Distributed MAC over A General Multi-packet Reception Channel

Yanru Tang, Faeze Heydaryan, and Jie Luo
ECE Dept., Colorado State Univ., Fort Collins, CO 80523
Email: {yrtang, faezeh66, rockey}@colostate.edu

Abstract—**This paper investigates the problem of distributed medium access control in a time slotted wireless multiple access network with an unknown finite number of homogeneous users. Assume that each user has a single transmission option. In each time slot, a user chooses either to idle or to transmit a packet. With a generally-modeled link layer channel, a distributed medium access control framework is proposed to adapt transmission probabilities of all users to maximize an arbitrarily chosen symmetric network utility. Probability target of each user in the proposed algorithm is calculated based on a channel contention measure, which is defined as the success probability of a virtual packet. It is shown that the proposed algorithm falls into the classical stochastic approximation framework with guaranteed convergence when the contention measure can be directly obtained from the receiver. On the other hand, when the contention measure is not directly available, computer simulations show that a revised medium access control algorithm can still help the system to converge to the same designed equilibrium.[1]**

I. INTRODUCTION

Opportunistic channel access in a distributed wireless network often leads to unavoidable packet collision that needs to be resolved using a medium access control (MAC) protocol. Distributed adaptive MAC protocols can generally be categorized into tree splitting algorithms [1][10] and exponential back-off approaches [5][2]. While splitting algorithms can often achieve a relatively high system throughput, their function depends on the assumptions of instant availability of noiseless channel feedback and correct reception of feedback sequence. Unfortunately, both of these conditions can be violated in a wireless environment. Back-off algorithms, on the other hand, has proven to enjoy more trackable analysis. In back-off algorithms such as the 802.11 DCF protocol, each user transmits its packets randomly according to packet availability and an associated probability parameter. A user should decrease its transmission probability in response to a packet collision (or transmission failure) event, and increase its transmission probability in response to a transmission success event. Distributed probability adaptation in a back-off algorithm often falls into the stochastic approximation framework [5], with rigorously developed mathematical and statistical tools available for its performance analysis. It is well known that convergence proof of these algorithms often hold in the existence of measurement noise and feedback delay [6]. Practical back-off algorithms can also be analyzed using Markov models to characterize the impact of discrete probability updates [2].

In [5], a stochastic approximation model was proposed for distributed networking over a collision channel with an unknown finite number of users, each having a saturated message queue. By setting the transmission probability target of each user as a function of a locally measurable system variable, such as the channel idling probability, it was shown that the system can be designed to converge to a unique stable equilibrium. In the case of throughput maximization with homogeneous users, it was proposed that idling probability of the channel should be controlled toward the asymptotically optimal value of $1/e$. Most of the existing analysis of the splitting and the back-off algorithms either assume a throughput optimization objective and/or a simple collision channel model. While significant research efforts have been made to revise collision resolution algorithms to incorporate wireless-related physical layer properties, such as capture effect [7] and multi-packet reception [4], not much progress has been reported since the 1980s on integrating these extensions with the insightful stochastic approximation-based frameworks, such as those introduced in [5].

In [9][12][8], a new channel coding theory was developed for physical layer distributed communication systems that feature opportunistic channel access and packet collision. The coding theory enabled the derivation of fundamental limits of distributed communication systems. It also supported the derivation of a link layer channel model based on the physical layer channel and the coding details of data packets. This motivated the investigation on the impact of a general link layer channel to the design and optimization of collision resolution algorithms. One such attempt was presented in [11], where a distributed MAC algorithm was proposed to optimize a class of utility functions for a multiple access network over a class of multi-packet reception channels with an unknown number of users. While it was shown that the proposed algorithm can help the distributed network to converge to the desired equilibrium, both the utility function and the channel model assumed in [11] were quite special. It was not clear whether similar MAC algorithms can be developed for multiple access networks with more general settings.

In this paper, we give a positive answer to the above question. As in [11], we consider a distributed multiple

[1]This work was supported by the National Science Foundation under Grants CCF-1420608 and CNS-1618960. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation.

access wireless networking with an unknown finite number of homogeneous users, each is backlogged with a saturated message queue. Assume that users intend to maximize a general symmetric network utility. Under a general link-layer channel model, we propose a distributed MAC framework for each user to adapt its transmission probability according to a channel contention measure defined as the success probability of a virtual packet. We show that the proposed MAC algorithm falls into the classical stochastic approximation framework with guaranteed convergence, if the system has a unique equilibrium and if two key conditions can be met. Without knowing the user number, we show that one can develop an MAC algorithm to satisfy the required conditions and to place the unique equilibrium at a point that is close to optimal with respect to the (arbitrarily) chosen utility. Our work extends the basic framework of [5] from a simple collision channel model to a general link-layer channel. Such extension is enabled by the following key ideas. First, as opposed to using a locally measurable variable such as the channel idling probability [5], we use the success probability of a specific virtual packet as the channel contention measure. Second, we show that, with the help of the channel contention measure and two key monotonicity properties, each user can estimate the unknown user number and set its transmission probability target as a function of the (estimated) user number. Compared with the approach of controlling channel contention at a pre-determined level, as suggested in [5], the MAC algorithm presented in this paper has the capability of helping the system to achieve a performance closer to optimal especially when the user number is not large in value.

II. A Stochastic Approximation Framework

Consider a link layer wireless multiple access network with K homogeneous users (transmitters) and one receiver. K is finite and is assumed to be known neither to the users nor to the receiver. Time is slotted such that each slot equals the length of one packet. Assume that each user has a saturated message queue, a single transmission option plus an idling option. In each time slot, each user, say user k, determines whether to transmit a packet or to idle according to an associated probability parameter p_k. We use transmission probability vector $\boldsymbol{p} = [p_1, \cdots, p_K]^T$ to denote the transmission probabilities of all users.

At the end of each time slot t, upon receiving channel feedback, each user, say user k, derives a probability target, denoted by $\tilde{p}_k(t)$. User k then updates its transmission probability by $p_k(t+1) = (1-\alpha(t))p_k(t) + \alpha(t)\tilde{p}_k(t)$, where $\alpha(t) \geq 0$ is the step size parameter of time slot t. Let $\boldsymbol{p}(t) = [p_1(t), \cdots, p_K(t)]^T$ and $\tilde{\boldsymbol{p}}(t) = [\tilde{p}_1(t), \cdots, \tilde{p}_K(t)]^T$ denote the transmission probabilities and probability targets of all users in time slot t, respectively. Transmission probability vector is updated by

$$\boldsymbol{p}(t+1) = \boldsymbol{p}(t) + \alpha(t)(\tilde{\boldsymbol{p}}(t) - \boldsymbol{p}(t)). \tag{1}$$

Note that (1) falls into the stochastic approximation framework [6][3], where $\tilde{\boldsymbol{p}}(t)$ is often calculated based upon noisy estimates of system variables.

Define $\hat{\boldsymbol{p}}(t) = [\hat{p}_1(t), \cdots, \hat{p}_K(t)]^T$ as the "theoretical value" of $\tilde{\boldsymbol{p}}(t)$ when there is no measurement noise and no feedback error in time slot t. Let $E_t[\tilde{\boldsymbol{p}}(t)]$ be the expectation of $\tilde{\boldsymbol{p}}(t)$ conditioned on system state at the beginning of time slot t. We express $E_t[\tilde{\boldsymbol{p}}(t)]$ as follows

$$E_t[\tilde{\boldsymbol{p}}(t)] = \hat{\boldsymbol{p}}(t) + \boldsymbol{g}(t) = \hat{\boldsymbol{p}}(\boldsymbol{p}(t)) + \boldsymbol{g}(\boldsymbol{p}(t)), \tag{2}$$

where $\boldsymbol{g}(t)$ represents a potential bias term in the probability target derivation. Given the communication channel, both $\hat{\boldsymbol{p}}(t)$ and $\boldsymbol{g}(t)$ are functions of $\boldsymbol{p}(t)$, which is the transmission probability vector in time slot t.

Next, we present two conditions that are typically required for the convergence of a stochastic approximation algorithm.

Condition 1: (Mean and Bias) There exists a constant $K_m > 0$ and a bounding sequence $0 \leq \beta(t) \leq 1$, such that

$$\|\boldsymbol{g}(\boldsymbol{p}(t))\| \leq K_m \beta(t). \tag{3}$$

We assume that $\beta(t)$ is controllable in the sense that, for any $\epsilon > 0$, one can design protocols to ensure $\beta(t) \leq \epsilon$ for a sufficiently large t.

Condition 2: (Lipschitz Continuity) There exists a constant $K_l > 0$, such that

$$\|\hat{\boldsymbol{p}}(\boldsymbol{p}_1) - \hat{\boldsymbol{p}}(\boldsymbol{p}_2)\| \leq K_l \|\boldsymbol{p}_1 - \boldsymbol{p}_2\|, \qquad \text{for all } \boldsymbol{p}_1, \boldsymbol{p}_2. \tag{4}$$

If Conditions 1 and 2 are met, and $\alpha(t)$ and $\beta(t)$ are small enough, according to stochastic approximation theory [6][3], trajectory of $\boldsymbol{p}(t)$ under probability update (1) can be approximated by the following associated ordinary differential equation (ODE),

$$\frac{d\boldsymbol{p}(t)}{dt} = -\left[\boldsymbol{p}(t) - \hat{\boldsymbol{p}}(t)\right], \tag{5}$$

where we used the same notation t to denote the continuous time variable. Because all entries of $\boldsymbol{p}(t)$ and $\hat{\boldsymbol{p}}(t)$ stay in the range of $[0,1]$, any equilibrium $\boldsymbol{p}^*$ of the associated ODE must satisfy $\boldsymbol{p}^* = \hat{\boldsymbol{p}}(\boldsymbol{p}^*)$.

Convergence of the distributed probability adaptation is stated in the following two theorems which are quite standard for stochastic control algorithms.

Theorem 1: [6, Theorem 4.3] Let Conditions 1 and 2 hold. Assume that the associated ODE given in (5) has a unique stable equilibrium at $\boldsymbol{p}^*$. If $\alpha(t)$ and $\beta(t)$ satisfy the following conditions

$$\sum_{t=0}^{\infty} \alpha(t) = \infty, \sum_{t=0}^{\infty} \alpha(t)^2 < \infty, \sum_{t=0}^{\infty} \alpha(t)\beta(t) < \infty, \tag{6}$$

then under distributed probability adaptation given in (1), $\boldsymbol{p}(t)$ converges to $\boldsymbol{p}^*$ with probability one.

Theorem 2: [3, Thorem 2.3] Let Conditions 1 and 2 hold. Assume that the associated ODE given in (5) has a unique stable equilibrium at $\boldsymbol{p}^*$. Under distributed probability adaptation given in (1), for any $\epsilon > 0$, there exists a constant

$K_w > 0$, such that, for any $0 < \underline{\alpha} < \overline{\alpha} < 1$ satisfying the following constraint

$$\exists T_0 \geq 0, \underline{\alpha} \leq \alpha(t) \leq \overline{\alpha}, \beta(t) \leq \sqrt{\overline{\alpha}}, \forall t \geq T_0, \quad (7)$$

$\boldsymbol{p}(t)$ converges weakly to $\boldsymbol{p}^*$ in the following sense

$$\limsup_{t \to \infty} Pr\{\|\boldsymbol{p}(t) - \boldsymbol{p}^*\| \geq \epsilon\} < K_w \overline{\alpha}. \quad (8)$$

With convergence of the system guaranteed by Theorems 1 and 2, key objectives of the system design are to develop the distributed MAC algorithm to satisfy Conditions 1 and 2 and to place the unique system equilibrium at the desired point. Because users are homogeneous, due to symmetry, if a system equilibrium $\boldsymbol{p}^*$ is indeed unique, it must take the form of $\boldsymbol{p}^* = p^*\mathbf{1}$, with $\mathbf{1}$ being the vector of all ones. In other words, transmission probabilities of all users at the equilibrium must be identical. We choose to enforce such a property by requiring that all users should obtain the same transmission probability target in each time slot, as introduced below.

We assume that there is a virtual packet being transmitted in each time slot. Virtual packets of different time slots are identical. A virtual packet is an assumed packet with known coding details but it is not physically transmitted by any user in the system, i.e., the packet is "virtual". We assume that, without knowing the transmission/idling status of the users, the receiver can detect whether the reception of a virtual packet should be regarded as successful or not, and therefore can estimate its success probability. Note that, such detection tasks and their performance bounds have been extensively discussed in the distributed channel coding theory [9][12][8].

Let $q_v(t)$ denote the success probability of the virtual packet in time slot t. We assume that the receiver should estimate $q_v(t)$ and feed it back to all transmitters. We term $q_v(t)$ the "channel contention measure" because it is designed to serve as a measurement of the contention level of the link-layer multiple access channel. In the collision channel case when $q_v(t)$ equals the channel idling probability, feeding back $q_v(t)$ may not be necessary. So long as each user k knows the success probability of its own packet, denoted by $q_k(t)$, idling probability of the channel can be estimated by $(1-p_k(t))q_k(t)$. With a general link layer channel, however, estimating $q_v(t)$ may not always be possible if it is not fed back directly by the receiver. Upon receiving $q_v(t)$, each user calculates its probability target as the same function of $q_v(t)$, denoted by $\hat{p}(q_v(t))$. In other words, the vector transmission probability target is given by $\hat{\boldsymbol{p}}(t) = \hat{p}(q_v(t))\mathbf{1}$. Consequently, any system equilibrium $\boldsymbol{p}^*$ must take the form of $\boldsymbol{p}^* = p^*\mathbf{1}$, where p^* satisfies $p^* = \hat{p}(p^*)$ with $\hat{p}(p^*)$ being the theoretical probability target of the users given that all users have the same transmission probability p^*.

In a practical system, the measurement of $q_v(t)$ is likely to be corrupted by noise. We assume that, if users keep their transmission probability vector $\hat{\boldsymbol{p}}$ at a constant, and q_v is measured over an interval of Q time slots, then the measurement should converge to its true value with probability one as Q is taken to infinity. Other than this assumption, measurement noise is not involved in the discussion of the design objectives of meeting Conditions 1 and 2 and placing the unique system equilibrium at the desired point. Therefore, in the following section, we assume that $q_v(t)$ can be measured precisely and be fed back to the users. This leads to $\tilde{\boldsymbol{p}}(t) = \hat{\boldsymbol{p}}(t) = \hat{p}\mathbf{1}$. We will also skip the time index t to simplify the notations.

III. Distributed MAC Algorithm

Given the physical layer channel and coding details of the real and the virtual packets, we characterize the link layer channel using two sets of derived channel parameters. Define $\{C_{rj}\}$ for $j \geq 0$ as the "real channel parameter set", where C_{rj} is the conditional success probability of a real packet should it be transmitted in parallel with j other real packets. Similarly, define $\{C_{vj}\}$ for $j \geq 0$ as the "virtual channel parameter set", where C_{vj} is the success probability of the virtual packet should it be transmitted in parallel with j real packets. We assume that $C_{vj} \geq C_{v(j+1)} \geq 0$ for all $j \geq 0$. This implies that an increased number of parallel real packet transmissions should not improve the chance of a virtual packet getting through the channel. Let $\epsilon_v \geq 0$ be a pre-determined small constant. Define J_{ϵ_v} as the minimum integer such that $C_{vJ_{\epsilon_v}}$ is strictly larger than $C_{v(J_{\epsilon_v}+1)} + \epsilon_v$, i.e.,

$$J_{\epsilon_v} = \arg\min_j C_{vj} > C_{v(j+1)} + \epsilon_v. \quad (9)$$

According to the distributed channel coding theory [9][12][8], $\{C_{rj}\}$ and $\{C_{vj}\}$ can be derived from the physical layer channel model and the coding details of the packets. Therefore, we assume that both $\{C_{rj}\}$ and $\{C_{vj}\}$ should be known to the users and to the receiver.

We assume that the users intend to maximize a symmetric utility, denoted by $U(K, p, \{C_{rj}\})$, which is defined as a function of the unknown user number K, the common transmission probability p of all users, and the real channel parameter set $\{C_{rj}\}$. Write $p = \frac{x}{K}$. We define x^* using the following asymptotic utility optimization

$$x^* = \operatorname*{argmax}_x \lim_{K \to \infty} U\left(K, \frac{x}{K}, \{C_{rj}\}\right). \quad (10)$$

Let $b \geq 1$ be a pre-determined design parameter whose value will be introduced later. Define $p_{\max}$ as

$$p_{\max} = \min\left\{1, \frac{x^*}{J_{\epsilon_v} + b}\right\}. \quad (11)$$

We will show next that, without knowing the actual user number K, it is possible to set the unique system equilibrium at $p^*\mathbf{1} = \min\{p_{\max}, \frac{x^*}{K+b}\}\mathbf{1}$.

We intend to design a distributed MAC algorithm to set the unique system equilibrium at $p^*\mathbf{1}$ by maintaining channel contention at an appropriate level. Note that, given the virtual channel parameter set $\{C_{vj}\}$, channel contention measure $q_v(\boldsymbol{p}, K)$ is a function of the transmission probability vector $\boldsymbol{p}$ and user number K. If all users transmit with the same probability p, i.e., $\boldsymbol{p} = p\mathbf{1}$, then $q_v(p\mathbf{1}, K)$ is given by

$$q_v(p\mathbf{1}, K) = \sum_{j=0}^{K} \binom{K}{j} p^j (1-p)^{K-j} C_{vj}. \quad (12)$$

Upon obtaining q_v from the receiver, each user should first obtain a user number estimate, denoted by $\hat{K}$, and then set the corresponding probability target at $\tilde{p} = \hat{p} = \min\left\{p_{\max}, \frac{x^*}{\hat{K}+b}\right\}$. In the following, we will show that, for any $x^* > 0$, without knowing K, one can always choose an appropriate b and design a distributed MAC algorithm to converge to the desired equilibrium $p^*\mathbf{1} = \min\{p_{\max}, \frac{x^*}{K+b}\}\mathbf{1}$.

Convergence of the MAC algorithm to be introduced depends on two monotonicity properties presented below. First, the following theorem shows that, given user number K, $q_v(p\mathbf{1}, K)$ is non-increasing in p.

Theorem 3: Under the assumption that $C_{vj} \geq C_{v(j+1)}$ for all $j \geq 0$, $q_v(p\mathbf{1}, K)$ given in (12) satisfies $\frac{\partial q_v(p\mathbf{1},K)}{\partial p} \leq 0$. Furthermore, $\frac{\partial q_v(p\mathbf{1},K)}{\partial p} < 0$ holds with strict inequality for $K > J_{\epsilon_v}$ and $p \in (0, 1)$.

The proof of Theorem 3 is given in Appendix A.

Next, we introduce the "theoretical channel contention measure", denoted by q_v^*, to characterize the desired contention level of the system. Let $\hat{p} = \frac{x^*}{\hat{K}+b}$, and $N = \lfloor \hat{K} \rfloor$ be the largest integer below $\hat{K}$. We define a continuous function $q_v^*(\hat{p})$, which can also be viewed as a function of $\hat{K}$, as follows

$$q_v^*(\hat{p}) = \frac{\hat{p} - p_{N+1}}{p_N - p_{N+1}} q_N(\hat{p}) + \frac{p_N - \hat{p}}{p_N - p_{N+1}} q_{N+1}(\hat{p}), \quad (13)$$

where $p_N = \min\left\{p_{\max}, \frac{x^*}{N+b}\right\}$, $p_{N+1} = \min\left\{p_{\max}, \frac{x^*}{N+1+b}\right\}$, and

$$q_N(p) = \sum_{j=0}^{N} \binom{N}{j} p^j (1-p)^{N-j} C_{vj},$$

$$q_{N+1}(p) = \sum_{j=0}^{N+1} \binom{N+1}{j} p^j (1-p)^{N+1-j} C_{vj}. \quad (14)$$

Note that if the user number indeed equals $K = \hat{K}$ with $\hat{K} \geq x^* - b$, then $q_v^*(\hat{p})$ defined in (13) equals the channel contention measure $q_v(\boldsymbol{p}^*, K)$ at the desired equilibrium $\boldsymbol{p}^* = \frac{x^*}{K+b}\mathbf{1}$.

The following theorem gives the second monotonicity property, which shows that, given an arbitrary $x^* > 0$, with an appropriate choice of b, $q_v^*(\hat{p})$ is non-decreasing in $\hat{p}$.

Theorem 4: Let $x^* > 0$. If $b \geq \max\{1, x^* - \gamma_{\epsilon_v}\}$, with γ_{ϵ_v} being given by

$$\gamma_{\epsilon_v} = \min_{N, N \geq J_{\epsilon_v}, N \geq x^* - b} \frac{\sum_{j=0}^{N} j \binom{N}{j} \left(\frac{p_{N+1}}{1-p_{N+1}}\right)^j (C_{vj} - C_{v(j+1)})}{\sum_{j=0}^{N} \binom{N}{j} \left(\frac{p_{N+1}}{1-p_{N+1}}\right)^j (C_{vj} - C_{v(j+1)})}, \quad (15)$$

then $q_v^*(\hat{p})$ defined in (13) is non-decreasing in $\hat{p}$. Furthermore, if $b > \max\{1, x^* - \gamma_{\epsilon_v}\}$ holds with strict inequality, then $q_v^*(\hat{p})$ is strictly increasing in $\hat{p}$ for $\hat{p} \in (0, p_{\max})$.

The proof of Theorem 4 is given in Appendix B. We want to point out that, if ϵ_v is small enough to satisfy $C_{vj} = C_{v(j+1)}$ for all $j < J_{\epsilon_v}$, then we have $\gamma_{\epsilon_v} = J_{\epsilon_v}$. Otherwise, $\gamma_{\epsilon_v} \leq J_{\epsilon_v}$ is generally true.

We now propose the following distributed MAC algorithm.

Distributed MAC Algorithm:

1) Initialize the transmission probabilities of all users.
2) Over an interval of Q time slots, with $Q \geq 1$, the receiver measures the success probability of the virtual packet, denoted by q_v, and feeds q_v back to all transmitters.
3) Upon receiving q_v, each user, say user k, derives a probability target $\hat{p}$ by solving the equation of $q_v^*(\hat{p}) = q_v$. If a $\hat{p} \in [0, p_{\max}]$ satisfying $q_v^*(\hat{p}) = q_v$ cannot be found, user k sets $\hat{p}$ at $\hat{p} = p_{\max}$ when $q_v > q_v^*(p_{\max})$, or at $\hat{p} = 0$ when $q_v < q_v^*(0)$.
4) User k then updates its transmission probability by $p_k = (1-\alpha)p_k + \alpha\hat{p}$, where α is the step size parameter for user k.
5) Go back to step 2 till convergence.

Theorem 5: Given $x^* > 0$, let b be chosen to satisfy $b > \max\{1, x^* - \gamma_{\epsilon_v}\}$. With the proposed MAC algorithm, the system has a unique equilibrium at $\boldsymbol{p}^* = \min\{p_{\max}, \frac{x^*}{K+b}\}\mathbf{1}$. Furthermore, given user number K, the probability target $\hat{p}(\boldsymbol{p})$ as a function of transmission probability vector $\boldsymbol{p}$ satisfies Conditions 1 and 2. Consequently, transmission probability vector $\boldsymbol{p}$ converges to $\boldsymbol{p}^*$ in the sense explained in Theorems 1 and 2.

The proof of Theorem 5 is given in Appendix C.

In the above analysis, we did not pose any design constraint on the coding detail of the virtual packet. Convergence of the distributed MAC algorithm is guaranteed so long as parameter b is chosen to satisfy $b > \max\{1, x^* - \gamma_{\epsilon_v}\}$, where $\gamma_{\epsilon_v} = J_{\epsilon_v}$ if ϵ_v is small enough. However, one should note that optimality of the MAC algorithm can be affected by the value of b and J_{ϵ_v}. Both b and J_{ϵ_v} are determined by the virtual channel parameter set $\{C_{vj}\}$ which is dependent on the virtual packet design. Because the proposed MAC algorithm sets system equilibrium at $\boldsymbol{p}^* = \min\{p_{\max}, \frac{x^*}{K+b}\}\mathbf{1}$, there are two optimality concerns. On one hand, for a large user number K, it is a general preference that one should design virtual packets to allow a relatively small value of b, which implies a J_{ϵ_v} value not much smaller than x^*. On the other hand, for a small user number K, one should design virtual packets to support a J_{ϵ_v} value not much larger than x^*, so that $p_{\max} = \min\{1, \frac{x^*}{J_{\epsilon_v}+b}\}$ can be as close to 1 as possible. Considering both optimality concerns, a general guideline is to design coding parameters of the virtual packet such that J_{ϵ_v} is slightly smaller than x^* and b is close to 1.

IV. Interpreting The Contention Measure

Classical MAC protocols often assume that a user should get feedback from the receiver on whether its own packets are successfully received or not [1]. This enables each user, say user k, to measure the conditional success probability of its own packet transmissions, denoted by q_k. In this section, we consider the case when q_k is the only feedback available to user k. To simplify the discussion, we also assume that a virtual packet should have the same communication parameters as those of a real packet. In order to apply the MAC algorithm proposed in Section III, user k will need to interpret the success probability of the virtual packet based on the measurement of q_k. Because transmission activities of the users are mutually independent, q_k equals the conditional success probability of the virtual packet given that user k idles. Consequently, user k can calculate the success probability of the virtual packet by

$$q_v = (1 - p_k)q_k + p_k d_k, \tag{16}$$

where p_k is the transmission probability of user k, and d_k is the conditional success probability of the virtual packet given that user k transmits a packet[2]. Note that d_k can be easily calculated in special cases. For example, under a collision channel model, we have $d_k = 0$. In this case, $q_v = (1-p_k)q_k$ is the actual success probability of the virtual packet. However, for a general channel, d_k may not always be available at the transmitters unless additional feedback information is provided. When d_k is not available, we propose a two-step approach for each user to interpret d_k and hence the success probability of the virtual packet q_v, and then to update its transmission probability accordingly.

To explain the detail of the two-step approach, we need to define two auxiliary functions. More specifically, for an arbitrary estimated number of users $\breve{K}$, let $\breve{N} = \lfloor \breve{K} \rfloor$ denote the largest integer below $\breve{K}$. Let $\breve{p} = \min\{p_{\max}, \frac{x^*}{\breve{K}+b}\}$, $p_{\breve{N}} = \min\{p_{\max}, \frac{x^*}{\breve{N}+b}\}$ and $p_{\breve{N}+1} = \min\{p_{\max}, \frac{x^*}{\breve{N}+1+b}\}$, where b is a constant satisfying $b > \max\{1, x^* - \gamma_{\epsilon_v}\}$. We define auxiliary functions $q^*(\breve{p})$ and $d^*(\breve{p})$ as follows

$$\begin{aligned} q^*(\breve{p}) = & \frac{\breve{p} - p_{\breve{N}+1}}{p_{\breve{N}} - p_{\breve{N}+1}} \sum_{j=0}^{\breve{N}-1} \binom{\breve{N}-1}{j} \\ & \times \breve{p}^j (1-\breve{p})^{\breve{N}-1-j} C_{vj} \\ & + \frac{p_{\breve{N}} - \breve{p}}{p_{\breve{N}} - p_{\breve{N}+1}} \sum_{j=0}^{\breve{N}} \binom{\breve{N}}{j} \breve{p}^j (1-\breve{p})^{\breve{N}-j} C_{vj}, \\ d^*(\breve{p}) = & \frac{\breve{p} - p_{\breve{N}+1}}{p_{\breve{N}} - p_{\breve{N}+1}} \sum_{j=0}^{\breve{N}-1} \binom{\breve{N}-1}{j} \\ & \times \breve{p}^j (1-\breve{p})^{\breve{N}-1-j} C_{v(j+1)} \\ & + \frac{p_{\breve{N}} - \breve{p}}{p_{\breve{N}} - p_{\breve{N}+1}} \sum_{j=0}^{\breve{N}} \binom{\breve{N}}{j} \breve{p}^j (1-\breve{p})^{\breve{N}-j} C_{v(j+1)}. \end{aligned} \tag{17}$$

In the case when $\breve{K}$ takes an integer value, $q^*(\breve{p})$ is the conditional success probability of the virtual packet under the assumptions that the system has $\breve{K}$ users, user k idles, and all other users have the same transmission probability of $\breve{p}$. Similarly, $d^*(\breve{p})$ represents the conditional success probability of the virtual packet under the assumptions that the system has $\breve{K}$ users, user k transmits a packet, and all other users have the same transmission probability of $\breve{p}$.

[2]Extensions can be made to the case when a virtual packet is equivalent to the combination of R real packets by decomposing q_k in a similar way as shown in (16).

Next, we present the two-step approach that is suggested for each user to obtain its transmission probability target.

Step 1: Over an interval of $Q \geq 1$ time slots, each user, say user k, measures its own conditional success probability q_k. User k then obtains an intermediate transmission probability $\breve{p}$ by solving the equation of $q^*(\breve{p}) = q_k$. If a $\breve{p} \in [0, p_{\max}]$ satisfying $q^*(\breve{p}) = q_k$ cannot be found, user k sets $\breve{p}$ at $\breve{p} = p_{\max}$ when $q_k > q^*(p_{\max})$, or at $\breve{p} = 0$ when $q_k < q^*(0)$.

Step 2: In the second step, user k interprets channel contention measure q_v as

$$q_v = (1 - p_k)q_k + p_k d^*(\breve{p}). \tag{18}$$

An updated transmission probability target $\hat{p}$ for user k is then determined by solving $q_v^*(\hat{p}) = q_v$. As before, if a $\hat{p} \in [0, p_{\max}]$ satisfying $q_v^*(\hat{p}) = q_v$ cannot be found, user k sets $\hat{p}$ at $\hat{p} = p_{\max}$ when $q_v > q_v^*(p_{\max})$, or at $\hat{p} = 0$ when $q_v < q_v^*(0)$.

Note that when $\hat{p}$ is obtained by the two step approach, a convergence proof of the MAC algorithm is no longer available. This is because the two step approach does not guarantee that transmission probability targets obtained by different users should be identical. Therefore, the assumption that any equilibrium $\boldsymbol{p}^*$ must take the form of $\boldsymbol{p}^* = p^*\mathbf{1}$ is no longer valid. Nevertheless, in the following theorem, we show that the two-step approach is equivalent to a simplified one-step approach where user k directly uses $\breve{p}$ obtained in Step 1 as its transmission probability target.

Theorem 6: Let $x^* > 0$, and $b \geq \max\{1, x^* - \gamma_{\epsilon_v}\}$, where γ_{ϵ_v} is defined in (15). Suppose that each user, say user k, first obtains an intermediate transmission probability $\breve{p}$ and then determines its transmission probability target $\hat{p}$ by following the two-step approach. Then $\breve{p} \geq p_k$ implies $\hat{p} \geq p_k$, while $\breve{p} \leq p_k$ implies $\hat{p} \leq p_k$.

The proof of Theorem 6 is presented in Appendix D.

Theorem 6 suggests that each user can simplify the two step approach into Step 1 only and simply set the transmission probability target at $\hat{p} = \breve{p}$. In cases when the two-step approach does lead the system to the designed equilibrium, the simplified one step approach should also lead the system to the same equilibrium.

V. Simulation Results

Example 1: (Optimality) We first consider a distributed multiple access network with K users and a simple fading channel. In each time slot, with a probability of 0.3, the channel can support no more than $M_1 = 4$ parallel real packet transmissions, and with a probability of 0.7, the channel can support no more than $M_2 = 6$ parallel real packet transmissions[3]. The real channel parameter set $\{C_{rj}\}$ in this case is given by $C_{rj} = 1$ for $j < 4$, $C_{rj} = 0.7$ for $4 \leq j < 6$, and $C_{rj} = 0$ for $j \geq 6$. Assume that users intend to optimize

[3]Note that such a channel can appear if there is an interfering user that transmits a packet with probability 0.3 in each time slot. One packet from the interfering user is equivalent to the combination of two packets from a regular user.

the symmetric throughput weighted by a transmission energy cost of $E = 0.3$. With the number of users being K and all users transmitting with the same probability p, system utility $U(K, p, \{C_{rj}\})$ is given by

$$U(K, p, \{C_{rj}\}) = -EKp + \sum_{j=0}^{K-1} K \binom{K-1}{j} p^{j+1}(1-p)^{K-1-j} C_{rj}. \quad (19)$$

Correspondingly, x^* can be obtained from (10) as $x^* = 3.29$. Assume that a virtual packet should have the same coding parameters as those of a real packet. The virtual channel parameter set $\{C_{vj}\}$ is therefore identical to the real channel parameter set, i.e., $C_{vj} = C_{rj}$ for all $j \geq 0$. With $\epsilon_v = 0.01$, we have $\gamma_{\epsilon_v} = J_{\epsilon_v} = 3$. Therefore, we can set $b = 1.01 > x^* - \gamma_{\epsilon_v}$.

In Figure 1, we illustrate three utilities all as functions of

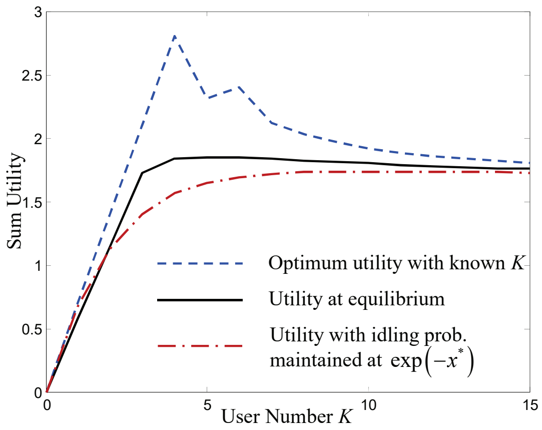

Fig. 1. Sum utility as functions of the number of users for distributed multiple access over a simple fading channel.

the number of users K. The solid curve represents the utility achieved by the proposed MAC algorithm at the designed equilibrium. The dashed curve represents the optimum utility under the assumption that number of users K is known, and this is not necessarily achievable without the knowledge of K. The dash-dotted curve represents the utility if we maintain the channel idling probability at its asymptotically optimal value of $\exp(-x^*)$, as suggested in [5][4]. This is equivalent to setting the transmission probabilities of all users at $1-\exp\left(-\frac{x^*}{K}\right)$. It can be seen that, the proposed MAC algorithm can achieve a higher utility value compared with the approach suggested in [5]. Achieved utility of the proposed MAC algorithm is also reasonably close to optimal when the number of users K is not close to M.

Example 2: (Convergence in a dynamic environment) Following Example 1, in each time slot, a channel state flag is randomly generated to indicate whether the channel can support the parallel transmissions of no more than 4 or 6 packets.

[4]While [5] also suggested to maintain other variables at their asymptotically optimal values, these alternative approaches do not lead to a better performance in this example.

Each user also randomly determines whether a packet should be transmitted according to its own transmission probability parameter. Whether a packet can go through the channel or not is then determined using the corresponding channel model. Assume that each user only knows the success/failure status of its own packets, and uses the simplified one step approach to calculate its transmission probability target. Each user k uses the following exponential moving average approach to measure q_k. q_k is initialized at $q_k = 1$. In each time slot, if user k transmits a packet, then q_k is updated by $q_k = (1-\frac{1}{300})q_k + \frac{1}{300}I_k$, where $I_k \in \{0, 1\}$ is an indicator of the success/failure status of the packet transmitted by user k in the current time slot. While this is different from the approach proposed in the distributed MAC algorithm, simulations show that an exponential averaging measurement of q_k can often lead the system to converge in a relatively smaller number of time slots. If user k does not transmit a packet in a time slot, then user k maintains the estimated q_k value during the time slot. The rest of probability adaptation proceeds according to the distributed MAC algorithm introduced in Section IV with a constant step size of $\alpha = 0.05$.

We assume that the system contains $K = 8$ users at the beginning. We say that the system starts with Stage 1. At the 3001th time slot, we assume that the system enters Stage 2 when 7 other users join the network. This leads to a total of $K = 15$ users. Each of the new users has its transmission probability initialized at zero and its packet conditional success probability initialized at one. Then at the 6001th time slot, we assume that the system enters Stage 3 when 5 users exit the network.

In Figure 2, we illustrate convergence behavior of the

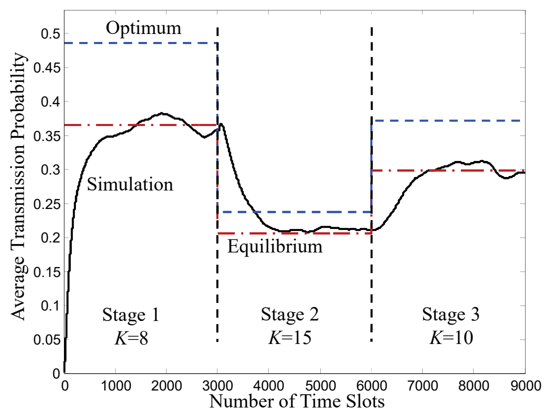

Fig. 2. Convergence behavior in average transmission probability of the active users of a multiple access network over three stages.

system in average transmission probability of the active users over the three stages. The corresponding optimal transmission probability (i.e., transmission probability that maximizes the symmetric utility) and the theoretical transmission probability at the designed equilibria are also illustrated in dashed lines and dash-dotted lines, respectively. While a theoretical convergence proof is not available in this case, we can see that in a dynamic environment when users join/exit the system, the proposed MAC algorithm has a reasonably good capability to help active users tracking the designed equilibrium.

APPENDIX

A. Proof of Theorem 3

The partial derivative of $q_v(p, K)$ with respect to p is given by

$$\frac{\partial q_v(p,K)}{\partial p} = \sum_{j=0}^{K} \binom{K}{j} j p^{j-1}(1-p)^{K-j} C_{vj} - \sum_{j=0}^{K} \binom{K}{j} p^j (K-j)(1-p)^{K-j-1} C_{vj} = \sum_{j=0}^{K-1} K \binom{K-1}{j} p^j (1-p)^{K-1-j} (C_{v(j+1)} - C_{vj}) \leq 0, \quad (20)$$

where the last inequality is due to the assumption that $C_{vj} \geq C_{v(j+1)}$ for all $j \geq 0$. Note that (20) holds with strict inequality if $K > J_{\epsilon_v}$ and $p(1-p) \neq 0$.

B. Proof of Theorem 4

We only prove the theorem under the assumption that $\frac{x^*}{N+b} \leq p_{\max}$. It is easy to extend the derivations to the case when $\frac{x^*}{N+b} > p_{\max}$.

According to the definition of $q_v^*(\hat{p})$ in (13), we have

$$\frac{dq_v^*(\hat{p})}{d\hat{p}} = \frac{q_N(\hat{p}) - q_{N+1}(\hat{p})}{p_N - p_{N+1}} + \frac{\hat{p} - p_{N+1}}{p_N - p_{N+1}} \frac{dq_N(\hat{p})}{d\hat{p}} + \frac{p_N - \hat{p}}{p_N - p_{N+1}} \frac{dq_{N+1}(\hat{p})}{d\hat{p}}. \quad (21)$$

Write $\hat{K} = N + 1 - \lambda$ with $\lambda \in (0, 1]$. We have

$$\hat{p} - p_{N+1} = \frac{x^*}{\hat{K}+b} - \frac{x^*}{N+1+b} = \frac{\lambda}{N+1+b}\hat{p}, \quad (22)$$

and

$$p_N - \hat{p} = \frac{x^*}{N+b} - \frac{x^*}{\hat{K}+b} = \frac{1-\lambda}{N+b}\hat{p}. \quad (23)$$

Meanwhile, because function $q_{N+1}(\hat{p})$ can be decomposed as

$$q_{N+1}(\hat{p}) = \sum_{j=0}^{N+1} \binom{N+1}{j} \hat{p}^j (1-\hat{p})^{N+1-j} C_{vj} = \hat{p} \sum_{j=0}^{N} \binom{N}{j} \hat{p}^j (1-\hat{p})^{N-j} C_{v(j+1)} + (1-\hat{p}) \sum_{j=0}^{N} \binom{N}{j} \hat{p}^j (1-\hat{p})^{N-j} C_{vj}, \quad (24)$$

we have

$$q_N - q_{N+1} = \hat{p} \sum_{j=0}^{N} \binom{N}{j} \hat{p}^j (1-\hat{p})^{N-j} (C_{vj} - C_{v(j+1)}). \quad (25)$$

Furthermore, the derivatives of $q_N(\hat{p})$ and $q_{N+1}(\hat{p})$ are given by

$$\frac{dq_N(\hat{p})}{d\hat{p}} = -\sum_{j=0}^{N} (N-j) \binom{N}{j} \times \hat{p}^j (1-\hat{p})^{N-j-1} (C_{vj} - C_{v(j+1)}), \quad (26)$$

and

$$\frac{dq_{N+1}(\hat{p})}{d\hat{p}} = -\sum_{j=0}^{N} (N+1) \binom{N}{j} \times \hat{p}^j (1-\hat{p})^{N-j} (C_{vj} - C_{v(j+1)}). \quad (27)$$

Substitute the above results into (21), we get

$$\begin{aligned}
&(p_N - p_{N+1}) \frac{dq_v^*(\hat{p})}{\hat{p}} = \\
&\hat{p} \sum_{j=0}^{N} \binom{N}{j} \hat{p}^j (1-\hat{p})^{N-j} (C_{vj} - C_{v(j+1)}) \\
&- \frac{\lambda}{N+1+b} \hat{p} \sum_{j=0}^{N} (N-j) \binom{N}{j} \hat{p}^j (1-\hat{p})^{N-j-1} \\
&\quad \times (C_{vj} - C_{v(j+1)}) \\
&- \frac{1-\lambda}{N+b} \hat{p} \sum_{j=0}^{N} (N+1) \binom{N}{j} \hat{p}^j (1-\hat{p})^{N-j} \\
&\quad \times (C_{vj} - C_{v(j+1)}) \\
&= \hat{p} \sum_{j=0}^{N} \binom{N}{j} \hat{p}^j (1-\hat{p})^{N-j-1} (C_{vj} - C_{v(j+1)}) \\
&\times \left(1 - \hat{p} - \frac{\lambda(N-j)}{N+1+b} - \frac{(1-\lambda)(1-\hat{p})(N+1)}{N+b}\right) \\
&= \hat{p} \sum_{j=0}^{N} \binom{N}{j} \hat{p}^j (1-\hat{p})^{N-j-1} (C_{vj} - C_{v(j+1)}) \\
&\times \left(\frac{\lambda((1-\hat{p})(N+1+b) - N + j)}{N+1+b} \right. \\
&\qquad \left. + \frac{(1-\lambda)(1-\hat{p})(b-1)}{N+b} \right).
\end{aligned} \quad (28)$$

Note that, for all $j \geq 0$, we have

$$\begin{aligned}
&\frac{\lambda((1-\hat{p})(N+1+b) - N + j)}{N+1+b} \\
&\geq \frac{\lambda((1-p_N)(N+1+b) - N + j)}{N+1+b} \\
&\geq \frac{\lambda(b - x^* + j)}{N+1+b}.
\end{aligned} \quad (29)$$

Therefore, $\frac{dq_v^*(\hat{p})}{d\hat{p}} \geq 0$ if $b \geq 1$ and the following inequality is satisfied.

$$\sum_{j=0}^{N} \binom{N}{j} \hat{p}^j (1-\hat{p})^{N-j-1} (C_{vj} - C_{v(j+1)})(b - x^* + j) \geq 0. \quad (30)$$

It is easy to see that (30) holds if $b \geq x^* - \gamma_{\epsilon_v}$, with γ_{ϵ_v} being defined in (15).

Furthermore, if we have both $b > 1$ and $b > x^* - J_{\epsilon_v}$ holding with strict inequality, and $C_{vj} > C_{v(j+1)}$ for at least one $j \le N$, then $\frac{dq_v^*(\hat{p})}{d\hat{p}} > 0$ should also hold with strict inequality for $\hat{p} \in (0, p_{\max})$.

C. Proof of Theorem 5

First, because $b > \max\{1, x^* - J_{\epsilon_v}\}$ holds with strict inequality, the theoretical channel contention measure $q_v^*(\hat{p})$ is strictly increasing in $\hat{p}$ for $\hat{p} \in (0, p_{\max})$. Given user number K, $q_v(\hat{p}, K)$ is non-increasing in $\hat{p}$. Therefore, if $K \ge J_{\epsilon_v}$, then $\hat{p} = p^* = \frac{x^*}{K+b}$ is the only solution to $q_v(\hat{p}, K) = q_v^*(\hat{p})$. When $K < J_{\epsilon_v}$ on the other hand, we have $q_v(\hat{p}, K) > q_v^*(\hat{p})$ for all $\hat{p} \in [0, p_{\max})$. This implies that $\boldsymbol{p}^* = \min\{p_{\max}, \frac{x^*}{K+b}\}\mathbf{1}$ should be the only equilibrium of the system.

Second, we show that there exists a constant $\epsilon > 0$, such that $\frac{dq_v^*(\hat{p})}{d\hat{p}} \ge \epsilon > 0$ for all $\hat{p} < p_{\max}$. Note that $\hat{p} < p_{\max}$ implies $\hat{K} > J_{\epsilon_v}$. From (28) and (29), we get

$$\begin{aligned}\frac{dq_v^*(\hat{p})}{\hat{p}} \ge & \frac{\hat{p}}{p_N - p_{N+1}} \binom{N}{J_{\epsilon_v}} \hat{p}^{J_{\epsilon_v}} (1-\hat{p})^{N-J_{\epsilon_v}-1} \\ & \times (C_{vJ_{\epsilon_v}} - C_{v(J_{\epsilon_v}+1)}) \\ & \times \left(\frac{\lambda(b - x^* + J_{\epsilon_v})}{N+1+b} + \frac{(1-\lambda)(1-\hat{p})(b-1)}{N+b} \right).\end{aligned} \tag{31}$$

Because the right hand side of (31) has a positive limit when $\hat{p} \to 0$, we can find two small positive constants $\epsilon_0, \epsilon_1 > 0$, such that $\frac{dq_v^*(\hat{p})}{\hat{p}} \ge \epsilon_0$ for all $\hat{p} \le \epsilon_1$. On the other hand, when $\epsilon_1 \le \hat{p} < p_{\max}$, because $b > \max\{1, x^* - \gamma_{\epsilon_v}\}$ holds with strict inequality, we can find a small positive constant $\epsilon_2 > 0$, such that the right hand side of (31) is no less than ϵ_2. Therefore, by choosing $\epsilon = \min\{\epsilon_0, \epsilon_2\}$, we have

$$\frac{dq_v^*(\hat{p})}{d\hat{p}} \ge \epsilon > 0, \qquad \text{for all } \hat{p} < p_{\max}. \tag{32}$$

Third, let $q_v^{*-1}(.)$ be the inverse function of $q_v^*(p)$. For any given transmission probability vector $\boldsymbol{p}$, transmission probability target $\hat{p}$ is obtained by

$$\hat{p} = q_v^{*-1}(q_v) = q_v^{*-1}(q_v(\boldsymbol{p}, K)). \tag{33}$$

Because $\frac{dq_v^*(\hat{p})}{d\hat{p}} \ge \epsilon > 0$, we can find a constant $K_{l1} > 0$ such that

$$|\hat{p}_1 - \hat{p}_2| \le K_{l1} |q_{v1} - q_{v2}|, \tag{34}$$

for all $\hat{p}_1 = q_v^{*-1}(q_{v1})$ and $\hat{p}_2 = q_v^{*-1}(q_{v2})$. In the meantime, since $q_v = q_v(\boldsymbol{p}, K)$ is Lipschitz continuous in $\boldsymbol{p}$ for any given K, there must exist a constant $K_{l2} > 0$ to satisfy

$$|q_{v1} - q_{v2}| \le K_{l2} \|\boldsymbol{p}_1 - \boldsymbol{p}_2\|, \tag{35}$$

for all $q_{v1} = q_v(\boldsymbol{p}_1, K)$ and $q_{v2} = q_v(\boldsymbol{p}_2, K)$. Consequently, by combining (34) and (35), we have

$$|\hat{p}_1 - \hat{p}_2| \le K_{l1} K_{l2} \|\boldsymbol{p}_1 - \boldsymbol{p}_2\|, \tag{36}$$

for all $\hat{p}_1 = q_v^{*-1}(q_v(\boldsymbol{p}_1, K))$ and $\hat{p}_2 = q_v^{*-1}(q_v(\boldsymbol{p}_2, K))$. This implies that the probability target function given in (33) satisfies the Lipschitz condition.

Finally, when the system is noisy, the receiver can choose to measure q_v over an extended number of time slots, namely increasing the value of Q introduced in Step 2 of the proposed MAC algorithm. If users maintain their transmission probabilities during the Q times slots, it is often the case that the potential measurement bias in the system can be reduced arbitrarily close to zero with a large enough Q. Therefore, the Mean and Bias condition is also satisfied.

Consequently, convergence of the distributed probability adaptation is supported by Theorems 1 and 2.

D. Proof of Theorem 6

According to the two-step approach, q_v is interpreted by $q_v = (1-p_k)q_k + p_k d^*(\breve{p})$. When $\breve{p} \ge p_k$, we should either have $q_k = q^*(\breve{p})$ when $\breve{p} < p_{\max}$, or $q_k \ge q^*(\breve{p})$ when $\breve{p} = p_{\max}$. Therefore

$$\begin{aligned} q_v &= (1-p_k)q_k + p_k d^*(\breve{p}) \\ &\ge (1-p_k)q^*(\breve{p}) + p_k d^*(\breve{p}) \\ &= q^*(\breve{p}) - p_k(q^*(\breve{p}) - d^*(\breve{p})) \\ &\ge q^*(\breve{p}) - \breve{p}(q^*(\breve{p}) - d^*(\breve{p})) = q_v^*(\breve{p}), \end{aligned} \tag{37}$$

where the last inequality is due to the fact that $q^*(\breve{p}) - d^*(\breve{p}) \ge 0$ should always hold.

Because $b \ge \max\{1, x^* - \gamma_{\epsilon_v}\}$, according to Theorem 4, $q_v^*(\hat{p})$ is non-decreasing in $\hat{p}$. Therefore, if $q_v > q_v^*(p_{\max})$, we have $\hat{p} = p_{\max} \ge p_k$. Otherwise, we have

$$q_v^*(\hat{p}) = q_v \ge q_v^*(\breve{p}) \ge q_v^*(p_k). \tag{38}$$

This also implies that we $\hat{p} \ge p_k$.

Similarly, when $\breve{p} \le p_k$, it can be shown that the two-step approach will yield $\hat{p} \le p_k$.

References

[1] D. Bertsekas and R. Gallager, "Data Networks," 2nd Ed., Prentice Hall, NJ, 1992.

[2] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," IEEE J. Sel. Areas Commun., Vol. 18, pp. 535-547, Mar. 2000.

[3] V. Borkar, and S. Meyn, "The O.D.E Method for Convergence of Stochastic Approximation and Reinforcement Learning," New York: Springer-Verlag, 1997.

[4] S. Ghez, S. Verdú, and S. Schwartz, "Stability Properties of Slotted ALOHA with Multipacket Reception Capability," IEEE Trans. on Auto. Contr., Vol. 33, pp. 640–649, Jul. 1988.

[5] B. Hajek, "Stochastic Approximation Methods for Decentralized Control of Multiaccess Communications," IEEE Trans. on Inform. Theory, Vol. IT-31, pp. 176-184, Mar. 1985.

[6] H. Kushner and G. Yin, "Stochastic Approximation Algorithms and Applications," New York: Springer-Verlag, 1997.

[7] C. Lau and C. Leung, "Capture Models for Model Packet Radio Networks," IEEE Trans. on Commun., Vol. 40, pp. 917-925, May 1992.

[8] J. Luo, "A Generalized Channel Coding Theory for Distributed Communication," IEEE Trans. on Commun., Vol. 63, pp. 1043-1056, Apr. 2015.

[9] J. Luo and A. Ephremides, "A New Approach to Random Access: Reliable Communication and Reliable Collision Detection," IEEE Trans. on Inform. Theory, Vol. 58, pp. 989-1002, Feb. 2012.

[10] X. Qin and R. Berry, "Opportunistic Splitting Algorithms for Wireless Networks," IEEE INFOCOM, Hong Kong, China, Mar. 2004.

[11] Y. Tang, F. Heydaryan, and J. Luo, "On Utility Optimization in Distributed Multiple Access over a Multi-packet Reception Channel," IEEE ISIT, Barcelona, Spain, Jul. 2016.

[12] Z. Wang and J. Luo, "Error Performance of Channel Coding in Random Access Communication," IEEE Trans. on Inform. Theory, Vol. 58, pp. 3961-3974, Jun. 2012.