# On Designing Distributed Auction Mechanisms for Wireless Spectrum Allocation

Shuo Yang, Dan Peng, Tong Meng, *Student Member, IEEE*, Fan Wu, *Member, IEEE*, Guihai Chen, *Senior Member, IEEE*, Shaojie Tang, Zhenhua Li, and Tie Luo, *Member, IEEE*

**Abstract**—Auctions are believed to be effective methods to solve the problem of wireless spectrum allocation. Existing spectrum auction mechanisms are all centralized and suffer from several critical drawbacks of the centralized systems, which motivates the design of distributed spectrum auction mechanisms. However, extending a centralized spectrum auction to a distributed one broadens the strategy space of agents from one dimension (bid) to three dimensions (bid, communication, and computation), and thus cannot be solved by traditional approaches from mechanism design. In this paper, we propose two distributed spectrum auction mechanisms, namely distributed VCG and FAITH. Distributed VCG implements the celebrated Vickrey-Clarke-Groves mechanism in a distributed fashion to achieve optimal social welfare, at the cost of exponential communication overhead. In contrast, FAITH achieves sub-optimal social welfare with tractable computation and communication overhead. We prove that both of the two proposed mechanisms achieve faithfulness, i.e., the agents' individual utilities are maximized, if they follow the intended strategies. Besides, we extend FAITH to adapt to dynamic scenarios where agents can arrive or depart at any time, without violating the property of faithfulness. We implement distributed VCG and FAITH, and evaluate their performance in various setups. Evaluation results show that distributed VCG results in optimal allocation, while FAITH is more efficient in computation and communication.

**Index Terms**—Wireless network, spectrum allocation, game theory, distributed algorithmic mechanism design, vcg mechanism, faithfulness

## 1 Introduction

The naturally limited radio spectrum is becoming increasingly scarcer due to the fast development of wireless technology. Unfortunately, traditional static spectrum allocation approaches are expensive and inefficient, causing newly emerged wireless services and applications unable to meet their demands for spectrum [1]. To tackle the limitations of traditional spectrum allocations, secondary spectrum market has been widely adopted where spectrum owners (i.e., primary users) can sell or lease idle spectrum to wireless applications (i.e., secondary users). Auctions have become natural choices for the secondary market due to their fairness and efficiency [2].

In recent years, a number of spectrum auction mechanisms (e.g., [2], [3], [4], [5], [6], [7], [8], [9], [10], [11]) have been proposed. These mechanisms achieve some attractive properties, such as strategy-proofness and approximate social welfare. Here, intuitively, strategy-proofness means that one can maximize her payoff by truthfully revealing her private valuation on the spectrum; social welfare means the sum of auction winners' valuations on the allocated spectrum. However, these existing spectrum auction mechanisms have to rely on a centralized and trusted authority to perform as an auctioneer and to process the auction procedures.

The centralized spectrum auction mechanisms have several critical drawbacks [12]. The first is that the functionality of the centralized mechanisms is based on the assumption that there exists a trusted central authority. But in practice, especially in the secondary spectrum market for wireless networks, a trusted central authority may not always exist. The second drawback is that the scalability of the centralized spectrum auctions can be poor. Since the centralized mechanisms usually need an auctioneer to collect all the bids in order to calculate the auction outcome, the agents need reliable ways to deliver their bids to the auctioneer. Unfortunately, such communication channels may not always exist between the auctioneer and the agents in wireless networks, especially when the wireless network is not fully connected. The third drawback, which is not only limited to spectrum auction mechanisms, but also applies to centralized systems in general, is robustness. Once the central authority breaks down, the entire system collapses.

To tackle the above drawbacks of the centralized spectrum auction mechanisms, we propose to implement distributed spectrum auction mechanisms. However, designing a distributed spectrum auction mechanism is much more challenging due to the following three reasons.

Most of all, without the management of a central authority, the roles of agents are now two-fold. They need not only to compete with each other for the wireless spectrum (as they do in centralized mechanisms), but also to cooperate in determining the outcome of the auction. This greatly broadens the strategy space of the agents from *one dimension* (i.e.,

- *S. Yang, D. Peng, T. Meng, F. Wu, and G. Chen are with the Shanghai Key Laboratory of Scalable Computing and Systems, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200000, China. E-mail: {wnmmxy, pd347, mengtong}@sjtu.edu.cn, {fwu, gchen}@cs.sjtu.edu.cn.*
- *S. Tang is with the Department of Information Systems, University of Texas at Dallas, Richardson, TX 75080. E-mail: tangshaojie@gmail.com.*
- *Z. Li is with the School of Software, Tsinghua University, Beijing 100084, China. E-mail: lizhenhua1983@tsinghua.edu.cn.*
- *T. Luo is with the Institute for Infocomm Research, A*STAR, Singapore. E-mail: luot@i2r.a-star.edu.sg.*

*Manuscript received 30 Oct. 2015; revised 4 June 2018; accepted 29 Aug. 2018. Date of publication 0 . 0000; date of current version 0 . 0000.*
*(Corresponding author: Fan Wu.)*
*For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.*
*Digital Object Identifier no. 10.1109/TMC.2018.2869863*

bid reporting) to *three dimensions* (i.e., bid reporting, message passing, and computation) [13], and thus are beyond the scope of traditional mechanism design perspective.

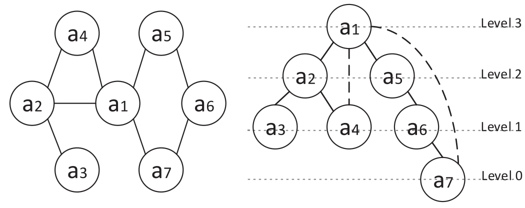
Second, unlike conventional goods, wireless spectrum can be spatially reused by multiple agents as long as their transmissions do not reduce each other's Signal to Interference and Noise Ratio (SINR) below a predefined threshold. Such a unique property makes it computationally intractable when calculating an optimal spectrum allocation for a large scale wireless network, even in a centralized manner. Due to lack of global information of inter-agent interferences, optimizing the spectrum allocation with local knowledge in a distributed wireless network is really challenging.

Third, due to wireless devices' limited computation capability and communication bandwidth, traditional secure multiparty computation cannot be directly applied, given its high computation and communication overhead. Therefore, the problem of designing a manipulation-resistant distributed auction mechanism need to be carefully considered.

In this paper, we consider the spectrum allocation problem from the perspective of *distributed algorithmic mechanism design* (DAMD) [12], and adopt the solution concept of faithfulness to characterize three-dimensional manipulation-proofness of distributed mechanisms. We propose two complementary distributed auction mechanisms, i.e., distributed VCG and FAITH. Distributed VCG is an extension of the celebrated Vickrey-Clarke-Groves (VCG) mechanism [14], [15], [16] to the distributed scenario. It collects bidding information bottom-up based on a carefully constructed pseudo-tree, and disseminates the optimal allocation top-down following the same tree structure. The payment for using the allocated spectrum is determined in the VCG manner. However, the optimal spectrum allocation is achieved at the cost of high communication overhead. Therefore, distributed VCG can only work in sparse secondary spectrum markets. Then, we present FAITH, which achieves sub-optimal spectrum allocation with bounded computation and communication overhead in general cases. We further extend FAITH to adapt dynamic network scenarios where agents may arrive and departure at any time. Our analysis shows that all the three proposed mechanisms are faithfulness.

Our main contributions are listed as follows.

- To the best of our knowledge, we are the first to consider the problem of distributed algorithmic mechanism design for secondary wireless spectrum markets. We extend the celebrated VCG mechanism to a distributed scenario, and prove that our extension is a faithful implementation of spectrum auction mechanism, achieving optimal social welfare.
- Second, we propose a more practical and efficient faithful distributed spectrum auction mechanism, called FAITH, which achieves sub-optimal social welfare with bounded computation and communication overhead.
- Third, we further extend FAITH to adapt to a dynamic network environment, where agents can arrive at and depart from the spectrum market at any time. Extended FAITH also achieves faithfulness with low communication overhead.
- Finally, we implement distributed VCG and FAITH, and extensively evaluate their performance in various topologies. Our evaluation results well demonstrate the properties of distributed VCG and FAITH.




(a) Conflict graph (b) Pseudo-tree

Fig. 1. Pseudo-tree construction.

The rest of the paper is organized as follows. In Section 2, we present the technical preliminaries, including the auction model and solution concepts. The distributed VCG and FAITH are presented in Sections 3 and 4, respectively. Then, we extend FAITH to adapt to a dynamic environment in Section 5. In Section 6, we present further discussions on our proposed mechanisms. In Section 7, we evaluate distributed VCG and FAITH, and present evaluation results. We briefly review related work in Section 8. Finally, we conclude this paper in Section 9.

## 2 PRELIMINARIES

In this section, we describe our auction model for wireless spectrum allocation, and present related solution concepts.

### 2.1 Model of Distributed Spectrum Auction

We model the problem of channel allocation in the secondary spectrum market as a distributed auction, in which there are a number of orthogonal channels to be leased out and a set of channel buyers, called *agents*, who want to lease the channels to serve their subscribers and make profits. Multiple agents can share the same channel as long as they do not interfere with each other [17]. Without the control of an auctioneer, a distributed auction is conducted by the autonomous and rational agents themselves in the secondary spectrum market. The objective of this auction is to efficiently select winners among the agents satisfying their interference constraints, and also to prevent the agents from manipulating the auction outcome.

Specifically, we consider a set $\mathbb{C} = \{c_1, c_2, \ldots, c_m\}$ of orthogonal and homogeneous channels. Information of the channels is public and known to the agents. Each channel can be simultaneously allocated to multiple non-conflicting agents, i.e., they can provide services to their subscribers simultaneously with an adequate SINR. Following the conventions of spectrum allocation auction [9], [10], [18], the interference between the agents is represented by a *conflict graph*, where an edge between two nodes/agents represents channel inference between them. An example of conflict graph is shown in Fig. 1a. We assume that a practical conflict graph has already been measured with techniques such as [10], and the underlying distributed system/application has informed each agent of her neighbors. Other alternative interference models will be discussed in Section 6.3.

We assume that the agents in one auction belong to the same connected component in the conflict graph. For a conflict graph with multiple connected components, each connected component can conduct an independent distributed spectrum auction. We also assume that conflicting agents can communicate with each other through a commonly known control channel, i.e., the communication range of the agents on the control channel is larger than the interference range of them on working channels. This is backed by the existing communication protocols, e.g., the communication range of IEEE 802.11b at a data rate of 1 Mbps is normally larger than the interference range of IEEE 802.11n at 150 Mbps.

We also consider a set $\mathbb{A} = \{a_1, a_2, \ldots, a_n\}$ of agents. Each agent $a_i \in \mathbb{A}$ has a per-channel valuation $v_i$, which is commonly known as *type* in the literature and is private to the agent herself. In a distributed auction, $a_i$ needs to report her per-channel bid $b_i$ to other agents. We note that rational agents $a_i$ may cheat her bid $b_i \neq v_i$ in order to win the spectrum auction. The agent $a_i$ also has a strict demand of $d_i$ channels. Any winning agent $a_i$ has to pay $p_i$ for allocated channel(s). We define the utility of agent $a_i$ to be the difference between her total valuation and payment, i.e., $u_i \triangleq d_i \times v_i - p_i$. Similar to papers [12], [19], [20], [21], [22], we assume that there is a Credit Clearance Service (CCS), who neither participates in the auction to determine the allocation and payment, nor needs to be always online during the auction. In distributed VCG, the CCS only collects the payments from the agents through an intermittently connected wireless overlay network. In FAITH, the CCS subtly controls agents' manipulated strategies on computation and communication by conducting an audit process.

In this paper, we consider that the agents are rational but helpful, meaning that although self-interested, each of the agents follows the prescriptions of the spectrum auction mechanism, if no unilateral deviation can lead to a better utility. We assume that there is no collusion among the agents, and tend to leave the design of collusion-resistant mechanisms to our future work.

In contrast to the agents' individual objectives, the overall objective of the spectrum auction is to maximize social welfare (*SW*), which is the sum of each winning agent $a_i$'s valuation $v_i$ on her allocated channel(s), i.e.,

$$SW \triangleq \sum_{a_i \in \mathbb{W}} (d_i \times v_i), \tag{1}$$

where $\mathbb{W} \subseteq \mathbb{A}$ is the set of winners.

### 2.2 Solution Concepts

Given the auction model, we review some important solution concepts used in this paper. First, we recall the definition of *distributed mechanism*.

**Definition 1 (Distributed Mechanism [12], [23]).** *A distributed mechanism $\mathcal{M} = (\boldsymbol{\Sigma}, \boldsymbol{s}^M, g)$ defines a feasible strategy space of agents $\boldsymbol{\Sigma} = \Sigma_1 \times \Sigma_2 \times \cdots \times \Sigma_n$, a prescribed strategy profile $\boldsymbol{s}^M = (s_1^M, s_2^M, \ldots, s_n^M) \in \Sigma$, and a determination rule $g : \boldsymbol{\Sigma} \rightarrow \mathcal{K}$ executed by the mechanism, where $\mathcal{K}$ is the set of possible outcomes.*

For any agent $a_i$, her prescribed strategy $s_i^M \in \Sigma_i$ is composed of three sub-strategies, i.e., *information-revelation strategy*, *message-passing strategy*, and *computation strategy* [23].

**Definition 2 (IC, CC, AC [23]).** *A distributed mechanism achieves IC (resp. CC, AC) if no agent can gain higher utility by deviating from her prescribed information-revelation strategy (resp. message-passing strategy and computation strategy) in an equilibrium.*

**Definition 3 (Dominant Strategy Equilibrium [24]).** *A strategy profile $\boldsymbol{s}^*$ is a dominant strategy equilibrium, if for any agent $i$, any strategy $s_i' \neq s_i^*$, and any other agents' strategy profile $\boldsymbol{s}_{-i}$, we have*

$$u_i(g(s_i^*, \boldsymbol{s}_{-i})) \geq u_i(g(s_i', \boldsymbol{s}_{-i})).$$

Dominant strategy equilibrium is a strong solution concept achieved in some traditional centralized auction mechanisms, e.g., Vickrey-Clarke-Groves mechanism [14], [15], [16]. However, it may not be achieved in distributed settings due to the agents' three-dimensional manipulations. Therefore, we turn to seek for ex-post Nash equilibrium, which is a weaker but effective solution concept in game theory.

**Definition 4 (Ex-Post Nash Equilibrium [12], [13]).** *A strategy profile $\boldsymbol{s}^*$ is an ex-post Nash equilibrium of a distributed mechanism, if for any agent $a_i$, any $s_i' \neq s_i^*$, we have*

$$u_i(g(s_i^*, \boldsymbol{s}_{-i}^*)) \geq u_i(g(s_i', \boldsymbol{s}_{-i}^*)).$$

We now introduce the concept of *faithful implementation*.

**Definition 5 (Faithful Implementation [13], [23]).** *A distributed mechanism $\mathcal{M} = (\boldsymbol{\Sigma}, \boldsymbol{s}^M, g)$ is a faithful implementation of outcome $g(\boldsymbol{s}^M)$ when prescribed strategy profile $\boldsymbol{s}^M$ is an ex-post Nash equilibrium.*

Intuitively, under a faithful distributed mechanism, the agents' individual utilities are maximized, if they follow the prescribed strategies.

## 3 Distributed VCG

In this section, we present a distributed implementation of the celebrated VCG auction mechanism. We first briefly review the concept of VCG mechanism.

**Definition 6 (VCG mechanism [14], [15], [16]).** *A mechanism $(f, p_1, \ldots, p_n)$ is a Vickrey-Clarke-Groves mechanism if*

- *Outcome function $f : (v_1, \ldots, v_n) \rightarrow \mathcal{K}$, ends up with $\boldsymbol{k}^* = argmax_{\boldsymbol{k} \in \mathcal{K}} \sum_i v_i(\boldsymbol{k})$, where $\mathcal{K}$ is the set of possible outcomes.*
- *Payment function $p_i(v_1, \ldots, v_n) = h_i(\boldsymbol{v}_{-i}) - \sum_{j \neq i} v_j(\boldsymbol{k}^*)$, where $h_i : V_{-i} \rightarrow \mathcal{R}$ (i.e., $h_i$ does not depend on $v_i$).*

We note that the outcome function of VCG outputs the optimal channel allocation $\boldsymbol{k}^*$, and the payment of each agent $a_i$ is calculated independent of $a_i$.

### 3.1 Design Rationale

To prevent the agents' manipulations, our distributed VCG mechanism is based on the *partition principle* proposed by [23]. Intuitively, the calculation process of each agent's payment is separated from the agent, *s.t.*, each agent cannot influence the calculation of her payment. Thus, it is in the best interest of every agent to follow the suggested protocol to calculate the optimal channel allocation outcome $\boldsymbol{k}^*$ (detailed proof is in Section 3.5).

To implement the VCG mechanism in a distributed manner, we also need a distributed algorithm to calculate the optimal spectrum allocation. One possible approach is to employ the algorithm of Distributed Pseudo-tree Optimization Procedure (DPOP) [25], which is the state-of-the-art solution to the distributed constrained optimization problem [26]. However, the original DPOP algorithm cannot handle the agents' multi-channel requests and does not take the spatial reusability of spectrum into consideration, thus it cannot be directly applied to the spectrum allocation scenarios. To address the limitations of the original DPOP algorithm, we extend the original DPOP algorithm by (1) proposing a concept of "constraint view" to handle the agents' multi-channel requests, and (2) reconstructing the conflicting graph into a pseudo-three to facilitate the search of optimal channel assignment. After that, the payment calculation of every agent can be implemented using our extended DPOP algorithm on a modified graph.

In this section, we propose the design of our distributed VCG mechanism, which has three phases: pseudo-tree construction, channel assignment, and payment determination.

### 3.2 Pseudo-Tree Construction

Before running the channel assignment algorithm, we first construct a pseudo-tree from the conflict graph, *s.t.*, we can exploit the problem structure of channel allocation to detect independent subproblems that can be solved separately. A pseudo-tree [27] of a graph is an arrangement of the graph with the property that adjacent vertices fall in the same branch of the tree. The relative independence of nodes lying in different branches of the pseudo-tree facilitates parallel searches for global optimal result [28], [29]. It is known that a Depth-First Search (DFS) tree is a pseudo-tree (though the inverse may not hold [25]). Fig. 1 shows an example of the pseudo-tree construction, where Fig. 1b is a pseudo-tree constructed from the conflict graph shown in Fig. 1a.

The pseudo-tree consists of tree edges, shown as solid lines, and back edges, shown as dashed lines. For each agent $a_i$ in the pseudo-tree, her parent $P(a_i)$ and children $C(a_i)$ are the set of agent(s) that are located in higher and lower levels than $a_i$ respectively and directly connected to $a_i$ through tree edges. We further define $PP(a_i)$ and $PC(a_i)$ as the set of *pseudo parents* and *pseudo children* of agent $a_i$, respectively. In contrast, an agent is connected to her pseudo parents and pseudo children through back edges. For example, in Fig. 1b, agent $a_1$ has 2 children and 2 pseudo children, i.e., $C(a_1) = \{a_2, a_5\}$ and $PC(a_1) = \{a_4, a_7\}$. Agent $a_7$ has a parent and a pseudo parent, i.e., $P(a_7) = \{a_6\}$ and $PP(a_7) = \{a_1\}$.

To construct a pseudo-tree in a distributed manner, we can employ a distributed DFS tree construction protocol (e.g., [30], [31] with polynomial time and space complexity). We note that there are multiple pseudo-trees that can be constructed by applying a given distributed DFS tree construction protocol. However, no matter which pseudo-tree is constructed, our following algorithm for channel assignment can derive an allocation profile with optimal social welfare. Due to limitations of space, we do not present here a detailed algorithm for constructing a pseudo-tree. We assume that the pseudo-tree has already been constructed and every agent has known her parent, children, pseudo parents, pseudo children, and their levels in the pseudo-tree.

### 3.3 Channel Assignment

Our channel assignment algorithm consists of two phases: a bottom-up *social welfare aggregation* and a top-down *channel choice propagation*. The former one aggregates the social welfare achieved by each subtree to calculate the optimal social welfare, while the latter let each agent select her channel allocation based on her parent's and pseudo-parents' channel selections. Our algorithm supports both single-channel demands and multi-channel demands. For clarity of presentation, we only discuss single-channel demands here, i.e., $\forall a_i \in \mathbb{A}, d_i = 1$. In this case, agents' selection domains are the same, i.e., $\forall a_i \in \mathbb{A}, D_i = \{c_1, c_2, \ldots, c_m, NULL\}$. We put $NULL$ in agents' selection domains, *s.t.*, agents can choose nothing when they do not want to lease any channel.

Here, we define agent $a_i$'s *constraint view* $CV(a_i)$ to be the set of $a_i$'s parent, $a_i$'s pseudo parents, and any other agent satisfying the following two conditions: (1) having higher level than $a_i$ and (2) having a pseudo child located in the subtree rooted at $a_i$ (e.g., $CV(a_4) = \{a_1, a_2\}$ and $CV(a_6) = \{a_1, a_5\}$). The constraint view of each agent will be obtained in the follow-up social welfare aggregation phase. In our algorithm, "$a_i : k_i$" means "when $a_i$ is allocated $k_i$" and $v_i(a_i : k_i, a_j : k_j)$ is $a_i$'s valuation over the channel allocation that $a_i$ is allocated $k_i$ and $a_j$ is allocated $k_j$, where $k_i \in D_i$ and $k_j \in D_j$. Note that $a_i$'s valuation function equals $a_i$'s per-channel valuation $v_i$ when and only when $a_i$ is allocated a channel and none of $a_i$'s neighbors are allocated the same channel.

**Algorithm 1.** Social Welfare Aggregation ($a_i$)

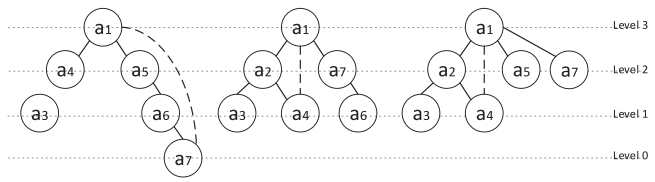
**if** $C(a_i) = \varnothing$ and $P(a_i) \neq \varnothing$ **then**
  $CV(a_i) \leftarrow P(a_i) \cup PP(a_i)$;
  **foreach** $\boldsymbol{k}_{CV(a_i)} \in \Pi_{j \in CV(a_i)} D_j$ **do**
    $SW_i(CV(a_i) : \boldsymbol{k}_{CV(a_i)}) \leftarrow$
    $max_{k_i \in D_i}(v_i(a_i : k_i, CV(a_i) : \boldsymbol{k}_{CV(a_i)}))$;
  Send $SW_i$ to $P(a_i)$;
**else**
  **if** $C(a_i) \neq \varnothing$ and $P(a_i) \neq \varnothing$ **then**
    Collect aggregation messages $\{SW_j | j \in C(a_i)\}$ ;
    Extract $CV(a_i)$ from received $SW$ messages;
    **foreach** $\boldsymbol{k}_{CV(a_i)} \in \Pi_{j \in CV(a_i)} D_j$ **do**
      $SW_i(CV(a_i) : \boldsymbol{k}_{CV(a_i)}) \leftarrow$
      $max_{k_i \in D_i}(v_i(a_i : k_i, CV(a_i) : \boldsymbol{k}_{CV(a_i)})$
      $+ \sum_{a_j \in C(a_i)} SW_j(a_i : k_i, CV(a_i) : \boldsymbol{k}_{CV(a_i)}))$;
  Send $SW_i$ to agents in $P(a_i)$;

#### 3.3.1 Social Welfare Aggregation

The bottom-up social welfare ($SW$) aggregation, as shown in Algorithm 1, starts from leaf agents of the pseudo-tree and goes up towards the root following tree edges. For agent $a_i$, $SW_i$ is the set of possible optimal social welfare that can be achieved by the subtree rooted at $a_i$, under each possible channel assignment of $CV(a_i)$. After collecting social welfare messages from her children, an agent can compose her aggregation message, and, if she is not the root, send it to her parent.

For a leaf agent $a_i$, if $P(a_i) = \{a_j\}$ and $PP(a_i) = \varnothing$, then $CV(a_i) = \{a_j\}$ and the social welfare that can be achieved at $a_i$ would only depend on her parent $a_j$. Thus the $SW_i$ sent from $a_i$ to $a_j$ would be a vector of the optimal social welfare that can be achieved at $a_i$, under each possible channel assignment of $a_j$. However, if $PP(a_i) \neq \varnothing$, then $CV(a_i) = \{a_j\} \cup PP(a_i)$ and the social welfare that can be achieved at $a_i$, would depend on both her parent and pseudo parents. Thus, the $SW_i$ would be a hypercube of $1 + |PP(a_i)|$ dimensions (one dimension for parent and the other $|PP(a_i)|$ for pseudo parents) of the tuple $\langle P(a_i), PP(a_i) \rangle$.

For an intermediate agent $a_i$, the social welfare that can be achieved by the subtree rooted at $a_i$ would be constrained by agents in her constraint view. After receiving all the $SW$ messages from her children, an intermediate agent can examine the $SW$ messages and get her children's constraint views and then extract her own constraint view




(a) DFS($A_{-2}$) (b) DFS($A_{-5}$) (c) DFS($A_{-6}$)

Fig. 2. DFS($A_{-i}$).

$CV(a_i)$. After that, under each possible channel assignment of $CV(a_i)$, say $\boldsymbol{k}_{CV(a_i)}$, $a_i$ calculates the optimal social welfare that can be achieved by the subtree rooted at $a_i$, which is $SW_i(CV(a_i) : \boldsymbol{k}_{CV(a_i)})$.

**Algorithm 2.** Choice Propagation ($a_i$)

**if** $P(a_i) = \varnothing$ **then**
  $k_i^* \leftarrow argmax_{k_i \in D_i} \sum_{a_x \in C(a_i)} SW_x(a_i : k_i)$;
  Send choice message $\langle a_i, k_i^* \rangle$ to agents in $C(a_i)$;
**else**
  Collect choice message from $P(a_i)$;
  Extract $CV(a_i)$'s channel assignment $\boldsymbol{k}^*_{CV(a_i)}$;
  $k_i^* \leftarrow argmax_{k_i \in D_i} SW_i(a_i : k_i, CV(a_i) : \boldsymbol{k}^*_{CV(a_i)})$;
  **foreach** $a_j \in C(a_i)$ **do**
    Extract $CV(a_j)$'s channel assignment $\boldsymbol{k}^*_{CV(a_j)}$;
    Send choice message $\langle CV(a_j), \boldsymbol{k}^*_{CV(a_j)} \rangle$ to $a_j$;

#### *3.3.2 Choice Propagation*

The top-down choice propagation, as shown in Algorithm 2, starts from root agent and moves towards the leaves. After receiving all the $SW$ messages, the root agent calculates the overall social welfare under each of her own channel choices, then picks the optimal choice, and sends her choice message down to her children. For any non-root agent $a_i$, based on the received choice message from her parent, $a_i$ picks her own channel choice $k_i^*$ that maximizes the social welfare for the subtree rooted at $a_i$, and sends the decision down to her children. The choice message received by $a_i$ from $P(a_i)$, contains not only her parent's choice, but also the choices of other agents in $CV(a_i)$.

When all the leaf agents have made their choices, the algorithm terminates. The channel assignment outcome $\boldsymbol{k}^* = (k_1^*, k_2^*, \ldots, k_n^*)$, where $k_i^* \in D_i$, is the one that maximizes the overall social welfare.

### 3.4 Payment Determination

After determining the optimal channel assignment $\boldsymbol{k}^*$, we calculate the payment for each winner. We set $h_i(\boldsymbol{v}_{-i})$ in VCG payment function to $max_{\boldsymbol{k} \in \mathcal{K}} \sum_{j \neq i} v_j(\boldsymbol{k})$, then the payment of agent $a_i$ is

$$p_i = \max_{\boldsymbol{k} \in \mathcal{K}} \sum_{j \neq i} v_j(\boldsymbol{k}) - \sum_{j \neq i} v_j(\boldsymbol{k}^*).$$

We define $\boldsymbol{k}^*_{-i} = argmax_{\boldsymbol{k} \in \mathcal{K}} \sum_{j \neq i} v_j(\boldsymbol{k})$, then

$$p_i = \sum_{j \neq i} v_j(\boldsymbol{k}^*_{-i}) - \sum_{j \neq i} v_j(\boldsymbol{k}^*) = \sum_{j \neq i} (v_j(\boldsymbol{k}^*_{-i}) - v_j(\boldsymbol{k}^*)).$$

From the above payment scheme, we observe that the agent $a_i$'s payment can be calculated without $a_i$. We define DFS($A$) as the DFS tree constructed from the conflict graph with all the agents $A$, and DFS($A_{-i}$) as the DFS tree with the agent $a_i$ being removed from the original conflict graph. To calculate payment for $a_i$, we first exclude $a_i$ from the conflict graph and create DFS($\mathbb{A}_{-i}$) by modifying DFS($\mathbb{A}$): the highest descendant of $a_i$ that has a back edge with an ancestor of $a_i$ turns the back edge into a tree edge. If such descendant does not exist, we exclude $a_i$ and her adjacent edges. For example, Fig. 2 shows the DFS($A_{-2}$), DFS($A_{-5}$) and DFS($A_{-6}$) after agent $a_2$, $a_5$ and $a_6$ are removed respectively from Fig. 1b. Then, we run channel assignment algorithm on modified DFS($\mathbb{A}_{-i}$) to get $\boldsymbol{k}^*_{-i}$. If excluding $a_i$ causes more connected components, then we run channel assignment algorithm on each connected component once. Afterwards, each agent $a_j \neq a_i$ is asked to report $v_j(\boldsymbol{k}^*_{-i}) - v_j(\boldsymbol{k}^*)$ to the CCS, who then extracts payments from agents' accounts. We run this procedure for each $a_i \in W$, where $W$ is the set of winners, thus $|W|$ times, to calculate payments for all agents.

### 3.5 Mechanism Analysis

**Theorem 1.** *The proposed distributed VCG mechanism is a faithful implementation.*

**Proof.** When agents follow the prescribed strategies $\boldsymbol{s}^* = (s_1^*, \ldots, s_n^*)$, the optimal allocation $\boldsymbol{k}^*$ can be achieved, then for any agent $a_i$, $a_i$'s utility is

$$\begin{aligned} u_i(g(s_i^*, \boldsymbol{s}^*_{-i})) &= v_i(\boldsymbol{k}^*) - p_i \\ &= v_i(\boldsymbol{k}^*) - \sum_{j \neq i} v_j(\boldsymbol{k}^*_{-i}) + \sum_{j \neq i} v_j(\boldsymbol{k}^*) \\ &= \sum_{j \in \mathbb{A}} v_j(\boldsymbol{k}^*) - \sum_{j \neq i} v_j(\boldsymbol{k}^*_{-i}). \end{aligned}$$

If agent $a_i$ personally chose to deviate from $s_i^*$ to $s_i' \neq s_i^*$, then the channel assignment outcome may change to $\boldsymbol{k}'$. Since $\boldsymbol{k}^*$ maximizes social welfare, then $\sum_{j \in \mathbb{A}} v_j(\boldsymbol{k}') \leq \sum_{j \in \mathbb{A}} v_j(\boldsymbol{k}^*)$. We also note that $a_i$'s payment would not be influenced by her manipulation, then $a_i$'s utility under this situation is

$$\begin{aligned} u_i(g(s_i', \boldsymbol{s}^*_{-i})) &= \sum_{j \in \mathbb{A}} v_j(\boldsymbol{k}') - \sum_{j \neq i} v_j(\boldsymbol{k}^*_{-i}) \\ &\leq \sum_{j \in \mathbb{A}} v_j(\boldsymbol{k}^*) - \sum_{j \neq i} v_j(\boldsymbol{k}^*_{-i}) = u_i(g(s_i^*, \boldsymbol{s}^*_{-i})) \end{aligned}$$

which means that under the prescribed strategy profile, following the prescribed strategy maximizes one's utility. Thus the strategy profile is an ex-post Nash equilibrium and the distributed VCG is a faithful distributed mechanism. □

We also note that the number of messages that distributed VCG produces is polynomial but the size of the largest message produced is exponential to the largest $|CV(a_i)|, \forall a_i \in \mathbb{A}$.

## 4 FAITH

In this section, we propose a more practical distributed spectrum auction, namely FAITH, to incentivize the rational agents towards an efficient spectrum allocation in ex-post Nash equilibrium. FAITH overcomes the computation and communication intractability of the distributed VCG spectrum auction, and thus can be extended to large scale spectrum markets.

### 4.1 Design Rationale

In most of the strategy-proof centralized spectrum auctions, an auctioneer sorts the agents in a non-increasing order of bids, greedily allocates channels to agents without violating the conflict constraints, and charges each winning agent with *critical price* [32]. The greedy allocation guarantees the feasibility of the algorithm, while the critical price-based payment schemes ensures the strategy-proofness. Based on this rationale, for each agent $a_i$, we divide the set of her neighbors $\mathbb{N}_i$ into *preemptive neighbor* set $\mathbb{PN}_i = \{a_j | a_j \succ a_i, a_j \in \mathbb{N}_i\}$ and *feedback neighbor* set $\mathbb{FN}_i = \{a_j | a_i \succ a_j, a_j \in \mathbb{N}_i\}$. We note that $\succ$ defines a priority order, i.e., $a_i \succ a_j$, if $b_i > b_j$, or $b_i = b_j$ and $a_i$ has a smaller index than $a_j$, where $b_i$ is $a_i$'s per-channel bid.

The greedy allocation strategy of the centralized spectrum auctions indicates that in distributed scenarios, an agent's channel allocation is only affected by her preemptive neighbors, and her allocation will directly influence the channel selections of her feedback neighbors. Thus propagating and gathering information in a well-designed order can enable the agents to determine their channel allocations in a fully distributed way.

Although the property of incentive compatibility can be achieved by enforcing the critical price-based payment scheme, simply allowing the agents themselves to handle the whole auction decision process may give them opportunities to manipulate the auction outcome by deviating from their prescribed computation and communication actions. Therefore, besides incentive compatibility, a distributed auction should also resist agents' manipulations in communication and computation. We observe that in a distributed spectrum auction, the computation and communication of an agent can be responded and confirmed by at least one of her neighbors, i.e., every agent acts both as a principal for herself, and as a witness for all of her neighbors. Exploiting agents' dual roles can provide necessary information for the CCS to verify agents' behaviors and to enable a "catch and punish" scheme (i.e., check the consistency of the information and penalize a deviation with a fine much heavier than what one can gain).

### 4.2 Design Details

FAITH has two phases: (1) Bid Exchange and (2) Channel Selection and Payment Calculation. Agents carry out the two phases autonomously and independently without the participation of any centralized party.

#### *4.2.1 Bid Exchange*

In this phase, the agents exchange bid statement messages (MSGBs) with neighbors to get local bidding information. Each agent $a_i \in \mathbb{A}$ sends her bid statement message, which is formatted as

$$\mathrm{MSGB}_i = <\mathrm{BID}, i, b_i, d_i>,$$

to all of her neighbors $\mathbb{N}_i$. Upon receiving a bid statement message $\mathrm{MSGB}_j$ from a neighbor $a_j$, agent $a_i$ adds agent $a_j$ into her preemptive neighbor set $\mathbb{PN}_i$, if $a_j \succ a_i$; otherwise, $a_i$ adds $a_j$ into her feedback neighbor set $\mathbb{FN}_i$. After the bid exchange phase, each of the agents gets her preemptive neighbor set and feedback neighbor set.

#### *4.2.2 Channel Selection and Payment Calculation*

Although logically separated, the processes of channel selection and payment calculation can be integrated together in order to reduce the number of messages involved in the distributed spectrum auction mechanism. The pseudo-code of this integrated process is shown in Algorithm 3.

We start from describing the distributed channel selection algorithm based on the locally collected bidding information, and then specify how to combine information needed for payment calculation.

In the process of channel selection, each agent $a_i$ uses channel selection message (MSGC) to inform neighbors of her selected channel set $\mathbb{C}_i^*$, in the format as

$$\mathrm{MSGC}_i = <\mathrm{CHL}, i, \mathbb{C}_i^*>.$$

As discussed in Section 4.1, the channel selection of one agent is only affected by the selection of her preemptive neighbors. Thus, agent $a_i$ first collects MSGCs from her preemptive neighbors in $\mathbb{PN}_i$, and updates her available channel set $\mathbb{AC}_i$ by deactivating the channels that are already selected by her preemptive neighbors (Lines 2 to 7). Then, if there are enough channels left, she selects the first $d_i$ channel(s) from $\mathbb{AC}_i$ as her own selected channel set

$$\mathbb{C}_i^* \leftarrow \mathrm{First}(\mathbb{AC}_i, d_i).$$

If $\mathbb{C}_i^* \neq \varnothing$, then $a_i$ is a winning agent (Lines 8 to 11).

The next step is to calculate each winning agent's payment. We employ the critical price as winning agent $a_i$'s payment, i.e., the minimum price for $a_i$ to win in the spectrum auction. In our cases, $a_i$'s critical price is the bid of her critical neighbor $a_j$, where if $a_i$ bids lower than $a_j$, $a_i$ will not be allocated, and if $a_i$ bids higher than $a_j$, $a_i$ will be allocated. Since $a_i$ cannot influence the channel selection of her preemptive neighbors, $a_i$'s critical neighbor (if exists) must be one of her feedback neighbors. Thus, to calculate $a_i$'s payment, $a_i$'s feedback neighbors are required to provide necessary information, which is their channel selection if the agent $a_i$ does not participate in the spectrum auction. Each winning agent $a_i$ sends a payment determination request message

$$\mathrm{MSGP}_i = <\mathrm{PAY}, i>$$

to her feedback neighbors (Line 10). Since the channel selection of $a_i$'s feedback neighbors can be affected by those agents that do not directly connect to $a_i$, the payment determination request message $\mathrm{MSGP}_i$ may need to be further forwarded (Line 21). We note that the total number of forwarding is bounded by the number of agents.

Upon receiving a payment determination request message $\mathrm{MSGP}_k$, agent $a_i$ first checks whether there are sufficient channels left, given her preemptive neighbors' selection if agent $a_k$ does not participate in the spectrum auction, i.e.,

$$\mathbb{AC}_{i|-k} \leftarrow \mathbb{C} - \bigcup_{j \in \mathbb{PN}_i} \mathbb{C}_{j|-k},$$

where $\mathbb{C}_{j|-k}$ denotes agent $a_j$'s channel selection if agent $a_k$ is absent from the auction (Lines 14 to 17). If $|\mathbb{AC}_{i|-k}| \geq d_i$, agent $a_i$ sets $\mathbb{C}_{i|-k} \leftarrow \mathrm{First}(\mathbb{AC}_{i|-k}, d_i)$; otherwise, $\mathbb{C}_{i|-k} \leftarrow \varnothing$ (Line 18 and 19). The agent $a_i$ encapsulates this selection into her reply message $\mathrm{MSGR}_i$ (Line 21), i.e.,

$$\mathrm{MSGR}_i \leftarrow \mathrm{MSGR}_i \mid\mid \mathrm{MSGR}_{i,k},$$

where

$$\mathrm{MSGR}_{i,k} = <\mathrm{RPY}, i, k, \mathbb{C}_{i|-k}>.$$

We note that sending the three different kinds of messages (i.e., $\mathrm{MSGC}_i$, $\mathrm{MSGP}_i$, and $\mathrm{MSGR}_i$) separately may

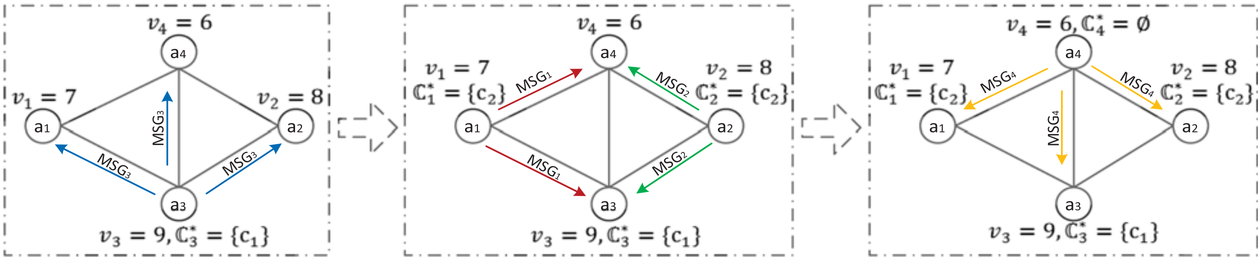



Fig. 3. Message flow for channel selection and payment calculation with four agents and two channels.

introduce extra overhead for MAC layer coordination, we combine all of these three kinds of messages together

$$\mathrm{MSG}_i \leftarrow \mathrm{MSGC}_i \mid\mid \mathrm{MSGP}_i \mid\mid \mathrm{MSGR}_i,$$

and utilize the broadcast of the wireless communication media to send the integrated messages in a single shot (Line 23).

**Algorithm 3.** Channel Selection and Payment Calculation ($a_i$)

1 $\mathbb{N}'_i \leftarrow \varnothing, \mathbb{AC}_i \leftarrow \mathbb{C}, p_i = 0$;
2 **foreach** $a_j \in \mathbb{PN}_i$ **do**
3 Receive $\mathrm{MSG}_j$ from agent $a_j$;
4 **foreach** $\mathrm{MSGP}_k = <\mathrm{PAY}, k>$ in $\mathrm{MSG}_j$ **do**
5 $\mathbb{N}'_i \leftarrow \mathbb{N}'_i \cup \{a_k\}$;
6 Extract $\mathrm{MSGC}_j = <\mathrm{CHL}, j, \mathbb{C}^*_j>$ from $\mathrm{MSG}_j$;
7 $\mathbb{AC}_i \leftarrow \mathbb{AC}_i \backslash \mathbb{C}^*_j$;
8 **if** $|\mathbb{AC}_i| \geq d_i$ **then**
9 $\mathbb{C}^*_i \leftarrow \mathrm{First}(\mathbb{AC}_i, d_i)$;
10 $\mathrm{MSGC}_i \leftarrow <\mathrm{CHL}, i, \mathbb{C}^*_i>, \mathrm{MSGP}_i \leftarrow <\mathrm{PAY}, i>$;
11 **else** $\mathbb{C}^*_i \leftarrow \varnothing$;
12 **foreach** $a_k \in \mathbb{N}'_i$ **do**
13 $\mathbb{AC}_{i|-k} \leftarrow \mathbb{C}$;
14 **foreach** $a_j \in \mathbb{PN}_i$ **do**
15 Extract $\mathrm{MSGR}_{j,k} = <\mathrm{RPY}, j, k, \mathbb{C}_{j|-k}>$ from $\mathrm{MSG}_j$;
16 **if** $\mathrm{MSGR}_{j,k}$ exists **then** $\mathbb{AC}_{i|-k} \leftarrow \mathbb{AC}_{i|-k} \setminus \mathbb{C}_{j|-k}$;
17 **else** $\mathbb{AC}_{i|-k} \leftarrow \mathbb{AC}_{i|-k} \setminus \mathbb{C}^*_j$;
18 **if** $|\mathbb{AC}_{i|-k}| \geq d_i$ **then** $\mathbb{C}_{i|-k} \leftarrow \mathrm{First}(\mathbb{AC}_{i|-k}, d_i)$;
19 **else** $\mathbb{C}_{i|-k} \leftarrow \varnothing$;
20 $\mathrm{MSGR}_{i,k} \leftarrow <\mathrm{RPY}, i, k, \mathbb{C}_{i|-k}>$;
21 $\mathrm{MSGR}_i \leftarrow \mathrm{MSGR}_i \mid\mid \mathrm{MSGR}_{i,k}$;
22 $\mathrm{MSGP}_i \leftarrow \mathrm{MSGP}_i \mid\mid \mathrm{MSGP}_k$;
23 Send $\mathrm{MSG}_i \leftarrow \mathrm{MSGC}_i \mid\mid \mathrm{MSGP}_i \mid\mid \mathrm{MSGR}_i$ to $\mathbb{N}_i$;
24 **if** $\mathbb{C}^*_i \neq \varnothing$ **then**
25 Sort agents in $\mathbb{FN}_i$ in decreasing order of bids as $\overline{\mathbb{FN}}_i$;
26 **foreach** $a_j \in \overline{\mathbb{FN}}_i$ **do**
27 Receive $\mathrm{MSG}_j$ from agent $a_j$;
28 Extract $<\mathrm{RPY}, j, i, \mathbb{C}_{j|-i}>$ from $\mathrm{MSG}_j$;
29 $\mathbb{AC}_i \leftarrow \mathbb{AC}_i \setminus \mathbb{C}_{j|-i}$;
30 **if** $|\mathbb{AC}_i| < d_i$ **then** $p_i \leftarrow b_j \times d_i$; **break**;
31 **Return** $\mathbb{C}^*_i$ and $p_i$;

After collecting replies from all her feedback neighbors (Lines 23 to 29), agent $a_i$ can calculate her payment, if she is a winning agent. Here, she sorts her feedback neighbors in a decreasing order of bids as $\overline{\mathbb{FN}}_i$ (Line 25), and then follows the order to determine her critical price $b_j$, if it exists (Lines 26 to 30). The payment is $p_i \leftarrow b_j \times d_i$ (Line 30).

### 4.2.3 A Toy Example

Fig. 3 shows a toy example for channel selection and payment calculation. In this example, we consider four agents $\mathbb{A} = \{a_1, a_2, a_3, a_4\}$, and 2 channels $\mathbb{C} = \{c_1, c_2\}$ for sale. The per-channel valuation for each agent is 7, 8, 9, and 6 respectively. For clarity, we assume that each of the agents demands one channel.

Each agent keeps a local ranking of agents (e.g., $a_1$ gets $a_3 \succ a_1 \succ a_4$, and $a_3$ gets $a_3 \succ a_2 \succ a_1 > a_4$) after the bid exchange phase. Based on the ranking, each agent sequentially selects one channel. For agent $a_3$, she does not need to consider any preemptive selections, since she ranks the highest in her neighborhood. So she selects a channel $c_1$, broadcasts her message $\mathrm{MSG}_3 = \mathrm{MSGC}_3||\mathrm{MSGP}_3$, and waits for feedback neighbors' replies to calculate her payment. Upon receiving $\mathrm{MSG}_3$, agents $a_1$ and $a_2$ can run Algorithm 3 concurrently, since they are out of conflict. Agent $a_1$ then updates her available channel set, selects a channel $c_2$, selects a payment determining channel $c_1$ assuming that $a_3$ is absent, and broadcasts her message $\mathrm{MSG}_1 = \mathrm{MSGC}_1||\ \mathrm{MSGP}_1||\mathrm{MSGR}_1$. Agent $a_2$ runs the same process. Finally, agent $a_4$ collects messages from all her preemptive neighbors and responds her own message $\mathrm{MSG}_4$. Thereafter, winning agents extract critical price from feedback messages and calculate payment (i.e., $p_1 = 0, p_2 = 0, p_3 = 6$). Table 1 lists the contents of corresponding messages.

### 4.2.4 Consistency Check

To guarantee faithfulness, the consistency of the communication and computation should be checked. Note that each message sent in the spectrum auction has at least two copies (i.e., one at the sender and the other at the receiver) in the network. We require each of the agents to submit the messages she sent and received to the CCS, when a transaction is cleared. After collecting all the messages, the CCS can check the messages, authorize the channel allocations, and collect the payments. If a mismatch is detected, the involved agents have to pay a penalty which is much higher than the largest possible utility one can gain by cheating. We note that the CCS does not always need to have a reliable communication channel with each agent, or participate in the process of distributed spectrum auction. The CCS only needs to check the consistency and clears the transaction when a connection is available after the auction.

## 4.3 Mechanism Analysis

In this subsection, we show that FAITH meets our design requirements for a distributed mechanism, especially in terms of network complexity and faithfulness.

### 4.3.1 Network Complexity

Feigenbaum et al. [12] proposed the concept of network complexity with respect to five metrics to measure the complexity of a distributed algorithm over an interconnected network $G = (\mathbb{V}, \mathbb{E})$, where $\mathbb{V} = \mathbb{A}$ is the set of agents and $\mathbb{E}$

TABLE 1
Messages Transmitted in the Network

| Msg \ Agent | $a_1$ | $a_2$ | $a_3$ | $a_4$ |
|---|---|---|---|---|
| Ranking | $a_3 \succ a_1 \succ a_4$ | $a_3 \succ a_2 \succ a_4$ | $a_3 \succ a_2 \succ a_1 \succ a_4$ | $a_3 \succ a_2 \succ a_1 \succ a_4$ |
| MSGB | $<\mathrm{BID}, 1, 7, 1>$ | $<\mathrm{BID}, 2, 8, 1>$ | $<\mathrm{BID}, 3, 9, 1>$ | $<\mathrm{BID}, 4, 6, 1>$ |
| MSGC | $<\mathrm{CHL}, 1, \{c_2\}>$ | $<\mathrm{CHL}, 2, \{c_2\}>$ | $<\mathrm{CHL}, 3, \{c_1\}>$ | $<\mathrm{CHL}, 4, \varnothing>$ |
| MSGP | $<\mathrm{PAY}, 3>$ $<\mathrm{PAY}, 1>$ | $<\mathrm{PAY}, 3>$ $<\mathrm{PAY}, 2>$ | $<\mathrm{PAY}, 3>$ | $<\mathrm{PAY}, 3>$ $<\mathrm{PAY}, 1>$ $<\mathrm{PAY}, 2>$ |
| MSGR | $<\mathrm{RPY}, 1, 3, \{c_1\}>$ | $<\mathrm{RPY}, 2, 3, \{c_1\}>$ | | $<\mathrm{RPY}, 4, 3, \{c_2\}>$ $<\mathrm{RPY}, 4, 2, \varnothing>$ $<\mathrm{RPY}, 4, 1, \varnothing>$ |

contains all the communication links among the agents in $G$. Here we demonstrate the network complexity of FAITH, in terms of the following five metrics.

- Total number of messages sent over $G$: Every agent broadcasts two messages, i.e., one for bid exchange, and the other for integrated channel selection and payment calculation, resulting in $4|\mathbb{E}|$ messages.
- Maximum number of messages sent over any link in $G$: There are 4 messages on each link due to mutual message exchanges in the two phases.
- Maximum size of a message: In the worst case, the agent with the lowest bid may inherit all the payment determination request messages from her preemptive neighbors (i.e., the agent that ranks lowest in a ring topology will extract all other agents' payment determination request messages when there are more than one channels being auctioned), which will result in a merged $\mathrm{MSG}$ with $2|\mathbb{V}|$ sub-messages (i.e., 1 for $\mathrm{MSGC}$, $|\mathbb{V}|$ for $\mathrm{MSGP}$, and $|\mathbb{V}|-1$ for $\mathrm{MSGR}$). Since each sub-message has a maximum length of $c$-byte, the maximum size of a message is $O(2c|\mathbb{V}|)$, where $c$ is a constant.
- Local computation overhead: The most computation consuming part throughout the mechanism is the payment determining channel reselection, which takes $O(\delta|\mathbb{V}|)$ time in the worst case, where $\delta$ is the maximum degree of the network.
- Local storage overhead: Every agent is required at most $O(\delta|\mathbb{V}|)$ space to store propagated messages and local outcome in the worst case.

### 4.3.2 *Faithfulness*

To prove the faithfulness of FAITH, we begin by presenting the definition of *strong*-CC and *strong-AC*, followed by an important lemma.

**Definition 7 (Strong-CC/Strong-AC [13]).** *A distributed mechanism $\mathcal{M} = (\mathbf{\Sigma}, \boldsymbol{s}^M, g)$ satisfies strong-CC/strong-AC if no agent can gain higher utility by deviating from the prescribed message-passing strategy/computation strategy, whatever the other two strategies are, when other agents follow the prescribed strategies.*

**Lemma 1 (Faithful Implementation [13]).** *A distributed mechanism $\mathcal{M} = (\mathbf{\Sigma}, \boldsymbol{s}^M, g)$ is a faithful implementation of outcome $g(\boldsymbol{s}^M)$ if the corresponding centralized mechanism is strategyproof and $\mathcal{M}$ satisfies strong-CC and strong-AC.*

It suggests that given a centralized mechanism which is strategyproof (also known as dominant strategy incentive compatible), we can prove that a distributed mechanism is faithful by combining the properties of strong communication compatibility (*strong*-CC) and strong algorithm compatibility (*strong-AC*). We assume that, for each agent, a complete implementation of the auction is much preferable than dropping out without any affirmed outcome.

In FAITH, the intended strategy for each agent is to report bidding information truthfully, pass messages correctly, and calculate channel selection, reselection and payment correctly. A rational agent $a_i$ may deviate from the intended strategy to increase her utility by performing the following actions:

- Misreport: to report false bidding information, i.e., $b_i \neq v_i$ (reporting false number of demanded channels will obviously hurt $a_i$ herself).
- Miscommunication: to drop or distort messages received from her neighbor $a_j$ (e.g., $\mathrm{MSGB}_j$ or $\mathrm{MSG}_j$), or withhold her own messages.
- Miscalculation: to divide neighbors into wrong sets, or incorrectly determine channel selection $\mathbb{C}_i^*$, channel reselection $\mathbb{C}_{i|-k}^*$ or payment $p_i$.

**Theorem 2.** *FAITH is a faithful distributed implementation of the critical price-based spectrum allocation mechanism.*

**Proof.** To prove the faithfulness of FAITH, we show that FAITH satisfies centralized strategyproofness, strong-CC and strong-AC respectively.

The corresponding centralized auction mechanism is strategyproof. The critical price-based centralized spectrum auction is proved to be strategyproof in [9].

FAITH satisfies strong-CC. Based on the redundancy principle and "catch and punish" scheme, any miscommunication behavior will be detected and punished. On one hand, each agent has no incentive to drop or distort her neighbors' messages, since doing so will cause message mismatch and will be caught and punished by the CCS in consistency check. On the other hand, agent $a_i$ will not withhold her own messages, because doing so will block the auction and thus prevent herself from participating in the auction. Hence, each agent $a_i$ has no incentive to deviate from her intended message-passing strategy.

FAITH satisfies strong-AC. In the bid exchange phase, each agent $a_i$'s neighbors are divided into two sets with different priorities. Any unilateral miscalculation will breach the determination order and cause communication chaos. In the channel selection and payment calculation phases, agent $a_i$ selects channel sets based on her preemptive neighbors' choices $\mathbb{C}_j$ ($\mathbb{C}_{j|-k}$), and calculates payments based on her feedback neighbors' choices $\mathbb{C}_{j|-i}$. All the necessary information is packed in $\mathrm{MSG}_j$ and sent to the CCS. Since any miscalculation will be caught and punished, $a_i$ has no incentive to deviate from the intended computation strategy.

Therefore, FAITH is a faithful distributed implementation of the critical price-based spectrum allocation mechanism. □

## 5 Adaption to Dynamic Environment

In previous sections, agents are considered to be static in the auction. A more practical scenario is that agents may come and go at any time. An intuitive adaptation is a "reboot" scheme, i.e., to rerun the entire auction process, whenever a new arrival or a departure occurs. However, this scheme is inflexible and costly in terms of computation and communication overheads. In this section, we extend FAITH to support agents' dynamics by only updating the smallest part of affected allocation profile.

We observe that the design rationale of FAITH can also be applied to dynamic scenarios. Intuitively, for a newly arrived agent, her neighborhood information is enough for her to determine her channel selection and the corresponding payment. Besides, the arrival or departure of an agent normally affects only a part of the existing agents.

**Algorithm 4.** Extended FAITH for Newly Arrived Agent $a_i$

1 Send $\text{MSGB+}_i \leftarrow <\text{BID}, i, b_i, d_i>$ to $\mathbb{N}_i$;
2 $\mathbb{PN}_i \leftarrow \varnothing, \mathbb{FN}_i \leftarrow \varnothing, \mathbb{N}'_i \leftarrow \varnothing, \mathbb{AC}_i \leftarrow \mathbb{C}, \mathbb{FC}_i \leftarrow \varnothing, p_i \leftarrow 0$;
3 **foreach** $a_j \in \mathbb{N}_i$ **do**
4 Receive $\text{MSG}_j$ from agent $a_j$;
5 Extract $\text{MSGB}_j = <\text{BID}, j, b_j, d_j>$ from $\text{MSG}_j$;
6 **if** $a_j \succ a_i$ **then** $\mathbb{PN}_i \leftarrow \mathbb{PN}_i \cup \{a_j\}, \mathbb{AC}_i \leftarrow \mathbb{AC}_i \setminus \mathbb{C}_j^*$;
7 **foreach** $\text{MSGP}_k = <\text{PAY}, k>$ in $\text{MSG}_j$ **do**
8 $\mathbb{N}'_i \leftarrow \mathbb{N}'_i \cup \{a_k\}$;
9 **else** $\mathbb{FN}_i \leftarrow \mathbb{FN}_i \cup \{a_j\}, \mathbb{FC}_i \leftarrow \mathbb{FC}_i \cup \mathbb{C}_j^*$;
10 Sort agents in $\mathbb{FN}_i$ in decreasing order of bids as $\overline{\mathbb{FN}_i}$;
11 **if** $|\mathbb{AC}_i| < d_i$ **then** $\mathbb{C}_i^* \leftarrow \varnothing$;
12 **else**
13 **if** $|\mathbb{AC}_i \setminus \mathbb{FC}_i| \geq d_i$ **then**
$\mathbb{C}_i^* \leftarrow \text{Random}(\mathbb{AC}_i \setminus \mathbb{FC}_i, d_i)$;
14 **else** $\mathbb{C}_i^* \leftarrow \text{LIP}(\overline{\mathbb{FN}}_i, \mathbb{AC}_i, d_i), \text{MSGP+}_i \leftarrow <\text{PAY}, i>$;
15 $\text{MSGC+}_i \leftarrow <\text{CHL}, i, \mathbb{C}_i^*>$;
16 **foreach** $a_k \in \mathbb{N}'_i$ **do**
17 $\mathbb{AC}_{i|-k} \leftarrow \mathbb{C}, \mathbb{FC}_{i|-k} \leftarrow \varnothing$;
18 **foreach** $a_j \in \mathbb{PN}_i$ **do**
19 **if** $\text{MSGR}_{j,k}$ exists **then** $\mathbb{AC}_{i|-k} \leftarrow \mathbb{AC}_{i|-k} \setminus \mathbb{C}_{j|-k}$;
20 **else** $\mathbb{AC}_{i|-k} \leftarrow \mathbb{AC}_{i|-k} \setminus \mathbb{C}_j^*$;
21 **foreach** $a_j \in \mathbb{FN}_i$ **do**
22 **if** $\text{MSGR}_{j,k}$ exists **then** $\mathbb{FC}_{i|-k} \leftarrow \mathbb{FC}_{i|-k} \cup \mathbb{C}_{j|-k}$;
23 **else** $\mathbb{FC}_{i|-k} \leftarrow \mathbb{FC}_{i|-k} \cup \mathbb{C}_j^*$;
24 **if** $|\mathbb{AC}_{i|-k}| < d_i$ **then** $\mathbb{C}_{i|-k} \leftarrow \varnothing$;
25 **else if** $|\mathbb{AC}_{i|-k} \setminus \mathbb{FC}_{i|-k}| \geq d_i$ **then**
26 $\mathbb{C}_{i|-k} \leftarrow \text{Random}(\mathbb{AC}_{i|-k} \setminus \mathbb{FC}_{i|-k}, d_i)$;
27 **else** $\mathbb{C}_{i|-k} \leftarrow \text{LIP}(\overline{\mathbb{FN}}_i, \mathbb{AC}_{i|-k}, d_i)$;
28 $\text{MSGR+}_{i,k} \leftarrow < \text{RPY}, i, k, \mathbb{C}_{i|-k} >$;
29 $\text{MSGR+}_i \leftarrow \text{MSGR+}_i \mid\mid \text{MSGR+}_{i,k}$;
30 $\text{MSGP+}_i \leftarrow \text{MSGP+}_i \mid\mid \text{MSGP}_k$;
31 Send $\text{MSG+}_i \leftarrow \text{MSGC+}_i \mid\mid \text{MSGP+}_i \mid\mid \text{MSGR+}_i$ to $\mathbb{N}_i$;
32 **if** preemption occurs **then**
33 **foreach** $a_j \in \overline{\mathbb{FN}}_i$ **do**
34 Receive $<\text{RPY}, j, i, \mathbb{C}_{j|-i}>$ from agent $a_j \in \overline{\mathbb{FN}}_i$;
35 $\mathbb{AC}_i \leftarrow \mathbb{AC}_i \setminus \mathbb{C}_{j|-i}$;
36 **if** $|\mathbb{AC}_i| < d_i$ **then**$p_i \leftarrow b_j \times d_i$; **Break**;
37 **Return** $\mathbb{C}_i^*$ and $p_i$;

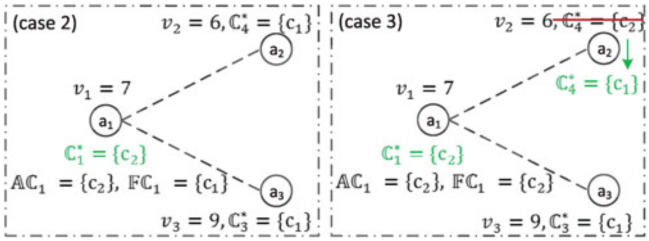



Fig. 4. An example of Case (2) and Case (3).

Algorithm 4 shows our proposed procedures for a newly arrived agent $a_i$. When agent $a_i$ arrives the market, she first broadcasts her bid statement message (MSGB+), in the format of

$$\text{MSGB+}_i = <\text{BID}, i, b_i, d_i>,$$

to her neighbors in $\mathbb{N}_i$ to inform her arrival (Line 1), and collects their MSG messages, based on which she then selects her required channels and calculate the corresponding payment.

Same as FAITH, agent $a_i$ divides her neighbors $\mathbb{N}_i$ into preemptive neighbor $\mathbb{PN}_i$ and feedback neighbor $\mathbb{FN}_i$ according to their bids. Then, agent $a_i$ updates the set of her available channels $\mathbb{AC}_i$ by deactivating the channels selected by her preemptive neighbors, and stores the set of channels selected by her feedback neighbors into $\mathbb{FC}_i$ for possible preemption. There are three cases needed to be considered:

(1) If $|\mathbb{AC}_i| < d_i$, agent $a_i$ gets nothings (Line 11); Otherwise, she can always meet her demand.
(2) If $|\mathbb{AC}_i \setminus \mathbb{FC}_i| \geq d_i$, agent $a_i$ can randomly select a subset of $d_i$ channels out of $\mathbb{AC}_i \setminus \mathbb{FC}_i$, without disturbing her neighbors (Line 13);
(3) If $|\mathbb{AC}_i| \geq d_i$ and $|\mathbb{AC}_i \setminus \mathbb{FC}_i| < d_i$, agent $a_i$ needs to preempt channels from her feedback neighbors. We propose a Least Impactive Preemption (LIP) scheme, shown in Algorithm 5, i.e., agent $a_i$ preempts channels from one of her feedback neighbors, who critically leads to the channel unavailability of $a_i$.

In case (1) and case (2), agent $a_i$'s payment should be zero, since she has no critical neighbor. However, in case (3), agent $a_i$'s critical neighbor is not straightforward to see. For example, in Fig. 4, there are 2 channels $\mathbb{C} = \{c_1, c_2\}$ for sale and 3 agents $\mathbb{A} = \{a_1, a_2, a_3\}$, where each agent demands one channel. We assume that agent $a_1$ arrives after the other two agents having been allocated channels. Since the priority is $a_3 \succ a_1 \succ a_2$, agent $a_1$ can only preempt the channel from agent $a_2$. We note that agent $a_2$ may have selected $c_1$ (case 2), or $c_2$ (case 3). In the former case, agent $a_1$ figures out that $\mathbb{AC}_1 \setminus \mathbb{FC}_1 = \{c_2\}$, and directly selects $c_2$ as her allocation with zero payment; In the latter case, $\mathbb{AC}_1 = \{c_2\}$, but $\mathbb{AC}_1 \setminus \mathbb{FC}_1 = \varnothing$, which means that agent $a_2$'s selection forces agent $a_1$ to preempt a channel. However, agent $a_2$ is not agent $a_1$'s critical neighbor, since agent $a_2$ can reselect channel $c_1$ when channel $c_2$ is preempted by agent $a_1$. Therefore, if a preemption occurs, agent $a_i$ needs to check the reply messages from her feedback neighbors to determine her payment (Lines 32 to 36). Besides determining her own payment, agent $a_i$ also needs to compose the reply message $\text{MSGR+}_{i,k}$ for every agent $a_k$, whose $\text{MSGP}_k$ message is received by $a_i$, to help $a_k$ to calculate her payment, since $a_i$'s arrival may change $a_k$'s critical neighbor (Lines 17 to 31).

Under dynamic network environment, any existing agent $a_j$ should keep listening to the control channel for incoming messages. We provide the procedures for an existing agent $a_j$ upon the arrival of a new agent $a_i$ in Algorithm 6. If agent $a_j$ is one of the agent $a_i$'s neighbors, agent $a_j$ will receive agent $a_i$'s bid statement message $\mathrm{MSGB}+_i$, based on which she can mark agent $a_i$ as her preemptive neighbor or feedback neighbor. Then, agent $a_j$ sends agent $a_i$ her information $\mathrm{MSG}_j$ to help agent $a_i$ to select channels, and wait for agent $a_i$'s $\mathrm{MSG}+_i$ message (Lines 2 to 6). Upon receiving agent $a_i$'s response, if agent $a_j$ is agent $a_i$'s preemptive neighbor, then agent $a_j$ only needs to recalculate her payment $p_j$. Otherwise, agent $a_j$ needs to check if her preoccupied channels have been preempted, to update her channel selection $\mathbb{C}_j^*$ and reselection $\{\mathbb{C}_{j|-n}\}$, to recalculate the corresponding messages ($\mathrm{MSGC}_j$, $\mathrm{MSGP}_j$, and $\mathrm{MSGR}_j$), and to send the updated $\mathrm{MSG}'_j$ to her neighbors (Lines 7 to 12). If agent $a_j$ is not agent $a_i$'s neighbor, she will not directly respond to the newly arrived agent, but only react to the updated messages $\{\mathrm{MSG}'_k\}$ received from her neighbors (Lines 13 to 18).

Once an agent $a_i$ finishes her job, she leaves the market. Before the departure, she broadcasts the leaving message (MSGL-), in the format of

$$\mathrm{MSGL}-_i = <\mathrm{LVE}, i, \mathbb{C}_i^*>,$$

to inform her neighbors to recycle her channels and recalculate their payments. We observe that if agent $a_i$ did not win the auction, her departure will not influence the remaining agents' channel selections. In this case, agents only need to update their payments. If agent $a_i$ won the auction, then every other agent $a_j$ has already known her allocation $\mathbb{C}_{j|-i}^*$ upon $a_i$'s departure, since agent $a_i$'s payment is calculated based on the allocation profile when $a_i$ is absent. In this case, every existing agent $a_j$ will directly change her channel selection to $\mathbb{C}_{j|-i}^*$. Newly allocated agents and $a_i$'s preemptive neighbors will calculate their payments by broadcasting their updated MSG messages.

**Algorithm 5.** Least Impactive Preemption (LIP)

**Input**: $\overline{\mathbb{FN}}_i, \mathbb{AC}_i, d_i$; **Output**: $\mathbb{C}_i^*$;
**foreach** $a_j \in \overline{\mathbb{FN}}_i$ **do**
  **if** $|\mathbb{AC}_i \setminus \mathbb{C}_j^*| < d_i$ **then**
    $\mathbb{C}_i^* \leftarrow \mathrm{Random}(\mathbb{AC}_i, d_i)$; **Break**;
  $\mathbb{AC}_i \leftarrow \mathbb{AC}_i \setminus \mathbb{C}_j^*$;
**return** $\mathbb{C}_i^*$;

Extended FAITH follows the same design rationale as FAITH, and thus is still faithful with polynomial computation and communication overhead. Comparing with FAITH, extended FAITH is likely to result in more communication overhead due to the dynamic arrival/departure of agents.

**Theorem 3.** *Extended FAITH is a faithful distributed mechanism in a dynamic environment.*

**Proof.** (Sketch) Similar to the proof of FAITH, we show that extended FAITH satisfies centralized strategyproofness, strong-CC and strong-AC.

First, the channel allocation of extended FAITH follows the critical price-based method on either a new arrival or a new departure, thus is still centralized strategyproof.

Second, extended FAITH satisfies strong-CC. For a newly arrived agent, she has to follow the prescribed message-passing strategy, since withholding her own messages will prevent herself from being allocated, and any other deviation in communication will be detected and punished by the CCS. The newly departured agent also cannot benefit from manipulating her prescribed message-passing strategy. For any other agent, the "catch and punish" scheme ensures that she has no incentive to deviate from her intended message-passing strategy under the intended strategy profile.

Finally, extended FAITH satisfies strong-CC. For a newly arrived agent, miscalculation may cause chaos in channel allocation and get punished. The newly departured agent no longer participates in the auction and thus has no computation strategy. We note that any deviation from the prescribed strategy will be detected and punished by the CCS, thus no agent has the incentive to deviate from the intended computation strategy under the intended strategy profile.

Therefore, extended FAITH is a faithful distributed mechanism in a dynamic environment. □

**Algorithm 6.** Extended FAITH for Existing Agent $a_j$ upon a Newly Arrived Agent $a_i$

1 **switch** *Received message* **do**
2   **case** $\mathrm{MSGB}+_i$
3     **if** $a_i \succ a_j$ **then** $\mathbb{PN}_j \leftarrow \mathbb{PN}_j \cup \{a_i\}$;
4     **else** $\mathbb{FN}_j \leftarrow \mathbb{FN}_j \cup \{a_i\}$;
5     $\mathrm{MSG}_j \leftarrow \mathrm{MSGB}_j||\mathrm{MSGC}_j||\mathrm{MSGP}_j||\mathrm{MSGR}_j$;
6     Send $\mathrm{MSG}_j$ to $a_i$; **Break**;
7   **case**$\mathrm{MSG}+_i$
8     **if** $a_i \in \mathbb{PN}_j$ **then**
9       Update $\mathbb{C}_j^*$, $\{\mathbb{C}_{j|-n}\}$, $\mathrm{MSGC}_j$, $\mathrm{MSGP}_j$, $\mathrm{MSGR}_j$, $\mathrm{MSG}_j$ based on $\mathrm{MSG}+_i$;
10       Send updated $\mathrm{MSG}'_j$ to $\mathbb{N}_j$;
11     **else** Update $p_j$ based on $\mathrm{MSG}+_i$;
12     **Break**;
13   **case**$\mathrm{MSG}'_k$
14     **if** $a_k \in \mathbb{PN}_j$ **then**
15       Update $\mathbb{C}_j^*$, $\{\mathbb{C}_{j|-n}\}$, $\mathrm{MSGC}_j$, $\mathrm{MSGP}_j$, $\mathrm{MSGR}_j$, $\mathrm{MSG}_j$ based on $\{\mathrm{MSG}'_k\}$;
16       Send updated $\mathrm{MSG}'_j$ to $\mathbb{N}_j$;
17     **else** Update $p_j$ based on $\{\mathrm{MSG}'_k\}$;
18     **Break**;
19 **Return** $\mathbb{C}_j^*$ and $p_j$;

# 6 DISCUSSION

In this section, we present further discussions of this work.

## 6.1 Optimality and Convergence

*Optimality.* Distributed VCG mechanism achieves optimal social welfare, while FAITH achieves sub-optimal social welfare based on a greedy channel allocation algorithm. However, it is infeasible to provide an approximation ratio for FAITH. The key reason is that the social welfare of FAITH are determined by multiple factors (whose instances could be arbitrary), including the agents' bids, agents' channel requests, and the conflicting relationships between agents. In this case, analyzing the approximation ratios for spectrum allocation algorithms, even in centralized scenarios, is extremely difficult and cannot be addressed by this work. In fact, due to the infeasibility, the optimality analysis of is often missing in current researches of spectrum allocation, such as [9], [11]. Thus, we do believe that it would be the best to treat it as a potential future work.

*Convergence.* The convergence of distributed VCG is clear, since it terminates after a bottom-up and then a top-down traversal. The convergence of FAITH and extended FAITH is also guaranteed. Note that the auctions determine the channel allocation results sequentially, i.e., the agent with the highest bid will be allocated first, and then the one with the second highest bid, and so on. The channel assignment of each agent will be determined eventually. As for the payment determination, each agent first collects channel assignment information from her neighbors, and determines which one of her feedback neighbors is her critical neighbor. These calculations require finite searches, and will be done in finite time.

### 6.2 Other Mechanism Design Properties

We also analyze our mechanisms with several widely used properties of algorithmic mechanism design. The properties of our proposed mechanisms are summarized in Table 2.

TABLE 2
Properties of Our Mechanisms

| | Faithful | Opt | Conv | DSE | IR | CS | NPT | VP |
|---|---|---|---|---|---|---|---|---|
| DVCG | √ | √ | √ | × | √ | √ | √ | √ |
| FAITH | √ | × | √ | × | √ | √ | √ | √ |
| E-FAITH | √ | × | √ | × | √ | √ | √ | √ |

*Dominant Strategy Equilibrium* (DSE). As discussed in Section 2.2, distributed VCG mechanism cannot guarantee dominant strategy equilibrium. Weaker mechanisms—FAITH and extended FAITH cannot achieve it as well.

*Individual Rationality* (IR). Each participating agent will have a non-negative utility, i.e., $\forall i, u_i \geq 0$. It has been proved that VCG mechanism satisfies individual rationality [33]. Since our distributed VCG mechanism follows the payment rule of original VCG mechanism, it also satisfies individual rationality. As for FAITH and extended FAITH, since the critical neighbor of each winner, if it exists, is one of her feedback neighbors, the payment of each winner is always no higher than her bid.

*Consumer Sovereignty* (CS). The mechanism cannot arbitrarily exclude any agent, and the mechanism has to allow an agent to win if she is willing to pay a sufficiently high payment (while others' bids are fixed). The consumer sovereignty of VCG mechanism has already been proved [33]. As for FAITH and extended FAITH, agents cannot be arbitrarily rejected by the mechanisms. Due to the greedy allocation rule, if an agent has the highest bid, she will win the auction. Thus, FAITH and extended FAITH also satisfy consumer sovereignty.

*No Positive Transfer* (NPT). The payments are nonnegative, i.e., $\forall i, p_i \geq 0$. This property obviously holds for each of our proposed mechanism.

*Voluntary Participation* (VP). An agent who does not participate the auction will not be charged, and an agent who wins the auction will not be charged more than her bid. In our proposed mechanisms, only winning agents will need to pay. Also, each winning agent's payment must be no more than her bid, thus voluntary participation holds.

### 6.3 Alternative Interference Models

This work, following the convention of spectrum auction literature, adopts the conflict graph to model the physical interference conditions among agents. The benefit of using a conflict graph model is the great simplification of the spectrum allocation design, *s.t.*, one can focus on the development of highly efficient allocation mechanisms with nice game-theoretical properties and polynomial complexity. Other alternative physical inference models, such as SINR-based and power-based models [2], are also promising but require specific catering of the current problem settings, including redefining the optimization and constraint terms and reconsidering the agents' manipulation strategies, thus may not be directly solved by our proposed distributed VCG and FAITH. Despite that, the design intuitions of our proposed distributed mechanisms could provide implications for subsequent designs of faithful distributed spectrum allocation methods for these models.

### 6.4 Synchronization

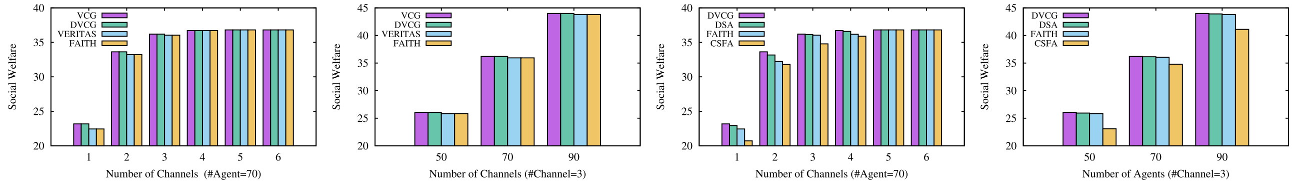
Synchronization is an important issue in distributed algorithms. Based on the way the agents update their local information, distributed algorithms can be generally classified as *synchronous* or *asynchronous* [34]. In asynchronous algorithms, each agent has its own view of the problem and updates their local variables independently from the actual decisions of other agents. In contrast, synchronous algorithms update the agents' decisions in a particular order, which is usually enforced by the representation structure adopted. It tends to delay the decisions of some agents guaranteeing their local view of the problem is always consistent with that of the other agents.

The proposed distributed VCG mechanism synchronous, since its execution follows a pre-defined order (bottom-up and then top-down) and the update of an agent's decision is postponed until all the dependent agents have been updated. As for FAITH, the channel assignment part is synchronous, as it determines the channel assignments of agents in the decreasing order of bids. In the payment determination part, each winning agent first sends payment determination request message to her feedback neighbors, and waits until all her feedback neighbors have sent her channel assignment messages. After that, each agent can calculate her own payment. Since each agent only needs to update her own payment, there's no shared agreement in this part, *s.t.*, no synchronization technique is needed. One particular part we need to deal with is that in the extended FAITH, the handling of a new arrival or departure is postponed if the auction is in progress (due to the efficiency of our algorithm, this delay is short).

## 7 Evaluation Results

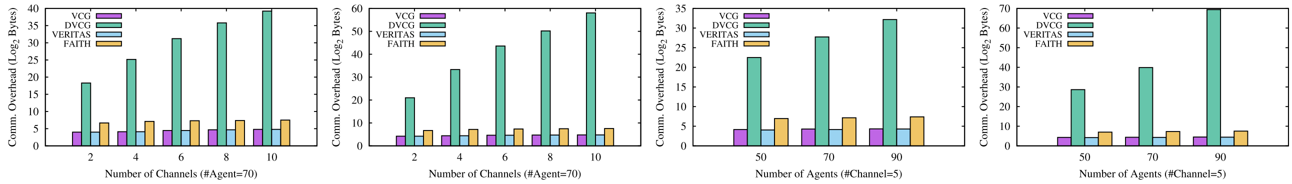
In this section, we employ NS-2 to evaluate the performance of distributed VCG and FAITH on allocation efficiency and transmission overhead. First, we explore the social welfare and communication overhead of distributed VCG and FAITH in a small spectrum market. Second, we further examine the performance of FAITH by comparing it to a non-faithful distributed allocation algorithm [35]. Finally, we investigate the efficiency of extended FAITH in a dynamic environment by comparing it to the "reboot" scheme.

We consider both real network data and simulated data. The real network data, collected by [10], records 78 access points (AP) in a $7\,\text{km}^2$ area of the Google WiFi network in Mountain View, California. The simulated data consists of




(a) Distributed vs. Centralized (b) Distributed vs. Centralized (c) Ours vs. Benchmarks (d) Ours vs. Benchmarks

Fig. 5. Social welfare comparison: (a)-(b) compare distributed mechanisms with centralized mechanisms and (c)-(d) compare DVCG and FAITH with two benchmarks.




(a) In Sparse Topologies (b) In Random Topologies (c) In Sparse Topologies (d) In Random Topologies

Fig. 6. Transmission overhead comparison: (a)-(b) vary the number of channels and (c)-(d) vary the number of agents.

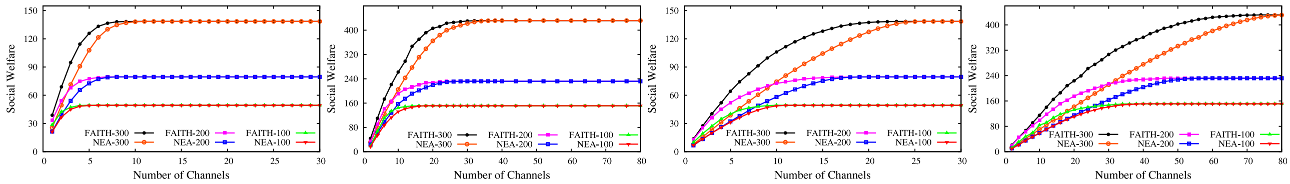
three different kinds of topologies, i.e., sparse topologies, random topologies and clustered topologies. In sparse topologies and random topologies, agents are randomly distributed in a square area of $2{,}500\,\mathrm{m} \times 2{,}500\,\mathrm{m}$. We restrict that each connected component in sparse topologies has no more than 10 agents. For clustered topologies, same as [9], we initially distributed 100 agents in a square area of $1{,}200\,\mathrm{m} \times 1{,}200\,\mathrm{m}$, and then increase agents up to 300 by adding 100 agents in the center iteratively. We apply a widely used distance-based interference model [9], [10], [18] to generate the simulated conflict graphs. In our setting, any two agents within $250\ \mathrm{m}$ will conflict with each other and thus cannot utilize the same channel simultaneously. Without loss of generality, we uniformly distribute the bids of agents in $(0, 1]$, and the channel demands in $\{1, 2, 3, 4, 5\}$. The results are averaged over 1000 runs.

### 7.1 Distributed VCG versus FAITH

We evaluate the social welfare (i.e., the sum of winning agents' valuations) and transmission overhead (i.e., the size of the largest message) of distributed VCG and FAITH. We first compare the proposed distributed mechanisms with centralized mechanisms. For distributed VCG, we compare it to the centralized VCG mechanism, and for FAITH, it is compared to a strategyproof centralized spectrum auction mechanism [9], namely VERITAS. Then, we compare distributed VCG and FAITH with two additional benchmarks. One of the benchmarks is distributed stochastic search algorithm (DSA) [36], which is a incomplete and synchronous algorithm for distributed constraint optimization problem (DCOP). It can achieve near-optimal solution with polynomial time and space complexity, and has often been used as a benchmark algorithm for DCOP [26]. The other benchmark is a centralized strategyproof and fair auction mechanism for secondary spectrum markets [37]. For simplicity, we refer to this algorithm as "CSFA" in the evaluation. Due to the exponential complexity of distributed VCG, we evaluate it only in a small scale spectrum market with sparse network topologies. The number of agents ranging from 50 to 90 and the number of channels from 1 to 10. We also assume that each agent only requires a single channel.

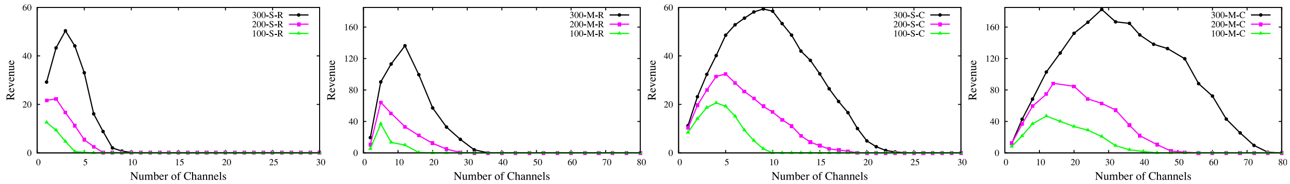
Fig. 5 shows the comparisons of social welfare between our proposed mechanisms and the benchmarks. From Figs. 5a, and 5b, we can see that distributed VCG achieves the same social welfare as VCG. This is because the distributed VCG implements the same outcome function and payment function as the original VCG mechanism. FAITH and VERITAS also have the same social welfare, since they both follow the greedy-based channel allocation rule and the critical price-based payment determination rule. Figs. 5c, and 5d shows the social welfare comparison between our proposed distributed mechanisms and additional two benchmarks: DSA and CSFA. We can observe that DSA achieves lower social welfare than distributed VCG but higher social welfare than FAITH. That is due to the reason that although DSA does not guarantee an optimal solution, it is based on a carefully designed searching method that allows the agents iteratively update their variables to achieve a near-optimal solution. Besides, CSFA achieve the lowest social welfare, because it sacrifices a portion of social welfare to ensure the fairness of the spectrum allocation. In addition, we observe that social welfare grows as the number of channels increases and reaches saturation when the number of channels is 5. When the number of channels is fixed, a larger number of agents leads to higher social welfare. Under the same parameters, distributed VCG achieves higher social welfare than FAITH before saturation, which is because that distributed VCG is designed to choose the optimal allocation, while FAITH prefers the greedy policy.

Fig. 6 compares the transmission overhead of distributed VCG, FAITH, and the centralized mechanisms in sparse topologies and random topologies respectively. We note that the $y$-axis is in logarithmic form, i.e., $log_2 S$, where $S$ is the size of the largest message in units of bytes. It can be seen that centralized mechanisms have the lowest transmission overhead since they only need basic communications (i.e., bid reporting, outcome and payment announcement) between the agents and the centralized auctioneer. As for distributed mechanisms, we observe that the transmission overhead of distributed VCG grows dramatically as the number of channels/agents increases, due to the fact that distributed VCG requires each agent to enumerate every possible channel assignment of her constraint view, which results in exponential space complexity. In contrast, the transmission overhead of FAITH grows slightly with the increase in the number of agents/channels, supporting our



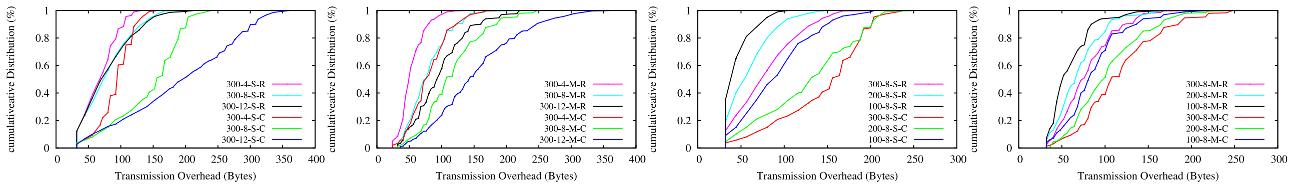

(a) Single-Channel Demand (b) Multi-channel Demand (c) Single-Channel Demand (d) Multi-Channel Demand

Fig. 7. Social welfare of FAITH *versus* NEA: (a)-(b) in random topologies and (c)-(d) in clustered topologies.




(a) Single-Channel Demand (b) Multi-Channel Demand (c) Single-Channel Demand (d) Multi-Channel Demand

Fig. 8. Revenue of FAITH: (a)-(b) in random topologies and (c)-(d) in clustered topologies.




(a) Single-Channel Demand (b) Multi-Channel Demand (c) Single-Channel Demand (d) Multi-Channel Demand

Fig. 9. Transmission overhead of FAITH: where (a)-(b) vary the number of channels and (c)-(d) vary the number of agents.

claim that FAITH has bounded transmission overhead. Despite the slight loss of social welfare, the polynomial communication complexity makes FAITH more feasible and practical, especially in large spectrum markets.

## 7.2 More Evaluations on FAITH

In Sections 7.2.1 and Section 7.2.2, we evaluate the allocation efficiency and transmission overhead of FAITH in both random topologies and clustered topologies. The number of agents ranges from 100 to 300 and the number of channels ranges from 1 to 80. We also evaluate the performance of FAITH in a real network topology in Section 7.2.3.

### *7.2.1 FAITH versus Non-Faithful Distributed Allocations*

We measure the allocation efficiency of FAITH in terms of social welfare and revenue (i.e., the sum of payments), and compare it with a non-faithful distributed Nash equilibrium based channel allocation algorithm [35] (denoted by NEA in our evaluation).

Fig. 7 shows the comparison results on social welfare between FAITH and NEA in random topologies and clustered topologies. We observe that social welfare of both algorithms grows as the number of channels gets larger and finally reaches saturation, where every agent's demand is satisfied. Besides, FAITH outperforms NEA with much higher social welfare when saturation has not been reached, due to the fact that FAITH allocates channels to higher bidders with higher priorities, while NEA allows the agents to compete for the channels in an arbitrary way.

Compared with cases in clustered topologies, FAITH saturates at fewer channels in random topologies (e.g., under single-channel demand, 9 channels for 300 agents in random topologies, while 23 channels for 300 agents in clustered topologies). This is because agents in clustered topologies are densely located, resulting in more intensive conflicts. Besides, the multi-channel demand cases also show a lag of increase compared with single-channel demand scenarios (e.g., in clustered topologies, FAITH saturates at 22 channels for 300 agents with single-channel demand, while 80 channels for 300 agents with multi-channel demand).

Fig. 8 presents the revenue of FAITH in both random topologies and clustered topologies. We do not show the results of NEA because it does not have a pricing scheme. Different from the growth trend of social welfare, revenue cannot always stay at a high level with the increment of the number of channels. This non-monotonic growth trend is caused by our critical price-based payment scheme. At first, few agents get satisfied when the number of channels is small. In this case, increasing the number of channels improves the percentage of winning agents, thus increases revenue. However, a large number of available channels alleviates the auction competition, *s.t.*, some agents no longer have critical neighbors and thus are charged zero payments. Finally, the revenue decreases to zero when every agent is satisfied. Due to different intensity levels of competition, the four subfigures of Fig. 8 show different growth and saturation speed.

### *7.2.2 Transmission Overhead of FAITH*

We also measure FAITH's per agent transmission overhead, which is defined as the total size of messages each agent generates. Fig. 9 shows the cumulative distribution of transmission overhead in bytes, where "n-m-S/M-R/C" denotes that n agents bid for m channels with single-channel(S)/multi-channel(M) demand in random(R)/clustered(C) topologies.

According to Figs. 9a, and 9b, we observe that more channels lead to heavier transmission overhead (e.g., over 85

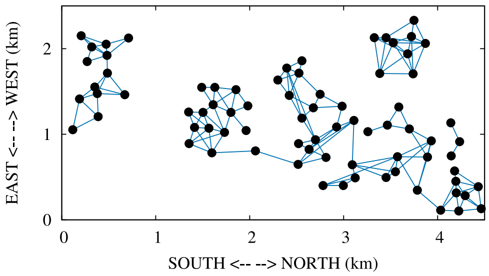



Fig. 10. The conflict graph of Google WiFi network.

percent of agents transmit no more than 100 bytes in "300-4-S-R" while the percentage is only 70 percent in "300-12-S-R"). This is because more channels result in more winners and thus more messages are generated to perform channel reselection and payment determination. Another observation is that the cumulative distribution of clustered topologies grows more slowly than that of random topologies. For example, while there is only a tiny portion of agents generating transmission overhead over 200 bytes in 300-12-S-R, the percentage is about 50 percent in "300-12-S-C". The reason is that some agents in the cluster have the most intense conflicts and thus have to transmit more messages.

Figs. 9c, and 9d present the transmission overhead with various numbers of agents. We can see that the transmission overhead grows with the increasing number of agents (e.g., over 70 percent of agents transmit no more than 50 bytes in "100-8-S-R" while the percentage is down to 30 percent in "300-8-S-R"), due to the fact that more agents lead to more intensive conflicts. Besides, single-channel demand causes larger transmission overhead than multi-channel demand. That is because that under the single-channel request of each agent, conflicting agents will have more opportunities to select channels than they do under multi-channel request, which will further result in more winners and thus more message exchanges.

#### 7.2.3 *FAITH in a Real Network Topology*

Besides simulated network topologies, we also evaluate the performance of FAITH based on a Google WiFi dataset, which was collected by Zhou et al.in April 2010 [10]. The dataset covers a 7 $\text{km}^2$ residential area of the Google WiFi network in Mountain View, California. They recorded the detailed signal strength values of 78 APs, and built a measured conflict graph, which is shown in Fig. 10. The black dots represent the APs, and the blue edges represent the conflicting relationship between the APs. We present the statistics of the number of each node's neighbors in Table 3. We can see that the number of each node's neighbors ranges from 2 to 8, while the average and the mode are 4.92 and 5, respectively. In our simulation, we treat each AP as an agent,

TABLE 3
Statistics on the Number of Agents' Neighbors in Google WiFi Network

| #Neighbors | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|
| Counts | 4 | 12 | 13 | 21 | 15 | 11 | 2 |

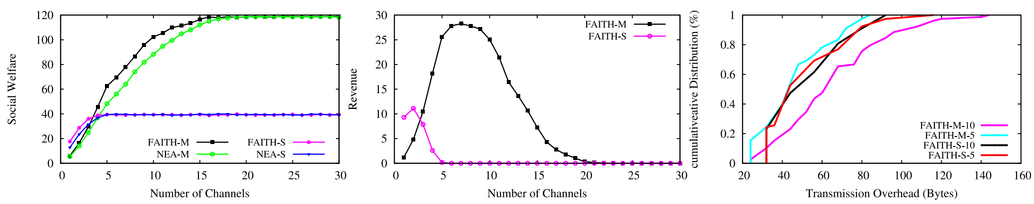
and apply FAITH to the conflict graph. The number of channels is set to 5. The agents' bids are uniformly distributed in $(0, 1]$, and their channel demands in $\{1, 2, 3, 4, 5\}$.

Fig. 11 shows the social welfare, revenue, and transmission overhead of FAITH in the real network topology. We observe that these metrics follow the similar patterns as in simulated network topologies. Specifically, Fig. 11a shows that under either single-channel demand (denoted by "FAITH-S") or multi-channel demand (denoted by "FAITH-M"), FAITH achieves higher social welfare than NEA before saturation is reached. We observe that in average, saturation can be reached when the number of channels is over 5 for single-channel demand, and 20 for multi-channel demand. In Fig. 11b, the revenue of FAITH first increases and then decreases as the number of channels grows, and finally saturates at zero. This growth trend is similar to that in simulated network topologies (shown in Fig. 8), due to the critical price-based pricing scheme of FAITH. The transmission overhead of FAITH is shown in Fig. 11c, where "FAITH-M-10" denotes the transmission overhead of FAITH under multi-channel demand with 10 channels. We see that under either single-channel demand or multi-channel demand, the transmission overhead of each agent is low (e.g., no agent needs to transmit over 160 bytes).

### 7.3 Adaptation to Dynamic Environment

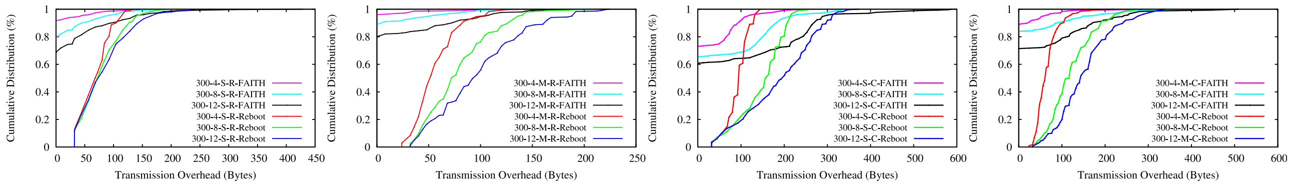
We also implement extended FAITH to evaluate its performance in dynamic environment. We compare the transmission overhead of extended FAITH to the "reboot" scheme (Section 5) on both agent arrival and agent departure. The evaluation results are presented in Figs. 12 and 13.

Fig. 12 shows the comparisons of extended FAITH and the "reboot" scheme on agent arrival, where Figs. 12a and 12b are in random topologies and Figs. 12c and 12d in clustered topologies. We observe that extended FAITH significantly reduces the transmission overhead required by the "reboot" scheme. For example, in "300-4-S-R-Reboot", about 70 percent of agents transmit over 50 bytes, while the percentage is only about 5 percent in "300-4-S-R-FAITH". That is because extended FAITH takes advantage of the existing information, instead of regenerating this information and rerunning the whole allocation. Besides, the transmission overhead under single-channel demand is heavier than under multi-channel demand, due to the fact that under single-channel demand, a new arrival usually lead to a larger variation in



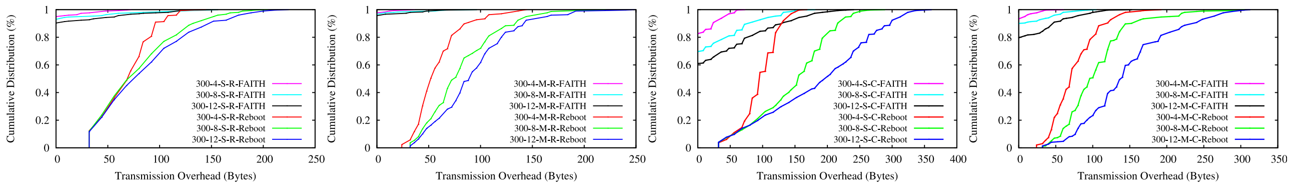

(a) Social Welfare (b) Revenue (c) Transmission Overhead

Fig. 11. Performance of FAITH in a real conflict graph.




(a) Single-Channel Demand (b) Multi-Channel Demand (c) Single-Channel Demand (d) Multi-Channel Demand

Fig. 12. Transmission overhead of extended FAITH *versus* reboot scheme on agent arrival: (a)-(b) in random topologies and (c)-(d) in clustered topologies.




(a) Single-Channel Demand (b) Multi-Channel Demand (c) Single-Channel Demand (d) Multi-Channel Demand

Fig. 13. Transmission overhead of extended FAITH *versus* reboot scheme on agent departure: (a)-(b) in random topologies and (c)-(d) in clustered topologies.

the allocation profile. We also observe that clustered topologies generate heavier transmission overhead than random topologies
(e.g., about 80 percent of agents do not transmit messages in "300-12-M-R-FAITH", while only 70 percent in "300-12-M-C-FAITH"). This is because the clustered topologies have more conflicts than random topologies, *s.t.*, a newly arrived agent will disturb more agents in clustered topologies than in random topologies.

Fig. 13 presents the comparisons of the transmission overhead of extended FAITH and the "reboot" scheme on agent departure, where Figs. 13a, and 13b are in random topologies and Figs. 13c, and 13d in clustered topologies. It can be observed that extended FAITH greatly reduced the transmission overhead of the "reboot" scheme. Comparing Figs. 13 to Fig. 12, we can see that extended FAITH requires less transmission overhead on agent departure than agent arrival. This is because that the allocation profile under the absence of an agent is known when calculating the agent's payment, *s.t.*, remaining agents only need to update necessary payments.

## 8 Related Work

*Spectrum Allocation Protocols.* There are many research works addressing the problem of spectrum allocation in various network settings, such as cellular networks [38], [39], wireless LANs [40], wireless mesh networks [41], [42], mobile ad-hoc networks [43], 5G networks [44], [45], heterogeneous networks [46], and vehicular networks [47]. Most of them assume that agents in the networks strictly follow the prescribed protocols, thus cannot be applied to scenarios where agents are rational and only interested in maximizing their own utilities.

*Auction-Based Spectrum Allocation.* Auction-based spectrum allocation mechanisms, which model the problem as a game over rational agents, have been extensively studied to improve spectrum utilization and allocation fairness. Following the pioneer work of Zhou et al. [9], various researchers have addressed the problem from from different perspectives listed as follows:

- *Double Auction:* TRUST [11] considers the incentive problem of both sellers and buyers, and elegantly extends spectrum market to double auction. SMALL [8] further improves TRUST to achieve a higher channel utilization ratio. TAHES [6] addresses heterogeneous spectrum in a double auction. Dong et al. [48] proposed a double auction based spectrum allocation algorithm that can achieve truthfulness, individual rationality, and budget-balance.
- *Combinatorial Auction:* Dong et al. [5] studied combinatorial auction in cognitive radio networks. Zheng et al. [49] further modeled the heterogeneous spectrum market as a combinatorial auction.
- *Online Auction:* Deek et al. [4] designed a truthful online spectrum auction mechanism. Li et al. [50] proposed an online spectrum allocation mechanism for secondary wireless communication, which can dynamically evaluate the true value of the spectrum channels and achieve sub-optimal social welfare. Hyder et al. [51] extended the online spectrum auction design to dynamic spectrum markets with varying transmission deadlines and random availability of spectrum units.
- *Revenue Maximization:* Al-Ayyoub and Gupta [3] proposed a truthful spectrum auction to maximize the total revenue with polynomial-time complexity.
- *Privacy Preserving Auction:* SPRING [7] is a strategy-proof and privacy preserving spectrum auction mechanism. DEAR [52] is a differentially private spectrum auction with approximate revenue maximization.
- *Collusion-Resistent Mechanism:* Gao and Wang [53] proposed a min-max coalition-proof Nash equilibrium channel allocation mechanism for multi-channel allocation in multi-hop wireless networks. THEMIS et al. [54] is a truthful and collusion-resistent online spectrum auction mechanism that provides price fairness under unknown and dynamic spectrum supply.
- *Other:* Gopinathan et al. [37] considered the balance between social welfare and fairness in spectrum markets. Li et al. [55] proposed an extensible and flexible truthful auction framework for heterogeneous spectrum market. ALETHEIA [56] is a large-scale strategy-proof spectrum auction mechanism that can prevent false-name bidding. Nadendla et al. [57] considered the problem of optimal spectrum auction

design under the scenarios where the spectrum availability is not always certain. Yang et al.[58] applied the group buying strategy into secondary spectrum market, and proposed two group buying auctions that can dramatically improve the utility of spectrum users.

A good survey of spectrum auctions can be found in [59]. However, all of the existing spectrum auction mechanisms are centralized, and may suffer from the critical drawbacks discussed in Section 1. In contrast, we consider the design of distributed spectrum auction mechanisms. A preliminary version of this work appears in INFOCOM 2015 [60], while this work has substantial revisions over the previous one, including the design of extended FAITH, additional technical details, and more comprehensive evaluations.

*Early Researches on DAMD.* To overcome the limitations of centralized mechanisms, Feigenbaum et al. [12] initiated the study of *distributed algorithmic mechanism design*, and pointed out two key aspects that DAMD differed from traditional centralized algorithmic mechanism design (AMD): agents' additional ways of manipulations and the measure of network complexity. To prevent the agents' manipulations in distributed implementation of VCG mechanism, they proposed the idea of *replication*, i.e., breaking the agents into two groups and letting each group compute its own version of the outcomes and payments. Then, a central *enforcer* will conduct consensus check and penalize all agents if the outcomes and payments do not agree. Feigenbaum and her colleges also articulated the concept of *network complexity*, and proposed efficient distributed mechanisms for multicast transmissions [61] and interdomain routing [62] without addressing agents' additional manipulations. Later, Parkes and Shneidman [23] proposed several general principles, such as partition principle, information-revelation principle, and redundancy principle, to guide the distribution of mechanisms, which shaped our faithfulness implementation of distributed VCG and FAITH. Shneidman and Parks also [13] studied the agents' strategy space in distributed scenarios, and introduced the notions of *communication compatibility* and *algorithm compatibility*. In [63] and [13], Shneidman and Parkes extended the interdomain routing mechanism proposed by [62] to address the agents' manipulations in communications and computations based on the idea of redundancy and "catch and punish" scheme. Different from the existing studies on distributed VCG mechanism, where the implementation of the outcome function was based on standard protocols, we focused on the design of a distributed spectrum allocation mechanism, and in particular, customized a distributed social welfare optimization algorithm that takes both the agents' multi-channel requests and the spatial reusability of the spectrum into consideration.

*Recent Applications of DAMD.* Yang et al. [64] considered the problem of stochastic data collection in mobile phone sensing systems, and proposed a distributed mechanism. However, they only considered the agents' manipulations in information-revelation actions and assumed that the agents are obedient in message-passing and computations. Mhanna et al. [65] considered the problem of sharing the cost of electricity among a large number of strategic agents, and proposed faithful distributed mechanisms to determine the price of each consumer. Similar to [61], they focused on the network complexity part without addressing the agents' additional ways of manipulations.

## 9 Conclusion

In this paper, we have modeled the problem of wireless spectrum allocation as a distributed auction, and have proposed two faithful distributed auction mechanisms, namely distributed VCG and FAITH. In addition, we have extended FAITH to adapt to dynamic scenarios where agents can come and go at any time. We have analyzed their economic properties and complexities, and implemented them in various settings. Our evaluation results well demonstrate the properties of distributed VCG and FAITH in terms of social welfare and transmission overhead. As for our future work, we are interested in designing similar distributed mechanisms that can prevent collusion among multiple agents.

## Acknowledgments

The authors would like to thank the anonymous reviewers for their efforts in improving the quality of the paper. This work was supported in part by the National the Key R&D Program of China under grant 2018YFB1004700, in part by China NSF grant 61672348, 61672353, and 61472252, and in part by Shanghai Science and Technology fund 15220721300. The opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies or the government.

## References

[1] R. Berry, M. L. Honig, and R. Vohra, "Spectrum markets: Motivation, challenges, and implications," *IEEE Commun. Mag.*, vol. 48, no. 11, pp. 146–155, Nov. 2010.

[2] J. Huang, R. A. Berry, and M. L. Honig, "Auction-based spectrum sharing," *Mobile Netw. Appl.*, vol. 11, no. 3, pp. 405–418, 2006.

[3] M. Al-Ayyoub and H. Gupta, "Truthful spectrum auctions with approximate revenue," in *Proc. 30th IEEE Int. Conf. Comput. Commun.*, 2011.

[4] L. Deek, X. Zhou, K. Almeroth, and H. Zheng, "To preempt or not: Tackling bid and time-based cheating in online spectrum auctions," in *Proc. 30th IEEE Int. Conf. Comput. Commun.*, 2011, pp. 2219–2227.

[5] M. Dong, G. Sun, X. Wang, and Q. Zhang, "Combinatorial auction with time-frequency flexibility in cognitive radio networks," in *Proc. 31st IEEE Int. Conf. Comput. Commun.*, 2012, pp. 2282–2290.

[6] X. Feng, Y. Chen, J. Zhang, Q. Zhang, and B. Li, "TAHES: Truthful double auction for heterogeneous spectrums," in *Proc. 31st IEEE Int. Conf. Comput. Commun.*, 2012, pp. 3076–3080.

[7] Q. Huang, Y. Tao, and F. Wu, "Spring: A strategy-proof and privacy preserving spectrum auction mechanism," in *Proc. 32nd IEEE Int. Conf. Comput. Commun.*, 2013, pp. 827–835.

[8] F. Wu and N. Vaidya, "SMALL: A strategy-proof mechanism for radio spectrum allocation," in *Proc. 30th IEEE Int. Conf. Comput. Commun.*, 2011, pp. 81–85.

[9] X. Zhou, S. Gandhi, S. Suri, and H. Zheng, "eBay in the sky: Strategy-proof wireless spectrum auctions," in *Proc. 14th Annu. Int. Conf. Mobile Comput. Netw.*, 2008, pp. 2–13.

[10] X. Zhou, Z. Zhang, G. Wang, X. Yu, B. Y. Zhao, and H. Zheng, "Practical conflict graphs for dynamic spectrum distribution," in *Proc. ACM Int. Conf. Meas. Model. Comput. Syst.*, 2013, pp. 5–16.

[11] X. Zhou and H. Zheng, "TRUST: A general framework for truthful double spectrum auctions," in *Proc. 28th IEEE Int. Conf. Comput. Commun.*, 2009, pp. 999–1007.

[12] J. Feigenbaum, M. Schapira, and S. Shenker, *Distributed Algorithmic Mechanism Design*. Cambridge, U.K.: Cambridge Univ. Press, 2007, pp. 363–384.

[13] J. Shneidman and D. C. Parkes, "Specification faithfulness in networks with rational nodes," in *Proc. 23rd Annu. ACM Symp. Principles Distrib. Comput.*, 2004, pp. 88–97.

[14] E. H. Clarke, "Multipart pricing of public goods," *Public Choice*, vol. 11, no. 1, pp. 17–33, 1971.

[15] T. Groves, "Incentives in teams," *Econometrica: J. Econometric Soc.*, vol. 41, pp. 617–631, 1973.

[16] W. Vickrey, "Counterspeculation, auctions, and competitive sealed tenders," *J. Finance*, vol. 16, no. 1, pp. 8–37, 1961.

[17] M. M. Halldórsson, J. Y. Halpern, L. E. Li, and V. S. Mirrokni, "On spectrum sharing games," in *Proc. 23rd Annu. ACM Symp. Principles Distrib. Comput.*, 2004, pp. 107–114.

[18] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *Proc. 9th Annu. Int. Conf. Mobile Comput. Netw.*, 2003, pp. 66–80.

[19] W. Wang, S. Eidenbenz, Y. Wang, and X.-Y. Li, "OURS: optimal unicast routing systems in non-cooperative wireless networks," in *Proc. 12th Annu. Int. Conf. Mobile Comput. Netw.*, 2006, pp. 402–413

[20] S. Zhong, J. Chen, and Y. R. Yang, "Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks," in *Proc. 22nd IEEE Int. Conf. Comput. Commun.*, 2003, pp. 1987–1997.

[21] S. Zhong, L. E. Li, Y. G. Liu, and Y. R. Yang, "On designing incentive-compatible routing and forwarding protocols in wireless ad-hoc networks," *Wireless Netw.*, vol. 13, no. 6, pp. 799–816, 2007.

[22] S. Zhong and F. Wu, "On designing collusion-resistant routing schemes for non-cooperative wireless ad hoc networks," in *Proc. 13th Annu. Int. Conf. Mobile Comput. Netw.*, 2007, pp. 278–289.

[23] D. C. Parkes and J. Shneidman, "Distributed implementations of vickrey-clarke-groves mechanisms," in *Proc. 3rd Int. Conf. Autonomous Agents Multiagent Syst.*, 2004, pp. 261–268.

[24] R. B. Myerson, *Game Theory*. Cambridge, MA, USA: Harvard Univ. Press, 2013.

[25] A. Petcu and B. Faltings, "A scalable method for multiagent constraint optimization," in *Proc. 19th Int. Joint Conf. Artif. Intell.*, 2005, pp. 266–271.

[26] F. Fioretto, E. Pontelli, and W. Yeoh, "Distributed constraint optimization problems and applications: A survey," *J. Artif. Intell. Res.*, vol. 61, pp. 623–698, 2018.

[27] J. Larrosa, P. Meseguer, and M. Sanchez, "Pseudo-tree search with soft constraints," in *Proc. 15th Eur. Conf. Artif. Intell.*, 2002, pp. 131–135.

[28] R. Dechter, *Constraint Processing*. San Mateo, CA, USA: Morgan Kaufmann, 2003.

[29] E. C. Freuder, "A sufficient condition for backtrack-bounded search," *J. ACM*, vol. 32, no. 4, pp. 755–761, 1985.

[30] V. C. Barbosa, *An Introduction to Distributed Algorithms*. Cambridge, MA, USA: MIT Press, 1996.

[31] S. Makki and G. Havas, "Distributed algorithms for constructing a depth-first-search tree," in *Proc. Int. Conf. Parallel Process.*, 1994, pp. 270–273.

[32] R. B. Myerson, "Optimal auction design," *Math. Operations Res.*, vol. 6, no. 1, pp. 58–73, 1981.

[33] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic Game Theory*. Cambridge, U.K.: Cambridge Univ. Press, 2007.

[34] E. Arjomandi, M. J. Fischer, and N. A. Lynch, "Efficiency of synchronous versus asynchronous distributed systems," *J. ACM*, vol. 30, no. 3, pp. 449–456, 1983.

[35] M. Felegyhazi, M. Cagalj, S. S. Bidokhti, and J.-P. Hubaux, "Non-cooperative multi-radio channel allocation in wireless networks," in *Proc. 26th IEEE Int. Conf. Comput. Commun.*, 2007, pp. 1442–1450.

[36] W. Zhang, G. Wang, Z. Xing, and L. Wittenburg, "Distributed stochastic search and distributed breakout: Properties, comparison and applications to constraint optimization problems in sensor networks," *Artif. Intell.*, vol. 161, no. 1/2, pp. 55–87, 2005.

[37] A. Gopinathan and Z. Li, "Strategyproof auctions for balancing social welfare and fairness in secondary spectrum markets," in *Proc. IEEE Conf. Comput. Commun.*, 2011, pp. 3020–3028.

[38] I. Katzela and M. Naghshineh, "Channel assignment schemes for cellular mobile telecommunication systems: A comprehensive survey," *IEEE Personal Commun.*, vol. 3, no. 3, pp. 10–31, Jun. 1996.

[39] P.-Y. Chen, W. C. Ao, S.-C. Lin, and K.-C. Chen, "Reciprocal spectrum sharing game and mechanism in cellular systems with cognitive radio users," in *Proc. IEEE GLOBECOM Workshops*, 2011, pp. 981–985.

[40] A. Mishra, S. Banerjee, and W. Arbaugh, "Weighted coloring based channel assignment for wlans," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 9, no. 3, pp. 19–31, 2005.

[41] A. Raniwala, K. Gopalan, and T. C. Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 8, no. 2, pp. 50–65, 2004.

[42] M. Doraghinejad, H. Nezamabadi-Pour, and A. Mahani, "Channel assignment in multi-radio wireless mesh networks using an improved gravitational search algorithm," *J. Netw. Comput. Appl*, vol. 38, pp. 163–171, 2014.

[43] L. Cao and H. Zheng, "Distributed spectrum allocation via local bargaining," in *Proc. 2nd IEEE Conf. Sensor Mesh Ad Hoc Commun. Netw.*, 2005, pp. 475–486.

[44] N. U. Hasan, W. Ejaz, N. Ejaz, H. S. Kim, A. Anpalagan, and M. Jo, "Network selection and channel allocation for spectrum sharing in 5g heterogeneous networks," *IEEE Access*, vol. 4, pp. 980–992, 2016.

[45] L. Lei, D. Yuan, C. K. Ho, and S. Sun, "Power and channel allocation for non-orthogonal multiple access in 5g systems: Tractability and computation," *IEEE Trans. Wireless Commun.*, vol. 15, no. 12, pp. 8580–8594, Dec. 2016.

[46] B. Zhuang, D. Guo, and M. L. Honig, "Energy-efficient cell activation, user association, and spectrum allocation in heterogeneous networks," *IEEE J. Select. Areas Commun.*, vol. 34, no. 4, pp. 823–831, Apr. 2016.

[47] L. Sun, H. Shan, A. Huang, L. Cai, and H. He, "Channel allocation for adaptive video streaming in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 734–747, Jan. 2017.

[48] W. Dong, S. Rallapalli, L. Qiu, K. Ramakrishnan, and Y. Zhang, "Double auctions for dynamic spectrum allocation," *IEEE/ACM Trans. Netw.*, vol. 24, no. 4, pp. 2485–2497, Aug. 2016.

[49] Z. Zheng, F. Wu, S. Tang, and G. Chen, "Unknown combinatorial auction mechanisms for heterogeneous spectrum redistribution," in *Proc. 15th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2014, pp. 3–12.

[50] H. Li, C. Wu, and Z. Li, "Socially-optimal online spectrum auctions for secondary wireless communication," in *Proc. IEEE Conf. Comput. Commun.*, 2015, pp. 2047–2055.

[51] C. S. Hyder, T. D. Jeitschko, and L. Xiao, "Bid and time truthful online auctions in dynamic spectrum markets," *IEEE Trans. Cognitive Commun. Netw.*, vol. 3, no. 1, pp. 82–96, Mar. 2017.

[52] R. Zhu, Z. Li, F. Wu, K. Shin, and G. Chen, "Differentially private spectrum auction with approximate revenue maximization," in *Proc. 15th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2014, pp. 185–194.

[53] L. Gao and X. Wang, "A game approach for multi-channel allocation in multi-hop wireless networks," in *Proc. 9th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2008, pp. 303–312.

[54] Q. Wang, Q. Sun, K. Ren, and X. Jia, "Themis: Collusion-resistant and fair pricing spectrum auction under dynamic supply," *IEEE Trans. Mobile Comput.*, vol. 16, no. 7, pp. 2051–2064, Jul. 2017.

[55] W. Li, X. Cheng, R. Bie, and F. Zhao, "An extensible and flexible truthful auction framework for heterogeneous spectrum markets," in *Proc. 15th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2014, pp. 175–184.

[56] Q. Wang, B. Ye, B. Tang, T. Xu, S. Guo, S. Lu, and W. Zhuang, "Aletheia: Robust large-scale spectrum auctions against false-name bids," in *Proc. 16th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2015, pp. 27–36.

[57] V. S. S. Nadendla, S. K. Brahma, and P. K. Varshney, "Optimal spectrum auction design with 2-D truthful revelations under uncertain spectrum availability," *IEEE/ACM Trans. Netw.*, vol. 25, no. 1, pp. 420–433, Feb. 2017.

[58] D. Yang, G. Xue, and X. Zhang, "Group buying spectrum auctions in cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 810–817, Jan. 2017.

[59] Y. Zhang, C. Lee, D. Niyato, and P. Wang, "Auction approaches for resource allocation in wireless systems: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1020–1041, Jul.-Sep. 2013.

[60] D. Peng, S. Yang, F. Wu, G. Chen, S. Tang, and T. Luo, "Resisting three-dimensional manipulations in distributed wireless spectrum auctions," in *Proc. IEEE Conf. Comput. Commun.*, 2015, pp. 2056–2064.

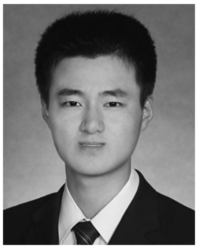
[61] J. Feigenbaum, C. H. Papadimitriou, and S. Shenker, "Sharing the cost of multicast transmissions," *J. Comput. Syst. Sci.*, vol. 63, pp. 21–41, 2001.

[62] J. Feigenbaum, C. Papadimitriou, R. Sami, and S. Shenker, "A BGP-based mechanism for lowest-cost routing," *Distrib. Comput.*, vol. 18, no. 1, pp. 61–72, 2005.

[63] J. Shneidman and D. C. Parkes, "Using redundancy to improve robustness of distributed mechanism implementations," in *Proc. 4th ACM Conf. Electron. Commerce*, 2003, pp. 276–277.

[64] S. Yang, U. Adeel, and J. McCann, "Backpressure meets taxes: Faithful data collection in stochastic mobile phone sensing systems," in *Proc. IEEE Conf. Comput. Commun.*, 2015, pp. 1490–1498.

[65] S. Mhanna, G. Verbič, and A. C. Chapman, "A faithful distributed mechanism for demand response aggregation," *IEEE Trans Smart Grid*, vol. 7, no. 3, pp. 1743–1753, May 2016.



**Shuo Yang** received the BS degree in computer science from Shanghai Jiao Tong University in 2014. He is working toward the PhD degree in the Department of Computer Science and Enginerring, Shanghai Jiao Tong University. He is a recipient of the 2017 Google PhD Fellowship. His research interests include mobile computing, machine learning, wireless networking, and algorithmic mechanism design. He is a student member of the ACM and CCF.

**Dan Peng** received the BS degree from the Huazhong University of Science & Technology, in 2012, and the PhD degree from Shanghai Jiao Tong University, in 2018, both in computer science and engineering. Her research interests include algorithmic game theory, wireless networking, and mobile computing.

**Tong Meng** received the BS degree in computer science from Shanghai Jiao Tong University, in 2013, and the MS degree in computer science from Shanghai Jiao Tong University, in 2016. His research interests encompass neighbor discovery, routing in wireless networks, and mobile social networks. He is a student member of the IEEE, ACM, and CCF.

**Fan Wu** received the BS degree in computer science from Nanjing University, in 2004, and the PhD degree in computer science and engineering from the State University of New York at Buffalo, in 2009. He is a professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University. He has visited the University of Illinois at Urbana-Champaign (UIUC) as a post doc research associate. His research interests include wireless networking and mobile computing, algorithmic game theory and its applications, and privacy preservation. He has published more than 170 peer-reviewed papers in technical journals and conference proceedings. He is a recipient of the first class prize for the Natural Science Award of China Ministry of Education, NSFC Excellent Young Scholars Program, ACM China Rising Star Award, CCF-Tencent "Rhinoceros bird" Outstanding Award, CCF-Intel Young Faculty Researcher Program Award, Pujiang Scholar, and Tang Scholar. He has served as the chair of CCF YOCSEF Shanghai, on the editorial board of *Elsevier Computer Communications*, and as the member of technical program committees of more than 60 academic conferences. For more information, please visit http://www.cs.sjtu.edu.cn/**fwu/. He is a member of the IEEE.

**Guihai Chen** received the BS degree from Nanjing University, in 1984, the ME degree from Southeast University, in 1987, and the PhD degree from the University of Hong Kong, in 1997. He is a distinguished professor with Shanghai Jiao Tong University, China. He had been invited as a visiting professor by many universities including the Kyushu Institute of Technology, Japan, in 1998, the University of Queensland, Australia, in 2000, and Wayne State University, during September 2001 to August 2003. He has a wide range of research interests with a focus on sensor networks, peer-to-peer computing, high-performance computer architecture, and combinatorics. He has published more than 250 peer-reviewed papers, and more than 170 of them are in well-archived international journals such as the *IEEE Transactions on Parallel and Distributed Systems*, the *Journal of Parallel and Distributed Computing*, *Wireless Networks*, *The Computer Journal*, the *International Journal of Foundations of Computer Science*, and *Performance Evaluation*, and also in well-known conference proceedings such as HPCA, MOBIHOC, INFOCOM, ICNP, ICPP, IPDPS, and ICDCS. He is a senior member of the IEEE.

**Shaojie Tang** received the PhD degree in computer science from the Illinois Institute of Technology, in 2012. He is currently an assistant professor with the Naveen Jindal School of Management, University of Texas at Dallas. His research interest includes social networks, mobile commerce, game theory, e-business, and optimization. He received the Best Paper Awards in ACM MobiHoc 2014 and IEEE MASS 2013. He also received the ACM SIGMobile service award in 2014. He served in various positions (as chairs and TPC members) at numerous conferences, including ACM MobiHoc and IEEE ICNP. He is an editor for *Elsevier Information Processing in the Agriculture* and the *International Journal of Distributed Sensor Networks*.

**Zhenhua Li** received the BSc and MSc degrees from Nanjing University, in 2005 and 2008, and the PhD degree from Peking University, in 2013, all in computer science and technology. He is an assistant professor with the School of Software, Tsinghua University. His research areas cover cloud computing/storage/download, big data analysis, content distribution, and mobile Internet.

**Tie Luo** received the PhD degree in electrical and computer engineering from the National University of Singapore. He is a programme lead and research scientist with the Institute for Infocomm Research (I2R), A*STAR, Singapore. His research interests are Internet of Things analytics with machine learning, trust management in IoT, incentives, and trust in mobile crowd sensing. He was a Best Paper Award nominee of IEEE INFOCOM 2015, Best Paper Award recipient of ICTC 2012, Best Student Paper Award recipient of AAIM 2018, and a tutorial presenter of IEEE ICC 2016. His research work on mobile crowdsourcing was featured by the *IEEE Spectrum* magazine in 2016. He is a member of the IEEE.