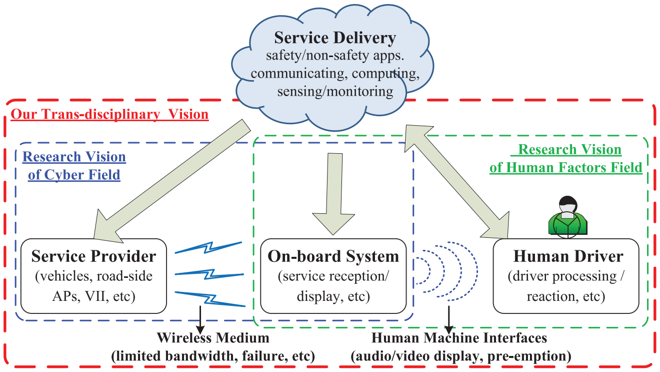


Fig. 1. Service delivery process in friver-in-the-loop VCPS.

Fig. 1 depicts an on-road service delivery process in Driver-in-the-Loop VCPS. Typically, a complete service delivery to a driver goes through three steps: 1) *service transmission*. A service provider (or sender) transmits a service (or message) to a driver (or receiver); 2) *service display*. The on-board system receives a service, and then *presents* the service to the driver through Human Machine Interfaces (HMI). Note that a service may be presented using multiple modalities (audio, visual, tactile, etc), but hereafter, we use the phrase "service display" to loosely refer to all possible service presentations; 3) *driver processing*. The driver tries to comprehend the service and decide appropriate actions as needed, e.g., maneuvering.

More specifically, a service, which could be for example, a safety warning message, is transmitted from a sender to some intended receivers via the wireless medium. Note that firstly, different services require different amount of transmission time. As an example, a text message (e.g., a price bulletin from a gas station) can take much less transmission time than a message with video content (e.g., an advertisement from a restaurant). In this work, we assume that network resources can be used to send only a limited number of services in a given time period.

Secondly, a service can be pre-empted during display (even though different services may be displayed on different modalities), and only a complete (and well-presented) service display can activate the processing of drivers' cognitive systems. More specifically, if a service is pre-empted, the driver's attention will shift to focus on the new service.

Thirdly, the amount of mental resources consumed by the

Manuscript received March 1, 2012; revised July 1, 2012.

X. Li, C. Qiao, and A. Wagh are with the Department of Computer Science and Engineering, the State University of New York at Buffalo, USA (e-mail: drxuli@gmail.com; qiao@buffalo.edu; aswagh@buffalo.edu).

R. Sudhaakar and S. Addepalli are with Cisco Systems, USA (e-mail: rs96@buffalo.edu; sateeshk@cisco.com).

C. Wu is with the Department of Industrial and System Engineering, the State University of New York at Buffalo, USA (e-mail: seanwu@buffalo.edu).

A. Sadek with the Department of Civil, Structural and Environmental Engineering, the State University of New York at Buffalo, USA (e-mail: asadek@buffalo.edu).

This work was supported in part by the U.S. NSF's CPS Program (Grant No. NSF-CNS-1035733) and the Cisco University Research Program.

Digital Object Identifier 10.1109/JSAC.2013.SUP.0513046

driver to process different services may also be different. e.g., a speed warning could be processed immediately whereas a driver may take more time (i.e., more mental resources) to process a more complicated message. Note that initially, the driver is idle and ready to process a service. When a service is being processed by the driver, his/her cognitive system turns into a busy status, during which he/she will not be able to react to any other services.

In this paper, we address a new Driver-centric Service Delivery Problem (DSDP) from a cross-disciplinary resource allocation standpoint. In this problem, a list of services needs to be delivered to a set of intended drivers and each service has a given service utility. In particular, a service can produce a utility income (or UI) at a receiver *if and only if* the receiver can fully process the service in a given time period (say K time units), without any interruption (i.e., service pre-emption). A solution to DSDP is primarily concerned with both service selection and scheduling in the given time period with the objective of maximizing system-wide UI, by considering the limited wireless network and human mental resources. We also address several important issues, such as wireless transmission failure, as well as distributed implementation of service delivery in a multi-sender scenario, where service collision may happen if a driver receives more than one service in a time unit. In addition, we also present a case study representing a realistic scenario by considering a number of practical aspects. Since DSDP has some unique challenges, which differentiate it from all existing problems studied in prior research, novel and holistic solutions are needed.

The main contributions of this work are as follows:

1) To the best of our knowledge, this work is the first that takes a cross-disciplinary and holistic approach to the on-road service delivery problem from a resource allocation standpoint.

2) We show that DSDP is NP-complete, and in particular, we discuss the novelty of DSDP by comparing it with many existing problems, such as the Knapsack Problem and its variations, the Job Scheduling Problem, the Multicast Packing Problem and other well-known theoretical problems.

3) We propose a list of heuristic algorithms for DSDP to address the challenges related to limited wireless network and human mental resources, service display pre-emption, transmission failure, as well as distributed implementation.

4) We present a case study of a small area in the city of Shanghai and examine practical issues, including vehicle mobility by utilizing real traces collected from taxis, dynamic service arrivals, and heterogonous driver processing capabilities.

5) We analyze the performances of the proposed algorithms based on large-scale simulations and provide useful insights.

The remainder of the paper is organized as follows. In Section II, we present assumptions, models and problem definitions. We discuss the problem novelty in Section III. The proposed algorithms are described in Section IV, followed by their performance evaluation in Section V. Section VI presents a case study in a more realistic scenario. We discuss related work in Section VII and Section VIII concludes the paper.

II. PROBLEM DESCRIPTION

We consider the service delivery problem in a given period T, which is divided into a set of K time units 1, 2,...t,...K. In this section, we will start with a basic scenario, where there is only one sender (e.g., a road-side unit/AP), which will use a wireless transmission channel to deliver a set of services to a set of receivers within its transmission range in the given K time units. Then, we introduce an extended scenario with multiple senders, followed by examples for the two scenarios. In this work, we are mainly concerned with one-hop service delivery because it is a typical and dominant transmission model in VANET. How to apply the basic ideas of the proposed solution framework to multi-hop communications is an open question left for future work.

A. Service and Utility

In describing the services and their utilities, we will largely borrow the notations from our previous work on the HF-aware Service Scheduling Problem (HFSSP) [1]. However, note that this work on DSDP is *much different* from and more challenging than HFSSP because in this case, we consider many new issues related to limited wireless network, display, and human mental resources, possible transmission failures, possible service display pre-emption, distributed implementation in the multi-sender scenario, as well as vehicle mobility, dynamic service arrivals and other practical issues.

We denote a set of N services by S=$\{s_i\}$ where i = 1 to N, and a set of M receivers by R=$\{r_j\}$ where j = 1 to M. Each service s_i is targeted for a subset of the M receivers, denoted by $R_i \subseteq R$. For each service s_i, which receivers it should be delivered to can be determined either by the sender of the s_i, or based on any existing service subscription made by the receivers. For simplicity, we assume that each service has identical utility to the different receivers it is targeted for, and the utility of service s_i is given by a function $\mu(i,t)$. We will examine the following two cases. In the first case, we assume a step function of time t which returns a constant during the K time units. That is:

$$\mu(i,t) = C_i, if 1 \leq t \leq K, and\ 0\ otherwise \qquad (1)$$

where $C_i \in [1, C]$ for a positive number C. In the second case, we assume a more general time-variant utility function, which is defined as:

$$\mu(i,t) \geq \mu(i,t'), if 1 \leq t \leq t' \leq K, and\ 0\ otherwise \qquad (2)$$

The utility function as defined in Eq. (1) generally models non-safety application scenarios, in which services (such as infotainment messages that will improve trip experience) always have some fixed (time-invariant) utility to receivers. The utility function as defined in Eq. (2), on the other hand, can often be used to model safety-related application scenarios, in which time-critical safety warning messages whose utility may decrease with time. Note that, it is a non-trivial task to determine the utility of each service. However, the focus and contributions of this work are the algorithms designed for service delivery, which are applicable to any reasonably defined utility functions [26].

B. Modeling

As mentioned earlier, a service delivery process includes service transmission, service display and driver processing. Below, we describe basic assumptions, some of which will be relaxed later in our case study.

a) Service transmission. Although IEEE 802.11p is a recently proposed MAC standard for VANETs, its disadvantages in supporting broadcast have been documented. For example, as indicated in [31], one of the disadvantages of the IEEE 802.11p standard is that, for broadcast frames, no RTS/CTS exchange is used and no acknowledgement is transmitted from any of the recipients of the frame. The lack of RTS/CTS exchange results in a hidden terminal problem which reduces the frame delivery ratio of the broadcast service. Accordingly, the use of TDMA for broadcasting in VANETs has been proposed [18][31][32]. Since in our basic scenario, there is only one sender having the capability of scheduling transmissions (e.g., a road-side AP), it is natural to use a TDMA-based model. In addition, the problem model and the proposed algorithms to be described can also be extended to other scenarios using different transmission models as well.

For each service s_i, we assume that it could be scheduled to transmit at the beginning of time unit t as a multicast operation (we denote such a transmission start time of s_i by $T_{Trans}(i)$), and then received by its receivers after $D_{Trans}(i)$ time units (i.e., at the end of time unit $(t+D_{Trans}(i)-1)$) if there is no transmission error. Since we assume that a sender will only deliver a service to its receivers when the sender comes into contact with receivers within 1-hop transmission range, the multicast operation could be performed by a single broadcast.

Note that in this study, we opt to focus on higher-layer issues and ignore the detailed lower-layer issues such as wireless channel modeling, MAC protocols or even routing and transport issues. Nevertheless, we consider the issue of lossy transmission by defining *tfp(i, j)* to be the transmission failure probability, or *tfp*, of transmitting service s_i to receiver r_j, which may be obtained based on statistics about current channel condition, service size, distance or other factors. In addition, we assume that a new transmission cannot be launched if there is an on-going service transmission.

Since the network resources are limited in the sense that not all the services can be sent out in the given time period, we consider the following two network models to describe the limited network resources:

Network Model I: Different services have different transmission times but each service takes at least one time unit to be transmitted (i.e., $D_{Trans}(i) \geq 1$ for any service s_i). Accordingly, all the K time units can be used for service transmissions (of less than or equal to K services) under this model.

Network Model II: Each service can always be transmitted in one time unit (i.e., $D_{Trans}(i) = 1$). However, a sender can only select fewer than K time units randomly for service transmissions. That is, although the given time period is still K time units, the sender can only transmit $Q < K$ services. This model considers the probability that the network resources are shared by other applications, and therefore, we need to avoid injecting too much traffic into the network. In either model, we assume that, at the sender side, a service can be sent at most once (this assumption will be relaxed in the multi-sender scenario, in which each service is allowed to be sent for more than once).

b) Service display. A service successfully transmitted will be displayed at the beginning of time unit $t + D_{Trans}(i)$ if $T_{Trans}(i) = t$. We denote the amount of time units to completely display service s_i by $D_{Disp}(i) \geq 1$. At the same time, we consider service pre-emption by assuming that if a newly received service s'_i appears while s_i is being displayed at receiver r_j, then s_i will be pre-empted by s'_i and r_j will start to focus on s'_i (and the incomplete/unsuccessful display of s_i cannot induce r_j's cognitive system to become busy for service processing).

It is worth noting that although we adopt the above service pre-emption model, a safety-critical warning will not be pre-empted by a non-safety message if doing so results in a utility loss. In addition, we are not concerned with service display buffering, which may be used to avoid service pre-emption. More specifically, at the receiver side, only one service can be displayed at a time, and a service will not be redisplayed regardless of whether it has been pre-empted in the past. These assumptions are not as restrictive as they appear to be. For example, if a service has been delayed/cached or has been transmitted/displayed for multiple times, in each time, it can be regarded as a new service (i.e., the utility of such a new service will be re-assigned) in our model.

c) Driver processing. A complete and well-presented service s_i turns a driver's cognitive system into a busy status and we denote by $D_{Proc}(i) \geq 1$ the receiver (driver) processing time for service s_i, which is the same for all receivers (This assumption will be generalized later to consider heterogeneous receivers). Let $t = T_{Trans}(i)$ be the time when service s_i is transmitted, and $T_{Complete}(i,t)$ be the time when the service is completely processed by the receiver, then we have $T_{Complete}(i,t) = t + D_{Total}(i) - 1$, where $D_{Total}(i)$ is the total time duration for service transmission, service display and driver processing, i.e., $D_{Total}(i) = D_{Trans}(i) + D_{Disp}(i) + D_{Proc}(i)$. Prior to $T_{Complete}(i,t)$, a receiver r_j will not be able to respond or pay attention to any other services while he/she is already in the busy status (processing the current service) due to limited mental resources. This provides the upper bound on the number of services that a receiver can process during the given K time units.

It is worth noting that while both network models assume limited network resources for service transmission, the effect of limited display and the driver's mental resources is more pronounced in Network Model II than in Network Model I since $D_{Trans}(i)$ accounts for a smaller percentage of $D_{Total}(i)$ in Network Model II. Accordingly, as to be shown in Section V, under Network Model II, it is more important and beneficial to design an algorithm to effectively deal with possible display pre-emption and driver overload.

C. Definitions

Based on the above notations and assumptions, one intuitive assumption is that a receiver will obtain an utility income of $\mu(i, T_{Complete}(i,t))$ when it completes the processing of a received service s_i. Since not all services transmitted can

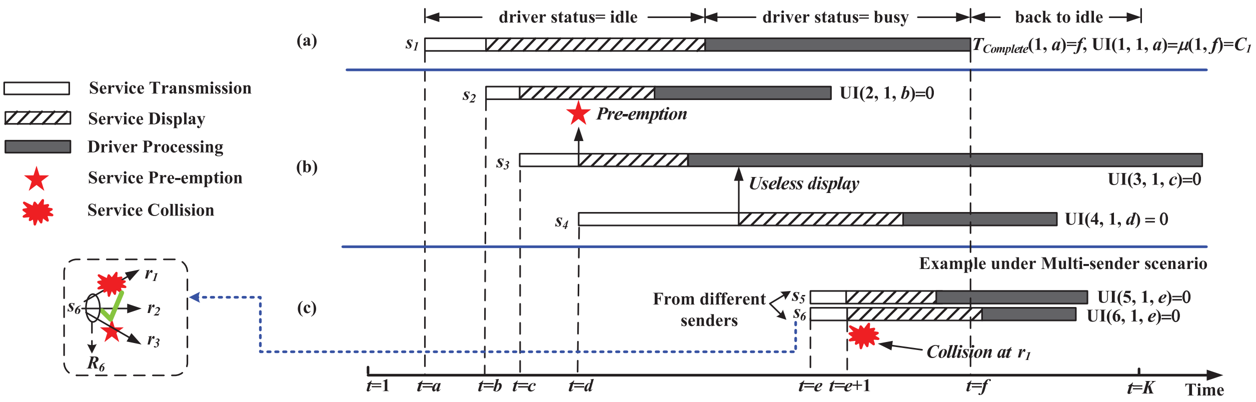


Fig. 2. Examples illustrating service delivery at receiver r_1.

be successfully displayed (due to possible pre-emption) and processed in time (due to the fact that the driver is busy when the service is displayed), we define $B(i,j,t)$ to be a binary function such that $B(i,j,t)=1$ if the service s_i is a) transmitted at time t, and b) completely processed in the given K time units; and $B(i,j,t)=0$ otherwise. More formally, we have:

Definition 1 *Utility Income*: A utility income $UI(i,j,t)$ produced by service s_i at receiver r_j is defined as:

$$UI(i,j,t) = B(i,j,t) \times \mu(i, T_{Complete}(i,t)) \tag{3}$$

Given limited network and mental resources and a list of services, a sender needs to select and schedule a subset of services to deliver, which calls for a service schedule:

Definition 2 *Service Schedule*: For an instance with given parameters S, R, T, a schedule Ω is defined as:

$$\Omega = (s_i, t), \Omega \subseteq S \times T \tag{4}$$

where the 2-tuple (s_i, t) means the sender starts to send service s_i from time unit t by satisfying the constraints and assumptions mentioned earlier, e.g., a service transmission cannot be launched if another service is currently being transmitted (However, the sender does not have to launch a new transmission even if the wireless medium is idle).

Definition 3 *Driver-centric Service Delivery Problem* (*DSDP*): Given a list of services $\{s_i\}$ to be sent in K time units and a set of receivers $\{r_j\}$, find a schedule Ω such that the *total utility income* (or *TUI*) can be maximized, i.e.:

$$\begin{aligned} TUI(\Omega) = \sum_{t=1}^{K}\sum_{i=1}^{N}\sum_{j\in R_i} UI(i,j,t) = \\ \sum_{t=1}^{K}\sum_{i=1}^{N}\sum_{j\in R_i} B(i,j,t) \times \mu(i, T_{Complete}(i,t)) \end{aligned} \tag{5}$$

In this work, we use the utility to model service priority. One reason for doing so is that we can easily change the priority of a service by using a time-variant utility function as in Eq. (2). Another reason is that considering utility makes more intuitive sense than considering priority. For example, a safety warning message in general has a higher utility value and thus is more likely to be delivered than a non-safety warning message, unless the former is outdated already. Also, based on our model, a message (even if it is important) should not be delivered to its receivers if they are currently busy processing other services. In both cases, the deciding factor is whether delivering a message will contribute to a higher TUI. A higher TUI means that more services are delivered at the right times and can thus benefit the receivers, which should be the ultimate goal anyway. Note that in this sense, a higher TUI is not directly related to the traditional network performance metrics in terms of a high throughput, low delay and high delivery ratio. We believe these traditional metrics do not matter as much when considering human drivers' information processing capability.

Definition 4 *Decision Version of DSDP*: does there exist a service schedule Ω such that the $TUI(\Omega)$ is more than a given positive number q?

Theorem 1: DSDP is NP-complete.

Proof: Please refer to our detailed technical report [30].

It is worth noting that in our previous work [1], it was assumed that *all the services can be scheduled and sent* (since there was no resource limitation, service pre-emption or transmission failure). Accordingly, minimizing total utility loss (For a scheduled service s_i, the utility loss is the difference between its initial utility, i.e., $\mu(i,1) \times |R_i|$, and its induced utility income at its receivers) was equivalent to maximizing the total utility income. In DSDP, however, only some of services can be scheduled/delivered such that a solution with a lower total utility loss does not mean a higher total utility income.

D. Multi-sender Case and Distributed Implementation

In the multi-sender case with Network Model II, more than one sender are allowed to launch a new transmission in the same time unit but a service collision may happen at a receiver r_j if he/she receives two or more services in the same time unit. If two or more services collide at r_j, they will be discarded and r_j will not be affected. For such a multi-sender case, we will extend the centralized algorithms proposed for the one-sender case in the basic scenario to their distributed versions (See Section IV.G).

E. Examples

Fig. 2 shows examples illustrating the service delivery process at receiver r_1. Fig. 2(a) shows that service s_1 is transmitted from time unit $t=a$ and it will produce $C_1 = \mu(1,f)$

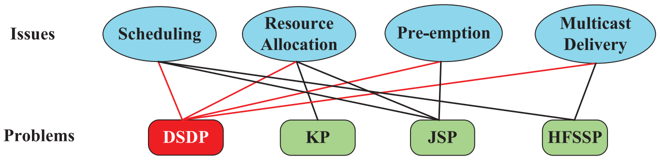


Fig. 3. Different problems and their involved issues.

units of UI since r_1 will be able to process it successfully and the driver processing will be complete at time unit $t = f$. In the example shown in Fig. 2(b), we cannot obtain any UI from either service s_2 or service s_3 because s_2 is pre-empted by s_3 and the driver processing of s_3 cannot be completed within the given time period (i.e., $[1, K]$). In addition, s_4 cannot produce any UI either because the driver is busy processing s_3 and will not pay attention to s_4 when it is displayed. Fig. 2(c) gives an example of the multi-sender case under Network Model II: if both s_5 and s_6, sent by two different senders, arrive at receiver r_1 at time unit $e + 1$, a service collision happens. Consequently, we cannot obtain any UI at r_1 from s_5 and s_6. Note that, since each service has multiple receivers (e.g., r_1, r_2, and r_3 are the receivers of s_6), delivering service s_6 could produce various UIs at different receivers. More specifically, as illustrated in inset in the bottom left corner of Fig. 2, s_6 may generate a UI at r_2 and lead to a service pre-emption at r_3.

III. PROBLEM NOVELTY

In this section, we compare DSDP with many existing problems, including the Job Scheduling Problem (JSP), Knapsack Problem (KP), and our own previous work HFSSP [1] etc. Generally speaking, DSDP differs from and is more complex than all those other problems as it involves all four issues: scheduling, resource allocation, pre-emption and multicast delivery, while others involve only some of the issues, as illustrated in Fig. 3 (see also [30]).

Nevertheless, we have strived to borrow some useful ideas from the solutions to JSP, KP and HFSSP when tackling the new challenges present in DSDP, and compare their performance as to be discussed next.

IV. ALGORITHM DESIGN FOR DSDP

In this section, we first study several heuristic algorithms for selecting the next service to schedule in DSDP assuming the single-sender case in the basic scenario, and then present the distributed versions of a couple of these algorithms suitable for the multi-sender case.

Although DSDP is different from many existing problems as discussed in the last section, we are interested in not only designing several new algorithms for DSDP, but also modifying the algorithms for existing problems (e.g., KP, JSP, and HFSSP) to solve DSDP, in order to gain insights into the effects of transmission loss, display pre-emption and driver status on the designs and performances of various algorithms. Table I shows the list of algorithms to be studied for the single-sender case and the related factors leading to utility loss. Compared to the JSP-based, KP-based and HFSSP-based algorithms, Table I shows that only the new algorithms, called *Pre-emption and Resources aware Algorithm* (*PRA*) and its *Feedback-free variation* (*FPRA*) consider the utility loss of a service due to transmission failures, display pre-emption and driver overload (being busy in processing a service when another service is displayed).

TABLE I
ALGORITHMS AND UTILITY LOSS FACTORS CONSIDERED
(×: NOT CONSIDERED √: CONSIDERED)

Algorithms	Trans. Loss	Display Pre-emption	Driver's Busy Status	Online/ Offline
Random	×	×	×	Off
KP-based	√	×	×	Off
JSP-Based	×	Avoid	×	Off
HFSSP-Based	√	×	√	On
PRA	√	√	√	On
FPRA	√	√	×	Off

Note that all the algorithms below follow the same general framework: starting with $t = 1$, a service si is selected for transmission based on an algorithm-specific strategy. The algorithm then selects the next service when or after the current service transmission completes. This continues until either the time runs out (i.e., K time units) or $Q < K$ services have already been scheduled under Network Model II. Below, we briefly describe the basic ideas of each algorithms, whose detailed description can be found in [30].

A. *Random Algorithm*

This is a baseline solution, which is used mainly for comparison. For a given time unit t without on-going transmission, this algorithm will randomly choose whether to launch a new transmission. If so, it arbitrarily selects an unscheduled service to deliver.

B. *KP-based Algorithm (KPA)*

The central idea of classical algorithms for KP is to check the profit and weight for each item, and choose the one having the best benefit-cost ratio at each time.

C. *JSP-based Algorithm (JSPA)*

An effective heuristic for JSP is to schedule a job with the shortest completion time first. Accordingly, in the JSP-based algorithm (JSPA) for DSDP, we order all unscheduled services s_i in ascending order using $D_{Total}(i)$, and then select the first (i.e., with the smallest $D_{Total}(i)$) that *will not lead to any service pre-emption*. This requirement is imposed to avoid service display pre-emption in JSPA since a pre-empted job must be resumed in JSP but a pre-empted service cannot be re-displayed in DSDP.

D. *HFSSP-based Algorithm (HFSSPA)*

The basic idea of the utility income-dominant algorithm proposed for HFSSP in [1] is to transmit a service that will produce the largest UI. But note that, HFSSP ignores possible utility loss due to pre-emption, or in other words, the possibility that the display of service s_i may pre-empt an existing display.

E. Pre-emption and Resource-aware Algorithm (PRA)

We introduce a new algorithm for DSDP called PRA which selects a service using 1) *its absolute potential UI*; 2) *its relative potential UI*; and 3) *its benefit-cost ratio* (similar to that used in KPA) in succession. The basic idea is that PRA will choose a service which can contribute to TUI and has relatively high relative potential UI (i.e., the produced UI is large compared to its intrinsic utility), and has the best benefit-cost ratio by considering the consumed resources.

F. Feedback-free PRA (FPRA)

In order to simplify the implementation of PRA and also to understand the performance degradation by eliminating the need for feedback information on the service transmission and driver status, we design and evaluate a variation of PRA called Feedback-free PRA (FPRA).

The main challenge in designing FPRA is to estimate the absolute potential UI of sending s_i at time t because the receiver status is no longer known. FPRA overcomes this challenge by using conservative estimates of its potential UI at each of its receivers.

G. Distributed PRA and FPRA (DPRA and DFPRA)

Due to limited space, we mainly focus on the distributed versions of PRA and FPRA (denoted by DPRA and DFPRA) when services are provided by multiple senders (denoted by $p_y, y = 1, 2, ...V$) in the vicinity. We assume Network Model II wherein each sender could deliver up to Q_y services during K time units, but the total service delivery opportunities are still limited, i.e., $Q = Q_y < K$. In addition, each sender only has some static global knowledge, e.g., the total number of senders (i.e., V) in the vicinity, and will and will not obtain feedback information from all the receivers in the case of DPRA and DFPRA, respectively. In particular, senders do not coordinate with each other.

A main challenge in the distributed system with multiple senders is how to avoid service collision at a receiver caused by having multiple senders transmit to the receiver at the same time, while also avoiding waste of various resources by letting go service delivery opportunities. The main idea is that in each time unit t, one of the senders is regarded as a master sender, which still follows PRA or FPRA introduced in the previous section to select a service. In the meantime, other senders (which are regarded as secondary users) will take a probabilistic approach to decide whether they should also send a service.

V. A BASIC SCENARIO AND PERFORMANCE EVALUATION

In this section, we consider a basic scenario with some simplified assumption and evaluate the performances of the proposed algorithms through simulations of a large number of test cases. For a given parameter setting, we randomly generate 1000 test cases. We take the performance of the random algorithm as base line and define the *average performance improvement ratio* (*APIR*) of algorithm A to the random algorithm as:

$$APIR = \frac{(TUI_A - TUI_{RAN})}{TUI_{RAN}} \tag{6}$$

where TUI_A and TUI_{RAN} are the sum of the TUIs of algorithm A and the random algorithm over all test cases, respectively.

TABLE II
DEFAULT VALUES OF EXPERIMENT PARAMETERS

No. of Total Services (N)	30
No. of Total Receivers (M)	30
Avg. No. of Receivers Per Service (w)	10
Total Time Units (K)	50s
Maximum Utility of Service (C)	100
Service Utility Function	Eq. (1)
Threshold for Relative Potential Utility (α)	0.5
Maximum Ratio of D_{Total} to K (r)	0.5(∼25s)
Maximum Ratio of D_{Trans} to D_{Total} (r')	0.2
Serv. Delivery Opportunity in Net. Model II (Q)	10
No. of Senders in the Multi-sender Scenario (V)	3

A. Simulation Setup and Default Parameter Settings)

We evaluate the service delivery algorithms in a two-road intersection area, where one or more senders are assumed to deliver services to their one-hop receivers. For the time being, we do not address the vehicle mobility issue by assuming that all the vehicles involved are currently stopped by red lights. In addition, we assume that one time unit corresponds to one second and set $K = 50$ to be the duration of one time period for scheduling purposes. In the next section, we will consider vehicle mobility and also a larger K.

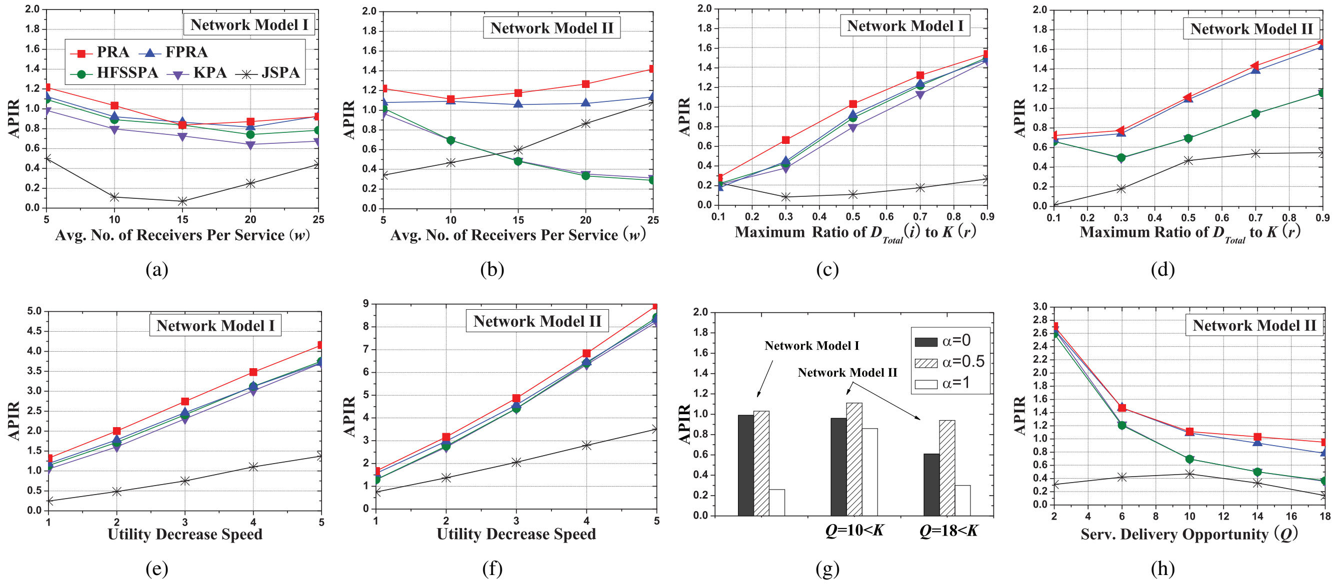
Table II lists the default parameter settings based on empirical data provided in [21-23] (see also [30] for detailed description).

B. Simulation Results

Fig. 4(a) shows the APIR of each algorithm as w increases under Network Model I. As can be seen, PRA performs best among all the algorithms. Moreover, FPRA also yields a satisfactory performance. This seems to indicate that the feedback information from receivers is not as helpful in this case as we might have expected. This is because, even if the sender chooses the best service for transmission next based on the current feedback, it cannot guarantee that the service would be well received, due to possible transmission failure.

Looking at Fig. 4(b) (which, along with the following subfigures, use the same legends for various algorithms), it can be seen that the relative performance of the different algorithms under Network Model II is somewhat different from Network Model I. In particular, the performance improvement (over Random) of PRA is more obvious due to its superior strategy for dealing with limited display and drivers' mental resources. In addition, JSPA appears to perform better than HFSSPA and KPA when $w > 15$ (however it is still worse than PRA and FPRA). This shows that when each service has a large number of receivers, it is more effective to not allow service pre-emption, as in JSPA, because the induced utility loss due to service pre-emption outweighs the obtained utility income.

Figs. 4(c) and (d) show that the APIRs of almost all the algorithms increase as r increases under the two network models. This is because a large r implies that each service consumes more resources on average, and therefore, the random algorithm, which does not adopt any rational strategy



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 4. Testing results of DSDP. (a)-(b) Performance with different w. (c)-(d) Performance with different r. (e)-(f) Performances with different utility decrease speeds. (g) Performance of PRA with different α. (h) Performance with different service delivery opportunity Q.

in service selection and scheduling, performs much worse when compared to other heuristic algorithms for a large r. In addition, similar to the observation from comparing Fig. 4(a) and (b), PRA provides better improvements under Network Model II than under Network Model I.

Next, we examine how each algorithm performs if we assume time-variant utility function as defined in Eq. (2) instead of Eq. (1) by default. We take the approach used in [1] to produce time-variant service utility, with which, the service utility reduces by $a \times \frac{t}{K}$ percent of its initial utility after t time units (where a is the utility decrease speed ranging from [1, 5] in our simulations).

Figs. 4(e) and (f) show that how the APIRs of algorithms change as we vary the utility decrease speed. PRA is still the best solution, and in fact, its APIR goes up to 5 and 9 under Network Models I and II, respectively (from about 2 in the case of a constant utility function). This indicates that, although the TUIs of all algorithms will decrease with the utility decrease speed (not shown), the random algorithm performs a lot worse in this case since a service having a large initial utility may not get a chance to be delivered before its utility significantly decreases, leading to a big utility loss.

In Fig. 4(g), we look at how PRA performs under different threshold values (α) used to qualify candidate services according to their relative potential UIs (see [30] for the formal description of relative potential UI and the use of α). As can be seen, the moderate policy ($\alpha = 0.5$) is the best of the three. In addition, the conservative one ($\alpha = 1$) is relatively the worst, especially if there is a reasonable amount of network resources available when under Network Model I (leftmost part of Fig. 4(g)) or under Network Model II with a large Q (rightmost part of Fig. 4(g)). This is because the conservative policy may give up too many service transmission opportunities while the other two can take advantage of the available resources.

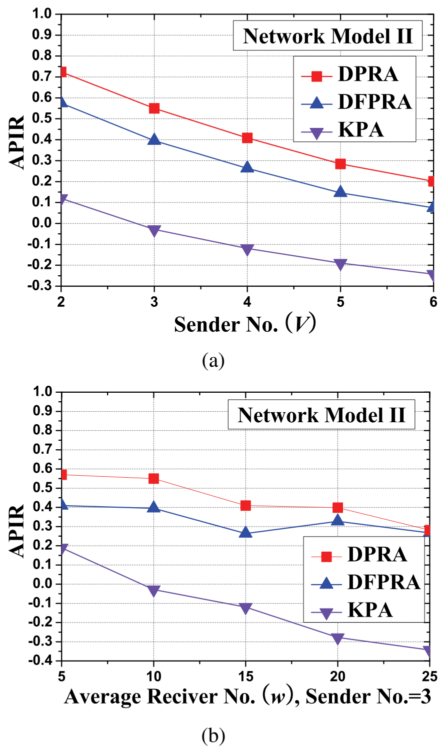
Fig. 4(h) further shows the APIR of each algorithm as Q increases under Network Model II. We can see that the APIRs of most of the algorithms decrease with Q. The reason being, that the random algorithm benefits more from a larger amount of network resources (i.e., a large Q).

Next, we show the performances of DPRA and DFPRA under the multi-sender scenario and compare them with KPA and the random algorithm, which can also be implemented in a distributed fashion. In these sets of simulations, we only consider Network Model II with $Q = 18$, and randomly assign the transmission opportunities to the multiple senders.

Fig. 5(a) shows the APIRs of DPRA, DFPRA and KPA as the number of senders (V) increases from 2 to 6. For DPRA, a $40 - 70\%$ improvement over the random algorithm can still be achieved under different settings, although its APIR decreases with V. While still significant, the improvement is less than the case of a single sender. This is mainly due to the following facts: 1) each sender can only perform local optimization when selecting a few services it will send; 2) the secondary senders take a probabilistic approach to decide whether to transmit a service, which may lead to either wasted transmission opportunities or service collisions.

Fig. 5(a) also shows that DFPRA performs slightly worse than DPRA. This can be explained by the fact that DFPRA lacks more information about which services have been transmitted to a given receiver. Finally, it can be seen that KPA performs even worse than the random algorithm (i.e., its $APIR < 0$) in this case. This is because, according to KPA, each sender launches a new transmission so long as it finishes a previous transmission, and then there will be too many service collisions and too much service pre-emption.

Fig. 5(b) shows the APIR of each algorithm as w increases in the multi-sender case. Again, DPRA and DFPRA are still better than the random algorithm although their APIRs decrease with w. When $w = 25$, DPRA still has 30% improvement over the random algorithm while KPA performs much worse than the random algorithm.



(a)

(b)

Fig. 5. Testing results under the basic multi-sender scenario. (a) Performance with different sender number V. (b) Performance with different w.

VI. A More Realistic Case Study

A. More Practical Assumptions and Issues

In this case study, we extend our results by addressing several more practical issues:1) Vehicle mobility; 2) Dynamic service arrivals; 3) Transmission failure model; 4) Differentiated service popularity and 5) Heterogeneous driver processing capability. Accordingly, we also modify algorithms KPA, PRA and FPRA and denote the modified versions by MKPA, MPRA and MFPRA, respectively. (Please refer to [30] for more discussions on these issues and the modifications).

Table III summarizes the new values of the major parameters used in this case study, and the parameters that do not appear in Table III will take the same default values as in Table II. Note that, the maximum D_{Total} is still 25s (although now we do not set its value according to K as in Section V.A).

B. Testing results

Fig. 6(b) shows the performances of MKPA, MPRA and MFPRA in the default setting. Generally speaking, both MPRA and MFPRA have a better performance than either the random algorithm or MKPA in this realistic scenario. We also observe that MKPA can perform worse than the random algorithm just as before but for a different reason than that explained in the multi-sender case of the basic scenario (see Fig. 5 and related discussion). Here the main cause is that now a service is allowed to be delivered more than once. Hence, since MKPA does not consider drivers' real-time status (see Table I) but only consider the number of receivers that are currently in sender's coverage, it will repeatedly transmit a same service. Most of such service transmissions will be useless and in fact waste transmission opportunities because they prevent the receivers from receiving other service deliveries.

TABLE III
Parameter Values in More Realistic and Mobile Scenario

Total Time Units	250
No. of Senders	5
No. of Total Involved Vehicles	140
Distances between Senders	350-500m
Service Popularity Model	Zipf
Transmission Range of Senders	150m

Fig. 6(c) shows how the transmission range of the senders affects the performance of each algorithm. It can be seen that each algorithm has an almost constant APIR over the random algorithm. The reason is that for a given transmission range, each algorithm has equal opportunities to deliver services to vehicles, i.e., the time frames when vehicles are traveling under senders' coverage. Therefore, TUI largely depends on service selection and scheduling algorithms.

Last but not least, similar to Fig. 4(d), Fig. 6(d) shows that the APIR of both MPRA and MFPRA increase as the maximum D_{total} increases because each service will consume more resources on average with a larger maximum D_{total}, and both the random algorithm and MKPA perform poorly in such a resource-limited situation.

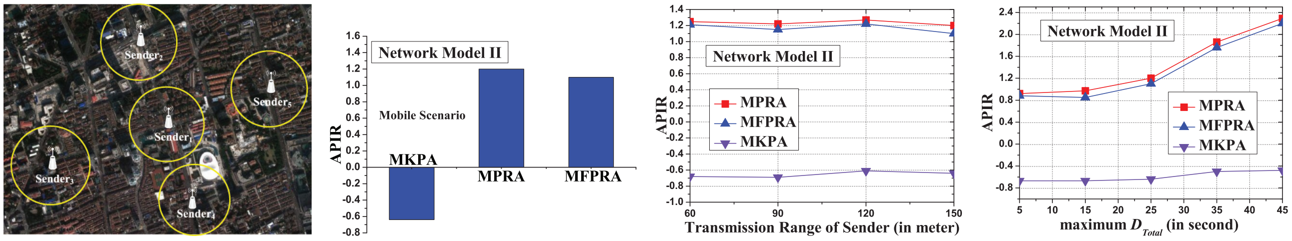
VII. Related Work

For existing Human Factors-related studies, the focus has primarily been on the human drivers [2, 3, 5-9]. On the other hand, several studies in the cyber field have focused on VANET. Typical research issues addressed include data delivery and access [10]-[13][28], reliable broadcasting [17][18], data aggregation[29], test beds [14], safety and security [15][20] and safety alert messaging [16][27], etc. Moreover, compared to our previous work [1], our current work considers a number of factors which were missing from [1]. Please refer to our technical report for more elaborations on the comparison between this work and existing works [30].

Overall, this work is, unlike any existing work on VANET, transportation systems or HFs in transportation systems, the first that takes a cross-disciplinary approach to the design of the Driver-in-the-Loop VCPS.

VIII. Conclusion

Vehicular Cyber-Physical Systems (*VCPS*) are expected to improve road safety, driving experience and on-road infotainments. Different from existing research, this work has adopted, for the first time, a cross-disciplinary and holistic approach to study the service delivery problem in the Driver-in-the-Loop VCPS by considering both the cyber and human aspects. More specifically, the study addressed a new *Driver-centric Service Delivery Problem* (*DSDP*) considering limited wireless communications network resources, on-board display resources and drivers' mental (cognitive/processing) resources, as well as practical issues such as possible transmission failure, service display pre-emption and distributed implementation in the multi-sender scenario. We have shown that DSDP is NP-Complete and that it is different from all existing problems. We have designed efficient heuristics for service selection and



(a) (b) (c) (d)

Fig. 6. (a) Our case study based on real vehicle traces collected in Shanghai. Testing results in a more realistic and mobile scenario. (b) Performance of three algorithms under default setting. (c) Performance with different transmission ranges of senders. (d) Performance with different upper bound of D_{total}.

scheduling in order to maximize the system-wide benefit in terms of the *total utility income* (*TUI*) to the drivers. A detailed performance evaluation and comparison study of upper-layer protocols and algorithms based on large-scale simulations have also been presented. Furthermore, we have performed a case study utilizing real traces, and more practical assumptions related to vehicle mobility, dynamic service arrivals, differentiated service popularity, service transmission failure model and heterogonous driver processing capabilities. Our studies and results have shown that it is not sufficient to borrow ideas from existing solutions to similar problems such as the Job Scheduling Problem or the Knapsack Problem, and that our proposed algorithms can achieve a much better performance than a random algorithm.

REFERENCES

[1] X. Li, C. Qiao, et al., "Human factors-aware service scheduling in vehicular cyber-physical systems," in *Proc. IEEE INFOCOM*, 2011.

[2] R. Gray, "Looming auditory collision warnings for driving," *Human Factors*, vol. 53, no. 1, 2011.

[3] SAE, "ITS in-vehicle message priority," Safety and Human Factors Committee, J2395, 2002.

[4] H. Sohn, J. D. Lee, et al., "A dynamic programming algorithm for scheduling in-vehicle messages," *IEEE Trans. Intell. Transp. Syst.*, 2008.

[5] S. Classen, et al., "Personality as a predictor of driving performance: An exploratory study," *Transp. Res. Part F: Traffic Psychology Behavior*, 2011.

[6] J. Whitmire, "The effect of in-vehicle warning systems on speed compliance in work zones," *Trans. Res. Part F: Traffic Psychology Behavior*, 2011.

[7] C. Gnardellis, et al., "An investigation of the effect of sleepiness, drowsy driving, and lifestyle on vehicle crashes," *Transportation Res. Part F: Traffic Psychology Behavior*, vol. 11, no. 4, 2008.

[8] W. Verwey, "On-line driver workload estimation: Effects of road situation and age on secondary task measures," *Ergonomics*, 2000.

[9] W. Brodsky, "The effects of music tempo on simulated driving performance and vehicular control," *Trans. Res. Part F: Traffic Psychology Behavior*, 2001.

[10] J. Eriksson, H. Balakrishnan, and S.Madden, "Cabernet: Vehicular content delivery using WiFi," in *Proc. ACM/IEEE MOBICOM*, 2008.

[11] Y. Zhang and G. Cao, "V-PADA: Vehicle-platoon-aware data access in VANETs," *IEEE Trans. Veh. Technol.*, 2011.

[12] H. Zhu, S. Chang, M. Li, S. Naik, and S. Shen, "Exploiting temporal dependency for opportunistic forwarding in urban vehicular network," in *Proc. IEEE INFOCOM*, 2011.

[13] J. Yoo, B. S. C. Choi, and M. Gerla, "An opportunistic relay protocol for vehicular road-side access with fading channels," in *Proc. IEEE ICNP*, 2010.

[14] H. Soroush, N. Banerjee, A. Balasubramanian, M. Corner, B. N. Levine, and B. Lynn, "DOME: A diverse outdoor mobile testbed," *ACM HotPlanet*, 2009.

[15] F. Kargl, P. Papadimitratos, and J.-P. Hubaux, "Secure vehicular communication systems: Implementation, performance, and research challenges," *IEEE Commun. Mag.*, 2008.

[16] R. Mangharam, R. Rajkumar, M. Hamilton, P. Mudalige, and F. Bai, "Bounded-latency alerts in vehicular networks," in *Proc. IEEE INFOCOM*, 2007.

[17] F. Farnoud and S. Valaee, "Reliable broadcast of safety messages in vehicular ad hoc networks," in *Proc. IEEE INFOCOM*, 2009.

[18] R. K. Lam and P. R. Kumar, "Dynamic channel partition and reservation for structured channel access in vehicular networks," in *Proc. IEEE VANETs*, 2010.

[19] [Online]. Available: http://www.cse.buffalo.edu/CTS/npc-proof-DSDP.pdf

[20] J. Yang, H. Liu, et al., "Detecting driver phone use leveraging car speakers," in *Proc. ACM/IEEE MOBICOM*, 2011.

[21] National Highway Traffic Safety Administration, "Vehicle safety communications project task 3 final report," U. S. Dept. of Transportation (USDOT), 2005.

[22] C. Wu and Y. Liu, "Queuing network modeling of driver workload and performance," *IEEE Trans. Intell. Transp. Syst.*, 2007.

[23] M. Green, "How long does it take to stop? Methodological analysis of driver perception-brake times," *Transp. Human Factors*, 2000.

[24] X. Li, et al., "A new paradigm for urban surveillance with vehicular sensor networks," *Comput. Commun.*, 2011.

[25] U. Feige, "A threshold of ln n for approximating set cover," *J. ACM*, vol. 45, no. 5, pp. 634–652, 1998.

[26] H. Sohn, J. D. Lee, et al., "A dynamic programming algorithm for scheduling in-vehicle messages," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 2, pp. 226–235, 2008.

[27] T. Tielert, D. Jiang, H. Hartenstein, et al., "Design methodology and evaluation of rate adaptation based congestion control for vehicle safety communications," in *IEEE Veh. Netw. Conf.*, 2011.

[28] C. Sommer and O. K. Tonguz, "Traffic information systems: Efficient message dissemination via adaptive beaconing," *IEEE Commun. Mag.*, 2011.

[29] B. Scheuermann, M. Mauve, et al., "A fundamental scalability criterion for data aggregation in VANETs," in *Proc. ACM/IEEE MOBICOM*, 2009.

[30] [Online]. Available: http://www.cse.buffalo.edu/CTS/DSDP-TR.pdf

[31] H. A. Omar, W. Zhuang, and L. Li, "VeMAC: A TDMA-based MAC protocol for reliable broadcast in VANETs," *IEEE Trans. Mobile Comput.*, June 2012.

[32] R. S. E. Chaput and A.-L. Beylot, "Comparison of CSMA and TDMA for a heartbeat VANET application," in *Proc. IEEE ICC*, 2010.

Xu Li is a senior research scientist at the State University of New York at Buffalo. He graduated from Shanghai Jiao Tong University, China, where he received his doctorate in 2009. His research interests include vehicular ad hoc networks (VANET), wireless sensor networks (WSN), and vehicular cyber physical systems (CPS), etc. He has published technical papers in mainstream international journals and conferences, such as INFOCOM, IEEE TPDS, IEEE JSAC, IEEE TVT, Elsevier Ad Hoc Networks, Elsevier Computer Communications, ICCCN, GLOBECOM, WCNC, VTC, etc. He is or has been on the program committee of a number of conferences, including ICC 2012, GLOBECOM 2012, ICC2011, ISCC 2011, ISCI 2011, GLOBECOM2010, IWCMC2010, VTC 2010-Fall, ISCC2010, ICUMT 2010, etc.

Chunming Qiao directs the Lab for Advanced Network Design, Analysis, and Research (LANDER), with current foci on optical networking and survivability/availability issues in cloud computing, active-safety and infotainment in transportation systems, and low-cost and low-power sensors and mobile (robotic) sensor networks. He has published about 100 and 160 papers in leading technical journals and conference proceedings, respectively, with an h-index of about 50 (according to Google Scholar). He has pioneered research on optical Internet, and in particular, optical burst switching (OBS). One of his papers on OBS has been cited more than 2000 times. In addition, his work on integrated cellular and ad hoc relaying systems (iCAR), started in 1999, is recognized as the harbinger for today's push towards the convergence between heterogeneous wireless technologies, and has been featured in *Business Week* and *Wireless Europe*, as well as the web sites of New Scientists and CBC. His research has been funded by 10 NSF grants including two ITR awards, and by major IT and telecommunications companies including Alcatel Research, Fujitsu Labs, Cisco, Google, NEC labs, Nokia, Nortel Networks, Sprint Advanced Technology Lab, and Telcordia, as well as the Industrial Technology Research Institute (Taiwan).

Dr. Qiao has given a dozens of keynotes and numerous invited talks on the above research topics. He has chaired or co-chaired a dozen of international conferences and workshops. He has been an editor of the IEEE TRANSACTIONS ON NETWORKING and the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, and a guest-editor for several issues of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS (JSAC). He was chair of the IEEE Technical Committee on High Speed Networks (HSN) and the IEEE Subcommittee on Integrated Fiber and Wireless Technologies (FiWi,) which he founded. He was elected an IEEE Fellow for his contributions to optical and wireless network architectures and protocols.

Aditya Wagh is a doctoral student at the State University of New York at Buffalo. He received his bachelors and master's degrees from the University of Mumbai in 2006 and 2008, respectively. His research is mainly focused on wireless and ad hoc networks. He is currently exploring the application of human factors considerations to the design of communication protocol and applications in vehicular ad hoc systems.

Raghuram Sudhaakar graduated with the Ph.D. degree in computer science from the State University of New York at Buffalo. He is currently a researcher at Cisco Systems. His research is mainly focused on vehicular and sensor networks.

Sateesh Addepalli is currently director of the Advanced Architecture and Research Group of Cisco Systems. Apart from a range of research and innovation activities at Cisco, he has also been responsible for launching and driving research and innovation initiatives related to mobile and wireless, sensor networks, and smart connected vehicle and transportation, respectively.

Changxu Wu (S'04-M'07) received the M.S. and the Ph.D. degrees in industrial and operational engineering from the University of Michigan-Ann Arbor, in 2004 and 2007, respectively. He also received the M.S. degree in engineering psychology and human-computer interaction from the Chinese Academy of Sciences in 2002 and the B.S. degree in psychology, with the focus on engineering and mathematical psychology, from Zhejiang University, P.R. China in 1999. He is an Assistant Professor of the Department of Industrial and Systems Engineering at the State University of New York (SUNY), Buffalo, starting from August 2007. Dr. Wu directs the Cognitive System Lab at SUNY and he is interested in integrating cognitive science and engineering system design, especially modeling human cognition systems, with its applications in system design, improving transportation safety, promoting human performance in human-computer interaction, and inventing innovative sustainable and smart energy systems with humans in the loop. He is an Associate Editor for the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATIONS SYSTEMS and *Behaviour and Information Technology*. He has published more than 30 journal papers in the field, including *Psychological Review*, the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS (PART A) (IEEE-SMCA), the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS (IEEE-ITS), the *ACM Transactions on Computer-Human Interaction* (TOCHI), the *International Journal of Human-Computer Studies*, *Acta Psychologica Sinica*, and *Ergonomics in Design*, as well as several other journals. He received the Outstanding Student Instructor Award from the American Society of Engineering Education (ASEE) at the University of Michigan in 2006. Dr. Wu is a member of the Human Factors and Ergonomics Society (HFES), Society of Automobile Engineers (SAE), Cognitive Science Society, and American Society of Engineering Education (ASEE).

Adel W. Sadek received the Ph.D. degree in civil engineering from the University of Virginia, Charlottesville. He is currently an Associate Professor of civil engineering with the University at Buffalo (UB), The State University of New York, Buffalo, and the Acting Chair of UB2020 Strategic Strength in Extreme Events. Prior to that, he was an Associate Professor with the School of Engineering and Department of Computer Science, University of Vermont, Burlington. He is a co-author of two transportation textbooks and numerous articles in peer-reviewed journals. Dr. Sadek is a member of the Transportation Research Board committee on Artificial Intelligence and Advanced Computing Applications and the committee on Surface Transportation Weather. He is also a member of the Advanced Technologies committee of the Transportation and Development Institute of the American Society of Civil Engineers. He received the 1998 Milton Pikarsky Award for the Best Dissertation in the field of transportation science and technology, a National Science Foundation CAREER Award, and an IBM Smarter Planet Faculty Innovation Award.