Software-Defined Device-to-Device (D2D) Communications in Virtual Wireless Networks With Imperfect Network State Information (NSI)

Yegui Cai, F. Richard Yu, *Senior Member, IEEE*, Chengchao Liang, Bo Sun, *Member, IEEE*, and Qiao Yan

***Abstract*—Software-defined networking (SDN) and network function virtualization (NFV) are a promising system architecture and control mechanism for future networks. Although some works have been done on wireless SDN and NFV, recent advancements in device-to-device (D2D) communications are largely ignored in this novel framework. In this paper, we study the integration of D2D communication in the framework of SDN and NFV. An inherent challenge in supporting software-defined D2D is the imperfectness of network state information, including channel state information (CSI) and queuing state information, in virtual wireless (QSI) networks. To address this challenge, we formulate the resource sharing problem in this framework as a discrete stochastic optimization problem and develop discrete stochastic approximation algorithms to solve this problem. Such algorithms can reduce the computational complexity compared with exhaustive search while achieving satisfactory performance. Both the static wireless channel and time-varying channels are considered. Extensive simulations show that users can benefit from both wireless network virtualization and software-defined D2D communications, and our proposed scheme can achieve considerable performance gains in both system throughput and user utility under practical network settings.**

***Index Terms*—Device-to-device (D2D) communications, network function virtualization (NFV), software-defined networking (SDN).**

I. Introduction

RECENTLY, *software-defined networking* (SDN) has been widely considered as a novel approach to promote innovations in communication networks [1]. In wireless networks, by identifying and abstracting the wireless primitives and functions, wireless SDN can provide mechanisms for developing innovative wireless networking methodologies [2]. There have been some works on applying the SDN concepts in wireless networks, ranging from radio access networks (RANs) to core mobile networks and from control plane to data plane. SoftRAN [3] and CellSDN [4] are software-defined control planes for RANs by abstracting all the access points (APs) in a local geographical region as a virtual big-AP. OpenRadio [5] provides a solution for programmable wireless data plane. For core networks (CNs), SoftCell [6] directs traffic to traverse a sequence of middle boxes while optimizing the performance according to network conditions.

Another promising technology is *network function virtualization* (NFV) [1], [7], [8]. In wireless networks with virtualization, network infrastructure can be decoupled from the services that it provides [8]. Consequently, multiple wireless virtual networks operated by different service providers (SPs) can dynamically share the physical substrate wireless networks operated by mobile network operators (MNOs). Since wireless network virtualization enables the sharing of infrastructure and radio spectrum resources, the capital expenses (CapEx) and operation expenses (OpEx) of wireless (radio) access networks (RANs), as well as CNs, can be significantly reduced. Moreover, mobile virtual network operators (MVNOs) who may provide some specific telecommunication services can help MNOs attract more users, whereas MNOs can produce more revenue by leasing the isolated virtualized networks to them [9].

SDN and NFV can complement each other in wireless networks [7], [10]. On one hand, for wireless network virtualization, SDN's decoupling principles can enhance performance, simplify compatibility with legacy deployments, and enable operation and maintenance procedures. On the other hand, for SDN controllers, virtual wireless networks provide the infrastructure to run the software on the top.

Although some excellent works have been done on wireless SDN and NFV, recent advances on *device-to-device* (D2D) communications are largely ignored in this novel framework of SDN and NFV. Recently, the benefits of D2D have been well recognized [11]–[13]. In terms of improving the systematic performance of mobile wireless networks, D2D communications can provide the following gains [14]. First, *reuse gain* allows radio resources to be simultaneously used by both cellular and D2D links. Second, *proximity gain* empowers high bit rates, low latency, and low power consumption due to the locations of the two nodes in the D2D link. Third, *hop gain* enables the source and destination nodes to use a single link rather than both uplink and downlink, which results in two-hop communications.

Manuscript received July 16, 2014; revised March 2, 2015 and July 20, 2015; accepted September 8, 2015. Date of publication September 29, 2015; date of current version September 15, 2016. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada; by Huawei Technologies Canada Co., Ltd.; and by the the National Natural Science Foundation of China under Grant 61402294, Grant 61170283, and Grant 61572330, and the Major Fundamental Research Project in the Science and Technology Plan of Shenzhen under Grant JCYJ20140828163633977. The review of this paper was coordinated by Dr. T. Taleb. *(Corresponding author: Qiao Yan.)*

Y. Cai, F. R. Yu, and C. Liang are with the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON K1S 5B6, Canada (e-mail: richard.yu@carleton.ca; ycai@sce.carleton.ca; chengchaoliang@sce.carleton.ca).

B. Sun is with the Department of Computer Science, Lamar University, Beaumont, TX 77710 USA (e-mail: bsun@lamar.edu).

Q. Yan is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: yanq@szu.edu.cn).

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/TVT.2015.2483558

Nevertheless, to support D2D, a significant amount of modification is needed in both the control plane and data plane of RANs and CNs. Even if it is possible to support D2D by modifying many network entities and protocols in current systems, it will be difficult to provide fast innovation and deployment in the future for new infrastructureless networking applications [15].

The integration of D2D communications in cellular networks can be greatly facilitated by SDN and NFV. As one of the systematic solutions, applying SDN techniques in virtual wireless networks can provide a versatile framework for the integration of new communications schemes in legacy cellular systems.

In this paper, we study the integration of D2D communications in the framework of SDN and NFV. Specifically, we consider how to exploit the radio resource pool among multiple infrastructure providers (InPs) and D2D communications in virtual wireless networks to maximize the network-wide welfare. The contributions of this paper are as follows.

- An inherent challenge in software-defined D2D communications in virtual wireless networks lies in the fact that the SDN controllers need to have a global view of the network states to optimize the network's performance. Network state information (NSI) consists of channel state information (CSI) and queuing state information (QSI). Due to packet delay and loss, NSI is usually imperfect [16]–[18]. Imperfect NSI has a significant impact on the performance of not only D2D networks but wireless networks in general as well. Indeed, the capacity of channels with incomplete NSI is largely unknown in wireless networks [19]. To the best of our knowledge, this work is the first to address the imperfectness of NSI for software-defined D2D communications in virtual wireless networks.
- User utility is modeled as a function of the resource sharing decision. For a particular unit of radio resource, different resource sharing decisions result in different levels of satisfaction due to the channel dynamics, the randomness of the traffic, and various prices via either D2D or a particular InP. We further formulate the resource sharing problem with imperfect NSI as a discrete stochastic optimization problem, in which the sum of long-term user utility is maximized.
- We develop discrete stochastic approximation (DSA) algorithms to solve the discrete stochastic optimization problem. We first develop aggressive DSA for static channels and then introduce adaptive step size DSA for time-varying channels.
- Extensive simulations show that wireless mobile users can benefit from both wireless network virtualization and software-defined D2D communications, and our proposed scheme can achieve considerable performance gain under practical network settings.

The rest of this paper is organized as follows. Section II shows the system model of the software-defined D2D communications in virtual wireless networks. Sections III and IV discuss the resource sharing problem under perfect and imperfect NSI, respectively. Section V discusses the simulation results. Finally, we conclude this study with future work in Section VI.

II. System Description

Here, we present a software-defined framework with virtual wireless networks, followed by the description of the SDN approach to support D2D communications. Then, we describe the resource sharing issues in this framework.

A. *Software-Defined Framework With Virtual Wireless Networks*

The management of the virtual networks is highly complicated. One solution to handle these complications is to decouple the control and data planes and to have the control logics located inside a central controller via the software-defined networking paradigm. Fig. 1 shows a possible system framework for this purpose. The physical networks owned by multiple InPs are first mapped into virtual infrastructures and elements. Then, the virtual entities are aggregated and sliced into different virtual networks by the virtual resource manager or the hypervisor [8]. With the architecture shown in Fig. 1, we can deploy flexible network controllers under arbitrary combinations of wireless technologies such as second generation/third generation/fourth generation/WiFi. The network controllers are on the top of the system. They define the supported communication modes. Since they are all software independent of the underlying hardware, the introduction of new communication schemes or policies is very flexible.

The network controller concept is similar to the network operator concept in [20]. However, the single network operator in [20] is only responsible for spectrum access control, and its main role is to provide an abstract underlying channel conditions for multiple SPs. In Fig. 1, network controllers can have more functionalities, including topology control of the whole network. In terms of supporting the integration of the infrastructureless networking schemes into legacy cellular networks, we can have software-defined D2D communications. Moreover, in the future, such a framework is flexible to support more communication schemes, for example, multihop wireless communications.

The controllers manage the network based on the NSI provided by the lower tier in Fig. 1. One of the challenges in such systems is that it requires NSI to make resource sharing decisions; however, the observation of the network state is inaccurate in general due to the limitation in measurement and the dynamics of the whole system. Take CSI as an example. The channel estimation and measurement can only represent the actual channel state within limited granularity even in a point-to-point wireless communication system. The imperfectness is even worse in the virtual wireless networks since the NSI is transmitted to the central controller via rate-limited backhaul networks, which introduce transmission delays.

B. *SDN Approach to Support D2D Communications*

Traditionally, in a simplified cellular network with D2D communications enabled, a gateway is responsible for detecting the user traffic so that the potential D2D users can be paired if it is feasible. To support D2D communications, it requires

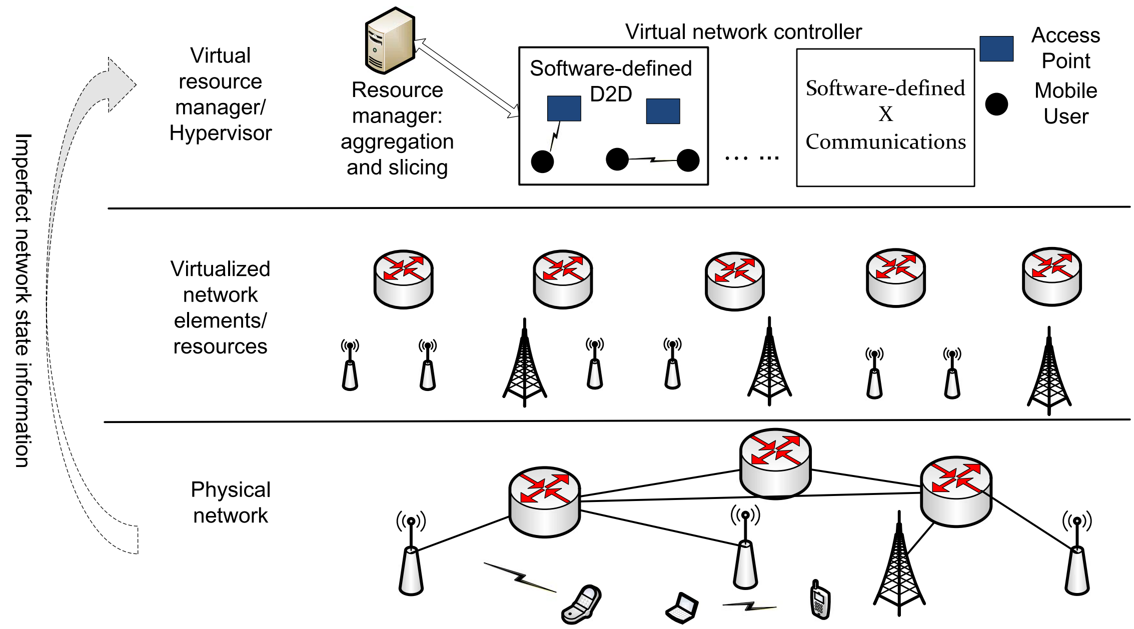


Fig. 1. Software-defined framework with virtual wireless networks.

introducing new functions (e.g., traffic earmarking for gateways and new radio resource management schemes for APs) and new signaling protocols between network entities under the legacy wireless mobile network framework. An example of the session setup and management approaches in a Long-Term Evolution Advanced context can be found in [15]. In the paper, the solution to address the limitation in terms of upgrading many elements is to have software-defined network architecture in a virtual environment, such as that shown in Fig. 1.

In particular, for D2D communications, the procedure of peer discovery and radio resource management can be done jointly by the network controller. In the peer discovery phase, the network controller accesses the routing table (if an open-flow type of implementation is adopted, it is the flow table) and decides if the session can be a potential D2D communication pair. If the source and the destination nodes are recognized as a potential D2D pair, the network controller further decides on whether to perform D2D transmission or to utilize an AP to relay the information. Moreover, since it is a virtual wireless network, it will also consider which AP is responsible for relaying the traffic if the AP relay mode is selected.

C. Resource Sharing in Virtual Wireless Networks With D2D

With wireless virtualization, InP and MVNO have become two important roles in the system, and the interactions between InPs and MVNOs are critical for the system performance. Specifically, InPs own the physical network infrastructure resources and physical radio resources. MVNOs lease the network resources from InPs, create virtual resources, and operate the virtual resources [2], [8]. Therefore, for ease of presentation, we only consider the interactions and optimizations between InPs and MVNOs in this paper. The proposed algorithms can be extended to consider other roles, e.g., SPs, in the virtual wireless network.

Without loss of generality, we assume that there is a single MVNO in the virtual network and that the MVNO manages the radio spectrum resource pool. The key question in such setting is how to make the best use of the precious radio resources. Fig. 2 shows an instance of the resource sharing problem. In the case study, there is a unit of radio resource, which can be used either in D2D modes or in relay mode (via different InP's infrastructure). Here, we have the degrees of freedom from the networks (InP i or InP j) and the user pairs (Pair A or Pair B) in utilizing the unit of radio resource. In particular, Fig. 2(b) and (c) shows the diversity from network selection, and Fig. 2(a) and (d) shows the diversity from user pair selection.

Assume that the InPs in the virtual wireless network are $\mathcal{I} = \{1, \ldots, i, \ldots, I\}$ with different prices $\mathbf{C} = [c_1, \ldots, c_i, \ldots, c_I]$ for each of them. That means to utilize InP i, the cost is c_i dollars.[1] If the radio resource is to be used in a D2D way, the cost is denoted as c_0. Note that the radio resource is in the control of MVNO; hence, $\mathbf{C}$ is independent of radio resource management mechanisms. For an InP i, it has a set of APs, which is denoted as $\mathcal{A}_i$. Then, the set of all APs is $\mathcal{A} = \bigcup_{i \in \mathcal{I}} \mathcal{A}_i$.

Since there usually are more resources allocated in downlink than in uplink, if the traffic turns out to be symmetrical, the uplink is the bottleneck from the point of view of end-to-end communications. Thus, this work focuses on the uplink when the relay mode is selected. Denote the set of source nodes as $\mathcal{S}$. The total number of wireless links concerned is $|\mathcal{S}| \times |\mathcal{A}| + |\mathcal{S}|$, in which there are $|\mathcal{S}| \times |\mathcal{A}|$ links in uplink and $|\mathcal{S}|$ links in D2D. Assume the *radio resource pool* is $\mathcal{N}$ with $|\mathcal{N}|$ units of radio resource at MVNO. Hence, the channel of the $|\mathcal{N}|$ units of radio resource at decision epoch t $\mathbf{H}^t$ is a vector of $(|\mathcal{S}| \times |\mathcal{A}| + |\mathcal{S}|) \times |\mathcal{N}|$ dimensions.

[1] We can further suppose that, even in the same InP, the price to access different APs is different according to geographic locations and time.

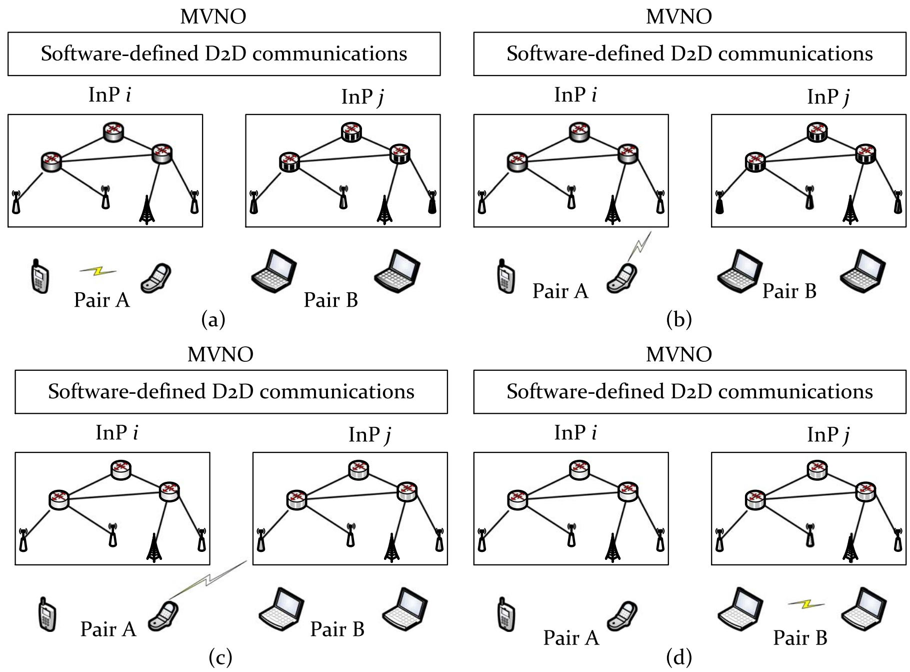


Fig. 2. Case studies of resource sharing in virtual wireless networks with D2D. (a) Radio resource is used in D2D communications for Pair A. (b) Pair A's uplink to InP i's network. (c) Pair A's uplink to InP j's network. (d) D2D for Pair B.

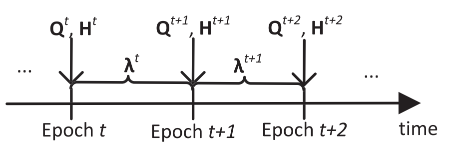


Fig. 3. Decision epochs, random traffic arrivals, and queuing dynamics in a time sequence schema.

III. Resource Sharing in Virtual Wireless Networks With Perfect Network State Information

The resource sharing problem is how to allocate the resources to the flows in virtual wireless networks. Provided perfect NSI related to the $|\mathcal{N}|$ units of virtual radio resources, the optimization problem is to search an optimal decision in a space of $\binom{|\mathcal{S}|\times|\mathcal{A}|+|\mathcal{S}|}{|\mathcal{N}|}$ options. Denote the set of all possible combinations as Ω and the decision variable at each decision epoch t $x^t \in \Omega$. When the traffic is non-full buffer, the queue dynamics need to be considered in resource sharing. Without loss of generality, we assume that there is one flow between each wireless mobile user pair. The queue lengths at the destination nodes are not relevant to the resource sharing problem since the data packets are delivered to the upper layer whenever they arrive at the destination nodes. Hence, the queue state information at epoch t is a $|\mathcal{S}| \times 1$ vector denoted as $\mathbf{Q}^t$. Denote the arrivals between two decision epochs t and $t+1$ as $\boldsymbol{\lambda}^t$, which is also a $|\mathcal{S}| \times 1$ vector. For a flow f originating from node s, the arrival between two epochs is denoted as λ_s^t. Fig. 3 shows the relationship of the decision epochs and the random variables.

Denote the length of a queue at time t in node $s \in \mathcal{S}$ as Q_s^t and the channel gain via radio resource $n \in \mathcal{N}$ from node s to AP $a \in \mathcal{A}$ or to destination node d as $h_{s,a}^{t,n}$ or $h_{s,d}^{t,n}$. The queueing dynamics of the source node is described as

$$Q_s^{t+1} = Q_s^t - \sum_{a\in\mathcal{A}}\sum_{n\in\mathcal{N}} \tilde{\mu}_{s,a}^{t,n}\left(x^t, Q_s^t, h_{s,a}^{t,n}\right) - \sum_{n\in\mathcal{N}} \tilde{\mu}_{s,d}^{t,n}\left(x^t, Q_s^t, h_{s,d}^{t,n}\right) + \lambda_s^t \quad (1)$$

in which $\tilde{\mu}_{s,a}^{t,n}(\cdot)$ is the actual service rate of the flow from source node s to AP $a \in \mathcal{A}$ and $\tilde{\mu}_{s,d}^{t,n}(\cdot)$ from s to d. λ_s is the traffic arrival rate at source node s.[2] $\tilde{\mu}_{s,a}^{t,n}(\cdot)$ is computed as follows:

$$\tilde{\mu}_{s,a}^{t,n}\left(x^t, Q_s^t, h_{s,a}^{t,n}\right) = \begin{cases} \min\left\{\mu_{s,a}^{t,n}\left(h_{s,a}^{t,n}\right), Q_s^t\right\}, & \text{if } x = s \to a \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where $\mu_{s,a}^{t,n}(h_{s,a}^{t,n})$ is the link capacity decided by the channel realization $h_{s,a}^{t,n}$, and $s \to a$ means the selected transmission mode is from node s to AP a. Likewise, the actual service rate via D2D is given as

$$\tilde{\mu}_{s,d}^{t,n}\left(x^t, Q_s^t, h_{s,d}^{t,n}\right) = \begin{cases} \min\left\{\mu_{s,d}^{t,n}\left(h_{s,d}^{t,n}\right), Q_s^t\right\}, & \text{if } x = s \to d \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

At each decision epoch, the software-defined D2D controller obtains a noisy observation of the NSI (merely CSI is needed for the full-buffer traffic case) and then decides whether the flows are transmitted via direct wireless links between the

[2] Admission control can be considered by controlling λ_s^t.

source–destination pairs or via the AP. The length of the period between two consecutive decision epochs is a tunable system parameter. In principle, the finer the granularity of the period, the better performance can be achieved with the higher overhead in control signaling.

A. Utility Function

The wireless users' behavior is heavily influenced by the pricing mechanism of MVNOs and InPs [21]. Thus, we consider the prices offered by the virtual wireless network in the utility function. The immediate user utility is defined as a function of the channel and queue states and the decision variable, i.e., $f(\mathbf{Q}^t, \mathbf{H}^t, x^t)$. Denote the flows in the network as a set $\mathcal{F}$. If a unit of radio resource is to be used by a flow $f \in \mathcal{F}$ via a specific communication mode (either via an InP or via D2D), the utility achieved is decided by the end-to-end data rate R_f, and the cost of the selected communication mode $c_f \in \{c_0, c_1, \ldots, c_I\}$.[3] Since we assume that when the relay mode is selected, the bottleneck link is in the uplink, we have the following:

$$R_f(Q_s^t, \mathbf{H}^t, x^t) = \sum_{a\in\mathcal{A}} \sum_{n\in\mathcal{N}} \tilde{\mu}_{s,a}^{t,n}\left(x^t, Q_s^t, h_{s,a}^{t,n}\right) + \sum_{n\in\mathcal{N}} \tilde{\mu}_{s,d}^{t,n}\left(x^t, Q_s^t, h_{s,d}^{t,n}\right). \quad (4)$$

Furthermore, the network-wide utility is defined as the ratio between the actual data rate and the price. With all the flows in $\mathcal{F}$, we have

$$f(\mathbf{Q}^t, \mathbf{H}^t, x^t) = \sum_{f\in\mathcal{F}} \frac{\sum_{a\in\mathcal{A}} \sum_{n\in\mathcal{N}} \tilde{\mu}_{s,a}^{t,n}\left(x^t, Q_s^t, h_{s,a}^{t,n}\right) + O}{c_f(x^t)} \quad (5)$$

where $O = \sum_{n\in\mathcal{N}} \tilde{\mu}_{s,d}^{t,n}(x^t, Q_s^t, h_{s,d}^{t,n})$. With imperfect NSI, $f(\mathbf{Q}^t, \mathbf{H}^t, x^t)$ is a random variable.

Essentially, the resource sharing problem is to select the optimal transmission mode and the resource utilization approach from Ω. If we have perfect knowledge of the channel and the queue states, we can have a formulation of the resource sharing problem via deterministic optimization. Such optimization can be solved by ranking all the possible values of the objective function (5) and then choosing the x with the biggest objective function value. Other system design aspects, such as power control, can also be jointly considered.

IV. Resource Sharing With Imperfect Network State Information

As we can see from the previous section, NSI is needed to make resource sharing decisions. However, the observation of the network state is inaccurate in general due to the limitation in measurement and the dynamics of the whole system. The imperfectness is even worse in the virtual wireless networks since the NSI is transmitted to the central controller via rate-limited backhaul networks, which introduce transmission delays. Therefore, here, we first formulate the resource sharing problem with full-buffer traffic and imperfect CSI, which does not need to take QSI into account. Then, we take the random traffic arrival into consideration where only noisy CSI and QSI can be obtained.

A. Full-Buffer Traffic and Imperfect CSI

The assumption of the availability of CSI is the same as in [22] and [23], where the inaccuracy in CSI is modeled as additive noise. The noisy version of the channel state $\hat{\mathbf{H}}$ is given by

$$\hat{\mathbf{H}} = \mathbf{H} + \mathbf{E} \quad (6)$$

in which the noise term $\mathbf{E}$ is usually assumed as independent and identically distributed Gaussian [24]. For each decision epoch, the network-wide utility defined in (5) is a random variable since $\mathbf{H}^t = \hat{\mathbf{H}}^t - \mathbf{E}^t$ is a random variable, and $\mathbf{Q}^t$ is ignored in this section. With imperfect CSI, we can only optimize the expectation of the sum utility, i.e.,

$$\underset{x\in\Omega}{\text{maximize}} \quad \mathbb{E}\left[f(\mathbf{H}^t, x^t)\right]. \quad (7)$$

A solution for the given discrete stochastic optimization is to exhaustively search the feasible set Ω [25]. To do so, we first need to compute the empirical average of the objective function assuming a particular action x is taken. Over a very large period of time T, according to strong law of large numbers, we have

$$\hat{f}(x) \triangleq \frac{1}{T}\sum_{t=1}^{T} f(\mathbf{H}^t, x) \to \mathbb{E}\left[f(\mathbf{H}^t, x)\right]. \quad (8)$$

Then, the optimal action is

$$x^* = \underset{x\in\Omega}{\text{argmax}}\, \hat{f}(x). \quad (9)$$

The exhaustive search approach has two disadvantages. First, it requires an extremely large amount of computation. To be exact, the number of times for computing the objective function value is $T \cdot \binom{|\mathcal{S}|\times|\mathcal{A}|+|\mathcal{S}|}{|\mathcal{N}|}$. Even if it might be possible to do so for a practical virtual network, the efficiency of the computation is very low because ideally, only the empirical average of the optimal objective function is needed. Second, the actual virtual wireless network is a dynamic system, and the channel states are time varying. However, to have statistically meaningful empirical average, we need to have a sufficiently large T, which is not achievable if the channels are changing in a time period less than T.

DSAs [26], [27] can address these drawbacks. Stochastic approximation algorithms are a type of iterative optimization algorithms to find the optimal solution of the objective functions, which cannot be explicitly computed, but there are estimations of them [28]. DSAs are a subset of general stochastic approximations where the decision variables are discrete.

[3]Rigidly, $\{c_0, c_1, \ldots, c_I\}$ is a tuple instead of a set since the price offered by different InPs can be the same.

1) DSA Algorithm for Static Channel: The DSA algorithm to solve the resource sharing problem (7) is illustrated in Algorithm 1. The basic spirit of the algorithm is to randomly explore Ω while keeping track of the number of visited resource sharing modes. In each round of iteration, the algorithm tends to pick up the resource sharing mode with higher network-wide user utility. We use a $|\Omega| \times 1$ vector π^t to record the relative frequency of each mode that has been visited up to time t, i.e.,

$$\pi^t[x^t] = \frac{\#\text{ of visits to } x^t}{t}. \tag{10}$$

In the initialization phase, the elements of π^0 are set to 0, except for that corresponding to the initial selected mode.

Each iteration in Algorithm 1 consists of four steps. In Step 1, a resource sharing mode $\tilde{x}^t$ is uniformly chosen, and the corresponding objective function value is evaluated. The same as in [26], we assume that the estimation of the network state is unbiased. We can achieve unbiased estimates by using different training preambles in channel estimation. In Step 2, if the randomly chosen objective function value is larger than that of the current state x^t, it is accepted as the next *visited state*. In Step 3, the state occupation probabilities are updated. e_x is a $|\Omega| \times 1$ vector, in which all the elements are 0 except for the xth element with a value of 1. The newly updated state occupation probability consists of two parts. The first part represents the probability computed from the last iteration, which is deteriorated by a factor of $(1-\nu^{t+1})$. The second part is the effect of the accepted state in step 2, in which $\mathbf{e}_{x^{t+1}}$ is a vector with all the elements being 0 except for the x^{t+1}th element, which has a value of 1. Note that the step size in each iteration is $1/t$, which means as time goes, the algorithm is getting more and more conservative to explore new resource sharing possibilities. This is the reason the algorithm is also called *aggressive* DSA [29]. In step 4, if the frequency of the newly visited state is higher than the resource sharing mode selected in the last iteration, it is chosen to be the new resource sharing mode.

In the virtual wireless network, each iteration of Algorithm 1 operates in a slot. At the end of each iteration, namely, the beginning of each slot, $\hat{x}^t$ will be selected as the resource sharing mode. It is easy to recognize from Algorithm 1 that the sequence of states visited x^t is a Markov chain on the state space Ω. Even if the states visited are not guaranteed to converge, we will show in the following that under a weak condition, the sequence of $\hat{x}^t$ converges almost surely to the global optimizer.

Algorithm 1 Aggressive DSA Algorithm for Software-Defined D2D Resource Sharing

{Step 0: Initialization}
  $t \leftarrow 0$
  Select initial mode $x^0 \in \Omega$, and $\pi^0[x^0] \leftarrow 1$
  $\pi^0[x] \leftarrow 0$, for all $x \neq x^0$
  Initialize estimate of optimal transmission mode as $\hat{x}^0 \leftarrow x^0$
for $t = 0, 1, \ldots$ **do**
  {Step 1: Sampling and evaluation}
  Given the channel gains $\hat{\mathbf{H}}^t$, obtain $f(\hat{\mathbf{H}}^t, x^t)$
  Choose $\tilde{x}^t \in \Omega \setminus x^t$ uniformly, obtain an independent observation $f(\hat{\mathbf{H}}^t, \tilde{x}^t)$
  {Step 2: Acceptance}
  if $f(\hat{\mathbf{H}}^t, x^t) < f(\hat{\mathbf{H}}^t, \tilde{x}^t)$ **then**
    $x^{t+1} \leftarrow \tilde{x}^t$
  else
    $x^{t+1} \leftarrow x^t$
  end if
  {Step 3: Adaptive filter for updating state occupation probabilities}

$$\pi^{t+1} \leftarrow (1-\nu^{t+1})\pi^t + \nu^{t+1}\mathbf{e}_{x^{t+1}}, \text{ with } \nu^t = \frac{1}{t} \tag{11}$$

  {Step 4: Update estimate of optimal transmission mode}
  if $\pi^{t+1}[x^{t+1}] > \pi^{t+1}[\hat{x}^t]$ **then**
    $\hat{x}^{t+1} \leftarrow x^{t+1}$
  else
    $\hat{x}^{t+1} \leftarrow \hat{x}^t$
  end if
end for

2) Verification of Convergence of Algorithm 1:

Theorem 1 [31]: For $\tilde{x} \neq x^*$, $x \neq x^*$, and independent observations of the corresponding objective function values, if

$$\Pr\left\{f(\boldsymbol{H}^t, x^*) > f(\boldsymbol{H}^t, x^t)\right\} > \Pr\left\{f(\boldsymbol{H}^t, x^t) > f(\boldsymbol{H}^t, x^*)\right\} \tag{12}$$

$$\Pr\left\{f(\boldsymbol{H}^t, x^*) > f(\boldsymbol{H}^t, \tilde{x}^t\right\} > \Pr\left\{f(\boldsymbol{H}^t, x^t) > f(\boldsymbol{H}^t, \tilde{x}^t)\right\} \tag{13}$$

then the sequence $\{x^t\}$ is a homogeneous irreducible and aperiodic Markov chain with state space Ω. Moreover, for sufficiently large t, the sequence $\{\hat{x}^t\}$ spends more time in x^* than any other states.

Using the similar techniques used in [26] and [31], we verify that Algorithm 1 converges to the global optimizer after sufficient numbers of iterations. Assume the mean value and the variance of the objective $f(\mathbf{H}^t, x^t)$ are $\mathbb{E}[f(\mathbf{H}^t, x^t)] = \theta_x$ and $\mathrm{Var}(f(\mathbf{H}^t, x^t)) = \sigma_x^2$, respectively. Consider the empirical cumulative distribution function $F_n(x) = (1/N)\sum_{i=1}^N 1_{X_i < x}$ of $f(\mathbf{H}^t, x^t)$, in which random variable X_i denotes the observations of the objective function. We can obtain empirical distributions of the objective function values. Fig. 4 is an example of the objective function values. The subplot on the left is the probability density function, and that on the right is the cumulative density function. By extensive simulations with MATLAB, we have verified that the distribution of the objective function can be approximated by a Gaussian distribution $N(\theta_x, \sigma_x^2)$ within a significance level of 5% based on the Kolmogorov–Smirnov test [32] if the variance of the channel estimation error is less than or equal to 20% of the mean value of the actual channel gains.

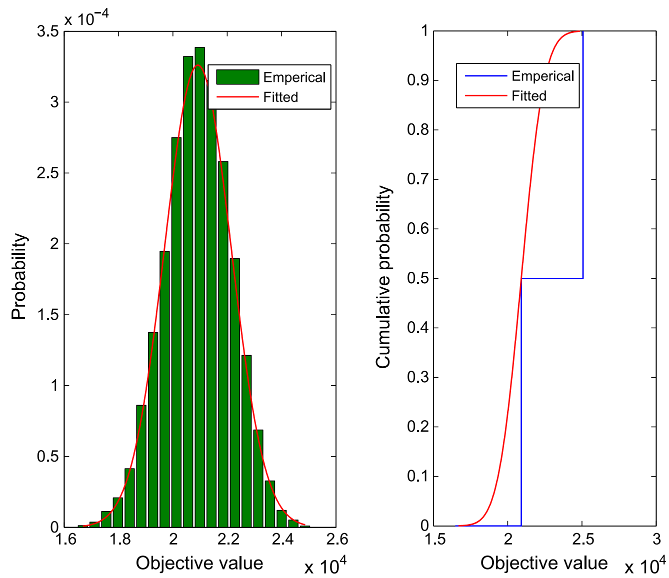


Fig. 4. Statistical distributions of the sum user utility values based on channel estimates: CSI noise variance is equal to 5% of the mean channel gains.

Therefore, we can use Gaussian distribution to denote the distribution of the objective function given that the channel estimation error is constrained within 20%. Consider three different transmission subsets $x_i = x^*$ and $x_j, x_l \in \{\Omega \setminus x^*\}$. From the above, we have $f(\mathbf{H}^t, x_i^t) \sim \mathcal{N}(\theta_{x_i}, \sigma_{x_i}^2)$, $f(\mathbf{H}^t, x_j^t) \sim \mathcal{N}(\theta_{x_j}, \sigma_{x_j}^2)$, and $f(\mathbf{H}^t, x_l^t) \sim \mathcal{N}(\theta_{x_l}, \sigma_{x_l}^2)$. Condition (12) can be rewritten as

$$\Pr\left\{f\left(\mathbf{H}^t, x_i^t\right) - f\left(\mathbf{H}^t, x_j^t\right) > 0\right\} > \Pr\left\{f\left(\mathbf{H}^t, x_j^t\right) - f\left(\mathbf{H}^t, x_i^t\right) > 0\right\}. \quad (14)$$

Due to the fact that samples of $f(\cdot)$ can be approximated by Gaussian distribution, the given inequality is equivalent to

$$\Pr\left\{\mathcal{N}\left(\theta_{x_i} - \theta_{x_j}, \sigma_{x_i}^2 + \sigma_{x_j}^2\right) > 0\right\} > \Pr\left\{\mathcal{N}\left(\theta_{x_j} - \theta_{x_i}, \sigma_{x_i}^2 + \sigma_{x_j}^2\right) > 0\right\}. \quad (15)$$

Since $x_i = x^*$ is the maximizer of the problem, we have $\max\{\theta_{x_i}, \theta_{x_j}, \theta_{x_k}\} = x_i$; hence, $(\theta_{x_i} - \theta_{x_j}) > \theta_{x_j} - \theta_{x_i}$. Therefore, (15) holds since both terms have the same variance. Based on the Gaussian distribution approximation, (13) can be rewritten as

$$\Pr\left\{\mathcal{N}\left(\theta_{x_i} - \theta_{x_j}, \sigma_{x_i}^2 + \sigma_{x_j}^2\right) > 0\right\} > \Pr\left\{\mathcal{N}\left(\theta_{x_l} - \theta_{x_j}, \sigma_{x_l}^2 + \sigma_{x_j}^2\right) > 0\right\}. \quad (16)$$

Furthermore, it is equivalent to

$$\frac{\theta_{x_i} - \theta_{x_j}}{\sqrt{\sigma_{x_i}^2 + \sigma_{x_j}^2}} > \frac{\theta_{x_l} - \theta_{x_j}}{\sqrt{\sigma_{x_l}^2 + \sigma_{x_j}^2}}. \quad (17)$$

Again, we verify the given inequality by extensive simulations.

3) Adaptive Step Size for Resource Sharing Optimization in Time-Varying Channels: In the previous section, Algorithm 1 solves the resource sharing optimization problem in static wireless channels, namely, the channel gains are fixed. However, the wireless channel in practical systems is time varying. In Algorithm 1, the step size ν^t is set as $1/t$, which makes the algorithm more and more conservative to stay at the most promising state as the algorithm proceeds. Moreover, the region of Ω is not sufficiently explored. In time-varying networks, the algorithm might stick to some suboptimal solution while getting more and more conservative. Therefore, ν^t has a prominent impact on the performance of the stochastic approximation algorithm. Intuitively, we should have $0 \le \nu^t \le 1$ to keep (11) as a probability. Consider two extreme cases. If $\nu = 1$, the previous states are all forgotten, which is equivalent to the exhaustive search; if $\nu = 0$, the state occupation probability is fixed at one value, which will miss the chance to find a better transmission subset.

To track the optimal step size ν^* in time-varying channels, we use an adaptive continuous least mean squares algorithm for the transmission optimization. Denote state occupation probability depending on the value of ν at iteration t as $\pi^{\nu,t}$ and its mean square derivative $(\partial/\partial\nu)\pi^{\nu,t}$ as $\mathbf{J}^{\nu,t}$, i.e.,

$$\lim_{\Delta \longrightarrow 0} \mathbb{E}\left\{\left|\frac{\pi^{\nu+\Delta,t} - \pi^{\nu,t}}{\Delta} - \mathbf{J}^{\nu,t}\right|^2\right\} = 0. \quad (18)$$

The error related to ν in iteration t is denoted as

$$\epsilon^{\nu,t} = \mathbf{e}_{x^{t+1}} - \pi^{\nu,t}. \quad (19)$$

Differentiating (11) with respect to ν, we have

$$\mathbf{J}^{\nu,t+1} = (1-\nu)\mathbf{J}^{\nu,t} + (\mathbf{e}_{x^{t+1}} - \pi^{\nu,t}). \quad (20)$$

Algorithm 2 is a modification of Algorithm 1. It adjusts the step size according to the error in (19) to minimize its expectation.

Algorithm 2 Adaptive DSA Algorithm

1: {Replace step 3 in Algorithm 1 with by}
2: {Step 3: Adaptive filter for updating state occupation probabilities}
3: $\epsilon^{\nu,t} \leftarrow \mathbf{e}_{x^{t+1}} - \pi^t$
4: $\pi^{t+1} \leftarrow \pi^t + \nu^t \epsilon^{\nu,t}$
5: $\nu^{t+1} \leftarrow \{\nu^t + \eta\epsilon^{\nu,t}\mathbf{J}^{\nu,t}\}_{\nu-}^{\nu+}$
6: $\mathbf{J}^{\nu,t+1} \leftarrow (1-\nu^t)\mathbf{J}^{\nu,t} + \epsilon^{\nu,t}$, $\mathbf{J}^{\nu,0} = 0$

In Algorithm 2, η is called the learning rate. If we set $\eta = 0$, the step size is constant. With $0 \le \nu^- < \nu^+ \le 1$, $\{X\}_{\nu-}^{\nu+}$ denotes the projection of X into $[\nu^-, \nu^+]$, i.e., $\{X\}_{\nu-}^{\nu+} = \min\{\max\{X, \nu^-\}, \nu^+\}$.

B. Stochastic Traffic Arrivals and Imperfect QSI

With random delay in transmitting QSI, we can only have noisy observation of the queue states, which is denoted as $\hat{\mathbf{Q}}$.[4] The optimization problem in (7) is updated as

$$\underset{x \in \Omega}{\text{maximize}} \quad \mathbb{E}\left[f(\mathbf{Q}^t, \mathbf{H}^t, x^t)\right]. \tag{21}$$

We apply the DSA algorithms previously developed to this problem. Even if it is intractable to show the convergence of the algorithms provided stochastic traffic and QSI delay, as we shall show in the following, we can still see considerable performance gains from our approach.

Discussion: In this paper, we assume unlimited buffer size. This means the latency is not considered in the formulation. However, this framework could be still applicable when limited buffer size is considered. For example, we can introduce the *Lyapunov function* [33] in the objective function (21).

V. Simulation Results and Discussions

In the simulations, we investigate a time-slotted system consisting of seven APs and 300 users. There are, in total, 150 pairs of potential peer-to-peer requests, with one flow for each pair. The intersite distance is 500 m; the users are uniformly dropped in regions 300 m away from their home APs. There are two InPs, which are denoted as InP_1 and InP_2. The default prices for InP_1, InP_2, and $D2D$ are 0.8$, 1.2$, and 0.4$, respectively. A single unit of radio resource of a bandwidth of 150 kHz is assumed. The transmission power is fixed at 0 dBm. The noise power in dBm is computed as $-174 + 10\log_{10}(\text{Bandwidth})$. The wireless channel samples are generated based on SCME channel model, in which the carrier frequency is 2.11 GHz, the user velocity is 6 Km/h, and the sample duration is 1 ms. The fading of the channels is assumed to be block fading with a block size of 100. For stochastic traffic arrivals, Poisson arrivals with a parameter λ are used, in which the size of packets is 512 Bytes. The achievable link data rate is computed based on Shannon bound. The CSI noise is modeled as a zero-mean Gaussian random variable with the value of standard deviation being 5% of mean channel gain by default. To model QSI noise, we add Gaussian noise to actual queue lengths and truncate them into nonnegative values. The standard deviation of the added Gaussian noise is also 5% of the average queue length by default.

In the following, we shall present the simulation results for various settings. The first two sections discuss the performance under various traffic intensities and different prices. Then, we shall discuss the convergence property of the proposed algorithms. At the end, we present some results to illustrate the effect of different levels of NSI noise.

A. Various Traffic Arrival Intensities

To show the effect of traffic arrival intensity λ, we range λ from 0 to 150 packets per second. Four schemes are compared.

[4]The modeling of imperfectness of QSI is not as straightforward as CSI noise because the queuing state values are integers.

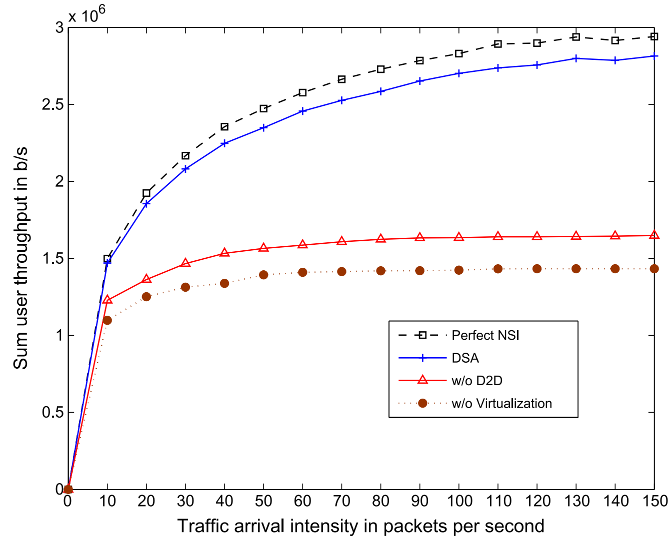


Fig. 5. Sum user throughput performance versus traffic arrival intensity.

The *Perfect NSI* is the one exhaustively searching the best resource sharing decision, given perfect NSI. The *DSA* is that which is using Algorithm 2, for which the parameters used in the simulations are as follows: The learning rate η is 0.9, and the lower bound and upper bound for ν are 0 and 0.9, respectively. In the *w/o Virtualization* scheme, the radio resources are shared among cells in a round-robin fashion, and either D2D or relay mode can be selected to utilize the resource but the resource sharing is constrained within a cell. To see the effect of data roaming, we assume that half of the users need to roam the InP they do not belong to. When roaming happens, an extra amount of money will be charged, which is the same as the visited InP's price. The *w/o D2D* scheme is that having wireless network virtualization but D2D communication is disabled. For all the schemes except for *DSA*, perfect NSI is used to compute the results. This provides an upper bound for the baseline schemes, including *Perfect NSI*, *w/o Virtualizatio*, and *w/o D2D* since only imperfect NSI is available in practical networks. In the simulations, we run the algorithm for 100 times, and each run has a different random seed. Then, we plot the average performance in Figs. 5–9 to compare the performance of different algorithms.

To see the throughput performance, we need to bypass the effect of pricing. Hence, we set the prices for all the resource utilization schemes to be 1.2$. As shown in Fig. 5, with the increase in the traffic arrival, the achievable system throughput of all the schemes will saturate. As for the proposed DSA-based scheme, it can utilize imperfect NSI, and its performance gap between the Perfect NSI scheme is rather small. While for the two schemes utilizing merely either D2D or virtualization, the former scheme outperforms the latter scheme in terms of system throughput.

Note that the higher system throughput cannot always bring more satisfaction to the users particularly when the prices for the resource sharing schemes are different. Fig. 6 shows the sum user utilities, in which the prices for D2D, InP_1, and InP_2 are 0.4$, 0.8$, and 1.2$, respectively. We can see that the DSA

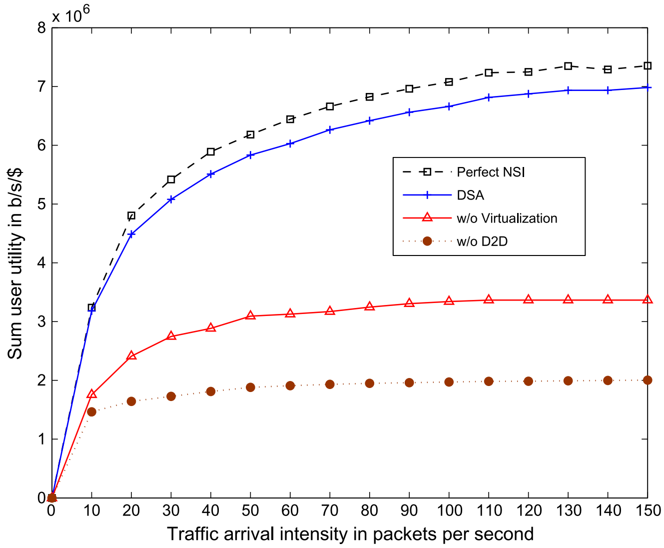


Fig. 6. Sum user utility performance versus traffic arrival intensity.

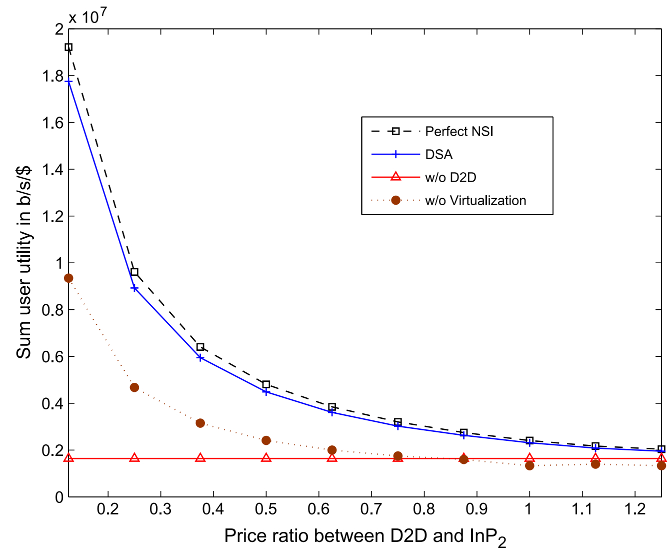


Fig. 8. Sum user utility performance versus the price ratio between D2D and InP_2.

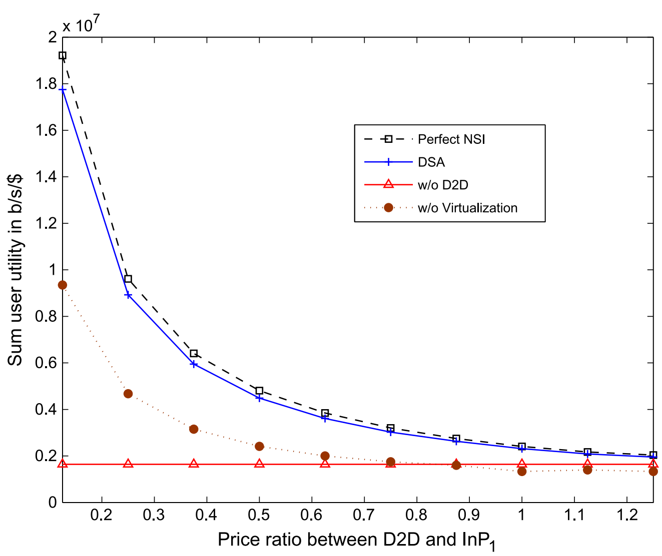


Fig. 7. Sum user utility performance versus the price ratio between D2D and InP_1.

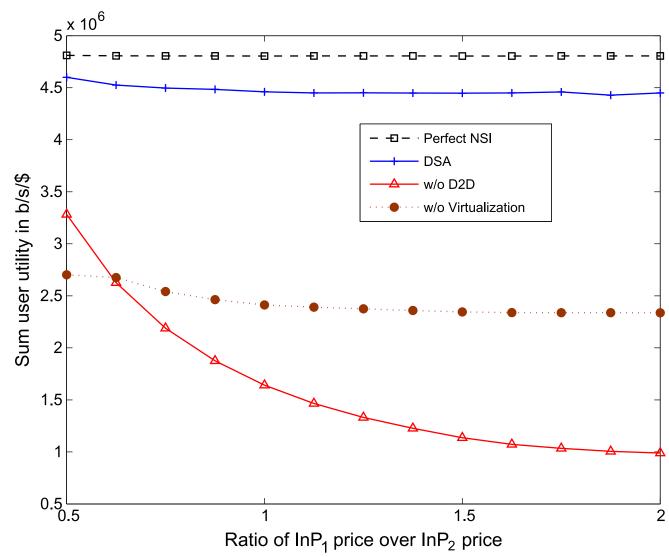


Fig. 9. Sum user utility performance versus the price ratio between InP_1 and InP_2.

scheme can have close performance as the Perfect NSI scheme. However, the scheme without virtualization outperforms that without D2D in terms of sum user utility in $b/s/\$$. Such contrast in the two figures is mainly caused by the price setting. We shall discuss the effect of prices in Section V-B.

B. Effect of Prices

Figs. 7 and 8 show the sum user utility under different D2D prices provided the traffic arrival being 20 packets per second and the same NSI noise setting as in Section V-A. In terms of sum user utility, when the ratios of the D2D price and the two InPs' prices are greater than 1, the difference between the four schemes is marginal. This means the benefits of virtualization and D2D will be weakened as the price of D2D increases. The scheme without D2D is independent of the D2D price. In contrast, the scheme without virtualization is sensitive to the D2D prices. In particular, when the ratio is greater than about 0.875%, the scheme without D2D has slightly higher sum utility than the scheme without virtualization.

Fig. 9 shows the effect of the ratio between InP_1's and InP_2's prices. The scheme without D2D is the most sensitive among the four schemes because the options among the users are limited inside the relay transmission mode. Without virtualization, each cell takes turns in utilizing the resource. However, since D2D is enabled, the users' utility slightly drops as the price of InP_1 increases. For the schemes employing both D2D and virtualization, the effect of a single InP's price is very marginal, since the resource sharing can be flexibly alternating between D2D and any InP bringing higher benefit to users.

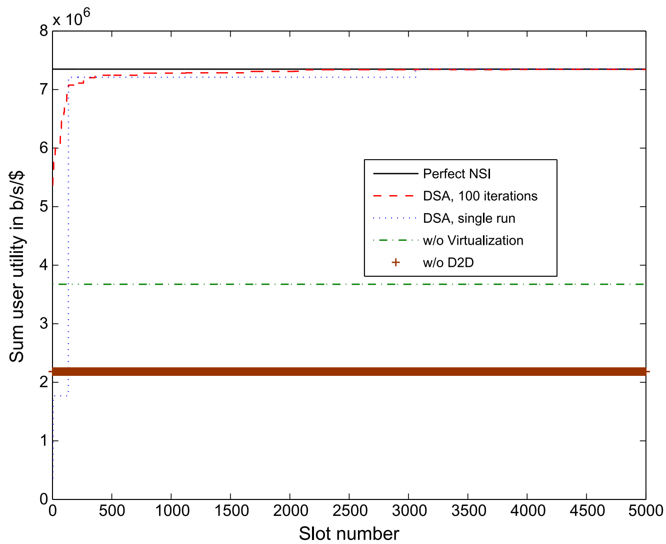


Fig. 10. Sum user utility under static channels.

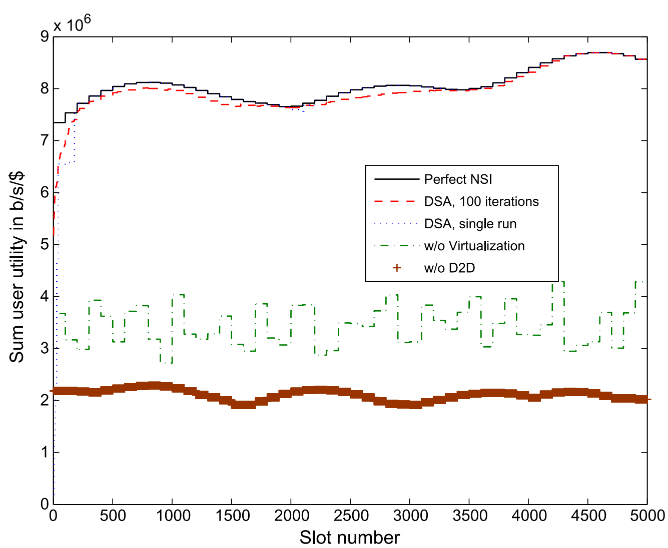


Fig. 11. Sum user utility under time-varying channels. Low CSI noise.

C. Convergence Properties

1) Static Channel and Full-Buffer Traffic: To show the convergence property, we plot the results from a single simulation run to compare different algorithms. We first investigate the case with static channels and full-buffer traffic discussed in Section IV-A, in which the inaccuracy of NSI merely comes from noisy CSI. Fig. 10 shows the achieved sum user utility with the standard deviation of CSI noise being 5% of the mean value of channel gains. There are two curves for DSA: The solid curve is the results averaged over 100 iterations, whereas the dotted curve is one shot of the algorithm. Clearly, we can see that the DSA algorithm is able to approach the optimal as the number of iteration increases. Such learning ability corresponds to the discussion in Section IV-A2.

2) Time-Varying Channels and Full-Buffer Traffic: An instance of sum user utility under time-varying channels and full-buffer traffic is demonstrated in Fig. 11. The CSI noise level is the same as in Section V-C1. Due to the block-fading channel, the performances of the schemes, except for DSA, tend to be static inside a block, whereas the proposed DSA-based scheme is able to approach the optimal performance within a fading block.

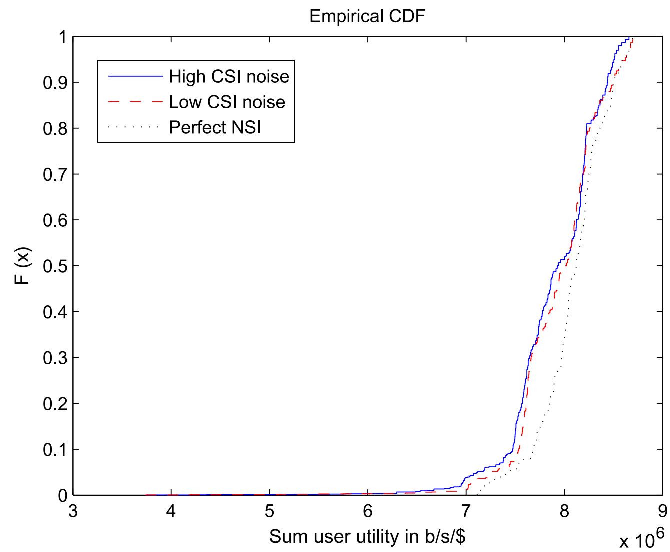


Fig. 12. Cumulative density function of sum user utility (full-buffer traffic and noisy CSI).

D. Effect of NSI Noise

In Fig. 12, we compare the cumulative density of the sum utility with low and high CSI noise in the full-buffer traffic scenario. The low and high CSI represent the standard deviation of CSI noise being 5% and 20% of the mean, respectively. In most cases, the DSA algorithm performs better, given low CSI noise than high CSI noise. However, the difference is still marginal. This implies that the proposed DSA algorithm is robust against CSI noise.

We fix the CSI noise at 5% to see the effect of QSI noise. We first add Gaussian noise to the actual queue length values and then round them up into integers to represent noisy QSI values. For the low and high QSI noise curves in Fig. 13, the standard deviation of the Gaussian noise are 5% and 20% of λ, respectively. The results in Fig. 13 are under traffic intensity λ of 20. We see that given noisy CSI and noisy QSI, the DSA algorithm still shows satisfactory performance. Moreover, with high QSI noise, its performance degrades very slightly. Another observation from comparing Figs. 12 and 13 is that the noise in CSI has a higher impact on DSA's performance than that in QSI.

VI. Conclusions and Future Work

In this paper, we have studied a framework to provide software-defined D2D communications in virtual wireless networks. To address the challenge introduced by imperfect NSI, including CSI and QSI, in this framework, we have proposed resource sharing algorithms based on recent advances in DSA. Both static channels and time-varying channels were considered. Extensive simulation results were presented to show that

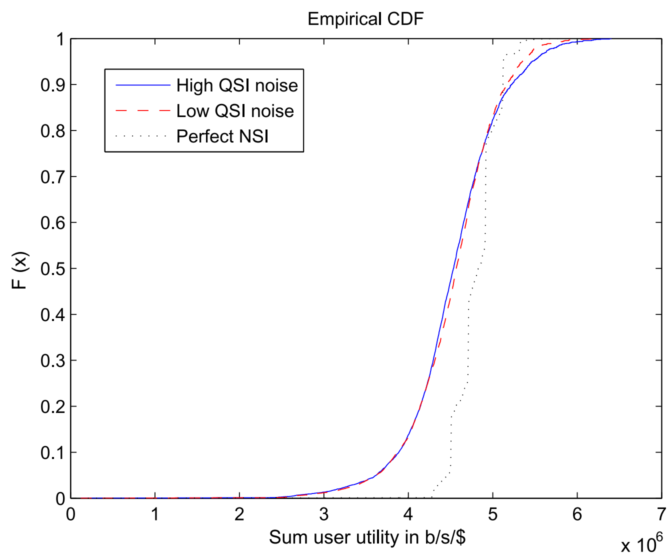


Fig. 13. Cumulative density function of sum user utility (Poisson arrivals, noisy CSI and QSI).

wireless mobile users can benefit from both wireless network virtualization and software-defined D2D communications, and the proposed scheme can achieve significant performance gain under practical network settings. Future work is in progress to consider full-duplex relaying [34] and information-centric networking [35] in the proposed framework.

Acknowledgment

The authors would like to thank the editors and reviewers for their careful examination of the manuscript and valuable comments, which have greatly helped improve the quality of this paper.

References

[1] R. Jain and S. Paul, "Network virtualization and software defined networking for cloud computing: A survey," *IEEE Commun. Mag.*, vol. 51, no. 11, pp. 24–31, Nov. 2013.
[2] X. Costa-Perez, J. Swetina, T. Guo, R. Mahindra, and S. Rangarajan, "Radio access network virtualization for future mobile carrier networks," *IEEE Commun. Mag.*, vol. 51, no. 7, pp. 27–35, Jul. 2013.
[3] A. Gudipati, D. Perry, L. E. Li, and S. Katti, "SoftRAN: Software defined radio access network," in *Proc. ACM HotSDN*, 2013, pp. 25–30.
[4] L. E. Li, Z. M. Mao, and J. Rexford, "CellSDN: Software-defined cellular networks," Princeton Univ., Princeton, NJ, USA, Tech. Rep., 2012.
[5] M. Bansal, J. Mehlman, S. Katti, and P. Levis, "OpenRadio: A programmable wireless dataplane," in *Proc. 1st Acm Workshop Hot Topics Softw. Defined Netw.*, 2012, pp. 109–114.
[6] X. Jin, L. E. Li, L. Vanbever, and J. Rexford, "SoftCell: Taking control of cellular core networks," *arXiv e-print 1305.3568*, May 2013.
[7] T. Taleb, "Toward carrier cloud: Potential, challenges, and solutions," *IEEE Wireless Commun.*, vol. 21, no. 3, pp. 80–91, Jun. 2014.
[8] C. Liang and F. R. Yu, "Wireless network virtualization: A survey, some research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 358–380, 1st Quart. 2015.
[9] C. Liang and F. R. Yu, "Wireless virtualization for next generation mobile cellular networks," *IEEE Wireless Commun.*, vol. 22, no. 1, pp. 61–69, Feb. 2015.
[10] M. Chiosi *et al.*, "Network Functions Virtualisation—Introductory White Paper," AT&T and others, Oct. 2012.
[11] A. Asadi, Q. Wang, and V. Mancuso, "A survey on device-to-device communication in cellular networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 4, pp. 1801–1819, 4th Quart. 2014.
[12] T. Taleb, A. Ksentini, A. Reznik, and T. Metsch, "Research and standards: Leading the evolution of telecom network architectures," *IEEE Wireless Commun.*, vol. 21, no. 3, pp. 10–11, Jun. 2014.
[13] T. Taleb, A. Ksentini, and A. Kobbane, "Lightweight mobile core networks for machine type communications," *IEEE Access*, vol. 2, pp. 1128–1137, Sep. 2014.
[14] N. Reider and G. Fodor, "A distributed power control and mode selection algorithm for D2D communications," *EURASIP J. Wireless Commun. Netw.*, vol. 2012, no. 1, p. 266, Dec. 2012.
[15] K. Doppler, M. Rinne, C. Wijting, C. Ribeiro, and K. Hugl, "Device-to-device communication as an underlay to LTE-advanced networks," *IEEE Commun. Mag.*, vol. 47, no. 12, pp. 42–49, Dec. 2009.
[16] R. Xie, F. R. Yu, and H. Ji, "Dynamic resource allocation for heterogeneous services in cognitive radio networks with imperfect channel sensing," *IEEE Trans. Veh. Technol.*, vol. 61, no. 2, pp. 770–780, Feb. 2012.
[17] Y. Dai, J. Wu, and A. Daniels, "Effective channel assignment based on dynamic source routing in cognitive radio networks," *Ad Hoc Sensor Wireless Netw.*, vol. 24, no. 3/4, pp. 219–247, 2015.
[18] S. Bu, F. R. Yu, and H. Yanikomeroglu, "Interference-aware energy-efficient resource allocation for heterogeneous networks with incomplete channel state information," *IEEE Trans. Veh. Technol.*, vol. 64, no. 3, pp. 1036–1050, Mar. 2015.
[19] A. Goldsmith *et al.*, "Beyond Shannon: The quest for fundamental performance limits of wireless ad hoc networks," *IEEE Commun. Mag.*, vol. 49, no. 5, pp. 195–205, May 2011.
[20] F. Fu and U. Kozat, "Stochastic game for wireless network virtualization," *IEEE/ACM Trans. Netw.*, vol. 21, no. 1, pp. 84–97, Feb. 2013.
[21] S. Sen, C. Joe-Wong, S. Ha, and M. Chiang, *Smart Data Pricing*. Hoboken, NJ, USA: IEEE, 2014.
[22] C.-H. Yu, K. Doppler, C. Ribeiro, and O. Tirkkonen, "Resource sharing optimization for device-to-device communication underlaying cellular networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 8, pp. 2752–2763, Aug. 2011.
[23] K. Doppler, C.-H. Yu, C. Ribeiro, and P. Janis, "Mode selection for device-to-device communication underlaying an LTE-advanced network," in *Proc. IEEE WCNC*, 2010, pp. 1–6.
[24] T. Weber, A. Sklavos, and M. Meurer, "Imperfect channel-state information in MIMO transmission," *IEEE Trans. Commun.*, vol. 54, no. 3, pp. 543–552, Mar. 2006.
[25] G. C. Pflug, *Optimization of Stochastic Models: The Interface Between Simulation and Optimization*. Boston, MA, USA: IEEE, 1996.
[26] I. Berenguer, X. Wang, and V. Krishnamurthy, "Adaptive MIMO antenna selection via discrete stochastic optimization," *IEEE Trans. Signal Process.*, vol. 53, no. 11, pp. 4315–4329, Nov. 2005.
[27] X. Li, N. Zhao, Y. Sun, and F. R. Yu, "Interference alignment based on antenna selection with imperfect channel state information in cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 7, pp. 5497–5511, Jul. 2016.
[28] H. J. Kushner and G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. Berlin. Germany: IEEE, 2003.
[29] V. Krishnamurthy, X. Wang, and G. Yin, "Spreading code optimization and adaptation in CDMA via discrete stochastic approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 9, pp. 1927–1949, Sep. 2004.
[30] S. Andradottir, "A global search method for discrete stochastic optimization," *SIAM J. Opt.*, vol. 6, no. 2, pp. 513–530, 1996.
[31] Q. Guan, F. R. Yu, S. Jiang, and V. C. M. Leung, "Capacity-optimized topology control for MANETs with cooperative communications," *IEEE Trans. Wireless Commun.*, vol. 10, no. 7, pp. 2162–2170, Jul. 2011.
[32] G. R. Shorack and J. A. Wellner, *Empirical Processes With Applications to Statistics*. Hoboken, NJ, USA: IEEE, Apr. 1986.
[33] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," in *Synthesis Lectures on Communication Networks*, vol. 3. San Rafael, CA, USA: IEEE, 2010, pp. 1–211.
[34] G. Liu, F. R. Yu, H. Ji, V. Leung, and X. Li, "In-band full-duplex relaying: A survey, research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 2, pp. 500–524, 2nd Quart. 2015.
[35] C. Fang, F. R. Yu, T. Huang, J. Liu, and Y. Liu, "A survey of green information-centric networking: Research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 3, pp. 1455–1472, 3rd Quart. 2015.

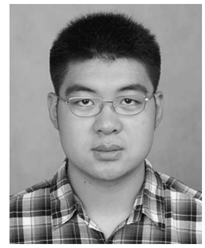
Yegui Cai received the B.Eng. and M.Eng. degrees in electrical engineering from South China University of Technology, Guangzhou, China, in 2008 and 2010, respectively, and the Ph.D. degree in electrical and computer engineering from Carleton University, Ottawa, ON, Canada, in 2014.

In 2015, he joined Irdeto Canada as a Mitacs Postdoctoral Research Intern, where he is currently a Software Developer, working on protecting online advertising.

F. Richard Yu (S'00–M'04–SM'08) received the Ph.D. degree in electrical engineering from the University of British Columbia, Vancouver, BC, Canada, in 2003.

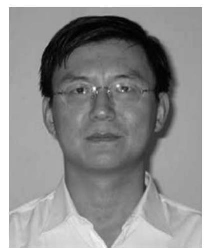
From 2002 to 2004, he was with Ericsson, Lund, Sweden, where he worked on the research and development of wireless mobile systems. From 2005 to 2006, he was with a startup in California, USA, where he worked on research and development in the areas of advanced wireless communication technologies and new standards. In 2007, he joined Carleton School of Information Technology and the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, Canada, where he is currently an Associate Professor. His research interests include cross-layer/cross-system design, security, green information technology, and quality-of-service provisioning in wireless-based systems.

Dr. Yu received the IEEE Outstanding Leadership Award in 2013, the Carleton Research Achievement Award in 2012, the Ontario Early Researcher Award (formerly Premier's Research Excellence Award) in 2011, the Excellent Contribution Award at the 2010 IEEE/IFIP International Symposium on Trusted Computing and Communications (TrustCom), the Leadership Opportunity Fund Award from the Canada Foundation of Innovation in 2009, and Best Paper Awards at the 2014 IEEE International Conference on Communications, the 2012 IEEE Global Telecommunications Conference, the 2009 IEEE/IFIP TrustCom, and the 2005 International Conference on Networking. He serves on the Editorial Board of several journals, including as a Coeditor-in-Chief for *Ad Hoc and Sensor Wireless Networks*; a Lead Series Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, the EURASIP *Journal on Wireless Communications Networking*, the Wiley *Journal on Security and Communication Networks*, and the *International Journal of Wireless Communications and Networking*; a Guest Editor for the IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTING Special Issue on Advances in Mobile Cloud Computing; and a Guest Editor for the IEEE SYSTEMS JOURNAL for the Special Issue on Smart Grid Communications Systems. He has served on the Technical Program Committee (TPC) of numerous conferences, such as a TPC Cochair of the 2015 IEEE International Conference on Green Computing and Communications (GreenCom), the 2015 IEEE INFOCOM Workshop on Mobile Cloud and Virtualization, the 2014 IEEE Global Telecommunications Conference (Globecom), the 2014 International Symposium on Wireless Vehicular Communications, the 2014 IEEE INFOCOM Workshop on Mobile Cloud Computing, the 2013 IEEE Globecom, the 2013 GreenCom, the 2013 IEEE Consumer Communications and Networking Conference, the 2012 IEEE INFOCOM Workshop on Communications and Control for Sustainable Energy Systems, the 2012 IEEE ICC Workshop on Green Communications and Networking, the 2012 IEEE Vehicular Technology Conference (VTC–Sring), the 2011 IEEE Globecom, the 2011 IEEE INFOCOM Workshop on Green Communications and Networking, the 2010 IEEE INFOCOM Workshop on Cognitive Wireless Communications and Networking, the 2009 IEEE International Wireless Communications and Mobile Computing Conference, the 2008 IEEE VTC (Fall), and the 2007 Wireless Networking for Intelligent Transportation Systems; a Publication Chair of the 2010 International ICST Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness; and a Cochair of the 2009 International Conference on Ultra Modern Telecommunications Cognitive Wireless Communications and Networking. He is a registered Professional Engineer in the province of Ontario, Canada.



Chengchao Liang received the B.Eng. degree in communication engineering and the M.Eng. degree in communication and information systems from Chongqing University of Posts and Telecommunications, Chongqing, China, in 2010 and 2013, respectively. He is currently working toward the Ph.D. degree with the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, Canada.

From 2011 to 2012, he was a visiting student with the Mobile Telecommunications Research Laboratory, Inha University, Incheon, Korea. His research interests include wireless network virtualization, radio resource allocation, interference management for heterogeneous networks and cellular systems, as well as applications of convex optimization in mobile networks.



Bo Sun (S'01–M'04) received the Ph.D. degree in computer science from Texas A&M University, College Station, TX, USA, in 2004.

He is currently an Associate Professor with the Department of Computer Science, Lamar University, Beaumont, TX. His research interests include mobile and wireless networking (with a focus on wireless sensor networks), distributed computing, network security, and privacy.

Qiao Yan received the Ph.D. degree in information and communication engineering from Xidian University, Xi'an, China, in 2003.

She is a Professor with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. From 2004 to 2005, she was a Postdoctoral Fellow with Tsinghua University, Beijing, China. From 2013 to 2014, she was a Visiting Scholar with Carleton University, Ottawa, ON, Canada. Her research interests include network security, cloud computing, and software-defined networking. Her current focus is research and development of security of software-defined networking.