Efficient and Scalable Distributed Autonomous Spatial Aloha Networks via Local Leader Election

Jiangbin Lyu, *Member, IEEE*, Yong Huat Chew, *Member, IEEE*, and Wai-Choong Wong, *Senior Member, IEEE*

***Abstract*—This paper uses a spatial Aloha model to describe a distributed autonomous wireless network in which a group of transmit–receive pairs (users) shares a common collision channel via slotted-Aloha-like random access. The objective of this study is to develop an intelligent algorithm to be embedded into the transceivers so that all users know how to self-tune their medium access probability (MAP) to achieve overall Pareto optimality in terms of network throughput under spatial reuse while maintaining network stability. While the optimal solution requires each user to have complete information about the network, our proposed algorithm only requires users to have local information. The fundamental of our algorithm is that the users will first self-organize into a number of nonoverlapping neighborhoods, and the user with the maximum node degree in each neighborhood is elected as the local leader (LL). Each LL then adjusts its MAP according to a parameter R, which indicates the radio intensity level in its neighboring region, whereas the remaining users in the neighborhood simply follow the same MAP value. We show that by ensuring $R \leq 2$ at the LLs, the stability of the entire network can be assured, even when each user only has partial network information. For practical implementation, we propose each LL to use $R = 2$ as the constant reference signal to its built-in proportional and integral controller. The settings of the control parameters are discussed, and we validate through simulations that the proposed method is able to achieve close-to-Pareto-front throughput.**

***Index Terms*—Control-theoretic tuning, distributed spectrum sharing, generalized Aloha game (GAG), Pareto front, spatial Aloha, stability.**

I. Introduction

GAME theory has been widely used to model the strategic interactions among intelligent devices sharing the same frequency band. In [2], Akkarajitsakul *et al.* provided a comprehensive review of the game models developed for different multiple-access schemes. In particular, Jin and Kesidis [3] proposed an Aloha game model whereby each user attempts to obtain a target rate by updating its medium access probability (MAP) in response to observed activities. It is further assumed in [4] that each user's target rate depends on its utility function and on its willingness to pay, and the same authors used a pricing strategy to control these target rates to be within the feasible region. More recently, for multichannel slotted Aloha, the joint MAP tuning and channel selection problem is studied in [5] under its proposed spectrum sharing model for *cognitive radio* [6] networks. The *stability* of the strategic interactions has been studied in these works; however, they are more suitably applied to the uplink random access channel in a fully connected centralized network.

Spatial reuse, which is also known as frequency reuse, is a powerful technique to improve the area spectral efficiency of multiuser communication systems. Cellular systems are examples whereby radios exploit the power falloff with distance and reuse the same frequency for transmission at spatially separated locations [7]. Similar ideas can be applied in the context of *dynamic spectrum access* (DSA) [8] and distributed *spectrum commons* [9] model, where different transmit–receive (Tx-Rx) pairs at a distance away are allowed to transmit simultaneously. Chen and Huang in [10] studied the distributed spectrum sharing problem in multichannel slotted Aloha with spatial reuse. The problem is formulated as a spatial channel selection game, under the assumption that the MAPs are fixed and preallocated. However, the efficiency, fairness, and scalability of such MAP allocations are not discussed. This paper addresses these issues for single-channel spatial Aloha networks.

In particular, we consider a distributed wireless network in which a group of Tx-Rx pairs shares a common collision channel via slotted-Aloha-like random access. These Tx-Rx pairs (users) are allowed to reuse the channel if they receive negligible interference from others. Such a network model is studied using stochastic geometry by Baccelli *et al.* in [11] and is called *spatial Aloha*. These approaches predict the achievable performance of the network but give no information about network stability during its operation. Indeed, it is mathematically challenging to obtain the stability conditions of the equilibrium solutions due to the *nonsymmetric* structure in the equations formulated for a partially connected network. How to enable the *autonomous* users to self-decide and yet achieve high efficiency, good fairness, and operation stability is the motivation behind this work.

For better utilization of spatial reuse and network scalability, *clustering* is used in our design. Clustering algorithms typically appear in the context of ad hoc [12] and wireless sensor networks (WSNs) [13]. These algorithms are mainly designed to perform cluster-based routing and to achieve scalable network management, whereas our proposed clustering method tries to resolve the concurrent transmission issue among Tx-Rx pairs

Manuscript received August 5, 2014; revised February 26, 2015, July 17, 2015, and October 27, 2015; accepted January 25, 2016. Date of publication February 8, 2016; date of current version December 14, 2016. This work was supported by the Singapore National Research Foundation through its IRC@SG Funding Initiative and administered by the Interactive and Digital Media Program Office. This paper was presented in part at the IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications, London, U.K., September 8–11, 2013. The review of this paper was coordinated by Dr. M. Dianati.

J. Lyu and W.-C. Wong are with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117583 (e-mail: jiangbin.lu@u.nus.edu; elewwcl@nus.edu.sg).

Y. H. Chew is with the Institute for Infocomm Research (I^2R), Agency for Science, Technology and Research, Singapore 138632 (e-mail: chewyh@i2r.a-star.edu.sg).

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.2016.2527058

and focuses on the attainable throughput and the stability of the spatial Aloha network. The effect of clustered topologies on the throughput of spatial Aloha is studied in [14] using stochastic geometry, which suggests that if the nodes can adjust their transmission parameters based on local information about their topological neighborhood, then the system performance can be improved. This also motivates our work.

Instead of viewing the distributed network from the statistical perspective, we zoom into the microlevel design of deployed networks. We specifically develop tailored algorithms so that the users have capability to enable the network to operate at one stable equilibrium solution that is close to the Pareto-front [15] throughput predicted by the *generalized Aloha games* (GAGs) [16]. In such games, each Tx-Rx pair in the distributed network is a player who competes to transmit using the slotted Aloha protocol. We prove the existence and uniqueness of the Nash equilibrium (NE) in terms of the MAP q of all players. We further derive the stability condition of the NE. However, this condition requires complete knowledge on the network topology, which is neither practical nor scalable. Although we further apply a heuristic algorithm in [1], which enables the autonomous users to heuristically search for the Pareto-front throughputs based on local information obtained from measurement, the approach nevertheless is limited by the long channel monitoring time and the poor convergence time, which significantly increase with the network size.

To provide *faster convergence* to a stable operating point, we apply the control-theoretic approach to tune the transmission parameters. Control theory has been used to provide a reliable and optimal configuration of 802.11 WLANs [17]. Similar approaches have also been applied to implement the distributed opportunistic scheduling algorithm [18], [19], however, without considering spatial reuse in their studies. Proportional and integral (PI) controllers are adopted in the aforementioned works and in feedback-based clock synchronization in WSNs [20].

The novelties of our proposed *spatial Aloha via local leader election* (SALE) scheme are listed as follows.

- Each user can self-regulate its transmission parameter to ensure that the network always operates in the stable region and yet achieves close-to-Pareto-front throughput by using only local information about its neighbors.
- Rigorous stability analysis is performed. By using the stability conditions derived in [16], we show that a local parameter *radio intensity metric* (RIM), which is denoted as R, can be used to indicate the cumulative radio intensity level of each user within its one-hop communication range—$R \leq 2$ for all users can guarantee network stability.
- The theory behind the design of the control system is presented. As commented by Patras *et al.* in [17], one of the key issues in building the control system is to discover a *constant reference signal* that relates to the desired system performance (e.g., maximum throughputs). In our case, a user that has the maximum node degree (ND) in a certain neighborhood is elected as the *local leader* (LL), and the remaining users in this neighborhood follow the same value of MAP. Each LL adjusts its MAP according to the value of RIM computed based on its local information and uses $R = 2$ as the constant reference signal in its built-in PI controller. As a result, the stability condition is satisfied, and the MAPs are adapted toward achieving Pareto-front throughput.
- The PI parameters are tuned to provide fast and smooth convergence of MAP, regardless of network size.
- Extensive simulations are performed to verify that the SALE scheme has a much faster convergence rate, better scalability, and better fairness than the heuristic algorithm, while achieving close-to-Pareto-front throughputs.

The SALE scheme may find its potential usage in DSA, where multiple Tx-Rx pairs share a common channel for transmission through spectrum commons [8], which is a spectrum resource that is owned or controlled jointly by a group of individuals [9]. Imagine if these users are not equipped with proper intelligence, it may end up that the users compete among themselves for transmission since all these selfish entities try to transmit aggressively. This could result in high contention probabilities and drive the system to function in the unstable region. Even if all users are willing to compromise, they will not know how to achieve this objective if there is a lack of rules and regulations. This work aims to develop an implementable methodology so that the users in a spectrum commons can work in a more controlled, more scalable, and fairer way through the embedded intelligence.

The rest of this paper is organized as follows. We review the spatial Aloha model in Section II. We dedicate Section III to giving the necessary background on GAGs and the heuristic algorithm. Then, we investigate the throughput optimality conditions and introduce the design parameter RIM in Section IV. Based on RIM, we design the control system for the SALE scheme in Section V. Then, we evaluate the system performance through simulations in Section VI. We conclude this paper in Section VII.

II. Spatial Aloha Model

Consider a distributed wireless network with N transmitters, where each transmitter has its unique designated receiver. Each Tx-Rx pair is a user who shares a common collision channel with other users, via slotted-Aloha-like random access. The conventional Aloha system is generalized to the scenario where there exists spatial reuse among the users. If the users are located sufficiently far apart, then they can transmit in the same frequency band simultaneously without causing any performance degradation to each other. Such a spatial reuse model can be characterized by an "interference graph" as in [10]. The interference estimation methods in [21] can be applied to obtain the interference graph.

As an example, three Tx-Rx pairs and the equivalent interference graph are shown in Fig. 1, where users 1 and 3 can transmit concurrently without collisions, but neither of them can transmit together with user 2. Such interference relations can be characterized by an *interference matrix* $\mathbf{A}$. For the topology given in Fig. 1, we have

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

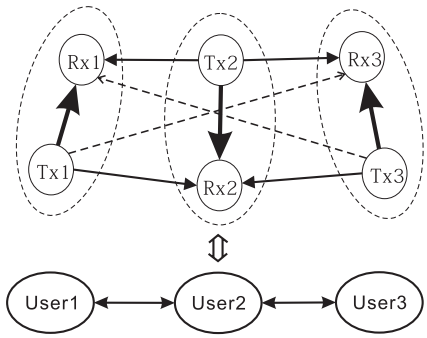


Fig. 1. Three Tx-Rx pairs and the interference graph [16]. In the upper part of the figure, the solid-thick arrow represents the transmission link from a transmitter to its designated receiver; the solid-thin and dash-thin arrows represent nonnegligible and negligible interference values, respectively.

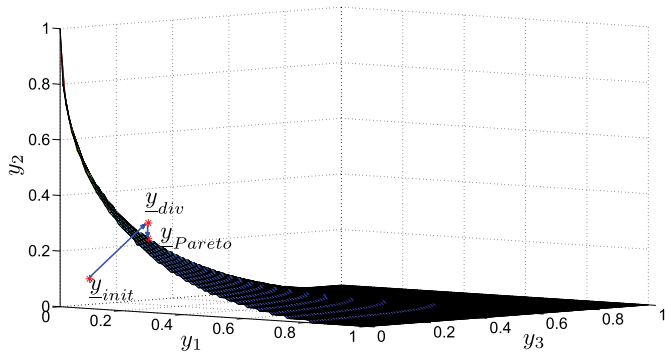


Fig. 2. Feasible target rate region topology in Fig. 1 [1].

in which $a_{12} = 1$ means user 2 is a *neighbor* of user 1, $a_{13} = 0$ means user 3 is not a neighbor of user 1, etc. We further assume that the interference graph is an undirected graph, then $\mathbf{A}$ is a symmetric matrix, i.e., $a_{ij} = a_{ji}\ \forall i, j$. By default, $a_{ii} = 0\ \forall i$.

The interference matrix characterizes the spatial distribution and frequency reuse capability of the users. Each user has different neighboring users who directly affect its transmission. For a successful transmission for user i, $i \in \mathcal{N} = \{1, 2, \ldots, N\}$, all of user i's neighbors (user j with $a_{ij} = 1$) should not transmit. We assume that every user's transmission queue is continuously backlogged, i.e., the transmitter of every user always has a packet to transmit to its designated receiver. Therefore, assuming that each user i chooses a MAP q_i, then the throughput θ_i can be obtained as

$$\theta_i = q_i \prod_{a_{ij}=1} (1 - q_j) \quad \forall i \in \mathcal{N}. \tag{1}$$

III. Close-to-Pareto-Front Throughputs

Here, we summarize some of the earlier results [1], [16] that form the basis of our design.

A. GAGs

If we assume that the users are selfish in nature and would choose MAP values to achieve their own objectives, then the spatial Aloha model can be formulated as a noncooperative game, which we name as the *generalized Aloha game* (GAG) [16]. In this game, each Tx-Rx pair is a player who competes for the channel to transmit. The objective of the game is for player i to select a suitable MAP q_i so that player i achieves its target rate $y_i\ \forall i \in \mathcal{N}$, with the lowest possible energy consumption, i.e., each player uses the smallest MAP to attain its target rate. The target rate combination $\underline{y} = [y_1, \ldots, y_N]$ is controlled by certain pricing strategies [4] or some commonly agreed adjusting rules that try to achieve Pareto efficiency [15].

We now formally state the GAG as follows.

Players: Distributed Tx-Rx pairs, $i \in \mathcal{N}$, who compete for a single collision channel to transmit via the slotted-Aloha-like random access scheme.

Actions: Each player i chooses a MAP $q_i \in [0, 1]\ \forall i \in \mathcal{N}$.

Objectives: Each player i $(i \in \mathcal{N})$ aims to minimize the energy consumption in attaining its target rate y_i, i.e.,

$$\begin{aligned} \min \quad & q_i \\ \text{s.t.} \quad & y_i = \theta_i = q_i \prod_{a_{ij}=1} (1 - q_j). \end{aligned} \tag{2}$$

To make the throughput θ_i approach the target rate y_i, player i's myopic best response strategy in the $(m+1)$th iteration is given as

$$q_i^{(m+1)} = \min \left\{ \frac{y_i}{\prod_{a_{ij}=1} \left(1 - q_j^{(m)}\right)}, 1 \right\} \quad \forall i \in \mathcal{N}. \tag{3}$$

Notice that we explicitly include the bound $q_i = 1$ in (3) to ensure that the mapping is within the compact domain $[0, 1]^N$. This would introduce an extraneous solution $\underline{q}^* = \underline{1}$, which happens when the system diverges to a dead-end situation with $\underline{q} = \underline{1}$ and $\underline{\theta} = \underline{0}$. Despite this undesirable situation, a stable NE solution would satisfy $q_i^* = y_i / \prod_{a_{ij}=1}(1 - q_j^*)\ \forall i \in \mathcal{N}$. Therefore, at such an operating point $\underline{q}^*$, the throughput θ_i is strictly equal to the target rate y_i, i.e., $\theta_i = q_i^* \prod_{a_{ij}=1}(1 - q_j^*) = y_i\ \forall i \in \mathcal{N}$. Apart from satisfying the equality constraints in (2), we have also proved that there exists a least fixed point that enables each player to operate with the minimal MAP concurrently. This optimal solution $\underline{q}^*$ corresponds to the unique NE of the GAG defined in (2).

The iteration process in (3) is then modified as a continuous-time game to study the convergence and stability of the NE. We construct a Lyapunov function and obtain a sufficient condition [16, Proposition 3] for the stability of the NE. Specifically, the NE $\underline{q}^*$ is stable if the matrix $\mathbf{C}(\underline{q}^*)$ is positive definite, whose entries are defined as

$$\left[\mathbf{C}(\underline{q}^*)\right]_{ij} = \begin{cases} 2, & i = j \\ -\frac{a_{ij} q_i^*}{1 - q_j^*} - \frac{a_{ji} q_j^*}{1 - q_i^*}, & i \neq j. \end{cases} \tag{4}$$

We now use the given condition to find a feasible region for the target rate $\underline{y}$. The word "feasible" here means that, at the stabilized operating point $\underline{q}^*$, the target rate $\underline{y}$ is achievable, i.e., the throughput $\underline{\theta} = \underline{y}$. As an illustration, we demonstrate in Fig. 2 the feasible target rate region (the region under the mesh surface) for the example in Fig. 1. The upper boundary of this region is the Pareto front [15], which is the upper bound of the feasible target rates. A network normally operates below the Pareto front to remain stable. How to drive the

users to self-regulate themselves under any topology so that the whole network operates in the feasible region yet close to the Pareto front is the interest of this study. Note that we introduce the stability condition derived from the GAG and use the results in our design. Thereafter, game theory is no longer the main theme; rather, the Tx-Rx pairs are trying to set their transmission rates autonomously to achieve close-to-Pareto-front throughputs predicted as if they are players in the GAG.

B. Autonomous Pareto Optimality Achieving Algorithm

The challenge is that the Pareto front is determined by the network as a whole, while each user usually has only limited local information and, hence, does not know how to achieve the Pareto-front throughput. As a result, if some or all the users are overdemanding, the resulting network can be unstable due to congestion. However, if the users set low target rates, the network is stable but the channel is not fully exploited.

In [1], a set of target-rate-adjusting rules is introduced to enable the users to improve their throughputs without affecting the network stability. Each user i only requires local information (e.g., measured throughput θ_i) to make a myopic best response and adjust its target rate. The distributed users heuristically search for Pareto-front target rates, and the system indeed settles down with a target rate $\underline{\hat{y}}_{\text{Pareto}}$ that is close to the Pareto front. However, the users using such a heuristic approach have to monitor the channel activities continuously and experience fluctuations before settling down since the network has to be driven into the unstable region to detect the crossing of the Pareto front. As the network size increases, the system experiences more fluctuations and takes longer to converge. In the following sections, we present the enabling theories behind our design of a fast self-adaptive network.

IV. Throughput Optimality Conditions

Here, we explain how a local parameter RIM can be defined and utilized by each user to judge for system optimality using only local information.

A. Optimal Conditions

For any MAP vector $\underline{q} = [q_1, q_2, \ldots, q_N] \in 0, 1]^N$, the N equations in (1) define a vector function $\underline{\theta}(\underline{q}) = [\theta_1, \theta_2, \ldots, \theta_N]$, whose value space is an N-dimensional region. The upper boundary ($\theta_i > 0\ \forall i \in \mathcal{N}$) of this region is formed by the critical values of $\underline{\theta}(\underline{q})$, with the critical points $\underline{q} \in (0,1)^N$. Mathematically, this corresponds to the Jacobian matrix $\mathbf{J} = \delta\underline{\theta}/\delta\underline{q}$ being singular [22], i.e., the determinant of $\mathbf{J}$ at the critical point is 0. Thus

$$\det(\mathbf{J}) = \left|\frac{\delta\underline{\theta}}{\delta\underline{q}}\right| = D(\underline{q}) \cdot \prod_{i=1}^{N} \frac{\prod_{a_{ij}=1}(1-q_j)}{1-q_i} = 0 \Rightarrow D(\underline{q}) = 0 \tag{5}$$

where

$$D(\underline{q}) = \begin{vmatrix} 1-q_1 & -a_{12}q_1 & \cdots & -a_{1N}q_1 \\ -a_{21}q_2 & 1-q_2 & \cdots & -a_{2N}q_2 \\ \vdots & \vdots & \ddots & \vdots \\ -a_{N1}q_N & -a_{N2}q_N & \cdots & 1-q_N \end{vmatrix}. \tag{6}$$

Similar derivations for the maximum throughputs of the original Aloha system (centralized, no spatial reuse) can be found in [23, Sec. III-B] by Abramson. Notice that when the network is fully connected, i.e., $a_{ij} = 1\ \forall i \neq j$, (5) is equivalent to [23, eqs. (26) and (27)]. In a specific case, if all MAPs are equal to q in a fully connected network of N users, then the maximum throughput is achieved at $q = 1/N$.

Since $D(\underline{q})$ involves all the MAPs and the complete interference matrix $\mathbf{A}$, it is not possible for an individual user to test for this optimal condition. In practice, generally only local information about neighbors is readily available for each user. Acquiring information beyond this will require large transmission overheads, and the design will suffer from large delay. We therefore look for certain suboptimal yet locally implementable testing conditions. The optimal condition in (5) gives the maximum throughput boundary and serves as a benchmark for any suboptimal schemes.

B. Suboptimal Conditions

From the analytical results in Section III-A, a sufficient condition for a target rate $\underline{y}$ to be feasible is that we can find a corresponding operating point $\underline{q}$ so that the matrix $\mathbf{C}(\underline{q})$ is positive definite. We retrospect on the analysis in [16] and find out a sufficient condition for $\mathbf{C}(\underline{q})$ to be positive definite, i.e., $\mathbf{C}(\underline{q})$ is strictly diagonally dominant [24]. Thus

$$R_i = \sum_{j=1, j\neq i}^{N} \left(\frac{a_{ij}q_i}{1-q_j} + \frac{a_{ji}q_j}{1-q_i} \right) < 2 \quad \forall i \in \mathcal{N}. \tag{7}$$

Here, we define R_i as the *radio intensity metric* (RIM) for user i. In other words, if (7) is satisfied, the corresponding target rate $\underline{y}$ is achievable, i.e., at the operating point $\underline{q}$, the throughput θ_i is equal to the target rate $y_i\ \forall i \in \mathcal{N}$, or the target rate falls within the feasible region (but not necessarily on the maximum throughput boundary). The converse of this, on the other hand, is not necessarily true.

We now examine the physical meaning of RIM. First, RIM is a local metric since R_i consists of q_i and the q_j terms with $a_{ij} = 1$, i.e., each user i only needs the information about its neighbors to calculate parameter R_i. Second, from (7), we observe that R_i is related to the number of user i's neighbors, i.e., $N_i = \sum_{j\in\mathcal{N}} a_{ij}$. Moreover, notice that R_i is monotonic with respect to q_i and q_j ($a_{ij} = 1$). In other words, the more neighbors with the higher MAPs, the larger the value of R_i. Therefore, RIM can be used to indicate the cumulative radio intensity level in the neighborhood of user i.

After all the $R_i\ \forall i \in \mathcal{N}$'s are obtained, condition (7) is suboptimal to (5) but is now locally implementable. The basic idea of the implementation is to tune the MAPs of all users so that condition (7) is *critically* satisfied, hence achieving a suboptimal network throughput. In other words, the MAPs of all users are tuned so that $R_l = 2\ \forall l \in \Omega$; $R_j < 2\ \forall j \notin \Omega$, where Ω is a certain subset of $\mathcal{N}$. Such an idea will be incorporated into our proposed scheme in Section V.

V. Spatial Aloha via Local Leader Election Scheme

Here, we present our proposed SALE scheme. The users first self-organize into a number of nonoverlapped neighborhoods.

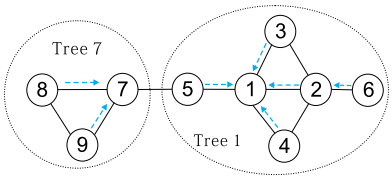


Fig. 3. Nine-user topology.

There are many ways to approach the Pareto-front surface defined in (5). One possible way is for the users in each neighborhood to adopt the same MAP to fulfill the fairness criterion, as it is also easier to analyze and implement.

A. LL Election Under Equal MAP

From (7), if we assume equal MAP in user i's neighboring region, then the value of R_i is related to the number of user i's neighbors given by $N_i = \sum_{j\in\mathcal{N}} a_{ij}$, which is also known as the ND of user i in graph theory [25]. If the MAPs of all users in a certain neighborhood are the same and gradually increase from zero, then that with the highest ND in this neighborhood will dissatisfy condition (7) first (assume an undirected link, i.e., $a_{ij} = a_{ji}, \forall\, i, j$). Therefore, unless the users have homogeneous NDs (regular graph), their RIM values cannot reach 2 at the same time. We call the user with the locally highest RIM value as a *local leader* (LL). The key principle behind the proposed scheme is to identify these LLs, since they are most likely to cause network instability due to interference from more neighbors. We next introduce how to identify these LLs, which consists of the following two steps.

1) Preliminary LL Election: Only two rounds of information exchange among each user and its neighbors are needed to complete the preliminary LL election process. In the first round, each user i broadcasts its identity number (ID) i to its neighbors. After the first round of broadcasting, each user i will be able to compute its ND N_i. In the second round, each user i broadcasts its ID i and its ND N_i and listens from its neighbors. User i then compares its ND and ID with those of its neighbors. If N_i is the largest, then user i will be aware of its role as an LL (for simplicity, when two or more candidates are connected, that with lower ID wins). Otherwise, user i would act as a *follower* of its neighbors k who has the highest ND (or the neighbor that has the same ND but lower ID). Such a user k is called the *parent* of user i, and user i is a *child* of user k. Note that a parent need not be the LL. For the example in Fig. 3, user 2 is the parent of user 6, but is also a child of LL 1. As a result, the whole network is grouped into several disjoint *trees* [25] ("neighborhood"), with each LL being the *root* of the tree and the users with no children being the *leaves*. The tree containing LL l is thus called tree l, which behaves similar to an independent neighborhood. The *height* H_l of tree l is the length from the root to a leaf that is farthest away. For the example in Fig. 3, there are two trees with LLs 1 and 7 being the roots, respectively. The child–parent relationship is denoted by dashed blue arrows. Tree 7 has a height of 1, whereas tree 1 has a height of 2, with the longest path being $6 \to 2 \to 1$.

Remark: There is no requirement for each user to have knowledge of neighbors beyond two hops. Although a neighborhood may have users two or more hops away, there is no need for each node to know who is in the neighborhood. Once the child–parent relationship has been identified, all users in the same neighborhood trace back to the same LL. The concept of a neighborhood or disjoint tree is just a virtual concept to explain the grouping of users who move as a group while adjusting its MAP.

2) Leadership Validation: The essential property of an LL l is that it has the locally highest RIM value R_l. Therefore, if $R_l \leq 2$, then all its neighbors would have a RIM value no greater than 2; hence, the stability condition in (7) is satisfied. The preliminary LL elected in the aforementioned process is that with the highest ND. When multiple candidates are connected, that with the smallest ID is elected. However, choosing candidates based on smaller ID might not always guarantee the LL to have the locally highest RIM value in cases when the tree under consideration is affected by other trees (more details will be discussed in Section V-D1). We therefore introduce a leadership validation mechanism to handle these exceptions.

During each iteration of update, all users monitor their own RIM values. If a user l_1 finds that $R_{l_1} > 2$, then it declares leadership and activates its PI controller to achieve $R_{l_1} = 2$. If there is a preliminary LL l_2 connected to user l_1, which hears the leadership declaration, l_2 should shut down its PI controller and regard user l_1 as its parent, i.e., both users exchange the leadership. If there is no neighboring preliminary LL, user l_1 is a new LL with a separate neighborhood that consists of all its followers. In cases when two or more connected users declare leadership, that with smaller ID wins.

An example will be given in Section V-D1 to illustrate the leadership validation process.

B. Control System Design

Although the theory predicts that the network can operate in the stable region, the MAP tuning process may still exhibit oscillatory behavior if improperly designed. An example is given by the heuristic algorithm [1], in which a user is able to detect the Pareto-front solution, provided that it detects the sudden drop in its throughput when it gradually increases its MAP. By doing so, the network is already driven out of the stable region, and the long monitoring process to collect the operating parameters will significantly affect the convergence rate. To improve the tuning process, a control-theoretic approach is used for each user to autonomously adapt themselves toward the suboptimal solution.

In each update iteration, each user i broadcasts its ID i, MAP q_i, and ND N_i to its neighbors. After the broadcasting, each user is able to compute its RIM value given in (7). Assume that the elected LLs make up the set Ω. Each LL $l \in \Omega$ sets its referenced RIM $R_{l,sp}$ to 2 and uses a PI controller [26, Ch. 10] to adjust its MAP q_l to achieve $R_l = 2$. Each follower $j \notin \Omega$ follows its parent k and sets $q_j(t+1) = q_k(t)$ (q_k ultimately follows the MAP q_l of its LL l). When $R_l = 2$ is achieved, the RIM of the followers in tree l will not be greater than 2. If $R_l = 2, \forall\, l \in \Omega$, then the conditions in (7) will be critically satisfied, thus providing a suboptimal network throughput. We will examine in Section V-E how close the design is to the Pareto-front solution predicted by (5).

The control system shown in Fig. 4 is designed as follows. For each LL l, the use of the PI controller is to eliminate the steady-state error [26, Ch. 10.1] while trying to achieve the desired reference signal $R_{l,sp} = 2$. The relationship between the input error signal $e_l(t) = R_{l,sp} - R_l(t)$ and the output $q_l(t)$

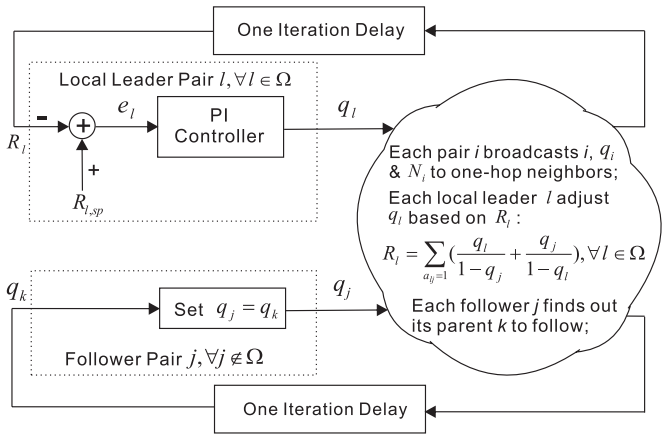


Fig. 4. SALE scheme.

of the PI controller in discrete time (sampling time interval = 1) can be expressed as

$$q_l(t) = K_{P,l} e_l(t) + K_{I,l} \sigma_l(t) \tag{8}$$

where $\sigma_l(t) = \sum_{n=0}^{t} e_l(n)$ is the integral function, and $\{K_{P,l} > 0, K_{I,l} > 0\}$ are the proportional and integral parameters for the PI controller in LL l. Alternatively, we can use the recursive form below that requires no memory on the integral value σ_l, i.e.,

$$q_l(t) = q_l(t-1) + K_{P,l}\left[e_l(t) - e_l(t-1)\right] + K_{I,l} e_l(t). \tag{9}$$

We can see from (8) that $q_l(t)$ stops changing after a certain time instant t_0, if and only if $e_l(t) = 2 - R_l(t) = 0 \; \forall \, t \geq t_0$ is achieved, i.e., $R_l = 2$ is achieved in the steady states. Taking the z-transform on both sides of (8), the transfer function (TF) for the PI controller can be obtained as

$$C_{PI}(z) = \frac{Q_l(z)}{E_l(z)} = K_{P,l} + \frac{K_{I,l}}{1 - z^{-1}}. \tag{10}$$

The PI controller enables smooth adaptation of the MAP to achieve this desired RIM value. As commented in [27, p. 174], when the goal is to asymptotically regulate the system output (RIM) to a "set point" ($R_{l,sp} = 2$), asymptotic regulation and disturbance rejection can be achieved by including an "integral action" in the controller. Moreover, by properly tuning the PI parameters, the PI controller can achieve a good tradeoff between response speed and steady-state convergence, which is a major challenge when designing adaptive algorithms.

In the following two sections, we perform the steady-state analysis, transient analysis, and PI parameter tuning for the given control system. We first consider a simple scenario with only one LL and then extend to multiple LLs.

C. Single-LL Case

With only one LL (thus only one tree) in the network, the ND is the highest at the LL and decreases toward the leaves. For example, if tree 7 in Fig. 3 does not exist and the network only comprises tree 1, the ND decreases from the root (LL 1) toward the leaves (users 3–6).

1) Steady-State Analysis: For the given scenario, the system finally settles down with all MAPs being equal, and the RIM for the only LL l at the steady state is given by

$$R_l = \sum_{j=1, j\neq l}^{N} \left(\frac{a_{lj} q_l}{1-q_j} + \frac{a_{jl} q_j}{1-q_l} \right) = \frac{2N_l \, q_l}{1-q_l} = R_{l,sp} = 2. \tag{11}$$

Therefore, when operating at the steady state, every user has a MAP equal to

$$\tilde{q}_l = \frac{1}{(N_l + 1)}. \tag{12}$$

Using tree 1 in Fig. 3 (i.e., without tree 7) as an example, the steady-state MAPs of all users are equal to $\tilde{q}_1 = 0.2$ as $N_1 = 4$.

In particular, for any fully connected network, the LL l is directly connected to all the remaining users; hence, $\tilde{q}_l = 1/(N_l + 1) = 1/N$, which coincides with the optimal condition in (5), i.e., Pareto-front throughput is achieved.

2) Throughput Sensitivity on RIM: The throughput at the LL is

$$\theta_l = q_l (1 - q_l)^{N_l}. \tag{13}$$

When analyzing the throughput sensitivity at the LL, we examine that the RIM value is perturbed by a small value ϵ, i.e., $R_l = 2 + \epsilon$, according to (11), we have

$$q_l = \frac{(2 + \epsilon)}{(2 + \epsilon + 2N_l)}. \tag{14}$$

By taking the derivative of θ_l on the perturbation ϵ, we can observe the local sensitivity [28, p. 251] of the throughput on the value of RIM at 2, i.e.,

$$\frac{\partial \theta_l}{\partial \epsilon} = \frac{\partial \theta_l}{\partial q_l} \cdot \frac{\partial q_l}{\partial \epsilon} = \frac{2N_l \cdot [1 - (N_l + 1) q_l] \cdot (1 - q_l)^{N_l - 1}}{(2 + \epsilon + 2N_l)^2}. \tag{15}$$

From (15), we have

$$\frac{\partial \theta_l}{\partial \epsilon}\Big|_{\epsilon=0} = \frac{\partial \theta_l}{\partial q_l}\Big|_{q_l = \frac{1}{N_l+1}} \cdot \frac{\partial q_l}{\partial \epsilon}\Big|_{\epsilon=0} = 0. \tag{16}$$

Therefore, the local sensitivity of the throughput θ_l on the value of RIM at 2 is 0, which means that the throughput θ_l is locally insensitive to small perturbations of the RIM value around 2. Similarly, we can obtain $\partial \theta_l / \partial \epsilon|_{\epsilon<0} > 0$ and $\partial \theta_l / \partial \epsilon|_{\epsilon>0} < 0$. These facts suggest that the throughput θ_l is being maximized when $\epsilon = 0$, i.e., when $R_l = 2$. Moreover, when $\epsilon < 0$ or, equivalently, $R_l < 2$, there is a margin for the throughput θ_l to be improved since $\partial \theta_l / \partial \epsilon > 0$, i.e., user l is operating in a slight-underload situation. Conversely, when $\epsilon > 0$ or, equivalently, $R_l > 2$, the throughput θ_l decreases with R_l since $\partial \theta_l / \partial \epsilon < 0$, i.e., user l is operating in a slight-overload situation.

Now, consider other users in the same neighborhood as user l. First, if other users have the same ND as user l (regular graph), then they will achieve the same MAP, RIM, and throughput. Therefore, their throughputs are also maximized with a RIM value of 2. The throughput with all users having a RIM value of 2 would be on the Pareto front. On the other hand, if another

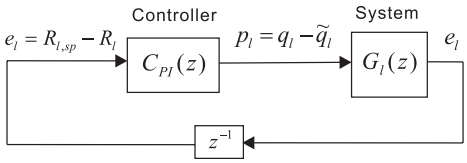


Fig. 5. Block diagram for the PI controller at the LL.

user j has a smaller ND N_j than that of user l, i.e., $N_j \leq N_l$, then the throughput of user j is

$$\theta_j = q_l(1-q_l)^{N_j} \tag{17}$$

where $q_l = \tilde{q}_l = 1/(N_l+1)$ is the common MAP value in this neighborhood. Since

$$\begin{aligned}\frac{\partial \theta_j}{\partial q_l}|_{q_l=\tilde{q}_l} &= [1-(N_j+1)\tilde{q}_l]\cdot(1-\tilde{q}_l)^{N_j-1} \\ &= \left[1-(N_j+1)\frac{1}{N_l+1}\right]\cdot\left(1-\frac{1}{N_l+1}\right)^{N_j-1} \geq 0.\end{aligned} \tag{18}$$

Hence, user j would be operating in the underload situation. As a result, since the followers normally do not fully exploit the transmission opportunities, the overall throughput solution is expected to be below the Pareto front. We will address this issue again in Section V-E.

3) Transient Analysis: Before the system converges to the steady-state operating point, there exists a transient period in which q is varying. Here, we use the control theory to derive a sufficient condition to guarantee system stability. The block diagram for the PI controller at the LL is shown in Fig. 5, where $C_{PI}(z)$ defined in (10) is the TF of the PI controller, $G_l(z)$ represents the TF of the spatial Aloha system to be controlled at the LL l, and z^{-1} represents one sample time delay in the z-domain.

Define $p_l = q_l - \tilde{q}_l$ and $e_l = R_{l,sp} - R_l$ as the input and output of the system $G_l(z)$, respectively, where $\tilde{q}_l$ is the desired operating point given in (12), and p_l is a small perturbation around $\tilde{q}_l$. For the simplicity of analysis, we would like to simplify $G_l(z)$ by assuming no information propagation delay between the LL and its neighbors, i.e., $q_j(t) = q_l(t)\ \forall\, j$ with $a_{l,j} = 1$. The TF $G_l(z)$ can then be obtained by linearizing the nonlinear function from (11), i.e.,

$$R_l = \frac{(2N_l q_l)}{(1-q_l)} \tag{19}$$

about the operating point $\tilde{q}_l$. Equation (19) can be expressed using p_l and e_l as

$$R_{l,sp} - e_l = \frac{(2N_l(p_l+\tilde{q}_l))}{(1-(p_l+\tilde{q}_l))}. \tag{20}$$

Taking the derivative on both sides of (20) and evaluating at the operating point $p_l = q_l - \tilde{q}_l = 0$, we have

$$\frac{de_l}{dt} = \frac{-2N_l}{[1-(p_l+\tilde{q}_l)]^2}|_{p_l=0}\cdot\frac{dp_l}{dt} = K_l\cdot\frac{dp_l}{dt} \tag{21}$$

where K_l is a constant related to N_l, i.e.,

$$K_l = \frac{-2N_l}{[1-\tilde{q}_l]^2} = \frac{-2N_l}{[1-1/(N_l+1)]^2} = \frac{-2(N_l+1)^2}{N_l}. \tag{22}$$

Then, we can discretize $e_l(t)$ and $p_l(t)$ and take the z-transform on both sides of (21), i.e.,

$$(1-z^{-1})E_l(z) = (1-z^{-1})K_l P_l(z). \tag{23}$$

The TF $G_l(z)$ can then be obtained as

$$G_l(z) = \frac{E_l(z)}{P_l(z)} = K_l. \tag{24}$$

In the following, we study the linearized model and ensure its stability by appropriately choosing the PI parameters. Note that the stability of the linearized model guarantees that our system is locally stable, which means that small perturbations around the desired operating point $\tilde{q}_l$ can all be absorbed, i.e., the control system will eventually converge to the operating point $\tilde{q}_l$ after being perturbed.

According to the control theory [29, eq. (6.22)], we need to check that the following TF is stable:

$$H(z) = \left[1-z^{-1}C_{PI}(z)G_l(z)\right]^{-1}C_{PI}(z). \tag{25}$$

Substituting (10) and (24) and after simplification, we have

$$H(z) = \frac{(K_{P,l}+K_{I,l})z^2 - K_{P,l}z}{z^2 - [1+K_l(K_{P,l}+K_{I,l})]z + K_l K_{P,l}}. \tag{26}$$

Applying the Schur–Cohn stability criterion [30, Sec. 3.2], a necessary and sufficient condition for a discrete-time system $H(z)$ to be stable is that its poles all lie within the unit circle, i.e., all the roots of the characteristic equation, i.e.,

$$C(z) = z^2 - [1+K_l(K_{P,l}+K_{I,l})]z + K_l K_{P,l} = 0 \tag{27}$$

should lie within the unit circle in the complex z-domain. Furthermore, for a second-order characteristic equation $A(z) = z^2 + a_1 z + a_2 = 0$, an equivalent stability condition is given by Jury's stability test [30, Th. 3.3], i.e.,

$$a_2 < 1; \qquad a_2 > -1 + a_1; \qquad a_2 > -1 - a_1. \tag{28}$$

If we apply Jury's stability test to (27), we have

$$K_l K_{P,l} < 1 \tag{29}$$

$$K_l K_{P,l} > -1 - [1+K_l(K_{P,l}+K_{I,l})] \tag{30}$$

$$K_l K_{P,l} > -1 + [1+K_l(K_{P,l}+K_{I,l})]. \tag{31}$$

Since $K_l < 0$, we only need $K_{P,l} > 0$ to satisfy (29). Equation (31) is equivalently reduced to $K_{I,l} > 0$. From (30), we have

$$-K_l(2K_{P,l}+K_{I,l}) < 2. \tag{32}$$

Hence, a sufficient condition to guarantee stability is obtained as follows:

$$\begin{cases} -K_l(2K_{P,l}+K_{I,l}) < 2 \\ K_{P,l} > 0, K_{I,l} > 0. \end{cases} \tag{33}$$

4) PI Parameter Tuning: In addition to guaranteeing stability, another consideration in selecting $\{K_{P,l}, K_{I,l}\}$ is to find a suitable tradeoff between fast convergence and transient oscillations. *Ziegler–Nichols* rules [26, Ch. 10.3] can be used for this purpose.

First, we compute parameter K_U, which is defined as the $K_{P,l}$ value that leads to instability when $K_{I,l}=0$, and parameter T_I, which is defined as the oscillation period under these conditions. According to Ziegler–Nichols rules, $K_{P,l}$ and $K_{I,l}$ can be configured as follows:

$$K_{P,l}=0.4K_U \tag{34}$$

$$K_{I,l}=\frac{K_{P,l}}{(0.85T_I)}. \tag{35}$$

To compute K_U, we first set $K_{I,l}=0$ in (32), and we have

$$K_{P,l}<\frac{1}{(-K_l)}. \tag{36}$$

From (36), we take K_U as the value where the system is about to turn unstable, i.e.,

$$K_U=\frac{1}{(-K_l)}. \tag{37}$$

Then, set $K_{P,l}$ according to (34), i.e.,

$$K_{P,l}=0.4K_U=\frac{0.4}{-K_l}=\frac{0.2N_l}{(N_l+1)^2}. \tag{38}$$

With the $K_{P,l}$ value that renders the system unstable, a given set of input values may take great changes up to every time interval, yielding an oscillation period of two time intervals ($T_I=2$). Then, from (35), we have

$$K_{I,l}=\frac{K_{P,l}}{0.85T_I}=\frac{K_{P,l}}{1.7}=\frac{2N_l}{17(N_l+1)^2}. \tag{39}$$

In summary, using the $\{K_{P,l},K_{I,l}\}$ values given in (38) and (39), the SALE scheme is guaranteed to converge fast to a stable steady-state operating point given in (12).

D. Multiple-LL Case

To study the scalability of the SALE scheme, we consider more general cases where there are multiple LLs in the network. We use the simple example given in Fig. 3, with users 1 and 7 being the LLs.

1) Steady-State Analysis: We have illustrated that after the LL election process, the whole network is partitioned into several disjoint trees. However, these trees are disjoint but their MAPs are not necessarily independent of each other. In our SALE scheme, the MAPs of all users in tree l are controlled by LL l, who adjusts its q_l based on R_l and only involves its neighbors. Therefore, if LL l is not directly connected to a user in other trees, then the MAP in tree l will not be affected by other trees. In Fig. 3, the MAP in tree 1 is not affected by tree 7; however, the steady-state MAP of LL 7 is affected by user 5, which is a follower in tree 1.

The steady state of the SALE scheme can then be determined as follows. For those independent trees, the analysis in Section V-C is readily applied to obtain the steady states. For LL l_1 who is affected by user m in tree l_2 (which normally happens when $N_{l_1}\le N_{l_2}$), it should wait until the steady state of $\tilde{q}_m=\tilde{q}_{l_2}$ is calculated before calculating $\tilde{q}_{l_1}$ based on $R_{l_1}=2$. It can be shown that to achieve $R_{l_1}=R_{l_2}=2$ with $N_{l_1}\le N_{l_2}$, LL l_1 should have $\tilde{q}_{l_1}\ge 1/(N_{l_1}+1)\ge\tilde{q}_{l_2}$ in general. In Fig. 3, where $l_1=7$, $l_2=1$, and $m=5$, the steady-state MAP in tree 1 can be obtained from (12) as $\tilde{q}_i=1/(N_1+1)=0.2\ \ \forall i\in\{1,\ldots,6\}$. Then, for LL 7, $R_7=\sum_{j=5,8,9}((q_7/(1-q_j))+(q_j/(1-q_7)))=4\cdot(q_7/(1-q_7))+(q_7/(1-\tilde{q}_5))+(\tilde{q}_5/(1-q_7))$. By setting $R_7=2$ and substituting in $\tilde{q}_5=0.2$, we have $\tilde{q}_7=0.2598$. Thus, the steady-state MAP in tree 7 is $\tilde{q}_i=0.2598,\forall i\in\{7,8,9\}$. It is easily verified that $\tilde{q}_7>1/(N_7+1)=0.25>\tilde{q}_1$. Notice that the steady-state MAP in tree 7 is *not* trivially equal to $1/(N_7+1)$ as in the single-LL case. Fortunately, our control system is able to converge to the steady state *automatically*.

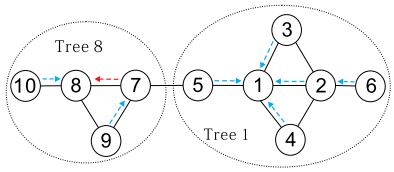


Fig. 6. Ten-user topology.

Now, we illustrate the leadership validation process by adding another user 10 as a neighbor of user 8, as shown in Fig. 6. Here, user 1 and user 7 are the preliminary LLs. User 8 now has the same ND as user 7, but user 7 is still elected as the preliminary LL due to its smaller ID. Using the given analysis, the preliminary LL 7 would push the MAP in the tree to be $\tilde{q}_i=0.2598\ \forall i\in\{7,8,9,10\}$, which achieves $R_7=2$, but this would push the RIM of user 8 to $R_8=2.1>2$, which should be avoided. Somewhere in the process of following the MAP of user 7, user 8 should find itself a more suitable LL than user 7. This example shows that the leadership should not be finalized simply based on smaller ID. In such a case, the leadership should be handed over to user 8. For the given example, in the midst of updating, when user 8 detects $R_8>2$, it declares leadership and activates its PI controller to achieve $R_8=2$. The preliminary LL 7 becomes a follower whose parent is the new LL 8. For simplicity, the child–parent relationship is only adjusted between the new and old LLs, whereas other followers stick to their original parent. For example, follower 9's parent is still user 7. The new child–parent relationship is shown in Fig. 6. Notice that after the leadership handover, the new LL 8 is not connected to any users in tree 1; hence, the tuning of MAP in tree 8 becomes independent of tree 1. The equilibrium MAP in the tree led by LL 8 is 0.25, whereas $R_8=2$, and $R_7=1.91$.

2) Transient Analysis and PI Parameter Tuning: The stability is guaranteed through the following argument. When all trees are independent, the transient analysis and PI parameter tuning in each tree follow those in Section V-C; hence, stability is guaranteed. Each LL l has its own PI controller, with the parameters $K_{P,l}$, $K_{I,l}$ given in (38) and (39), respectively. Again, the PI parameters only rely on local information N_l and, hence, can be immediately set after the LL election. The system TF to be controlled at LL l_1 follows (24) after applying (22), i.e.,

$$G_{l_1}(z)=K_{l_1}=-2\left(N_{l_1}+1\right)^2/N_{l_1}. \tag{40}$$

If LL l_1 is affected by user m in tree l_2 (e.g., see Fig. 3), this normally happens when $N_{l_1}\le N_{l_2}$. In this case, tree l_2 is independent of tree l_1, but tree l_1 depends on tree l_2 through

user m. Tree l_2 would reach the steady state first, with $\tilde{q}_m = \tilde{q}_{l_2} \leq 1/(N_{l_1}+1)$, which remains constant afterward. For tree l_1, due to the impact of $\tilde{q}_m$, from (19), the RIM for LL l_1 becomes

$$R'_{l_1} = \sum_{j=1, j\neq l_1}^{N} \left(\frac{a_{l_1 j} q'_{l_1}}{1-q_j} + \frac{a_{j l_1} q_j}{1-q'_{l_1}} \right) = \frac{2\left(N_{l_1}-1\right) q'_{l_1}}{1-q'_{l_1}} + \frac{q'_{l_1}}{1-\tilde{q}_m} + \frac{\tilde{q}_m}{1-q'_{l_1}} \quad (41)$$

where $\tilde{q}_m$ is now a constant, i.e., $d\tilde{q}_m/dt = 0$. In (41), we have used $'$ to denote the respective parameters to differentiate from the independent case. As a result of *fewer neighboring followers*, R'_{l_1} reacts more slowly to q'_{l_1} than in the independent case; hence, we expect the absolute gain $|K_{l_1}|$ of the system $G_{l_1}(z)$ to decrease.

Specifically, if we apply the same linearization procedure in Section V-C3 to (41), then the system $G'_{l_1}(z)$ to be controlled at LL l_1 is

$$G'_{l_1}(z) = K'_{l_1} = -\frac{2\left(N_{l_1}-1\right)}{\left(1-\tilde{q}'_{l_1}\right)^2} - \frac{1}{1-\tilde{q}_m} - \frac{\tilde{q}_m}{\left(1-\tilde{q}'_{l_1}\right)^2} \quad (42)$$

where $\tilde{q}'_{l_1}$ is the operating point that achieves $R'_{l_1} = 2$ in (41). It can be verified using Mathematica [31] that $|K'_{l_1}| < |K_{l_1}|$ for all $N_{l_1} \geq 2$ and $0 \leq \tilde{q}_m \leq 1/(N_{l_1}+1)$. Similar results can be verified if LL l_1 is affected by more neighbors $m_1, m_2, \ldots$ that belong to other trees.

Then, the PI parameters $\{K'_{P,l_1}, K'_{I,l_1}\}$ that guarantee system stability can be set from (38) and (39), i.e.,

$$K'_{P,l_1} = \frac{0.4}{-K'_{l_1}} > \frac{0.4}{-K_{l_1}} = K_{P,l_1} = \frac{0.2 N_{l_1}}{\left(N_{l_1}+1\right)^2} \quad (43)$$

$$K'_{I,l_1} = \frac{0.4}{-1.7K'_{l_1}} > \frac{0.4}{-1.7K_{l_1}} = K_{I,l_1} = \frac{2N_{l_1}}{17\left(N_{l_1}+1\right)^2}. \quad (44)$$

Since $\{K_{P,l_1}, K_{I,l_1}\}$ are easy to obtain using N_{l_1} only, LL l_1 uses $\{K_{P,l_1}, K_{I,l_1}\}$ in practice. More importantly, since the current system $G'_{l_1}(z)$ has a smaller absolute gain $|K'_{l_1}|$ due to fewer neighboring followers, LL l_1 is using the less aggressive PI parameters $\{K_{P,l_1}, K_{I,l_1}\}$ (see Section V-C4), and hence, system stability is guaranteed.

3) Throughput Sensitivity on RIM: Following the given settings, assume that LL l_1 is affected by user m in tree l_2 whose steady-state MAP is $\tilde{q}_m$, its throughput is given by

$$\theta'_{l_1} = q'_{l_1}\left(1-q'_{l_1}\right)^{N_{l_1}-1}(1-\tilde{q}_m). \quad (45)$$

When the RIM value in (41) is perturbed by a small value ϵ around 2, i.e., $R'_{l_1} = 2+\epsilon$, by taking the derivative on both sides of (41), we have $\partial q'_{l_1}/\partial\epsilon = -(1/K'_{l_1}) > 0$. Notice that when $\epsilon = 0$, i.e., $R'_{l_1} = 2$, it can be verified from (41) that the operating point $\tilde{q}'_{l_1}$ should be no greater than $1/N_{l_1}$. Therefore, the local sensitivity of throughput θ'_{l_1} on RIM R'_{l_1} around 2 is

$$\frac{\partial\theta'_{l_1}}{\partial\epsilon}|_{\epsilon=0} = \frac{\partial\theta'_{l_1}}{\partial q'_{l_1}}|_{q'_{l_1}\leq\frac{1}{N_{l_1}}} \cdot \frac{\partial q'_{l_1}}{\partial\epsilon}|_{\epsilon=0} \geq 0. \quad (46)$$

Therefore, in such cases, there are still some margins for throughput θ'_{l_1} to be improved, i.e., the network around the location of LL l_1 is operating in a slight-underload situation. Similar conclusions can be drawn when LL l_1 is affected by more users in other trees. We next discuss how close the throughput obtained by SALE is to the Pareto front.

E. "Distance" to Pareto Front

The Pareto-front surface is obtained if we apply the sufficient and necessary testing criterion (5) to the network. Any point on this surface gives a combination of throughputs achievable by all users while keeping the network operating in a stable condition. The solution obtained in SALE generally stays below the Pareto front for two reasons. First, the stability criterion (7) used in implementing the algorithm is only a sufficient condition. Second, some of the followers may not have fully exploited the transmission opportunities. Hence, the feasible throughput region obtained by SALE is only a subset to that obtained by using (5).

Based on the sensitivity analysis in Sections V-C2 and D3, in the homogeneous ND case (regular graph), all users have a RIM value of 2; hence, the throughput solution stays on the Pareto front. However, in most practical cases where there are variations in the NDs of users, the LL election allows for partitioning of the network into several local neighborhoods. Each LL, which has the highest ND in its neighborhood, uses a PI controller to achieve a RIM value of 2. The remaining nodes in the same neighborhood have a smaller ND than its LL and will have a RIM value no greater than 2. This suggests that, normally, the followers are operating under the underload condition or at a distance below the Pareto front obtained by using (5).

We attempt to characterize such a throughput margin with the optimal one obtained in (5) by defining the "distance to Pareto" d_{pareto}. When we obtain a solution $\underline{\theta} = [\theta_1, \theta_2, \ldots, \theta_N]$ in the SALE scheme, we continue to move in the direction $d \cdot \underline{\theta}$ ($d \geq 1$, i.e., proportionally increase the throughput of all users) until we find an operating point $\underline{q}$ that achieves $d_{\text{pareto}} \cdot \underline{\theta}$; beyond this point, there is no stable solution. In particular, when $d_{\text{pareto}} = 1$, the solution is on the Pareto front. In the simulation results, the SALE scheme achieves a close-to-Pareto-front throughput, with d_{pareto} below 1.05 for most of the topologies, i.e., less than 5% below the Pareto front.

F. Complexity, Scalability, and Overhead of SALE

We summarize the SALE scheme in Algorithm 1. The proposed scheme shows the following advantages.

1) Low Implementation Complexity: a) It takes only two rounds of information exchange among each user and its neighbors to complete the preliminary LL election. *b*) In each iteration, each user only needs to broadcast its ID, MAP, and ND to its neighbors. *c*) Each user only uses information about its neighbors to update its MAP in each iteration. *d*) Each LL l implements a simple PI controller to adjust its MAP q_l to achieve $R_l = 2$, which corresponds to a throughput close to the Pareto front. *e*) The PI parameter tuning can be autonomously done by the LL alone based on its ND N_l, which guarantees stability and fast convergence. *f*) Each follower j only needs to find out its parent k and simply sets $q_j(t+1) = q_k(t)$ in each iteration. *g*) Only in some situations, there is a need to change

the LL. Therefore, SALE can be autonomously implemented with low complexity.

Algorithm 1 The SALE Scheme

1: **Preliminary LL Election**:
2: Each user i broadcasts its ID to its neighbors;
3: Each user i computes its ND N_i;
4: Each user i broadcasts its ID and N_i to its neighbors;
5: Each user i compares N_i with the NDs of its neighbors. If N_i is the largest, then user i is elected as an LL. Otherwise, user i is a follower, whose parent is the neighbor with the largest ND. In cases when two or more candidates are connected, the one with smaller ID wins. Assume the LLs make up a set $\Omega \subset \mathcal{N}$.
6: **Control System**:
7: $R_{l,sp}=2,\ K_{P,l}=0.2N_l/(N_l+1)^2,\ K_{I,l}=2N_l/[17(N_l+1)^2], \forall l \in \Omega$;
8: $t=0$, $\underline{q}(t)=\underline{0}$; Declare$_i=0, \forall\, i \in \mathcal{N}$;
9: **repeat**:
10: Each user i broadcasts its MAP $q_i(t)$ to its neighbors;
11: Each user i computes $R_i(t)=\sum_{j=1,j\neq i}^{N}[a_{ij}q_i(t)/(1-q_j(t))+a_{ji}q_j(t)/(1-q_i(t))]$;
12: **for** each LL $l \in \Omega$ **do**:
13: $e_l(t+1)=R_{l,sp}-R_l(t)$;
14: $q_l(t+1)=q_l(t)+K_{P,l}[e_l(t+1)-e_l(t)]+K_{I,l}e_l(t+1)$;
15: **end for**
16: **for** each follower $j \notin \Omega$ **do**:
17: $q_j(t+1)=q_k(t)$, where user k is the parent of user j;
18: **end for**
19: **Leadership Validation**:
20: **for** each user $i \in \mathcal{N}$ **do**:
21: **if** Declare$_i=1$ **then**
22: User i joins Ω and becomes an LL. In cases when two or more connected users declare leadership, the one with smaller ID wins.
23: **else if** user $i \in \Omega$ and its neighbor declares leadership **then**
24: the preliminary LL quits from Ω, and follows the newly declared LL;
25: **end if**
26: **if** $R_i(t)>2$ **then**:
27: mark Declare$_i=1$ and declare leadership in the next round of broadcast;
28: **else** Declare$_i=0$;
29: **end if**
30: **end for**
31: $t=t+1$;
32: **until** Convergence, i.e., $R_l(t)=R_{l,sp}, \forall l \in \Omega$.

2) High Scalability: a) SALE is fully autonomous without any centralized controller. *b*) The whole network is grouped into several disjoint trees, and there is an LL controlling the MAPs in each tree. *c*) The Ziegler–Nichols rules adapt the PI parameters in (38) and (39) to various *user densities* (UDs; associated with different ND values N_l at the LL), thus guaranteeing fast and smooth convergence of MAPs at the LLs. *d*) Given a certain UD, if the number of users increases, the number of LLs also increases correspondingly, i.e., the whole network is grouped into more trees. As a result, the average number of users in a tree, as well as the tree height, will not change significantly as the network size grows. Therefore, when an LL l reaches the steady state ($R_l=2$), the corresponding MAP q_l would not take too many hops to reach all the followers in tree l. *e*) As we will see in Section VI-D, the SALE scheme converges in around 40 iterations, regardless of the UD or the number of users in the network. Therefore, SALE provides fast convergence with high scalability.

3) Overhead of Information Exchange: SALE requires local information exchange for leader election and MAP adaptation. The LL election requires two rounds of local information exchange about node ID and ND. Thereafter, the MAP adaptation requires each user to broadcast its ID and MAP to its one-hop neighbors.

As mentioned in [32], to handle the case with information exchange, slotted Aloha usually has a framed structure consisting of a control phase for the information exchange and a normal phase for data transmission. In our case, we embed the message in the packet header. In a simplified model, assume that each packet originally (i.e., in the heuristic approach) has a header field and constant packet size, which, at least, contains the user ID or address. For the need of our algorithm, we add three subfields to the header, i.e., ND subfield, MAP subfield, and leadership declaration subfield, and each occupies 8 bits, 16 bits, and 1 bit, respectively. We further assume that each packet occupies a time slot and has a packet size of L_S bits, e.g., 250 bytes = 2000 bits, then the newly added fractional overhead is then given by $25/L_S=25/2000=0.0125$.

The message exchange is realized in the following way. In each time slot, each user either sends a message to its neighbors according to its MAP or listens to the channel to receive a message from its neighbors. Assume that each iteration in the SALE scheme corresponds to packet transmissions in L_F slots, which is known as a frame, and all users are frame synchronized. Since the transmission of a packet is subject to collision, some packets (and hence the added subfields) will not be received correctly by some users. However, for a sufficiently large L_F value, each user is likely to receive at least one packet from all its neighbors and, hence, gather enough information about all its neighbors through the subfields embedded in the received packets. By the end of each iteration, all users then update the MAP subfields for the next iteration according to the SALE scheme. Since the SALE scheme relies on an accurate estimation of ND, we assume that each user counts and updates its ND in every L_{ND} slots. L_{ND} can be chosen to be sufficiently large to guarantee an accurate estimation of ND. In our simulations, we choose $L_{ND}=10L_F$ slots (i.e., ten iterations) to guarantee an accurate estimation.

VI. Performance Evaluation

In the simulations, we set the channel bit rate at $r_b=$ 20 Mb/s. Each packet in slotted Aloha has $L_S=2000$ bits. The slot time is $L_S/r_b=0.1$ ms. Each iteration in SALE corresponds to a frame of $L_F=100$ slots (i.e., 10 ms), and the users are frame synchronized. Each user counts and updates its ND in every $L_{ND}=10L_F=1000$ slots (i.e., ten iterations).

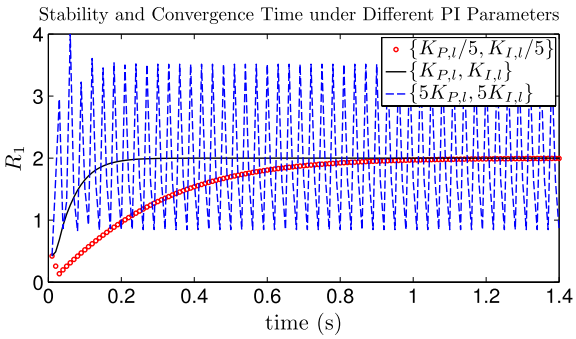


Fig. 7. Stability and convergence time under different PI parameters.

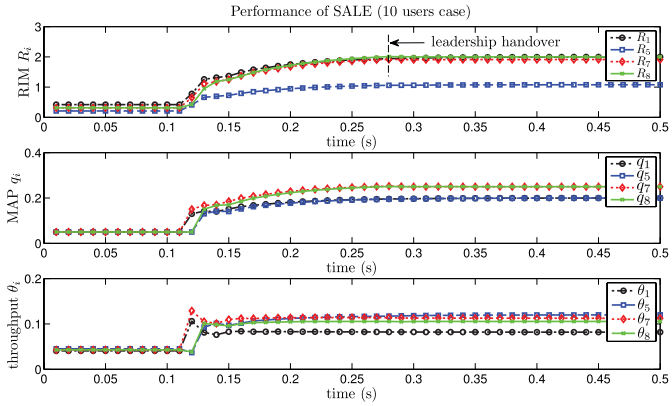


Fig. 8. Performance of SALE, topology in Fig. 6.

A. Parameter Tuning: Stability and Convergence Time

In Section V, we claim that the PI parameters given in (38) and (39) are able to guarantee system stability and fast convergence to the steady-state operating point. Here, we illustrate this by using the example in Fig. 6. Three sets of PI parameters are used in the PI controllers, respectively: $\{K_{P,l}/5, K_{I,l}/5\}$, $\{K_{P,l}, K_{I,l}\}$, and $\{5K_{P,l}, 5K_{I,l}\}$, where $\{K_{P,l}, K_{I,l}\}$ are obtained by (38) and (39). The algorithm starts with small initial MAPs, e.g., $q_i = 0.05\ \forall\, i$. The transient behaviors of R_1 are plotted in Fig. 7. Each iteration in SALE transmits $L_F = 100$ packets. The value of L_F is arbitrary and with the purpose of safeguarding the correct reception of the neighbors' information. This is similar to the use of L_{ND}. In fact, instances of message-passing failure in a frame are rarely captured in our simulations. In the rare occasion, if it happens, the transient variations caused by the delay of MAP feedback are well handled by the control system.

In Fig. 7, we can see that $\{K_{P,l}, K_{I,l}\}$ obtained by (38) and (39) enable the system to converge to the steady state $(R_1 = 2)$ within 30 iterations, i.e., 0.3 s. In contrast, the conservative PI parameters $\{K_{P,l}/5, K_{I,l}/5\}$ take around 120 iterations (i.e., 1.2 s) for the system to converge, whereas the aggressive PI parameters $\{5K_{P,l}, 5K_{I,l}\}$ render the system unstable. Similar results are observed for more complicated topologies, e.g., the 50-user case in Fig. 10 that will be introduced in Section VI-D1. Therefore, the PI parameters given by (38) and (39) indeed guarantee system stability with fast convergence.

B. Steady State, Optimality, and Fairness

For the same example given in Fig. 6, we demonstrate more details about the steady state, throughput optimality, and fairness among the users. The transient behaviors of RIM R_i, MAP q_i, and throughput θ_i are plotted in Fig. 8 for users 1, 5, 7, and 8, respectively.

It takes around 40 iterations (i.e., 0.4 s) for the system to converge. The preliminary LL election can be completed in the first ten iterations, in which each user counts its own ND and exchanges information with its neighbors. Starting from the 11th iteration, the PI controllers in LL 1 and LL 7 start working. At the 28th iteration, user 8 (green line in Fig. 8) detects $R_8 > 2$ and takes over the leadership from the preliminary LL 7 (red diamond in Fig. 8). In the steady state, both LL 1 and LL 8 have $R_1 = R_8 = 2$; follower 5 has $R_5 = 1.08$ and has MAP $\tilde{q}_5 = \tilde{q}_1 = 0.2$; follower 7 has $R_7 = 1.91$ and MAP $\tilde{q}_7 = \tilde{q}_8 = 0.25$. Regarding the throughput, generally, the users with a higher ND would have a lower throughput due to contentions from more neighbors. In this example, $\theta_1 < \theta_8 < \theta_7 < \theta_5$.

The distance to Pareto front d_{pareto} is used to evaluate throughput optimality. By applying our SALE scheme to Fig. 6, $d_{\text{pareto}} = 1.02$, which suggests only 2% loss between the achieved throughput $\underline{\theta}$ and the Pareto front. Therefore, although the suboptimal condition (7) is used in our design, the result is very close to the actual optimal.

We also evaluate the throughput fairness among the users. When spatial reuse is considered, different users at different spatial locations usually have different connectivity levels. As a result, those users with a higher ND usually receive more interference and, consequently, a lower throughput than those with a lower ND. Therefore, it is difficult to give an exact measurement of fairness in such a heterogeneous network. We make an attempt to take this spatial characteristic into consideration and weigh each user i's throughput by $N_i + 1$ (including user i and its neighbors), i.e., we define the *weighted throughput* for each user i as

$$\tilde{\theta}_i = (N_i + 1) \cdot \theta_i, \forall\, i \in \mathcal{N}. \tag{47}$$

Then, we compute Jain's fairness index [33] for the weighted throughput $\underline{\tilde{\theta}}$, i.e.,

$$\text{Jain}(\underline{\tilde{\theta}}) = \frac{\left(\sum_{i=1}^{N} \tilde{\theta}_i\right)^2}{N \cdot \sum_{i=1}^{N} \tilde{\theta}_i^{\,2}}. \tag{48}$$

Jain's index rates the fairness of an array of values $\underline{\tilde{\theta}} = [\tilde{\theta}_1, \tilde{\theta}_2, \ldots, \tilde{\theta}_N]$, where there are N users, and $\tilde{\theta}_i$ is the weighted throughput for the ith user. The result ranges from $1/N$ (worst case) to 1 (best case), and it is maximum when all users receive the same allocation. For our SALE scheme applied to Fig. 6, $\text{Jain}(\underline{\tilde{\theta}}) = 0.9921$, which is close to 1 and suggests good fairness among the users.

C. Comparison With Heuristic Algorithm

Here, we apply the heuristic algorithm in [1] to the same example in Fig. 6 and compare performance with that shown in Fig. 8. The algorithm in [1] assumes no information exchange among the users, and the users adapt their MAPs based on the measured throughput and channel idle rate, which require

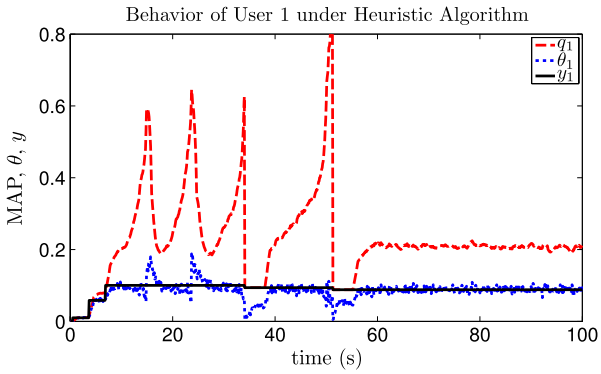


Fig. 9. Performance of the heuristic algorithm, topology in Fig. 6.

TABLE I
COMPARISON BETWEEN SALE AND HEURISTIC ALGORITHM

Topology	Scheme	$\overline{\theta_i}$	$\Sigma\theta_i$	$\overline{\theta}_{net,i}$	$Jain(\overline{\theta})$	d_{Pareto}	T_{conv}
Fig. 6	SALE	0.1246	1.246	0.1230	0.9921	1.02	∼0.4s
	Heuristic	0.1158	1.158	0.1158	0.9685	1.045	∼60s
Fig. 10	SALE	0.0578	2.889	0.0571	0.9906	1.025	∼0.4s
	Heuristic	0.0535	2.673	0.0535	0.9593	1.04	∼90s

a relatively accurate estimation. In the simulations, we choose each estimation period to consist of $L_I = 1000$ slots so that the adaptation in the heuristic algorithm works properly. Since the slot time is 0.1 ms, each estimation period lasts 0.1 s.

The transient states of the heuristic algorithm for user 1 are plotted in Fig. 9. As the users heuristically search for the Pareto front, the system experiences several fluctuations before settling down. The convergence time T_{conv} takes around 600 estimation periods, i.e., around 60 s, which is much longer than that when SALE is used. Since other users experience similar transient states as user 1, their behaviors are not plotted for brevity. The steady-state performance is summarized in the upper part of Table I for the SALE scheme and the heuristic algorithm. Both schemes achieve a throughput $\underline{\theta}$ close to the Pareto front ($d_{\text{pareto}} = 1.02$, 1.045, respectively), whereas SALE provides better fairness for the users (the heuristic algorithm has a lower Jain's index $= 0.9685$). Since SALE has an additional information exchange overhead of $25/L_S = 25/2000 = 0.0125$, we also compare the average net throughput $\overline{\theta}_{\text{net},i}$ in the table. Note we assume that both approaches use the same header, except for the three additional subfields in the SALE scheme, and the common parts of the header are included in computing the net throughputs.

D. Scalability of SALE

1) Fifty-User Case: Consider a distributed network with N users, which are randomly placed in a square region of a given area. For simplicity, we assume that all the distances between any transmitter and its designated receiver are much smaller than the distances between any two transmitters so that a Tx-Rx pair (user) can be represented by a single node in the topology. We further assume that all users have a transmission range of 5 unit length, and those users who are in each other's transmission range will have significant interference with each other, and the two users are said to be connected. Based on these assumptions, we can generate a random connected topology with 50 users in a square region of area 500 (units), as plotted in Fig. 10.

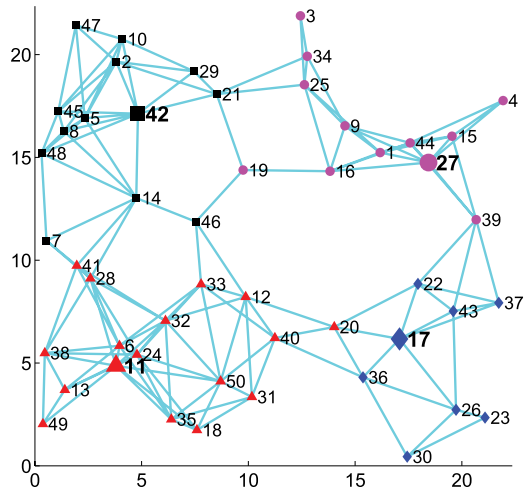


Fig. 10. Randomly generated connected topology with 50 users.

We apply the SALE scheme to Fig. 10, and the whole network is shown to be grouped into four trees, governed by LLs $l \in \Omega = \{11, 17, 27, 42\}$, respectively. The users in different trees are marked by different shapes and colors, e.g., the largest tree is marked with red triangles, which is governed by LL 11 with the highest ND $N_{11} = 10$. For comparison, we also apply the heuristic algorithm to the example in Fig. 10. The performance of the two schemes is summarized in the lower part of Table I.

The heuristic algorithm converges in around 90 s, which is longer than the 10-user case in Fig. 6. Therefore, the convergence time increases with the network size in the heuristic algorithm. In contrast, SALE still converges in around 0.4 s and, thus, is more efficient as the network size increases. After the system converges, both schemes achieve throughputs $\underline{\theta}$ close to the Pareto front ($d_{\text{pareto}} = 1.025$, 1.04, respectively), whereas SALE provides better fairness for the users, with a higher Jain's index $0.9906 > 0.9593$.

2) 100 Users With Various UDs: We define the UD as the number of users per unit area. For the 50-user case above, the UD is $50/500 = 0.1$. Now, we generate a sequence of random connected topologies with various UDs, by randomly scattering 100 users in a square region of various areas. In particular, when the square region has an area of 12.5, the network would become fully connected (the diagonal line length is equal to the transmission range of 5). We apply the SALE scheme to these topologies and summarize the results in the upper part of Table II.

As the UD increases from 0.1 to 8 (tend to a fully connected network), the network is grouped into fewer but bigger trees, and the maximum height of all trees gradually decreases to 1. As the UD increases, the total throughput decreases due to increased interference from more neighbors experienced by each user. However, regardless of the UD, the number of iterations for convergence is still around 40. Such fast convergence is guaranteed by the Ziegler–Nichols rules, which adapt the PI parameters in (38) and (39) to various UDs (associated with different ND values N_l at the LL).

TABLE II
SCALABILITY OF THE SALE SCHEME

Users	Area	UD	$\Sigma\theta_i$	$Jain(\tilde{\underline{\theta}})$	d_{Pareto}	T_{conv}	Leaders	$\max H_l$
100	12.5	8	0.370	1.0000	1	∼0.8s	1	1
	31.25	3.2	0.407	0.9998	1.00	∼0.8s	1	2
	62.5	1.6	0.524	0.9912	1.03	∼0.8s	1	2
	125	0.8	0.925	0.9881	1.05	∼0.4s	1	2
	250	0.4	1.492	0.9795	1.055	∼0.4s	1	4
	500	0.2	2.515	0.9856	1.04	∼0.4s	2	4
	1000	0.1	5.193	0.9823	1.02	∼0.4s	5	5
200	2000	0.1	9.950	0.9800	1.01	∼0.4s	12	5
400	4000		18.88	0.9692	1.01	∼0.4s	22	8
600	6000		27.26	0.9757	1.02	∼0.4s	32	5
800	8000		37.22	0.9771	1.015	∼0.4s	49	5
1000	10000		45.73	0.9799	1.01	∼0.4s	54	5

As the UD increases above 1.6, the individual throughput significantly drops, and packet collision probability increases and affects the success rate of passing the subfield information. Hence, we choose a larger frame of $L_F = 200$ slots in each iteration for the proposed SALE scheme. As a result, each iteration in these high-density topologies now takes 20 ms, and the convergence time is around 0.8 s. Note that similar problems exist for the heuristic algorithm in dense topologies, in which the individual user throughput is relatively small. To acquire a relatively accurate estimation for a small throughput value, a longer estimation period is required to suppress the relative error and keep the variance of the estimated throughput at a low level, in order for the MAP adaptation in the heuristic algorithm to work properly.

Regardless of the UD, the network still achieves close-to-Pareto-front throughput, with d_{pareto} around 1.05, i.e., only 5% below the Pareto front. In particular, d_{pareto} is equal to 1 in the fully connected case, thus verifying the statement in Section V-C1 that our SALE scheme achieves a throughput on the Pareto front in a fully connected network. Finally, the SALE scheme provides good fairness for all users, with Jain's index around 0.98 (close to 1).

3) Two-Hundred–1000-User Cases: Here, we keep the UD as 0.1 and increase the number of users by enlarging the area under consideration. Then, we generate a sequence of random connected topologies with $N = 200, 400, 600, 800,$ and 1000, respectively. We apply the SALE scheme to these topologies and summarize the results in the lower part of Table II.

As the number of users increases, the number of LLs also increases correspondingly, i.e., the whole network is grouped into more trees. As a result, the average number of users in a tree, as well as the tree height, will not change significantly as the network size grows. It is verified in Table II that the maximum tree height H_l remains around 6, regardless of the network size. Therefore, when an LL l reaches the steady state ($R_l = 2$), the corresponding MAP q_l would only take around six hops to reach all the followers in tree l. The SALE scheme converges in around 40 iterations, i.e., 0.4 s, regardless of the network size, hence providing fast convergence with high scalability.

In the steady state, SALE achieves a throughput $\underline{\theta}$ close to the Pareto front, with d_{pareto} around 1.02 for all cases. Consequently, the total throughput $\Sigma\theta_i$ increases almost linearly with the number of users in the network. Meanwhile, the Jain's index is around 0.97 for all cases, suggesting good fairness among the users.

VII. CONCLUSION AND FUTURE WORK

This paper has focused on spatial Aloha networks and attempted to approach global optimization of network throughput based on the limited spread of local information and realized the model's quick convergence and stability. Our proposed SALE scheme is introduced, which can be autonomously implemented by users using only local information. Specifically, a user with maximum ND in a certain neighborhood is elected as the LL, and the remaining users in this neighborhood simply follow the same MAP. The SALE scheme makes use of a sufficient condition previously derived for the spatial Aloha network, which ensures the network to operate in the stable region if the RIM parameter R at the LL(s) satisfies $R \leq 2$. In our design, the LL adjusts its MAP by a built-in PI controller to achieve $R = 2$. The resulting control system is sustained by mathematical foundations from control theory, which guarantees fast convergence and network stability. Most importantly, RIM is a local parameter based only on local information, which makes the SALE scheme easy and systematic to implement with high scalability. Through simulations, we validated the fast convergence of the system to a steady-state operating point with close-to-Pareto-front throughputs and good fairness among the users, while comparing with our previous heuristic algorithm. Future work can extend the SALE scheme to scenarios with dynamic topology changes [34] and with multiple channels [35], [36].

ACKNOWLEDGMENT

This work was carried out at the Sensor-Enhanced Social Media (SeSaMe) Center.

REFERENCES

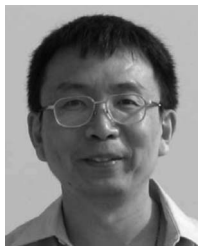
[1] J. Lyu, Y. H. Chew, and W.-C. Wong, "An autonomous Pareto optimality achieving algorithm beyond Aloha games with spatial reuse," in *Proc. IEEE PIMRC*, Sep. 2013, pp. 2674–2678.
[2] K. Akkarajitsakul, E. Hossain, D. Niyato, and D. I. Kim, "Game theoretic approaches for multiple access in wireless networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 13, no. 3, pp. 372–395, 3rd. Quart. 2011.
[3] Y. Jin and G. Kesidis, "Equilibria of a noncooperative game for heterogeneous users of an ALOHA network," *IEEE Commun. Lett.*, vol. 6, no. 7, pp. 282–284, Jul. 2002.
[4] Y. Jin and G. Kesidis, "A pricing strategy for an ALOHA network of heterogeneous users with inelastic bandwidth requirements," in *Proc. CISS*, Princeton, NJ, USA, Mar. 2002.
[5] K. Cohen, A. Leshem, and E. Zehavi, "Game theoretic aspects of the multi-channel ALOHA protocol in cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 11, pp. 2276–2288, Nov. 2013.
[6] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.
[7] D. Gutierrez-Estevez, I. Akyildiz, and E. Fadel, "Spatial coverage cross-tier correlation analysis for heterogeneous cellular networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 8, pp. 3917–3926, Oct. 2014.
[8] Q. Zhao and B. Sadler, "A survey of dynamic spectrum access," *IEEE Signal Process. Mag.*, vol. 24, no. 3, pp. 79–89, May 2007.
[9] J. Brito, "The spectrum commons in theory and practice," in *Proc. Stanford Technol. Law Rev.*, 2006, pp. 1–22.
[10] X. Chen and J. Huang, "Distributed spectrum access with spatial reuse," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 3, pp. 593–603, Mar. 2013.
[11] F. Baccelli, B. Blaszczyszyn, and P. Muhlethaler, "Stochastic analysis of spatial and opportunistic Aloha," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 7, pp. 1105–1119, Sep. 2009.
[12] J. Yu and P. Chong, "A survey of clustering schemes for mobile ad hoc networks," *IEEE Commun. Surveys Tuts.*, vol. 7, no. 1, pp. 32–48, 1st. Quart. 2005.

[13] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Comput. Commun.*, vol. 30, no. 14/15, pp. 2826–2841, Oct. 2007.
[14] J. Hoydis, M. Petrova, and P. Mahonen, "Effects of topology on local throughput-capacity of ad hoc networks," in *Proc. IEEE PIMRC*, Sep. 2008, pp. 1–5.
[15] M. J. Osborne and A. Rubinstein, "Terminology and Notation," in *A Course in Game Theory*. Cambridge, MA, USA: MIT Press, 1994, ch. 1.7.
[16] J. Lyu, Y. H. Chew, and W.-C. Wong, "Aloha games with spatial reuse," *IEEE Trans. Wireless Commun.*, vol. 12, no. 8, pp. 3932–3941, Aug. 2013.
[17] P. Patras, A. Banchs, P. Serrano, and A. Azcorra, "A control-theoretic approach to distributed optimal configuration of 802.11 WLANs," *IEEE Trans. Mobile Comput.*, vol. 10, no. 6, pp. 897–910, Jun. 2011.
[18] A. Garcia-Saavedra, A. Banchs, P. Serrano, and J. Widmer, "Distributed opportunistic scheduling: A control theoretic approach," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 540–548.
[19] A. Banchs, A. Garcia-Saavedra, P. Serrano, and J. Widmer, "A game-theoretic approach to distributed opportunistic scheduling," *IEEE/ACM Trans. Netw.*, vol. 21, no. 5, pp. 1553–1566, Oct. 2013.
[20] J. Chen, Q. Yu, Y. Zhang, H.-H. Chen, and Y. Sun, "Feedback-based clock synchronization in wireless sensor networks: A control theoretic approach," *IEEE Trans. Veh. Technol.*, vol. 59, no. 6, pp. 2963–2973, Jul. 2010.
[21] X. Zhou *et al.*, "Practical conflict graphs for dynamic spectrum distribution," *SIGMETRICS Perform. Eval. Rev.*, vol. 41, no. 1, pp. 5–16, Jun. 2013.
[22] V. I. Arnold, S. M. Gusein-Zade, and A. N. Varchenko, "Critical points and critical values of smooth maps," in *Singularities of Differentiable Maps*. Boston, MA, USA: Birkhaüser, 1985, vol. 1, ch. 1.2, p. 5.
[23] N. Abramson, "The throughput of packet broadcasting channels," *IEEE Trans. Commun.*, vol. COMM-25, no. 1, pp. 117–128, Jan. 1977.
[24] R. A. Horn and C. R. Johnson, "Positive definite matrices," in *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1990, ch. 7, pp. 396–405.
[25] R. Diestel, *Graph Theory*, 4th ed. New York, NY, USA: Springer-Verlag, 2010.
[26] K. J. Åström and R. M. Murray, *Feedback Systems*. Princeton, NJ, USA: Princeton Press Univ., 2008.
[27] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Englewood Cliffs, NJ USA: Prentice Hall, 2002.
[28] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge Univ. Press, 2004.
[29] T. Glad and L. Ljung, *Control Theory: Multivariable and Nonlinear Methods*. New York, NY, USA: Taylor and Francis, 2000.
[30] K. J. Åström and B. Wittenmark, *Computer-Controlled Systems: Theory and Design*, 3rd ed. Englewood Cliffs, NJ USA: Prentice-Hall, 1997.
[31] 'Mathematica 9.0 Student Edition," Wolfram Res. Inc., Champaign, IL, USA, 2012.
[32] C. Kissling, "On the stability of contention resolution diversity slotted ALOHA (CRDSA)," in *Proc. IEEE GLOBECOM*, Dec. 2011, pp. 1–6.
[33] R. Jain, D.-M. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer system," DEC Res. Rep. TR-301, 1984.
[34] Y. Zhang, S. He, and J. Chen, "Data gathering optimization by dynamic sensing and routing in rechargeable sensor networks," *IEEE/ACM Trans. Netw.*, vol. 24, no. 3, pp. 1632–1646, Jun. 2016.
[35] J. Chen *et al.*, "Dynamic channel assignment for wireless sensor networks: A regret matching based approach," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 1, pp. 95–106, Jan. 2015.
[36] J. Lyu, Y. H. Chew, and W.-C. Wong, "Multi-leader Stackelberg games in multi-channel spatial Aloha networks," in *Proc. IEEE VTC—Spring*, May 2015, pp. 1–6.

Jiangbin Lyu (S'12–M'16) received the B.Eng. degree (Honors) in control science and engineering from and completed the Chu Kochen Honors Program at Zhejiang University, Hangzhou, China, in 2011 and received the Ph.D. degree from the Graduate School for Integrative Sciences and Engineering (NGS), National University of Singapore (NUS), Singapore, in 2015, under the NGS scholarship.

He is currently a Postdoctoral Research Fellow with the Department of Electrical and Computer Engineering, NUS, where he uses game theory, control theory, and optimization techniques to design stable and efficient communication systems, particularly in the subfields of cognitive radios, device-to-device communication, and cross-layer network optimization.

Dr. Lyu received the Best Paper Award at the Singapore–Japan International Workshop on Smart Wireless Communications in 2014.



Yong Huat Chew (M'97) received the B.Eng., M.Eng., and Ph.D. degrees from National University of Singapore (NUS), Singapore, all in electrical engineering.

Since 1996, he has been with the Institute for Infocomm Research (I^2R, formerly known as the Center for Wireless Communications, NUS, and the Institute for Communications Research), which is an institute under the Agency for Science, Technology and Research, where he is currently a Senior Scientist. From 1999 to June 2013, he was also an Adjunct Faculty Member with the Department of Electrical and Computer Engineering, NUS. His research interests are in technologies related to highly spectrally efficient wireless communication systems.

Wai-Choong Wong (S'76–M'77–SM'93) received the B.Sc. (first-class honors) and Ph.D. degrees in electronic and electrical engineering from Loughborough University, Loughborough, U.K., in 1976 and 1980, respectively.

From 1980 to 1983, he was a Member of the Technical Staff with Crawford Hill Laboratory, AT&T Bell Laboratories, Holmdel, NJ, USA. From November 2002 to November 2006, he was an Executive Director of the Institute for Infocomm Research (I^2R), Agency for Science, Technology, and Research. He is currently a Professor with the Department of Electrical and Computer Engineering and the Deputy Director of the Interactive and Digital Media Institute, National University of Singapore (NUS), Singapore. Since joining NUS in 1983, he has been serving in various positions with the department, faculty, and university levels, including as the Head of the Department of Electrical and Computer Engineering from January 2008 to October 2009, the Director of the NUS Computer Center from July 2000 to November 2002, and the Director of the Center for Instructional Technology from January 1998 to June 2000. He is a coauthor of the book *Source-Matched Mobile Communications*. His research interests include wireless and sensor networks, ambient intelligent platforms, multimedia networks, and source matched transmission techniques, with over 250 publications and four patents.

Dr. Wong received the IEE Marconi Premium Award in 1989; the NUS Teaching Award (1989); the IEEE Third Millennium Award in 2000; the 2000 e-nnovator Awards, Open Category; and the Best Paper Award at the IEEE International Conference on Multimedia and Expo in 2006.