Conjecture-Based Load Balancing for Delay-Sensitive Users Without Message Exchanges

Hsien-Po Shiang and Mihaela van der Schaar, *Fellow, IEEE*

***Abstract*—In this paper, we study how multiple users can balance their traffic loads to share common resources in an efficient and distributed manner, without message exchanges. Specifically, we study a deployment scenario where users deploy delay-sensitive applications over a wireless multipath network and aim to minimize their own expected delays. Since the performance of a user's load balancing strategy depends on the strategies that are deployed by other users, it becomes important that a user considers the multiuser coupling when making its own load balancing decisions. We model this multiuser interaction as a load balancing game (LBG) and show that users can converge to a ε-consistent conjectural equilibrium by building near-accurate beliefs about the remaining capacities on each path. Based on these beliefs, users can make load balancing decisions without explicitly knowing the actions of the other users. In such a conjecture-based LBG, we analytically show that, if a leader is elected to build beliefs about how the users' aggregate transmission strategies affect the remaining resources, then this leader can use this knowledge to shape its traffic such that the multiuser interaction can achieve an efficient allocation across paths. Even if no leader is present in the game, as long as the users follow a set of prescribed rules for building beliefs, they can reach efficient outcomes in a distributed manner. Importantly, the proposed distributed load balancing solution can be also applied to other multiuser communication and networking problems where message exchanges are prohibited (or prohibitively expensive in terms of delay or bandwidth), ranging from multichannel selection in wireless networks to relay assignment in multivehicle networks.**

***Index Terms*—Conjectural equilibrium (CE), efficient resource management without message exchanges, load balancing.**

I. Introduction

LOAD BALANCING is a technique for distributing traffic across multiple resources of a communication system. In this paper, we study how multiple self-interested users can optimally distribute their traffic loads in an autonomous manner to minimize their individual delays of transmitting their packets through a multipath wireless network. Load balancing has been investigated in various multiuser transmission scenarios where users (nodes/transceivers) are obedient. Considered deployment scenarios included multipath routing in wired or wireless networks [1]–[6], nonoverlapping spectrum sharing in cognitive radio networks [7]–[10], or load sharing in multiprocessor systems [12].

In wireless networks, load balancing was studied to perform channel selection in cellular networks. Various channel assignment schemes have been proposed (see, e.g., [11] for an excellent survey). However, most of these channel assignment schemes are based on centralized solutions, which do not easily scale as the network size increases and/or are not suitable for wireless networks without a fixed infrastructure, such as ad hoc wireless networks. Moreover, centralized approaches are particularly not desirable for delay-sensitive applications such as the ones considered in this paper because they require exchanging control messages back and forth to a network coordinator, thereby incurring unacceptable delays for delay-sensitive applications [10]. To cope with these challenges, distributed schemes without a network manager have also been proposed in various types of wireless networks, such as ad hoc networks [2]–[6] or cognitive radio networks [7]–[10].

A. Related Works

In wireless ad hoc networks, Pham and Perreau [2] proposed a multipath routing protocol with load balancing by explicitly taking into account the congestion conditions over each network path. Zhang *et al.* [4] proposed a load balancing solution over multipath routing using weighted round-robin strategy based on measured roundtrip time. Jain *et al.* [6] proposed a multichannel carrier-sense multiple-access protocol that identifies the set of idle channels and selects the best channel for transmission based on the channel condition that is observed at the transmitter side. However, a common limitation of these solutions is that they are myopic, because the autonomous users only adapt to their latest network measurement (e.g., idle channel set, channel condition, or path congestion) and they do not predict the impact of their transmission actions on their long-term performance (utility). Since the individual users only react to the latest contention measurements that are experienced in the different wireless channels, the resulting multiuser interaction is often inefficient.

In emerging cognitive radio networks, Zheng and Cao [8] provided five rule-based spectrum management schemes where

Manuscript received April 20, 2012; revised October 19, 2012 and March 4, 2013; accepted April 14, 2013. Date of publication April 25, 2013; date of current version October 12, 2013. This work was supported by the National Science Foundation (NFS) under Grant NFS CCF 0830556. The review of this paper was coordinated by Prof. F. R. Yu.

H.-P. Shiang is with the Cisco Systems, Inc., San Jose, CA 95134 USA (e-mail: hshiang@cisco.com).

M. van der Schaar is with the Department of Electrical Engineering, University of California, Los Angeles, CA 90095-1594 USA (e-mail: mihaela@ee.ucla.edu).

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

Digital Object Identifier 10.1109/TVT.2013.2260188

users measure local interference patterns in wireless channels and independently act according to the prescribed rules. Huang *et al.* [9] proposed a resource sharing scheme where users can select multiple channels to transmit packets and exchange interference prices for each channel. Our previous work [10] proposed a distributed resource management solution where users learn the interference/congestion online, by using multiagent learning techniques based on fictitious play, and based on this knowledge, they balance their traffic loads across several shared channels and relays in a multihop cognitive radio network. All these distributed schemes assume that users cooperate to efficiently coordinate their load balancing strategies. However, as discussed in, for example, [16], individual users can decide to deviate from the rules that are prescribed by the protocols, as long as they derive a higher utility when deviating. Thus, self-interested users in the network may not have incentives to cooperate and maximize a network/system performance, because this does not necessarily maximize their own utilities.

To capture the behavior of these self-interested users, noncooperative games were proposed to characterize and analyze the performance of self-interested users interacting in different communication systems. For example, Lee *et al.* [32] showed that the current backoff-based medium-access-control protocols can be modeled as a noncooperative channel-access game. The noncooperative channel selection game was studied by Felegyhazi *et al.* [33], who showed that users autonomously selecting channels in multichannel wireless networks converge to a Nash equilibrium (NE). Similarly, multiuser transmission over multipath selection has been formalized and analyzed as a noncooperative game in [18]. However, it is well known that the NE can often be Pareto inefficient. For instance, it is possible that some of the selfish users will improve their performance at the cost of degrading the system-wide performance.

At the other spectrum of the existing multiuser networking research, a network utility maximization framework has been introduced in [31] to optimize the social welfare of a multiuser communication system. It has been shown that, by allowing users to exchange messages, they can determine a wireless channel-access strategy that reaches a Pareto-efficient solution in a distributed manner. Similar concepts have been proposed in [34] for distributed channel selection, where pricing has been deployed to get users to maximize the system throughput in a distributed manner. To determine the resource price, message exchanges among users are necessary. However, such message exchanges among users can be undesirable due to their increased computational and communication overhead or simply due to security issues, protocol limitations, etc. Moreover, the incentives for the users to add a penalty term in their utility functions that enables collaboration are not addressed. Alternatively, a distributed channel-access scheme using simple random-access algorithms without message exchanges was discussed in [24]. However, this solution can only achieve a near-optimal system-wise throughput if there are no message exchanges among the participating users.

In summary, existing centralized load balancing approaches [11], [13] provide efficient allocations, but they require extensive control information to be gathered by a central coordinator. Hence, such centralized approaches cannot be successfully deployed in distributed networks, where the participating users cannot exchange voluminous messages to a central network coordinator due to the resulting message overheads and the delay incurred from propagating messages back and forth to a central coordinator. On the other hand, distributed load balancing approaches [2]–[10], [15], [16] do not require message exchanges, but they often lead to inefficient allocations. Since users often respond in a self-interested and myopic manner to the measured local congestion in the network, these distributed approaches often result in a suboptimal solution from the users' or the communication system's perspective.

In this paper, we study an autonomous load balancing approach, which does not require any message exchanges but leads to a Pareto-efficient solution by enabling autonomous users to predict the implications of their load distribution on their expected future costs (delays in this paper) and thereby influence the multiuser interaction. We model the multiuser interaction as a load balancing game (LBG) that is played by users that are making conjectures about how their load distribution actions will impact other users and their responses and thus eventually impact their future performance. We endow the users with the ability to build beliefs about the aggregate response of the other users to their actions (in this paper, the aggregate response is the remaining capacity measured for each path using, for example, the bandwidth estimation method in [29]) and efficiently minimize their expected transmission delays. Specifically, we investigate the performance of the resulting ε-consistent conjectural equilibrium (ε-CE) in the LBG, which is a relaxed version of the conventional conjectural equilibrium (CE) [21] that allows us to characterize the equilibrium that is obtained when network users are able to build near-accurate conjectures. At equilibrium, the autonomous users will dynamically select the paths over which they should distribute their traffic in a distributed manner by estimating their expected utilities obtained from taking various transmission actions based on their near-accurate conjectures about the communication system.

B. Contributions and Organization of the Paper

Compared with the conventional distributed approach, we discuss two new concepts that enable the network users to minimize delays in distributed communication networks, without the need of message exchanges with other users.

1) *Active load balancing strategies.* As previously mentioned, the users' strategies are coupled in multiuser multipath networking environments because the load balancing decision of each user impacts and is impacted by the other users. Thus, users need to distribute their traffic loads by considering not only the impact of their actions on their immediate experienced utilities but also on their long-term utilities. For instance, a user's aggressive strategy may be rewarded in the short term, but this may trigger the other users to adapt their own strategies, which brings a negative impact to its long-term reward. Hence, active learners can build accurate models (conjectures) about how their actions are coupled with that of the other

users and, based on these models, make conjecture-based decisions on how to adapt their transmission strategies in real time. These learners are referred to as conjecturing learners in this paper.

2) *Learning accurate coupling models based on local information*. To build these coupling models, the conjecturing learners can adopt interactive learning approaches to update their beliefs about the expected response of the other users to their actions. Specifically, we propose learning approaches based on which the conjecturing learners can build their beliefs in a distributed manner, given only their local information (i.e., their own measurement history).

The goal of this paper is to develop belief formation techniques that allow the users to coordinate to reach efficient solutions, without message exchanges. We provide specific belief formation methods and conjecture-based load balancing strategies for the following two extreme communication scenarios: 1) when the system has only one conjecturing learner (e.g., an elected leader) and 2) when all users in the system are conjecturing learners. We are able to analytically show that, when the system has only one conjecturing learner, this user can deploy a linear belief function to model the aggregate response of the other users. We show that, when the leader is altruistic (e.g., it minimizes a system-wide utility), it can drive the system to a system-wide efficient solution by modeling the reactions of the other users. Alternatively, when the leader is self-interested (e.g., it minimizes its own delay), we show that this user will benefit itself at the expense of (some of) the other users' increased delays. If the system has multiple conjecturing learners that are simultaneously building beliefs, the simple linear belief formation becomes insufficient to capture the other users' behaviors. Therefore, to enable these conjecturing learners to build consistent beliefs at a low cost, the protocol designer prescribes for them a set of interaction rules. We then show how, if all the autonomous users in the network comply with the rules, the system reaches a Pareto-efficient resource allocation without exchanging messages among users.

This paper is organized as follows: Section II models the considered multiuser multipath network and formulates the conjecture-based load balancing problem for autonomous delay-sensitive users. We also define the conjecture-based load balancing game and the ε-CE of the game. In Section III, we investigate the case when there is only one conjecturing learner in the network. We provide a learning procedure for the conjecturing learner to update its belief. In Section IV, we present solutions for the case when all the users comply with the prescribed rules. The simulation results are shown in Section V. Section VI concludes this paper.

II. Load Balancing Problem Formulation

A. Network Model

We assume $\mathbf{V} = \{v_i, i = 1, \ldots, M\}$ as the set of M autonomous users sharing the same wireless multipath network. User v_i is composed of a source–destination pair, i.e., $v_i = (v_i^s, v_i^d)$, and each user has a delay-sensitive application with traffic rate λ_i (packet/second). We assume a wireless network

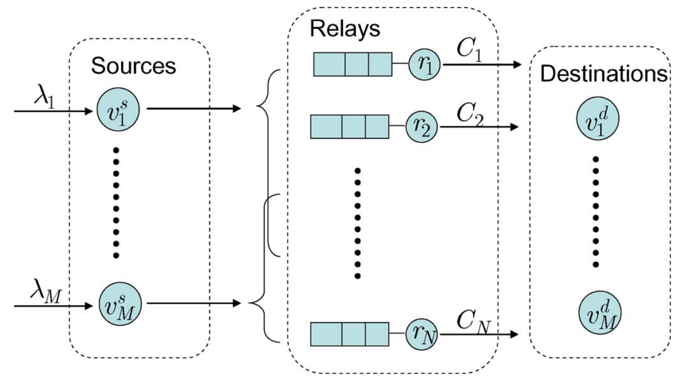

Fig. 1. Considered network model for multiuser multipath networks.

with N distinct relays from the sources to the destinations. Each relay can represent a mobile vehicle in the multivehicle relay network. We denote $\mathbf{r} = \{r_j, j = 1, \ldots, N\}$ as the set of these relays. Each relay r_j is associated with capacity C_j (packet/second)[1] representing how fast the relay can process/transmit the passing data. The multiuser multipath network model is shown in Fig. 1. Note that the relays in the multipath network abstract the limited network resources, which can represent not only bottlenecks in a multipath network but, for example, nonoverlapping frequency channels in a wireless network or parallel processors in a multiprocessor system as well.

The autonomous users aim to balance their traffic loads over the N shared relays such that the end-to-end delay for transmitting their applications are minimized. The traffic rate from user v_i through relay r_j is denoted as λ_{ij} (packet/second). Let λ_i represent the total traffic rate from user v_i. We denote $\boldsymbol{\sigma}_i = [\lambda_{ij}, j = 1, \ldots, N] \in \mathcal{X}_i$ as the traffic distribution of user v_i, and $\boldsymbol{\sigma}_{-i}$ as the traffic distribution for the other users except v_i ($\boldsymbol{\sigma} = [\boldsymbol{\sigma}_i, \boldsymbol{\sigma}_{-i}]$). $\mathcal{X}_i$ denotes all possible traffic distribution of user v_i, where $\sum_{j=1}^{N} \lambda_{ij} = \lambda_i$. We assume unsaturated network conditions, in which the total system capacity is more than the total traffic rate of the users, i.e., $\sum_{j=1}^{N} C_j > \sum_{i=1}^{M} \lambda_i$. Such unsaturated conditions can ensure that a user can always find an unsaturated relay to transmit its traffic, and hence, the delays of the applications are bounded. We assume that the expected delay through relay r_j can be modeled using an M/M/1 queuing model $E[D_j] = (C_j - \sum_{i=1}^{M} \lambda_{ij})^{-1}$, in which each path is modeled as a queue with the exponential service time and the Poisson arrival process [28]. Let U_{ij} represent the average delay when user v_i sends packets through r_j. The average end-to-end delay of user v_i is defined as

$$U_i(\sigma_i, \sigma_{-i}) = \sum_{j=1}^{N} \frac{\lambda_{ij}}{\lambda_i} U_{ij} = \frac{1}{\lambda_i} \sum_{j=1}^{N} \frac{\lambda_{ij}}{C_j - \lambda_j} \tag{1}$$

where $U_{ij} = E[D_j]$, and $\lambda_j \triangleq \sum_{i=1}^{M} \lambda_{ij}$ represents the total traffic loads that pass through relay r_j. Let $\mathbf{U}_i^t = \{U_{ij}^t, j = 1, \ldots, N\}$ denote the average delays that user v_i experiences over the paths at time t.

[1] For simplicity, we assume that the capacity of the relay is not changing over time nor changing for different users. However, the analysis provided in this paper can be generalized to the case when each relay has different capacities for different users by adopting a more sophisticated queuing model.

B. Centralized Coordination With Global Information

In general, centralized methods aim at implementing Pareto-efficient solutions, which optimize the "system welfare," e.g., they minimize the weighted summation of users' utilities, i.e., $U(\boldsymbol{\sigma}) = \sum_{i=1}^{M} w_i U_i(\boldsymbol{\sigma})$, where w_i represents the weighting parameters.

Definition 1—Pareto Boundary: Given different users' weights $\mathbf{w} = [w_i, i = 1, \ldots, M | w_i > 0, \sum_{i=1}^{M} w_i = 1]$, points on the Pareto boundary are formed by the solutions of the following multiuser multipath selection problem:

$$\boldsymbol{\sigma}^P(\mathbf{w}) = \arg \min_{\sigma_i \in_i, \forall v_i} \sum_{i=1}^{M} w_i U_i(\boldsymbol{\sigma}). \tag{2}$$

To perform the aforementioned centralized optimization, the network coordinator needs to determine weights $\{w_i, \forall v_i\}$ and collect the global network information $\mathcal{I}_g = [\{C_j, \forall r_j\}, \{\lambda_i, \forall v_i\}]$. Specifically, in this paper, we define the system-wide utility as $U^{\text{sys}}(\boldsymbol{\sigma}) = \sum_{i=1}^{M} \lambda_i U_i(\boldsymbol{\sigma}) = \sum_{j=1}^{N} (\sum_{i=1}^{M} \lambda_{ij}/C_j - \sum_{i=1}^{M} \lambda_{ij})$ (equivalent to the case using weights $\{w_i = (\lambda_i / \sum_{i=1}^{M} \lambda_i), \forall v_i\}$). Based on Little's formula [28], this utility represents the total queue size of these N M/M/1 queues for the N paths.

Definition 2—System-Wide Optimal Solution: The system-wide Pareto optimal (PO) solution is then defined as

$$\boldsymbol{\sigma}^P = \arg \min_{\sigma_i \in \mathcal{X}_i, \forall v_i} U^{\text{sys}}(\boldsymbol{\sigma}). \tag{3}$$

The system-wide optimal solution is PO where the users' weights are proportional to the traffic rates of the users. However, such a centralized approach may be undesirable in many delay-sensitive settings due to the message overhead required for exchanging the global network information. This motivates the adoption of distributed approaches.

C. Distributed Best Response

Without a centralized coordinator, the users can minimize their own delays, i.e., user v_i performs the following best response:

$$\sigma_i = \arg \min_{\boldsymbol{\sigma}_i \in \mathcal{X}_i} U_i(\boldsymbol{\sigma}_i, \boldsymbol{\sigma}_{-i}). \tag{4}$$

As indicated by (4), when a user performs the best response, the user requires knowledge about the other users' actions, i.e., the required information is still $\mathcal{I}_g = [\{C_j, \forall r_j\}, \{\lambda_i, \forall v_i\}]$. Such knowledge is usually acquired via message exchanges among the users. Applying the best response in the multiuser interaction, the NE $\boldsymbol{\sigma}^{NE} = \{\boldsymbol{\sigma}_i^{NE}, \forall v_i\}$ is defined by the well-known inequality $U_i(\boldsymbol{\sigma}_i^{NE}, \boldsymbol{\sigma}_{-i}^{NE}) \leq U_i(\boldsymbol{\sigma}_i, \boldsymbol{\sigma}_{-i}^{NE}), \forall \boldsymbol{\sigma}_i \in \mathcal{X}_i, \forall v_i$.

D. Distributed Decision Making Without Message Exchange

Without explicit message exchanges among users, user v_i does not know $\boldsymbol{\sigma}_{-i}^t$ when making decision on $\boldsymbol{\sigma}_i^t$ at time t. In other words, the user cannot know the exact average delay $U_i^t(\boldsymbol{\sigma}_i^t, \boldsymbol{\sigma}_{-i}^t)$ when making the decision at time t. However, the user is aware of the action-delay history in the past, i.e., $\{(\boldsymbol{\sigma}_i^1, \mathbf{U}_i^1), \ldots, (\boldsymbol{\sigma}_i^{t-1}, \mathbf{U}_i^{t-1})\}$ is known. For each time slot in the past, user v_i can infer the congestion that it experienced based on the action-delay history. The congestion is defined by $\sum_{i' \neq i} \lambda_{i'j}^k$ at time slot $k = 0, \ldots, t-1$, which is the aggregate load of the users other than v_i at a particular relay r_j. We refer to $C_{ij}^k \triangleq C_j - \sum_{i' \neq i} \lambda_{i'j}^k$ as the remaining capacity for time slot $k = 1, \ldots, t-1$. From (1), the remaining capacities in the past can be inferred by user v_i as $C_{ij}^k = (U_{ij}^k)^{-1} + \lambda_{ij}^k$, with $k = 1, \ldots, t-1$. Based on these, we define the congestion information history of user v_i at time t as

$$h_i^t = \left\{ \left(\lambda_{ij}^{t-S}, C_{ij}^{t-S}\right) \ldots \left(\lambda_{ij}^{t-1}, C_{ij}^{t-1}\right), j = 1, \ldots N \right\} \tag{5}$$

where $S < t$ represents the length of an observation window.

Although user v_i does not know $\boldsymbol{\sigma}_{-i}^t$ when making the decision, it can build a model (before making a decision) to conjecture the remaining capacity over each relay at time t based on the congestion information history h_i^t. We denote $\mathbf{B}_i^t = \{\tilde{C}_{ij}(h_i^t), j = 1, \ldots, N\} \in (\mathcal{B}_i)^N$ as the set of conjectured remaining capacities, where $\mathcal{B}_i$ represents a set of all possible conjectures $\tilde{C}_{ij}$ of user v_i over relay r_j. Based on $\mathbf{B}_i^t$, user v_i conjectures its expected delay when making a decision at time t without knowing the exact $\boldsymbol{\sigma}_{-i}^t$ value, which can be calculated by

$$\tilde{U}_i^t\left(\boldsymbol{\sigma}_i^t, \mathbf{B}_i^t\right) = \sum_{j=1}^{N} \frac{\lambda_{ij}^t}{\lambda_i} \frac{1}{\tilde{C}_{ij}\left(h_i^t\right) - \lambda_{ij}^t}. \tag{6}$$

User v_i then determines its load balancing decision based on the following conjecture-based best response.

Definition 3—Conjecture-Based Best Response: We define the conjecture-based best response of user v_i as

$$\pi_i\left(\mathbf{B}_i^t\right) = \arg \min_{\sigma_i \in \mathcal{X}_i} \tilde{U}_i^t\left(\boldsymbol{\sigma}_i, \mathbf{B}_i^t\right). \tag{7}$$

To perform the aforementioned best response, the user only needs to collect the local information $\mathcal{I}_i = [\{C_j, \forall r_j\}, h_i^t, \lambda_i]$. Fig. 2 shows the distributed load balancing of user v_i. In a time slot t, all users $v_i \in \mathbf{V}$ first observe their congestion information history h_i^t and then evaluate their conjectures $\tilde{C}_{ij}(h_i^t)$ on the remaining capacities of the paths. Using the corresponding conjecture functions, they determine their load balancing actions $\boldsymbol{\sigma}_i^t \in \mathcal{X}_i$ based on the conjecture-based best response in (7).

Although the users adopt the defined best response, they may adopt different learning methods to form their conjectures $\mathbf{B}_i^t = \{\tilde{C}_{ij}(h_i^t), j = 1, \ldots, N\}$. In this paper, we discuss the following two types of users:

1) **Naive learners:** A naive learner forms the conjectures independently of its action. For example, the naive learners can form the conjectures based on the average remaining capacities that they observed in history h_i^t. In this paper, we assume that the naive learners conjecture the remaining capacities simply based on the latest congestion information in h_i^t, i.e., $\tilde{C}_{ij}(h_i^t) = C_{ij}^{t-1}$, with $j = 1, \ldots, N$ [a special case when $S = 1$ in (5)].

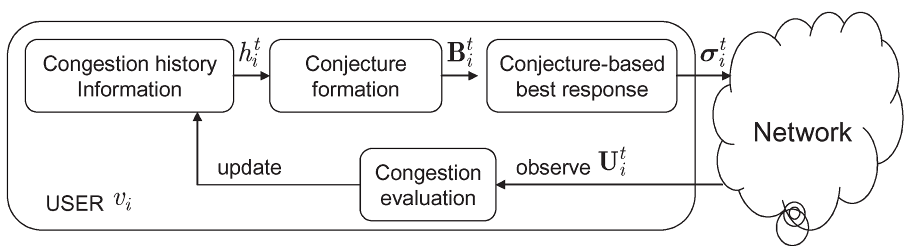

Fig. 2. Distributed load balancing in the conjectured-based LBG.

2) **Conjecturing learners:** A conjecturing learner forms the conjecture functions, depending on its action. In this paper, we encapsulate the forward-looking (foresighted) behavior of the active learning using a simple linear conjecture function, and we will show in Section V that it works well in practice.

Definition 4—Linear Conjecture Function: We design the conjecture function of a conjecturing learner v_i to be a linear function, i.e.,

$$\tilde{C}_{ij}\left(h_i^t\right) = \beta_{ij}^{(0)}\left(h_i^t\right) + \beta_{ij}^{(1)}\left(h_i^t\right)\lambda_{ij} \qquad \forall r_j \tag{8}$$

where $\beta_{ij}^{(k)}(h_i^t)$, $k = 0, 1$, are the coefficients of the conjecture functions.

E. Conjecture-Based LBG and the CE

We define the multiuser interaction in the distributed decision making in the previous subsection with the following game definition:

Definition 5—Conjecture-Based LBG: We consider the conjecture-based LBG as a stage game that is represented by the following tuple $\langle \mathbf{V}, \mathcal{X}, \mathcal{B}, \mathbf{U}\rangle$:

1) $\mathbf{V} = \{\mathbf{V}^N, \mathbf{V}^C\}$: the set of players (users), which can be either naive learners in a set, i.e., $\mathbf{V}^N \subset \mathbf{V}$, or conjecturing learners in a set, i.e., $\mathbf{V}^C \subset \mathbf{V}$;
2) $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_M$: the action space of the users;
3) $\mathcal{B} = \mathcal{B}_1 \times \cdots \times \mathcal{B}_M$: the conjecture space of the users;
4) $\mathbf{U} = \{\tilde{U}_i, \forall v_i\}$: a set of conjectured delays of the users.

In this paper, we assume that users minimize the conjectured delay by performing the conjecture-based best response in (7). Next, we discuss the equilibrium concepts that can emerge in the conjecture-based LBG.

Proposition 1—Unique NE: When $\mathbf{V} = \mathbf{V}^N$, a unique pure strategy NE that is described by $\boldsymbol{\sigma}^*$ exists. Given the remaining capacity C_{ij}^* at the equilibrium, the load balancing action of user v_i is given by

$$\lambda_{ij}^* = \max\{0, C_{ij}^* - \alpha_{ij}^* R\} \tag{9}$$

where $R \triangleq \sum_{j=1}^N C_j - \sum_{i=1}^M \lambda_i$ is a constant that represents the overall remaining capacity. $\alpha_{ij}^* = (\sqrt{C_{ij}^*}/\sum_{j=1}^N \sqrt{C_{ij}^*})$ represents the optimal fraction of the overall remaining capacity that user v_i should allocate over relay r_j to minimize its end-to-end delay.

Proof: If all the users are naive learners, at the equilibrium, they will passively form a correct belief $C_{ij}^{t-1} = C_{ij}^t = C_{ij}^*$. Hence, the best response of user v_i becomes $\boldsymbol{\sigma}_i^* = \arg\min_{\boldsymbol{\sigma}_i \in \mathcal{X}_i} \sum_{j=1}^N (\lambda_{ij}/C_{ij}^* - \lambda_{ij})$. The optimal actions in (9) can be obtained by solving this optimization, as shown in [17]. ■

Proposition 1 provides the equilibrium concept when all users are naive learners in the conjecture-based LBG. However, when there are conjecturing learners in the LBG, the equilibrium concept is captured by the CE. The CE was first discussed by Hahn in the context of a market model [21] and used in [20] for coordination among wireless users. We next discuss the CE in the conjecture-based LBG context.

Definition 6—CE of the LBG: Action $\boldsymbol{\sigma}^* \in \mathcal{X}$ is the CE of the LBG if, for each user $v_i \in \mathbf{V}$, the following two conditions are satisfied:

i) $\tilde{C}_{ij}^* = C_j - \sum_{i' \neq i} \lambda_{i'j}^* \; \forall r_j$.
ii) $\sigma_i^* = \arg\min_{\boldsymbol{\sigma}_i \in \mathcal{X}_i} \tilde{U}_i(\boldsymbol{\sigma}_i, \mathbf{B}_i^*(\boldsymbol{\sigma}_i)) \; \forall v_i$, where $\mathbf{B}_i^*$ denotes the conjecture function at the equilibrium.

Since the conjecture functions of the naive learners are independent of their actions, it can be easily seen that the aforementioned two conditions are satisfied at the NE when all the users are naive learners. Hence, the NE is a special case of CE. The first condition states that the conjectured remaining capacities at the equilibrium are consistent with the actual remaining capacities. The second condition states that action $\boldsymbol{\sigma}^* \in \mathcal{X}$ minimizes the expected end-to-end delay. However, as long as an action consistently optimizes the expected utility, a user can still keep selecting the same action given its imperfect conjectures. In this case, the first condition can be relaxed. For this, we define an extension of the conventional CE, where users' actions converge to the equilibrium based on imperfect conjectures.

Definition 7—ε-CE of LBG: The ε-CE is defined as $\boldsymbol{\sigma}^* \in \mathcal{X}$ if, for each user $v_i \in \mathbf{V}$, the following two conditions are satisfied:

$$i) \quad |\tilde{C}_{ij}^* - C_j + \sum_{i' \neq i} \lambda_{i'j}^*| \leq \varepsilon \qquad \forall r_j \quad \forall v_i.$$

$$ii) \quad \boldsymbol{\sigma}_i^* = \arg\min_{\boldsymbol{\sigma}_i \in \mathcal{X}_i} \tilde{U}_i(\boldsymbol{\sigma}_i, \mathbf{B}_i^*(\sigma_i)) \qquad \forall v_i. \tag{10}$$

The goal of this paper is to develop simple belief formation techniques in the conjecture-based LBG that allow the users to interact without message exchanges and reach efficient ε-CE. In Table I, we first summarize the solutions proposed in this paper. Unlike the centralized coordination solution and the distributed best response solution, which require explicit message exchange, we propose conjecture-based load balancing

TABLE I
SUMMARY OF THE INTRODUCED SOLUTIONS

	Coordinated elements	Information requirement	Equilibrium concept/efficiency	Introduced section
Centralized coordination	Coordinator determines actions of all the users	$\mathcal{I}_g = [\{C_j, \forall r_j\}, \{\lambda_i, \forall v_i\}]$	PO (Pareto Optimal)	II.B
Distributed best response	None	$\mathcal{I}_g = [\{C_j, \forall r_j\}, \{\lambda_i, \forall v_i\}]$	NE (Nash Equilibrium)	II.C
Single altruistic leader case	System designer defines belief formation of the leader	$\mathcal{I}_i = [\{C_j, \forall r_j\}, h_i^t, \lambda_i]$	ε -CE → PO	III.B
Single self-interested leader case	System designer defines belief formation of the leader	$\mathcal{I}_i = [\{C_j, \forall r_j\}, h_i^t, \lambda_i]$	ε -CE → SE (Stackelberg Equilibrium [25])	III.C
All conjecturing learner case	System designer defines belief formation of all learners	$\mathcal{I}_i = [\{C_j, \forall r_j\}, h_i^t, \lambda_i]$	ε -CE → PO	IV

methods, which are able to reach efficient outcomes (through appropriate belief formation) without message exchanges. We will investigate two cases that drive the load balancing solution $\boldsymbol{\sigma}^*$ to the ε-CE that corresponds to the system-wide optimal solution $\boldsymbol{\sigma}^P$ without the need to exchange messages. In Section III, we will focus on the case when the system has only one conjecturing learner, and then, in Section IV, we will study the case when every user in the LBG is a conjecturing learner.

III. Conjecture-Based Load Balancing When There Is Only One Conjecturing Learner

Without loss of generality, we assume in this section that user v_1 is the conjecturing learner and the other users are naive learners in the conjecture-based LBG. The conjecturing learner serves as a leader in the network and is elected based on the proportion of traffic that it generates.[2] We show that, in this scenario, a simple regression learning can be adopted by the conjecturing learner to drive the ε-CE to the Pareto boundary.

A. Linear Regression Learning to Model the Belief Function

The conjecturing learner v_1 repeatedly[3] updates its conjecture functions $\tilde{C}_{ij}(h_i^t)$ in (8) for all the paths based on its observation of the remaining capacities in its congestion history information h_1^t. Since there are S samples in the history (assuming $t > S$), the conjecturing learner can update the coefficient vector $\boldsymbol{\beta}_{1j}^t = [\beta_{1j}^{(0)t}, \beta_{1j}^{(1)t}]$ using the following update rule:

$$\boldsymbol{\beta}_{1j}^t = (1-\rho^t)\boldsymbol{\beta}_{1j}^{t-1} + \rho t\tilde{\boldsymbol{\beta}}_{1j}\left(h_1^t\right) \quad (11)$$

where $\tilde{\boldsymbol{\beta}}_{ij}^T = (X^T X)^{-1} X^T Y$, and

$$X = \begin{bmatrix} 1 & \lambda_{ij}^{t-1} \\ \vdots & \vdots \\ 1 & \lambda_{ij}^{t-S} \end{bmatrix}$$
$$Y = \begin{bmatrix} C_{ij}^{t-1} \\ \vdots \\ C_{ij}^{t-S} \end{bmatrix}. \quad (12)$$

Equation (12) is the standard regression for the degree-1 polynomial conjecture function [22]. ρ^t in (11) represents the adaptation rate $(0 \le \rho^t \le 1)$, which determines how rapidly a user is willing to change its conjecture on the remaining capacities. In this paper, the adaptation rate is determined by $\rho^t = 1 - e^{-\delta^t}$, where $\delta^t = (1/L)\sum_{k=1}^{L}\sqrt{(C_{1j}^{t-k} - C_{1j}^{t-k-1})^2 + (\lambda_{1j}^{t-k} - \lambda_{1j}^{t-k-1})^2}$ represents the average distance among the latest L samples in h_1^t $(L < S)$, which quantifies the diversity of the samples. The adaptation rate $\rho^t = 1 - e^{-\delta^t}$ ensures that the adaptation rate decreases when the latest L samples converge over time. We assume that the conjecturing learner adopts the simplest linear regression learning $\tilde{C}_{1j}(h_1^t) = \beta_{1j}^{(0)}(h_1^t) + \beta_{1j}^{(1)}(h_1^t)\lambda_{1j}$ and that it starts with an initial load balancing decision $\boldsymbol{\sigma}_1^{\text{Init}}$. If the responses of the rest of the naive learners are stable, the remaining capacity C_{1j}^t over path r_j concentrates to C_{1j}^* given the conjecturing learner's initial decision $\lambda_{1j}^{\text{Init}}$. Hence, the adaptation rate goes to 0 (since δ^t goes to 0), which leads to a fixed coefficient vector $\boldsymbol{\beta}_{1j}$. A new load balancing decision of the conjecturing learner can be subsequently made based on $\boldsymbol{\beta}_{1j}$. To estimate the error of the linear regression model with $\boldsymbol{\beta}_{1j}$, we also define the maximum residual error as follows:

Definition 8—Maximum Residual Error: The maximum residual error is defined as error $\bar{e}(\boldsymbol{\beta}_{1j}, h_1^t) = \max_{k=1,\ldots,S}| C_{1j}^{t-k} - (\beta_{1j}^{(0)}(h_1^t) + \beta_{1j}^{(1)}(h_1^t)\lambda_{1j}^{t-k})|$.

The maximum residual error represents the maximum difference between the remaining capacities of the history samples and the linear belief function $\tilde{C}_{1j}(h_1^t) = \beta_{1j}^{(0)}(h_1^t) + \beta_{1j}^{(1)}(h_1^t)\lambda_{1j}$ through path r_j. It quantifies how accurately the

[2]In the multipath setting, to enforce equilibrium, the leader is required to control a proportion of traffic above a certain predetermined threshold [17]. If no single users has traffic load above the threshold, users with larger traffic loads can be combined and elected as the leader with aggregate traffic load above the threshold.

[3]Different time scales can be applied for the conjecturing learners to make sure that the measured remaining capacities C_{ij}^t are the stable results of the other naive learners played in the game.

linear belief function can describe the remaining capacities after the other naive learners react to the leader's load balancing decision.

Proposition 2—Reaching the ε-CE Using the Linear Regression Learning: When $|\mathbf{V}^C| = 1$, if the linear regression learning converges, it converges to the ε-CE of the conjecture-based LBG with $\varepsilon = \max_{r_j}\{\bar{e}(\boldsymbol{\beta}_{1j}, h_1^t)\}$.

Proof: It is straightforward that, if ε is selected as the maximum mean residual error, we have $|\tilde{C}_{1j}^* - C_j + \sum_{i'\neq 1}\lambda_{i'j}^*| \leq \varepsilon, \forall r_j$. Hence, the first condition in Definition 7 can be satisfied. Regardless of whether a user is a naive learner or a conjecturing learner, all users are minimizing their delays with respect to their beliefs about the other users, and hence, such an equilibrium is a ε-CE. ■

Here, samples $\{(\lambda_{1j}^{t-k}, C_{1j}^{t-k}), k = 1, \ldots, S\}$ in the congestion information history of the conjecturing learner v_1 provide aggregate information about how the other naive learners react to the actions of the conjecturing learner in the past. The linear conjecture function is formed by using the linear regression based on these samples. In our simulation in Section V, we verify that the mean residual error of the linear regression is very small when there is only one conjecturing learner in the network. Next, we discuss in greater detail the ε-CE in two different cases, i.e., when the conjecturing learner is altruistic and when the conjecturing learner is self-interested.

B. Altrustic Conjecturing Learner

An altruistic conjecturing learner is usually the resource manager in a clustered network [7], e.g., the access point in the IEEE 802.11 network, or the routing leader in a hierarchical ad hoc network [14]. An altruistic conjecturing learner has an objective function that is aligned with system cost, e.g., the system-wide utility function in (3).

As the conjecturing learner v_1 applies the conjecture function $\tilde{C}_{1j}(\lambda_{1j})$, the system-wide utility function can become

$$\tilde{U}^{\text{sys}}(\boldsymbol{\sigma}_1, \mathbf{B}_1(\boldsymbol{\sigma}_1)) = \sum_{j=1}^{N}\frac{C_j - \beta_{1j}^{(0)} - \beta_{1j}^{(1)}\lambda_{1j} + \lambda_{1j}}{\beta_{1j}^{(0)} + \beta_{1j}(1)}\lambda_{1j} - \lambda_{1j}. \tag{13}$$

Then, the altruistic conjecturing learner v_1 directly minimizes the system cost[4] based on (13), whereas the rest of the naive learners perform myopic best responses. However, the conjecturing learner adopts a linear conjecture function, which may provide only an imperfect estimation of the remaining capacities. There will be a performance penalty (gap) experienced by the conjecturing learner between the resulting ε-CE $\boldsymbol{\sigma}_{\text{alt}}^*$ and the system-wide optimal solution $\boldsymbol{\sigma}^P$, which is defined as

$$\text{GAP}\left(\boldsymbol{\sigma}_{\text{alt}}^*, \boldsymbol{\sigma}^P\right) = U^{\text{sys}}\left(\boldsymbol{\sigma}_{\text{alt}}^*\right) - U^{\text{sys}}(\sigma^P). \tag{14}$$

Proposition 3—Reaching System-Wide Optimal Solution When Only One User Is Conjecturing Learner: When there is only one altruistic conjecturing learner v_i in the conjecture-based LBG, the gap between the resulting ε-CE $\boldsymbol{\sigma}_{\text{alt}}^*$ and $\boldsymbol{\sigma}^P$ will be bounded by

$$\text{GAP}\left(\boldsymbol{\sigma}_{\text{alt}}^*, \boldsymbol{\sigma}^P\right) \leq \varepsilon \sum_{\forall r_j}\frac{C_j}{\left(C_{ij,\text{alt}}^* - \lambda_{ij,\text{alt}}^*\right)^2}. \tag{15}$$

Proof: From the definition of an ε-CE $\boldsymbol{\sigma}_{\text{alt}}^*$, the worst case $\tilde{C}_{ij}^* \geq C_{ij}^*(\boldsymbol{\sigma}_{\text{alt}}^*) - \varepsilon$ can be considered to bound GAP($\boldsymbol{\sigma}_{\text{alt}}^*$, $\boldsymbol{\sigma}^P$). The worst case gap is bounded by GAP($\boldsymbol{\sigma}_{\text{alt}}^*$, $\boldsymbol{\sigma}^P$) $\leq \sum_{r_j}(C_j + \lambda_{ij}^* - C_{ij}^* + \varepsilon / C_{ij}^* - \lambda_{ij}^* - \varepsilon) - \sum_{r_j}(C_j + \lambda_{ij}^* - C_{ij}^* / C_{ij}^* - \lambda_{ij}^*)$. Let $K_{ij} = C_j + \lambda_{ij}^* - C_{ij}^*$ and $J_{ij} = C_{ij}^* - \lambda_{ij}^*$. For small ε, the first term of the right-hand side can be simplified as $\sum_{r_j}(K_{ij} + \varepsilon / J_{ij} - \varepsilon) \cong \sum_{r_j}(K_{ij} / J_{ij}) + \sum_{r_j}(K_{ij} + J_{ij} / (J_{ij})^2)\varepsilon$, and the gap will be bounded by GAP($\boldsymbol{\sigma}_{\text{alt}}^*, \boldsymbol{\sigma}^P$) $\leq \varepsilon \sum_{r_j}(K_{ij} + J_{ij} / (J_{ij})^2) = \varepsilon \sum_{r_j}(C_j / (C_{ij}^* - \lambda_{ij}^*)^2)$. ■

Proposition 3 implies that the conjecturing learner is able to drive $\boldsymbol{\sigma}_{\text{alt}}^*$ to the system-wide optimal solution when it is the only conjecturing learner in the conjecture-based LBG and ε is small.

C. Self-Interested Conjecturing Learner

If the conjecturing learner is self-interested, a conjecturing learner may have no incentive to sacrifice its own delay to minimize the system-wide cost. The objective function of the self-interested conjecturing learner is to minimize $U_i(\boldsymbol{\sigma}_i, \mathbf{B}_i(\boldsymbol{\sigma}_i)) = (1/\lambda_i)\sum_{j=1}^{N}(\lambda_{ij}/\beta_{ij}^{(0)} + \beta_{ij}^{(1)}\lambda_{ij} - \lambda_{ij})$. The following proposition provides the optimal action for the self-interested conjecturing learner.

Proposition 4—Solution of the Self-Interested Conjecturing Learner: Given the linear conjecture function $\tilde{C}_{1j}(h_1^t) = \beta_{1j}^{(0)}(h_1^t) + \beta_{1j}^{(1)}(h_1^t)\lambda_{1j}$, the optimal action is

$$\lambda_{ij}^* = \max\left\{0, \tilde{D}_{ij} - \alpha_{ij}^{(f)}\left(\sum_{r_j}\tilde{D}_{ij} - \lambda_i\right)\right\}. \tag{16}$$

Portion $\alpha_{ij}^{(f)}$ now becomes $\kappa_{ij} / \sum_{r_j}\kappa_{ij}$, where $\kappa_{ij} = \sqrt{\beta_{ij}^{(0)}} / (1 - \beta_{ij}^{(1)})$, and $\tilde{D}_{ij} = \beta_{ij}^{(0)} / (1 - \beta_{ij}^{(1)})$.

Proof: See Appendix A.

Note that, if the conjecturing learner is able to build a perfect belief on the remaining capacities (i.e., $\varepsilon = 0$), the resulting CE $\boldsymbol{\sigma}_{\text{self}}^*$ coincides with the Stackelberg equilibrium (SE) $\boldsymbol{\sigma}^S$ [25] of the game, since the conjecturing learner has perfect knowledge of the naive learners' reactions. Hence, we use the SE $\boldsymbol{\sigma}^S$ instead of the system-wide optimal solution $\boldsymbol{\sigma}^P$ to benchmark the self-interested conjecturing learner. The corresponding performance gap is defined as GAP($\boldsymbol{\sigma}_{\text{self}}^*, \boldsymbol{\sigma}^S$) $= U_i(\boldsymbol{\sigma}_{\text{self}}^*) - U_i(\boldsymbol{\sigma}^S)$.

Proposition 5—Reaching SE When Only One User Is Conjecturing Learner: When there is only one self-interested

[4]Note that only the system-wide optimal solution is on the Pareto boundary with weights $w_i = (\lambda_i / \sum_i \lambda_i)$. For the other solutions on the Pareto boundary, the conjecturing learner needs to know the corresponding weights.

TABLE II
SELF-INTERESTED CONJECTURE-BASED LOAD BALANCING ALGORITHM

Algorithm Self-interested conjecture-based load balancing

For user v_i **at time slot** t

Initialization: Set $t = 1$, $C_{ij}^0 = C_j$

Step 1. For all relay $r_j, j = 1,\ldots,N$, **calculate the remaining capacity** C_{ij}^{t-1} from U_{ij}^{t-1} **and record it to** h_i^t.

Step 2. Update $\boldsymbol{\beta}_{ij}^t$.

Calculate $\tilde{\boldsymbol{\beta}}_{ij}$ using least square error linear regression from samples $\{(C_{ij}^{t-k}, \lambda_{ij}^{t-k}), k = 1,\ldots,S\}$. Then set $\boldsymbol{\beta}_{ij}^t \in \mathcal{B}_i$ as in equation (11).

Step 3. Calculate the self-interested solution

$\sigma_{self}^t = [\lambda_{ij}^t, j = 1,\ldots,N]$.

λ_{ij}^t is calculated according to equation (16). Find the load balancing action $\mathbf{a}_i^t = \sigma_{self}^t / \lambda_i$

Step 4. Perform $\mathbf{a}_i^t$ **and observe the delays**

$U_{ij}^t(\boldsymbol{\sigma}_i^t, \boldsymbol{\sigma}_{-i}^t), j = 1,\ldots,N$.

Step 5. $t \leftarrow t + 1$, **and go back to Step 1.**

conjecturing learner v_i in the conjecture-based LBG, the gap between the resulting ε-CE and the SE will be bounded by

$$\mathrm{GAP}\left(\boldsymbol{\sigma}_{\mathrm{self}}^*, \boldsymbol{\sigma}^S\right) \leq \varepsilon \sum_{r_j} \frac{1}{\left(C_{ij,\mathrm{self}}^* - \lambda_{ij,\mathrm{self}}^*\right)^2}. \quad (17)$$

Proof: The gap can be shown to be bounded using a similar proof as in Proposition 3. The only difference is that the conjecturing learner is now minimizing its own delay instead of U^{sys} in Proposition 3.

Proposition 5 implies that the conjecturing learner is able to drive the ε-CE $\boldsymbol{\sigma}_{\mathrm{self}}^*$ to the SE $\boldsymbol{\sigma}^S$ when it is the only conjecturing learner in the conjecture-based LBG and the ε is small. We provide the load balancing algorithm in Table II that will be followed by the self-interested conjecturing learner. An illustrative example is given in Fig. 3 for the solutions introduced in Sections IV-C and D in the two-user case (v_i is the conjecturing learner and v_{-i} is the naive learner). Note that the SE $\boldsymbol{\sigma}^S$ provides a smaller delay compared with $\boldsymbol{\sigma}^P$ for the conjecturing learner v_i at the cost of increasing the delay of the naive learner. This is because it selfishly minimizes its own delay given that it knows the reaction of the other user, which is the best payoff that a self-interested conjecturing learner can achieve.

IV. CONJECTURE-BASED LOAD BALANCING WITH MULTIPLE CONJECTURING LEARNERS

As mentioned in Section II-E, when there is more than one conjecturing learner in the network, the multiuser interaction cannot always reach equilibrium. Moreover, even if the LBG converges, the CE may differ from the optimal solution desired by a protocol (see Fig. 3). Here, we discuss the case where multiple conjecturing learners interact.

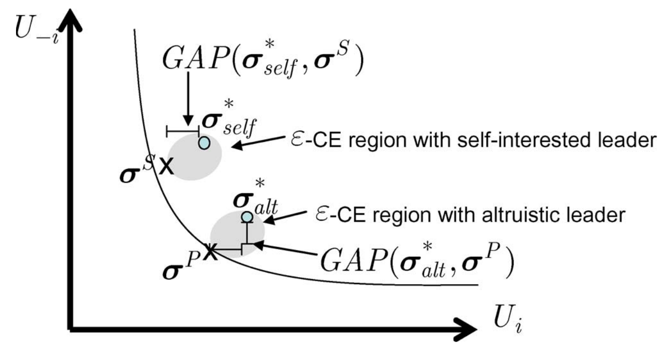

Fig. 3. Illustrative example of the solutions in the utility domain for a two-user case (v_i is the conjecturing learner).

A. Impact of Multiple Self-Interested Learners

When the number of self-interested conjecturing learners increases, larger errors in the belief function (ε in Proposition 5) tend to occur, which lead to a larger set of ε-CE, as shown in Fig. 3. Next, we determine the maximum number of self-interested learners that is allowed in the system to ensure that the resulting worst-case system performance is bounded.

Proposition 6—Maximum Tolerable Number of Self-Interested Users: The maximum number of self-interested learners that can be active in the system while keeping the worst-case system performance bounded is $\overline{N} = \max(1, \arg\max_n \Lambda(n)), s.t. \Lambda(n) \leq \min_j C_j$, where $\Lambda(n)$ represents the sum of n largest users' loads.

Proof: Let us consider the worst case scenario where all the self-interested users select the relay that has minimum capacity due to a bad belief function. If $\Lambda(n) > \min_j C_j$, then $(C_{ij,\mathrm{self}}^* - \lambda_{ij,\mathrm{self}}^*)$ in Proposition 5 becomes 0 in the worst-case scenario. Hence, the average delay of the self-interested users that select r_j becomes unbounded, as well as the worst-case system performance. ■

To improve the performance of the system when multiple self-interested learners are active in the system, these self-interested learners need to adhere to the collaborative rules determined by the protocol designer. Hence, we discuss next a rule-based linear conjecture mechanism that leads to the system-wide optimal solution without explicit message exchange among the users when all conjecturing learners comply to it.

B. Rule-Based Linear Conjecture Method

We propose an alternative rule-based belief function for the conjecturing learners in this section. Unlike the linear regression learning method proposed in Section IV-B computed by the leader, the rule-based belief function is set by the protocol designer. We prove that, as long as the users comply with the rule-based belief function, they can reach the system-wide optimal solution in a distributed manner, based on their local information. The following proposition gives the rule-based belief function parameters.

Proposition 7—Rule-Based Selection of Belief Function Parameters: A family of belief function parameters $\mathcal{B}_i^* = \{\boldsymbol{\beta}_{ij}^*\} \subseteq \mathcal{B}_i$ leads to the rule-based solution $\boldsymbol{\sigma}_{\mathrm{rule}}^* = \{\lambda_{ij}^*,$

$i=1,\ldots,M, j=1,\ldots,N\}$, where $\lambda_{ij}^{*}=\max\{0, C_{ij}-(\sqrt{C_j}/\sum_{j=1}^{N}\sqrt{C_j})R\}$. This solution $\boldsymbol{\sigma}_{\text{rule}}^{*}$ minimizes $U^{\text{sys}}(\boldsymbol{\sigma})$ and results in $\text{GAP}(\boldsymbol{\sigma}_{\text{rule}}^{*}, \boldsymbol{\sigma}^{P})=0$.

Proof: See Appendix B.

A straightforward example for the belief functions in Proposition 7 can be $\beta_{ij}^{*(0)}=((C_{ij})^2/C_j)$, $\beta_{ij}^{*(1)}=1-(C_{ij}/C_j)$, $\forall v_i\in\mathbf{V}$ [then (16) in Proposition 4 becomes $\lambda_{ij}^{*}=\max\{0, C_{ij}-(\sqrt{C_j}/\sum_{j=1}^{N}\sqrt{C_j})R\}$]. By forcing the users to use this belief function with $[\beta_{ij}^{*(0)}, \beta_{ij}^{*(1)}]$, the rule-based solution $\boldsymbol{\sigma}_{\text{rule}}^{*}$ can be obtained by the users based on the remaining capacities C_{ij}. Note that such a rule-based solution is not the equilibrium of the LBG. It is derived as an optimal rate allocation based on the utility function $U^{\text{sys}}(\boldsymbol{\sigma})$ that is defined in Section II-B (see Appendix B). Specifically, the rule-based solution λ_{ij}^{*} is determined when user v_i first joins the network, and C_{ij} can be regarded as the remaining capacities over path r_j, which user v_i determined by probing the network[5] before joining it. Intuitively, it can be seen from Proposition 7 that, as long as the overall remaining capacity R is distributed to path r_j with the exact fraction $(\sqrt{C_j}/\sum_{j=1}^{N}\sqrt{C_j})$, solution λ_{ij}^{*} is a system-wide optimal solution. Hence, every user needs to ensure such fractions when it joins the network. Unless the network setting changes (e.g., variation of C_j), a user's rule-based belief formation and the resulting action remain the same afterward.

If a new user joins a network and the other users present in the network are already complying with the rules (choosing $\boldsymbol{\beta}_{ij}^{*}$), the following condition ensures that the users will have no incentives to deviate from the rule-based solution.

Proposition 8—Sufficient Condition for Users to Comply With the Rule-Based Solution: When all the users in the network are conjecturing learners, i.e., $\mathbf{V}^{C}=\mathbf{V}$, no users will deviate from the rule-based solution $\boldsymbol{\sigma}_{\text{rule}}^{*}$ (i.e., the rule is self-enforcing), if $\lambda_{ij}>0, \forall v_i, \forall r_j$, and $C_j=C, \forall C_j$.

Proof: Let us assume that a new user joins the network and that the users already present in the network comply with the rule-based solution. Hence, the overall remaining capacity R is already allocated to different relays according to fraction $(\sqrt{C_j}/\sum_{j=1}^{N}\sqrt{C_j})$. The new user's remaining capacity can be calculated as $C_{ij}=(\sqrt{C_j}/\sum_j\sqrt{C_j})(\sum_j C_j-\sum_{i'\neq i}\lambda_{i'})$. When all the relays have the same capacity and they are shared by all the users (i.e., $\lambda_{ij}>0, \forall v_i, \forall r_j$), fraction $(\sqrt{C_j}/\sum_{j=1}^{N}\sqrt{C_j})=(1/N)$, and hence, $C_{ij}=C_{ij'}=(1/N)(\sum_j C_j-\sum_{i'\neq i}\lambda_{i'})$. Thus, fraction $\alpha_{ij}=(\sqrt{C_{ij}}/\sum_{j=1}^{N}\sqrt{C_{ij}})$ in the user's best response [see (9)] becomes $(\sqrt{C_j}/\sum_{j=1}^{N}\sqrt{C_j})$. Hence, the rule-based solution $\boldsymbol{\sigma}_{\text{rule}}^{*}$ is the best response for user v_i to minimize its own delay, $\forall v_i\in\mathbf{V}$, when the other users select the rule-based solution. ■

In general, when the condition in Proposition 8 is not satisfied, the rule-based solution is not the best response for the users. Hence, the system-wide optimal solution is not self-enforcing in this usage scenario.

[5]Probing can be done by using the similar method as calculating the remaining capacities in Section II-D. Here, we assume that the probability of two users simultaneously joining the network is very small.

TABLE III
CONSIDERED NETWORK SETTINGS

Network setting	Number of relays N	Number of users M	Total capacities (pkt/sec)	Total traffic rates (pkt/sec)
1 (Diverse network)	10	30	26K	21.2K
2 (Concentrated network)	2	8	10K	8K

So far, two linear conjecture formations are introduced for a conjecturing learner to build their conjecture functions, i.e., using $\boldsymbol{\beta}_{ij}^{t}=[\beta_{ij}^{(0)t}, \beta_{ij}^{(1)t}]$ that applies the linear regression learning in (11) and using $\boldsymbol{\beta}_{ij}^{*}=[\beta_{ij}^{*(0)}, \beta_{ij}^{*(1)}]$ that applies the rule-based solution. Importantly, there are two differences between these two approaches.

a) The first approach allows the conjecturing learners to build their conjectures about the *aggregate response of the other users* (C_{ij} in this paper) based on only local information. However, the second approach builds the conjectures for users to follow the *optimal rate allocation* that minimizes the system's cost.

b) The first approach is not suitable for the scenario when multiple conjecturing learners simultaneously build their conjectures, because the resulting remaining capacities become a highly nonlinear function of the loading. The *linear* conjecture functions are no longer able to capture the sample variation in the history, and the resulting solution becomes inefficient. On the contrary, applying the second approach is efficient but only when *all the users* are willing to comply with the rule-based solution. However, it is shown that the rule-based solution can only be self-enforcing in the case that each relay has the same capacity. Hence, an important topic for future research is determining how to build for the general-case self-enforcing rule-based solution without explicit message exchange among the users. A possible direction is deploying intervention functions [30].

V. SIMULATION RESULTS

Here, we simulate the conjecture-based LBG in a network with concentrated paths (two paths) and a network with diverse paths (ten paths), which are shown in Table III. The concentrated setup is representative of numerous network services that use a backup path for robustness to avoid single point of failure, see, for example, the similar setup discussed in [2] and [35]. The diverse setup can represent a larger ad hoc multipath network scenario, where multiple nodes in the same path are aggregated into one relay, similar to the setup that is simulated in [17] and [18], or in cognitive radio networks, where each relay represents a wireless channel, as in [7]–[10]. We assume an asymmetric network where the capacities of the relays are $W_1=8000$ pkt/s and $W_j=2000$ pkt/s, with $j=2,\ldots,N$. The users are assumed to experience traffic that is characterized by Poisson arrival rates $\lambda_1=3800$ pkt/s and $\lambda_i=600$ pkt/s, with $i=2,\ldots,M$.

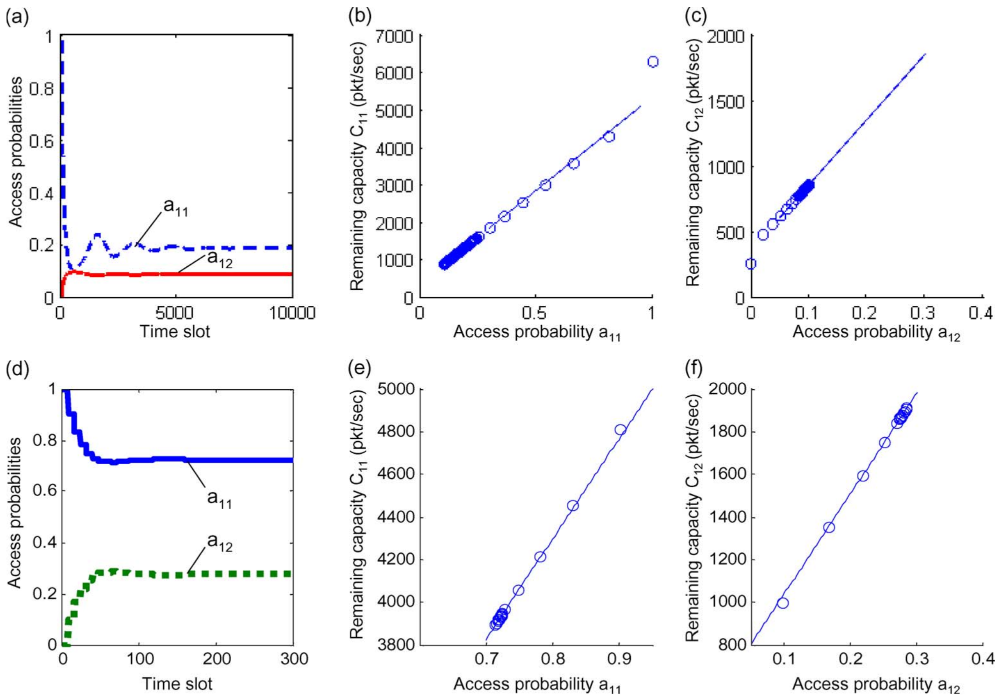

Fig. 4. Action of the conjecturing learner over time, while participating in the load balancing game [in network settings (a) 1 and (d) 2]. Actual remaining capacity C_{1j} and the estimated linear belief function $\tilde{C}_{1j}$, with $j = 1, 2$ [in network settings (b) and (c) 1 and (e) and (f) 2].

A. Single Conjecturing Learner Scenario

We first simulate the case when there is only one conjecturing learner. User v_1 is assumed to be the conjecturing learner, and the rest of the users are naive learners. Fig. 4(a) shows the evolution of the action of user v_1, i.e., $\boldsymbol{\sigma}_1$ (which is its load balancing ratio $a_{ij} \equiv (\lambda_{ij}/\lambda_i)$) until the system reaches the NE in network setting 1 (the diverse network). Since relay r_1 has a larger capacity, more traffic will be distributed to relay r_1 than to the other relays. Using the learning method proposed in Section IV-B, the conjecturing learner v_1 can determine its belief functions on the remaining capacities. The circles in Fig. 4(b) represent the measured remaining capacities C_{11} at different load balancing ratios a_{11} (the samples in h_1^t). The solid line represents the resulting linear regression. The resulting parameters of the linear belief function are $\boldsymbol{\beta}_{11} = [0.\ 375, 4962]$ when the linear regression learning converges. The resulting residual mean square error is $\bar{e}(\tilde{\boldsymbol{\beta}}_{ij}, h_i^t) = 0.\ 051$. Fig. 4(c) shows similar results in relay r_2. Similarly in network setting 2 (the concentrated network), Fig. 4(d) shows again the evolution of $\mathbf{a}_1$ in a network. The linear regression converges faster in this setting, since the number of users is smaller. The resulting parameters of the linear belief function are $\boldsymbol{\beta}_{11} = [0.\ 52, 4718]$ when the linear regression learning converges. The resulting residual mean square error is $\bar{e}(\tilde{\boldsymbol{\beta}}_{ij}, h_i^t) = 0.\ 012$. Based on the linear belief functions, user v_1 then performs the conjecture-based load balancing in the proposed algorithm in Table II.

Fig. 5 shows the utility domain (i.e., the experienced delays) when the users interact in the concentrated network setting. The x-axis is the delay of the conjecturing learner, and the y-axis is the average delay of the naive learners. By using the belief function, the simulation results show that the altruistic conjecturing learner is able to drive the system from the (system) inefficient NE to the system-wide optimal solution on the Pareto boundary (in which the system queue size U^{sys} is minimized) by using the belief function. If the conjecturing learner is selfish, it will drive the system from the NE to the SE. Table IV shows the results at different equilibriums. When the conjecturing learner is selfish, it puts more traffic into the efficient relay r_1 and forces the other naive learners to select the other relay, thereby benefiting its own utility. On the contrary, if the conjecturing learner is altruistic, it puts less traffic into relay r_1 and allows the other users to myopically select the efficient relay r_1, which will result in an optimal system performance. We also compared the performance against the well-known weighted round-robin solution provided in [4], in which the load balancing weight

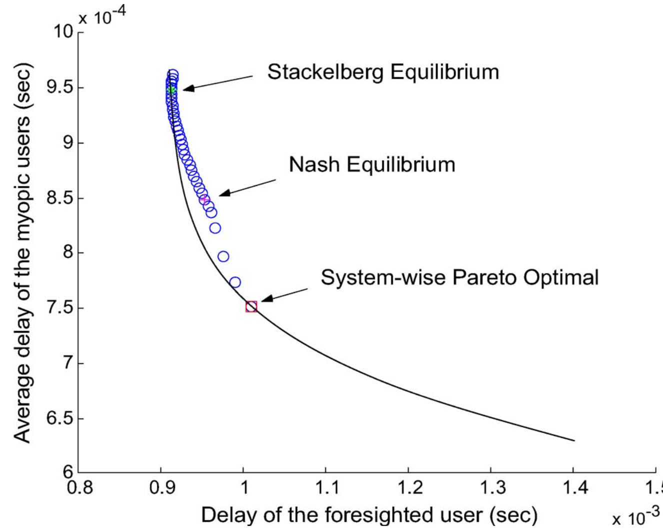

Fig. 5. Reaching the system-wide PO solution and the SE.

TABLE IV
RESULTS AT DIFFERENT EQUILIBRIUMS (CONCENTRATED NETWORK CASE)

	Action of the conjecturing learner a_{11}	Action of the naive learner a_{i1}	Delay of the conjecturing learner	Average delay of the naive learners	System Performance
NE	0.72	0.97	0.955 ms	0.848 ms	7.19
SE	0.95	0.78	0.914 ms	0.947 ms	7.45
System-wide optimal	0.66	1	1.011 ms	0.752 ms	7.00
Weighted-round robin [4]	0.875		1.00 ms		8.00

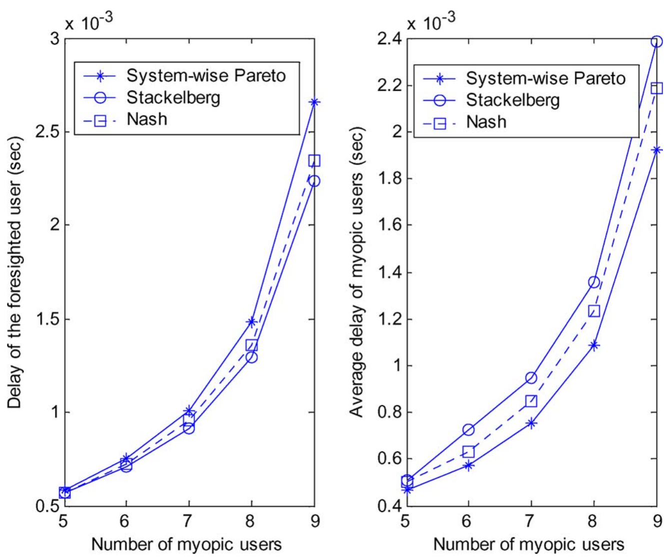

Fig. 6. Delay of the conjecturing learner at different equilibriums for various numbers of naive learners in the network.

over a path is proportional to the reciprocal of the delay, and based on the weights, users distribute more load to the path that provides lower delays. By following this load balancing solution, eventually, the delays and the remaining capacities become the same through the paths. However, our system-wide optimal solution outperforms these results and minimizes the system performance as proven in Proposition 7. Our system performance results outperform the existing solutions in all the various scenarios.

Next, we highlight the impact in terms of delay for the conjecturing learner (the foresighted user) and the naive learners (the myopic users), when there are different numbers of naive learners in the network. Fig. 6 shows the delay of the conjecturing learner at equilibrium, when there are various numbers of naive learners in the network. The results show that, as the number of naive learners in the network increases, the altruistic conjecturing learner will need to tolerate an increase in its experienced delay to reach the system-wide optimal solution. Beyond ten naive learners, the system-wide optimal solution is not reachable. This situation is also observed in network setting 1 (a diverse network setting). This is because the traffic ratio of the conjecturing learner to the total traffic in the network is not large enough to drive the equilibrium to the system-wide optimal solution (as discussed in [18]). On the contrary, the results also show that the conjecturing learner can benefit more in terms of delay when the number of the naive learners in the network increases.

TABLE V
SIMULATION RESULTS IN DIFFERENT SCENARIOS

Scenarios	Number of users using different solutions	Average delay of the conjecturing learners (ms)	Normalized system queue size (U^{sys}/total traffic rate)	GAP to the optimal system performance
1	Weighted round-robin [4]	1.0	1.0	0.125
2	All MY [18]	0.90	0.9	0.025
3	1 SF, 7 MY [18]	0.852	0.91	0.035
4	3 SF, 5 MY	0.877	0.918	0.043
5	5 SF, 3 MY	0.933	0.953	0.078
6	All RB	0.80	0.875	0
7	1 SF, 7RB	1.00	1.00	0.125
8	3 RB, 5 MY	0.864	0.911	0.034
9	3 RB, 2 SF, 3 MY	1.164	1.164	0.289

B. Multiple Conjecturing Learner Scenario

Here, we simulate the result when there are multiple conjecturing learners in the network. We simulate the resulting delays of the conjecture-based LBG using the concentrated network setting in the previous subsection. The only difference is that we now assume that all the eight users have traffic with the Poisson arrival rate $x_i = 1$ Mb/s. Hence, the total traffic rate is still 8 Mb/s (assuming 1000 bits/packet). These users can select three different load balancing solutions, i.e., the rule-based solution (RB) in Section IV-A, the self-interested conjecture-based solution (SF) in Section III-C, and the myopic solution (MY) in Section II-C. We discuss eight different scenarios in Table V. As a first benchmark (scenario 1), we deployed the weighted round-robin strategy proposed in [4]. In scenario 2, we simulate the case when all users are myopic (similar to the all-follower case in [18]). Then, we add a self-interested conjecturing learner, similar to the simulation results in the previous subsection. The self-interested conjecturing learner can have a smaller delay when the rest of the users are myopic (similar to the leader case in [18]). Next, we develop a worst case analysis. Based on Proposition 6, we can determine that the maximum tolerable number of self-interested learners is 2. When the number of these self-interested conjecturing learners is larger than 3, the average delay of these selfish conjecturing learners can be even worse than the average delay, which they experience when they adopt a myopic load balancing strategy. Hence, this gives incentives for these conjecturing learners to collaborate with each other. The rule-based solution (scenario 6) provides the minimum average delay for all the conjecturing learners and the minimum queue size of the system (minimum U^{sys}). However, we can see that, once a selfish user deviates from the rule, both the delay of the selfish user and the system queue size U^{sys} increase (scenario 7). Thus, if a conjecturing learner joins a network where the other users already comply with the rule-based solution, the users should collaborate with each other for their own benefit. Hence, their collaboration is self-enforcing rather than mandated by a protocol designer.

Moreover, comparing scenarios 8 and 4, we see that, even when the rest of the users are myopic, the three conjecturing learners will still have incentives to perform the collaborated rule-based solution. However, the delay performance seriously degrades when some conjecturing learners deliberately deviate from the prescribed rules, as we set two users to select SF in scenario 9 (these users can be categorized as malicious users). In this case, the rest of the conjecturing learners will have no incentive to comply with the rule-based solution. They will all become self-interested as in scenario 5.

VI. Conclusion

In this paper, we have studied the distributed load balancing problem in multiuser multipath networks. Although we have used a multipath network setting, it is important to note that the proposed method can be applied to other load balancing resource sharing systems. We have modeled the multiuser interaction using a conjecture-based LBG where naive learners and conjecturing learners coexist in the network. We have investigated two different operation scenarios. In the single conjecturing learner scenario, we have found that achieving the system-wide efficient solution is possible with no message exchanges among users, as long as the conjecturing learner is not selfish. In the scenario where multiple users are the conjecturing learners, we have shown that the resulting performance degrades when users are learning in an autonomous manner. Hence, we have discussed a rule-based solution for the conjecturing learners to collaboratively build the conjectures that minimize the system queue size in this paper. We have shown that, in such a multipath network, delay-sensitive users can efficiently minimize their delays when there is only one conjecturing learner managing the network or when all of the users comply with the rule-based solution. We have shown that, when each relay has the same capacity, the prescribed rule-based solution can be self-enforcing. Otherwise, the conjecturing learners can still minimize their own delays by autonomously building conjectures.

Appendix A
Proof of Proposition 5

First, we see that the objective function is a convex function, given that $0 \le \beta_{ij}^{(1)} \le 1$, $\beta_{ij}^{(0)} \ge 0$. Assume μ as the Lagrange multiplier. For $\forall r_j \in F_i$, the optimality conditions are

$$\frac{\beta_{ij}^{(0)}}{\left(\beta_{ij}^{(0)} + \beta_{ij}^{(1)}\lambda_{ij} - \lambda_{ij}\right)^2} = \mu \Rightarrow \lambda_{ij} = \tilde{D}_{ij} - \sqrt{\frac{1}{\mu}}\kappa_{ij}. \tag{18}$$

From constraint $\sum_{j=1}^{N} \lambda_{ij} = x_i$, we have

$$\sqrt{1/\mu} = \left(\sum_{r_j} \tilde{D}_{ij} - \lambda_i\right) / \sum_{r_j} \kappa_{ij}. \tag{19}$$

By substituting (19) into (18), we have $\lambda_{ij} = \tilde{D}_{ij} - \alpha_{ij}^{(f)}(\sum_{r_j} \tilde{D}_{ij} - \lambda_i)$ for the $\lambda_{ij} > 0$ case. ■

Appendix B
Proof of Proposition 8

Denote the total traffic through r_j as $\lambda_j = \sum_{i=1}^{M} \lambda_{ij}$. Assume $\boldsymbol{\mu} = [\mu_i, i = 1, \dots, M]$ as the Lagrange multipliers. The Lagrange function of minimizing $U^{\text{sys}}(\boldsymbol{\sigma})$ can be written as

$$\mathcal{L}(\boldsymbol{\sigma}, \boldsymbol{\mu}) = \sum_{j=1}^{N} \frac{\sum_{i=1}^{M} \lambda_{ij}}{C_j - \sum_{i=1}^{M} \lambda_{ij}} + \sum_{i=1}^{M} \mu_i(\lambda_i - \sum_{j=1}^{N} \lambda_{ij}). \tag{20}$$

For those $\lambda_{ij} > 0$, the optimality conditions are

$$\frac{C_j}{(C_j - \lambda_j)^2} = \mu_i \Rightarrow \lambda_j = C_j - \sqrt{\frac{C_j}{\mu_i}} \qquad \forall v_i \in \mathbf{V}. \tag{21}$$

Since we assume the nonsaturated condition, condition $\sum_{j=1}^{N} \lambda_j = \sum_{i=1}^{M} \lambda_i$ holds. Based on this, we can calculate the Lagrange multipliers, i.e.,

$$\sqrt{\frac{1}{\mu_i}} = \frac{\left(\sum_{r_j} C_j - \sum_{r_j} \lambda_j\right)}{\sum_{r_j} \sqrt{C_j}} \qquad \forall v_i. \tag{22}$$

Hence, the optimum solution will be

$$\lambda_j^* = C_j - \frac{\sqrt{C_j}}{\sum_{r_j} \sqrt{C_j}} \left(\sum_{r_j} C_j - \sum_{r_j} \lambda_j\right). \tag{23}$$

From the given $\beta_{ij}^{*(0)}=((C_{ij})^2/C_j)$ and $\beta_{ij}^{*(1)}=1-(C_{ij}/C_j)$, we have $\tilde{D}_{ij} = C_{ij}$ and $\kappa_{ij} = \sqrt{C_j}$ (see the definitions in Proposition 5). We see that $\lambda_{ij}^* = \max\{0, C_{ij} - (\sqrt{C_j}/\sum_{r_j} \sqrt{C_j})R\}$ is realized for all users. Then

$$\begin{aligned}\sum_{i=1}^{M} \lambda_{ij}^* &= \sum_{v_i \in \Psi} \left(C_{ij} - \frac{\sqrt{C_j}}{\sum_{r_j} \sqrt{C_j}} R_i\right) \\ &= \sum_{v_i \in \Psi} C_{ij} - \frac{\sqrt{C_j}}{\sum_{r_j} \sqrt{C_j}} \left(\sum_{v_i \in \Psi} \left(\sum_{r_j} C_{ij} - \lambda_i\right)\right)\end{aligned} \tag{24}$$

where Ψ represents a set of users whose $\lambda_{ij}^* > 0$. Denote $P = |\Psi|$ as the size of this set. Then, (24) can be viewed as

$$\begin{aligned}\lambda_j &= \sum_{i=1}^{M} \lambda_{ij}^* \\ &= PC_j - P\lambda_j + \lambda_j - \frac{\sqrt{C_j}}{\sum_{r_j} \sqrt{C_j}} \\ &\quad \times \left(\frac{\sum_{r_j} PC_j - \sum_{r_j} P\lambda_j + \sum_{r_j} \lambda_j - \sum_{v_i \in \Psi} x_i}{L}\right)\end{aligned}$$

$$\Rightarrow \lambda_j = C_j - \frac{\sqrt{C_j}}{\sum_{r_j} \sqrt{C_j}} \left(\sum_{r_j} C_j - \sum_{r_j} \lambda_j\right) = \lambda_j^*. \tag{25}$$

Hence, the solution is the optimal solution. ■

REFERENCES

[1] R. Rom, I. Cidon, and Y. Shavitt, "Analysis of multi-path routing," *IEEE/ACM Trans. Netw.*, vol. 7, no. 6, pp. 885–896, Dec. 1999.

[2] P. P. Pham and S. Perreau, "Increasing the network performance using multi-path routing mechanism with load balance," *Ad hoc Netw.*, vol. 2, no. 4, pp. 433–459, Oct. 2004.

[3] M. R. Pearlman, Z. J. Haas, P. Sholander, and S. S. Tabrizi, "On the impact of alternate path routing for load balancing in mobile ad hoc network," in *Proc. 5th ACM Int. Symp. MobiHOC*, 2000, pp. 3–10.

[4] L. Zhang, Z. Zhao, Y. Shu, L. Wang, and O. W. W. Yang, "Load balancing of multipath source routing in ad hoc networks," in *Proc. IEEE ICC*, 2002, vol. 5, pp. 3197–3201.

[5] Y. Ganjali and A. Keshavarzian, "Load balancing in ad hoc networks: Single path routing vs. multi-path routing," in *Proc. IEEE INFOCOM*, 2004, pp. 1120–1125.

[6] N. Jain, S. R. Das, and A. Nasipuri, "A multichannel CSMA MAC protocol with receiver-based channel selection for multihop wireless networks," in *Proc. IEEE Int. Conf. Comput. Commun. Netw.*, Scottsdale, AZ, USA, Oct. 2001, pp. 432–439.

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

[8] H. Zheng and L. Cao, "Device-centric spectrum management," in *Proc. IEEE DySPAN*, Nov. 2005, pp. 56–65.

[9] J. Huang, R. A. Berry, and M. L. Honig, "Spectrum sharing with distributed interference compensation," in *Proc. IEEE DySPAN*, Nov. 2005, pp. 88–93.

[10] H.P. Shiang and M. van der Schaar, "Distributed resource management in multi-hop cognitive radio networks for delay sensitive transmission," *IEEE Trans. Veh. Technol.*, vol. 58, no. 2, pp. 941–953, Feb. 2009.

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

[12] N. Mastronarde and M. van der Schaar, "A queuing-theoretic approach to task scheduling and processor selection for video decoding applications," *IEEE Trans. Multimedia*, vol. 9, no. 7, pp. 1493–1507, Nov. 2007.

[13] B. Awerbuch, Y. Azar, F. Grove, Y. Kao, P. Krishnan, and J. S.Vitter, "Load balancing in the L_p norm," in *Proc. 36th IEEE Comput. Soc. Press Symp. Found. Comput. Sci.*, Los Alamitos, CA, USA, 1995, pp. 383–391.

[14] E. M. Belding-Royer, "Multi-level hierarchies for scalable ad hoc routing," *Wireless Netw.*, vol. 9, no. 5, pp. 461–478, Sep. 2004.

[15] S. Suri, C. D. Toth, and Y. Zhou, "Selfish load balancing and atomic congestion games," in *Proc. ACM Symp. Parallel Algo. Arch.*, 2004, pp. 188–195.

[16] T. Roughgarden and E. Tardos, "How bad is selfish routing?" *J. ACM*, vol. 49, no. 2, pp. 235–259, Mar. 2002.

[17] Y. A. Korilis, A. A. Lazar, and A. Orda, "Achieving network optima using Stackelberg routing strategies," *IEEE/ACM Trans. Netw.*, vol. 5, no. 1, pp. 161–173, Feb. 1997.

[18] Y. A. Korilis, A. A. Lazar, and A. Orda, "Architecting noncooperative networks," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 7, pp. 1241–1251, Sep. 1995.

[19] S. T. Cheng and M. Wu, "Performance evaluation of ad-hoc WLAN by M/G/1 queuing model," in *Proc. Int. Conf. ITCC*, 2005, pp. 681–686.

[20] Y. Su and M. van der Schaar, "Conjectural equilibrium in multi-user power control games," *IEEE Trans. Signal Process.*, vol. 57, no. 9, pp. 3638–3650, Sep. 2009.

[21] F. H. Hahn, "Exercises in conjectural equilibria," *Scand. J. Econom.*, vol. 79, no. 2, pp. 210–226, Jun. 1977.

[22] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2000.

[23] D. P. Bertsekas, *Dynamic Programming and Optimal Control,* 3rd ed. Belmont, MA, USA: Athena Scientific, 2005.

[24] A. Proutiere, Y. Yi, and M. Chiang, "Throughput of random access without message passing," in *Proc. CISS*, Mar. 2008, pp. 509–514.

[25] D. Fudenberg and D. K. Levine, *The Theory of Learning in Games*. Cambridge, MA, USA: MIT Press, 1998.

[26] D. Fudenberg and J. Tirole, *Game Theory*. Cambridge, MA, USA: MIT Press, 1991.

[27] *Draft Supplement to Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS)*, IEEE Std. 802.11e/D5.0, Jun. 2003.

[28] L. Kleinrock, *Queuing Systems Volume I: Theory*. New York, NY, USA: Wiley, 1975.

[29] S. H. Shah, K. Chen, and K. Nahrstedt, "Available bandwidth estimation in IEEE 802.11-based wireless networks," in *Proc. ISMA/CAIDA 1st Bandwidth Estimation Workshop*, 2003, pp. 1–3.

[30] J. Park and M. van der Schaar, "Stackelberg contention games in multiuser networks," *EURASIP J. Adv. Signal Process., Spec. Issue Game Theory Signal Process. Commun.*, vol. 2009, no. 1, pp. 305978-1–305978-15, Jan. 2009.

[31] J. W. Lee, M. Chiang, and A. R. Calderbank, "Utility-optimal random-access control," *IEEE Trans. Wireless Commun.*, vol. 6, no. 7, pp. 2741–2751, Jul. 2007.

[32] J. W. Lee, A. Tang, J. Huang, M. Chiang, and A. R. Calderbank, "Reverse-engineering MAC: A non-cooperative game model," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 6, pp. 1135–1147, Aug. 2007.

[33] M. Félegyháazi, M. Cagalj, S. S. Bidokhti, and J.-P. Hubaux, "Noncooperative multi-radio channel allocation in wireless networks," in *Proc. IEEE INFOCOM*, May 2007, pp. 1442–1450.

[34] F. Wu, S. Zhong, and C. Qiao, "Globally optimal channel assignment for non-cooperative wireless networks," in *Proc. IEEE INFOCOM*, 2008, pp. 2216–2224.

[35] S.-J. Lee and M. Gerla, "AODV-BR: Backup routing in ad hoc networks," in *Proc. IEEE WCNC*, 2000, vol. 3, pp. 1311–1316.

Hsien-Po Shiang is currently working toward the Ph.D. degree with the Department of Electrical Engineering, University of California, Los Angeles, CA, USA.

In 2006, during his graduate study, he was with Intel Corporation, Folsom, CA, doing research on the overlay network infrastructure over wireless mesh networks. He published several journal papers and conference papers on these topics and was recently selected as one of the eight Ph.D. students for the 2007 Watson Emerging Leaders in Multimedia Workshop organized by IBM Research. His research interests are the cross-layer optimizations/adaptations for multimedia transmission over wireless mesh networks and dynamic resource allocation based on collaborative information exchange for delay-sensitive applications.

Mihaela van der Schaar (M'98–SM'04–F'10) received the M.S. and Ph.D. degrees in electrical engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 1996 and 2001, respectively.

She is currently an Associate Professor with the Department of Electrical Engineering, University of California, Los Angeles, CA, USA. Since 1999, she has been an active participant to the ISO MPEG standard to which she has made more than 50 contributions and for which she received three ISO recognition awards. She is the holder of 30 granted U.S. patents. She is also the editor (with P. Chou) of the book *Multimedia over IP and Wireless Networks: Compression, Networking, and Systems.*

Dr. van der Schaar received of the National Science Foundation CAREER Award in 2004, the IBM Faculty Award in 2005 and 2007, the Okawa Foundation Award in 2006, the Best IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY Paper Award in 2005, and the Most Cited Paper Award from the European Association for Signal Processing (EURASIP) Journal *Signal Processing: Image Communication* between 2004 and 2006.