A Stochastic Primal-Dual Algorithm for Joint Flow Control and MAC Design in Multi-hop Wireless Networks*

Junshan Zhang and Dong Zheng
Department of Electrical Engineering
Fulton School of Engineering
Arizona State University, Tempe, AZ 85287-7206

***Abstract*— We study stochastic rate control for joint flow control and MAC design in multi-hop wireless networks with random access. Most existing studies along this avenue are based on deterministic convex optimization and the corresponding distributed algorithms developed therein involve deterministic feedback control. In a multi-hop wireless network, however, the feedback signal is obtained using error-prone measurement mechanisms and therefore noisy in nature. A fundamental open question is that under what conditions these algorithms would converge to the optimal solutions in the presence of noisy feedback signals, and this is the main subject of this paper. Specifically, we first formulate rate control in multi-hop random access networks as a network utility maximization problem where the link constraints are given in terms of the persistence probabilities. Using the Lagrangian dual decomposition method, we devise a distributed primal-dual algorithm for joint flow control and MAC design. Then, we focus on the convergence properties of this algorithm under noisy feedback information. We show that the proposed primal-dual algorithm converges (almost surely) to the optimal solutions only if the estimators of gradients are asymptotically unbiased. We also characterize the corresponding rate of convergence, and our findings reveal that in general the limit process of the interpolated process, corresponding to the normalized iterate sequence generated from the primal-dual algorithm, is a reflected linear diffusion process, not necessarily the Gaussian diffusion process.**

I. INTRODUCTION

The utility maximization approach for QoS provisioning has recently garnered much attention in the network community since the seminar work [1]. Roughly speaking, in the network utility maximization (NUM) framework, network control is modeled as convex programming problems for maximizing users' utility functions, subject to resource constraints such as capacity constraints or power/energy constraints. The utilities reflect the social welfare of users, and are typically modeled as strictly concave functions of transmission rates. This modeling, together with nice structures of the constraint set such as separability and convexity, leads to distributed solutions.

In wireless networks, the feasible rate region depends heavily on the MAC parameters such as the persistence probability or the contention window size. Due to the broadcast nature of wireless transmissions, this region usually turns out to be non-separable (and even non-convex). There is a general consensus that joint optimization across multiple protocol layers can yield significant performance improvement for wireless network design. For instance, some initial steps for joint optimization of flow control and MAC layer using NUM can be found in [2], [3].

* This research is supported in part by Office of Naval Research through the grant N00014-05-1-0636 and National Science Foundation through the grant ANI-0208135.

Needless to say, in the NUM framework, feedback control signals play a critical role in the distributed implementation towards QoS provisioning. In a wireless network, the feedback control is usually measurement-based and therefore noisy in nature. For example, in the standard congestion control scheme, the price information from the routing nodes is needed to adjust the source rate; and a popular method for obtaining the price information is the packet marking technique. Simply put, packets are marked with certain probabilities which reflect the congestion level at routing nodes. The overall marking probability depends on these marking events along the path on which the packets traverse, and is estimated by observing the relative frequency of marked packets during a pre-specified time window at the source node. Thus, the price information obtained through the estimated marking probability is inevitably noisy. In general, one cannot hope to get perfect feedback information using measurement-based mechanisms in a practical scenario. It is therefore of great interest to examine the system dynamics and stochastic stability in the presence of the noisy feedback information.

In this paper, we explore stochastic rate control for joint flow control and MAC design in multi-hop random access networks. Specifically, we model the end-to-end flow control as a *(stochastic) network utility maximization* problem subject to rate constraints imposed by the MAC/PHY layer. We first present the deterministic NUM problem with the average throughput constraints, and show that this non-convex programming can be transformed into a convex optimization problem via "change of variables" (cf. [2], [3]). Different from the penalty function approach taken in [2], [3], we devise distributed primal-dual algorithms to obtain exact solutions to the original problem via using the Lagrange dual method.

Our main focus is then on the stochastic stability and the rate of convergence of the distributed algorithms with noisy feedback. Using a synergy of tools in stochastic approximation, convex programming, and stochastic analysis, we establish the convergence of these stochastic perturbed algorithms under mild technical conditions. Our finding reveals that the proposed primal-dual algorithm converges (almost surely) to the optimal solutions only if the estimator of gradients is asymptotically unbiased. Then, we examine the corresponding rate of convergence. It turns out that in general the interpolated process, corresponding to the normalized iterate sequence generated from the distributed algorithms, converges weakly to a stationary reflected linear diffusion process, in contrast to the standard Gaussian diffusion process; and the covariance matrix of the limit process gives a measure for the rate of convergence of the normalized iterate sequence.

There has recently been a surge of interest in using the utility maximization approach for cross-layer design in wireless networks. Most relevant to our study here is perhaps [2], [3], [4], [5], [6], [7], [8], [9], [10]. Notably, joint congestion control and resource allocation has been studied in [4], and recent work [7], [8] has investigated joint congestion control and link scheduling. We note that [10] has also studied the effect of the measurement error on the performance of dual-based algorithms, where the convergence is defined in terms of *attraction region*, using a deterministic model for the estimation error.

We have a few more words on stochastic stability. When dealing with the session level randomness where the number of flows changes, stochastic stability often refers to that the number of users and the queuing length across all the links remain finite [11], [7]. In contrast, in the presence of the stochastic perturbations of network parameters, stochastic stability is often used to examine if the proposed algorithms converge to the optimal solutions in some sense. The focus of this paper is to characterize the conditions under which the convergence of the stochastic rate control algorithms can be established. We show that the algorithm converges with probability one (w.p.1) to the optimal solutions provided that the gradient estimator is asymptotically unbiased. In contrast, if the estimator is biased, then we cannot hope that the iterates converge to the optimal solutions almost surely. Instead, we expect that the iterates would converge weakly to some neighborhood of the optimal solutions. It is worth noting that in the seminal works [1], [12], stochastic stability is examined using linear stochastic perturbation around the equivalent point, which implicitly assumes the convergence of the distributed algorithms under noisy feedback. Our study in this paper brings back the packet-level dynamics, and serves as an "intermediate" step towards understanding the stochastic dynamics.

The rest of the paper is organized as follows. In Section II and III, we present the problem formulation, and devise the corresponding deterministic distributed solutions via using Lagrange dual decomposition method. Then in Section IV, we examine the convergence performance of the above distributed algorithms under stochastic perturbations, and study its rate of convergence. Section V gives the numerical examples. Finally, Section VI concludes the paper.

II. Problem Formulation

Consider a wireless network modeled as an undirected graph $G = (N, E)$, where N is the set of the nodes and E is the set of the undirected edges. An edge exists between two neighboring nodes within one-hop distance. Let $\mathcal{S}$ denote the set of flows in the network. For flow s in $\mathcal{S}$, let $U_s(x_s)$ denote its utility function where x_s is the flow rate. Suppose that flow s traverses multiple hops towards the destination, and let $\mathcal{L}(s)$ denote the set of the links that this flow passes through and $\mathcal{S}(l)$ denote the set of flows using link l. Let $N_{to}^I(i)$ denote the set of nodes whose transmissions interfere with node i's reception. For convenience, we use $L_{in}(i)$ to denote the set of the nodes from which node i receives traffic, $L_{out}(i)$ to denote the set of nodes to which node i is sending packets and $L_{from}^I(i)$ to denote the set of nodes whose reception is interfered by node i's transmission.

We assume that the persistence transmission mechanism is used at the MAC layer, i.e., node i of link (i,j) contends the channel with a persistence probability $p_{(i,j)}$. Define $P_i = \sum_{j\in L_{out}(i)} p_{(i,j)}$. It is known that in the Aloha scheme, the average throughput of link (i,j) with persistence probability $p_{(i,j)}$ can be shown as $p_{(i,j)}(1-P_j)\prod_{k\in N_{to}^I(j)}(1-P_k)$ [13]. In light of this, we impose the constraint that the total flow rates using link (i,j) should be no more than $c_{(i,j)}p_{(i,j)}(1-P_j)\prod_{k\in N_{to}^I(j)}(1-P_k)$, where $c_{(i,j)}$ is the average link transmission rate. It follows that the joint optimization of the end-to-end flow control at the transport layer and the link scheduling at the MAC layer can be put together as follows:

$$
\begin{aligned}
\Xi : \max_{\{x_s\}} \quad & \sum_{s\in\mathcal{S}} U_s(x_s) \\
\text{subject to} \quad & \sum_{s\in\mathcal{S}((i,j))} x_s \le c_{(i,j)}(p_{(i,j)} \prod_{k\in N_{to}^I(j)} (1-P_k)),\ \forall\ (i,j) \\
& \sum_{j\in L_{out}(i)} p_{(i,j)} = P_i,\ \forall\ i \\
& 0 \le x_s \le M_s,\ \forall\ s \\
& 0 \le P_i \le 1,\ \forall\ i,
\end{aligned}
\tag{1}
$$

where M_s is the maximum for flow data rate of s, and the utility function $U_s(\cdot)$ takes the following general form [14]:

$$
U_\kappa(r_i) = \begin{cases} w_i \log r_i, & \text{if } \kappa = 1 \\ w_i(1-\kappa)^{-1} r_i^{1-\kappa}, & otherwise. \end{cases} \tag{2}
$$

For simplicity, in the following, we assume that $\{c_{(i,j)}\}$ is normalized to one.

III. Deterministic Rate Control

A. Equivalent Convex Optimization using Change of Variables

Clearly, Problem Ξ is non-convex because of the product term of $p_{i,j}$ in the capacity constraints. By using a change of variables $\tilde{x}_s = \log(x_s)$ (cf. [2], [3]), however, it can be transformed into the following convex programming problem under the condition $\kappa \ge 1$ (cf. (2)):

$$
\begin{aligned}
\mathbf{P} : \max_{\{\tilde{x}_s\}} \quad & \sum_{s\in\mathcal{S}} U_s'(\tilde{x}_s) \\
\text{subject to} \quad & \log(\textstyle\sum_{s\in\mathcal{S}((i,j))} \exp(\tilde{x}_s)) - \log(p_{(i,j)}) \\
& - \textstyle\sum_{k\in N_{to}^I(j)} \log(1-P_k) \le 0,\ \forall\ (i,j) \\
& \textstyle\sum_{j\in L_{out}(i)} p_{(i,j)} = P_i,\ \forall\ i \\
& -\infty \le \tilde{x}_s \le \tilde{M}_s,\ \forall\ s \\
& 0 \le P_i \le 1,\ \forall\ i,
\end{aligned}
\tag{3}
$$

where $U_s'(\tilde{x}_s) = U_s(\exp(\tilde{x}_s))$. Next we verify that Problem $\mathbf{P}$ is indeed a convex optimization problem when $\kappa \ge 1$. It is easy to show that $U_s'(\tilde{x}_s)$ is strictly concave in $\tilde{x}_s$ for utility functions in (2) of $\kappa \ge 1$ [3]. Furthermore, since both terms $\log(\sum_{s\in\mathcal{S}((i,j))}\exp(\tilde{x}_s))$ and $-\log(p_{(i,j)}) - \sum_{k\in N_{to}^I(j)}\log(1-P_k)$ are convex functions of $\tilde{x}_s$ and $p_{(i,j)}$, it follows that Problem $\mathbf{P}$ is strictly convex with a unique optimal point $\mathbf{x}^*$. In the following discussions, we assume that $\kappa \geq 1$.

B. Lagrange Dual Approach

In what follows, we use the Lagrange dual decomposition method to solve Problem $\mathbf{P}$. The Lagrangian function with the Lagrange multipliers $\{\lambda_{(i,j)}\}$ is given as follows:

$$\begin{aligned} & L(\tilde{\mathbf{x}}, \mathbf{p}, \lambda) \\ = & \left\{ \sum_s U_s'(\tilde{x}_s) - \sum_{(i,j)} \lambda_{(i,j)} \log \left(\sum_{s \in \mathcal{S}((i,j))} \exp(\tilde{x}_s) \right) \right\} \\ & + \sum_{(i,j)} \lambda_{(i,j)} \log \left(p_{(i,j)} \prod_{k \in N_{to}^I(j)} (1 - P_k) \right) \end{aligned} \quad (4)$$

Then, the Lagrange dual function is

$$Q(\lambda) = \max_{\substack{\sum_{j \in L_{out}(i)} p_{(i,j)} = P_i \\ \mathbf{0} \leq \mathbf{P} \leq \mathbf{1} \\ -\infty \leq \tilde{\mathbf{x}} \leq \tilde{\mathbf{M}}}} L(\tilde{\mathbf{x}}, \mathbf{p}, \lambda), \quad (5)$$

where $\tilde{\mathbf{M}}$ is a vector of $\tilde{M}_s, \forall\ s \in \mathcal{S}$. Thus, the dual problem is given by

$$\mathbf{D} : \min_{\lambda \geq \mathbf{0}} Q(\lambda) \quad (6)$$

To solve the dual problem, we rewrite $Q(\lambda)$ as

$$\begin{aligned} & Q(\lambda) \\ = & \max_{\tilde{\mathbf{m}} \leq \tilde{\mathbf{x}} \leq \tilde{\mathbf{M}}} \underbrace{\left\{ \sum_s U_s'(\tilde{x}_s) - \sum_{(i,j)} \lambda_{(i,j)} \log \left(\sum_{s \in \mathcal{S}((i,j))} \exp(\tilde{x}_s) \right) \right\}}_{\triangleq O_{\mathbf{x}}(\lambda)} \\ & + \max_{\substack{\sum_{j \in L_{out}(i)} p_{(i,j)} = P_i \\ 0 \leq P_i \leq 1,\ \forall\ i}} \underbrace{\sum_{(i,j)} \lambda_{(i,j)} \log (p_{(i,j)} \prod_{k \in N_{to}^I(j)} (1 - P_k))}_{\triangleq O_{\mathbf{p}}(\lambda)} \end{aligned}$$

which essentially "decomposes" the original utility optimization into two subproblems, i.e., maximizing $O_{\mathbf{x}}(\lambda)$ by flow control and maximizing $O_{\mathbf{p}}(\lambda)$ via MAC layer scheduling, which are coupled by the shadow price λ. It is shown in [3] that the MAC layer scheduling problem can be solved by

$$p_{(i,j)} = \frac{\lambda_{(i,j)}}{\sum_{k \in L_{out}(i)} \lambda_{(i,k)} + \sum_{(m,n): n \in L_{from}^I(i), m \in L_{in}(n)} \lambda_{(m,n)}}. \quad (7)$$

The flow control subproblem can be readily solved by the following gradient method

$$\tilde{x}_s(n+1) = [\tilde{x}_s(n) + \epsilon_n L_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n))]_{-\infty}^{\tilde{M}_s}, \quad (8)$$

where

$$\begin{aligned} & L_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n) \\ \triangleq & \dot{U}_s'(\tilde{x}_s(n)) - \exp(\tilde{x}_s(n)) \sum_{(i,j) \in \mathcal{L}(s)} \frac{\lambda_{(i,j)(n)}}{\sum_{s \in \mathcal{S}((i,j))} \exp(\tilde{x}_s(n))}, \end{aligned}$$

ϵ_n is the step size and $[x]_b^a$ stands for $\max(b, \min(a, x))$. Since $x_s = \exp(\tilde{x}_s)$, it follows that

$$\frac{\partial L}{\partial x_s} = \frac{1}{x_s} \frac{\partial L}{\partial \tilde{x}_s},$$

and therefore the flow control subproblem can also be solved in terms of x_s:

$$\begin{aligned} x_s(n+1) \quad = \quad & \Bigg[x_s(n) + \epsilon_n \Bigg(\dot{U}_s'(\log(x_s(n))) \\ & - x_s \sum_{(i,j) \in \mathcal{L}(s)} \frac{\lambda_{(i,j)(n)}}{\sum_{s \in \mathcal{S}((i,j))} x_s(n)} \Bigg) \Bigg]_0^{M_s} \end{aligned} \quad (9)$$

Similarly, the dual problem can be solved by using the subgradient method, and we have that

$$\lambda_{(i,j)}(n+1) = \left[\lambda_{(i,j)}(n) - \epsilon_n L_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)) \right]_0^\infty, \quad (10)$$

where

$$\begin{aligned} & L_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)) \\ \triangleq & \log(p_{(i,j)}(n)) + \textstyle\sum_{k \in N_{to}^I(j)} \log(1 - P_k(n)) \\ & - \log \left(\textstyle\sum_{s \in \mathcal{S}((i,j))} \exp(\tilde{x}_s(n)) \right). \end{aligned}$$

We note that from the perspective of distributed implementation of (9), the parameters $\frac{\lambda_{(i,j)}}{\sum_{s \in \mathcal{S}((i,j))} x_s}$ need to be generated at each link in $\mathcal{L}(s)$, and fed back to the source node along the routing path. It can be seen that the calculation of the ratio $\frac{\lambda_{(i,j)}}{\sum_{s \in \mathcal{S}((i,j))} x_s}$ at node i only requires the local information of the shadow price $\lambda_{(i,j)}$ and the total incoming traffic $\sum_{s \in \mathcal{S}((i,j))} x_s$. In the dual algorithm (10), in order to adapt the shadow price $\lambda_{(i,j)}$, it suffices to have the local information of the total incoming traffic and the two-hop information of $\{P_k, k \in N_{to}^I(j)\}$. Similarly, by examining (7), we can see that the calculation of $p_{(i,j)}$ only requires two-hop information of the shadow price.

C. Deterministic Primal-dual Algorithms

Summarizing, Problem $\mathbf{P}$ can be solved via the following deterministic distributed algorithm:

- *The source rates are updated by*

$$\tilde{x}_s(n+1) = [\tilde{x}_s(n) + \epsilon_n L_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n))]_{-\infty}^{\tilde{M}_s}. \quad (11)$$

- *The shadow prices are updated by*

$$\lambda_{(i,j)}(n+1) = \left[\lambda_{(i,j)}(n) - \epsilon_n L_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)) \right]_0^\infty. \quad (12)$$

- *The persistence probabilities are updated by*

$$\begin{aligned} & p_{(i,j)}(n+1) \\ = & \frac{\lambda_{(i,j)}(n)}{\sum_{k \in L_{out}(i)} \lambda_{(i,k)}(n) + \sum_{\substack{(l,m): m \in L_{from}^I(i), \\ l \in L_{in}(m)(n)}} \lambda_{(l,m)}} \end{aligned} \quad (13)$$

We note that in the above algorithms, we have used the same step size ϵ_n for both the primal algorithm and the dual algorithm.

IV. STOCHASTIC RATE CONTROL AND STABILITY UNDER NOISY FEEDBACK

In this section, we examine the convergence performance of the above distributed algorithms under stochastic perturbations, due to noisy feedback information.

A. Stochastic Primal-Dual Algorithm For Rate Control

In the presence of noisy feedback information, clearly, the gradients are estimators. More specifically, the stochastic version of the primal-dual algorithm is given as follows:

- *SA algorithm for source rate updating*:
$$\tilde{x}_s(n+1) = [\tilde{x}_s(n) + \epsilon_n \left(\hat{L}_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n))\right)]_{-\infty}^{\tilde{M}_s}, \tag{14}$$
where $\hat{L}_{\tilde{x}_s}(\cdot,\cdot,\cdot)$ is an estimator of $L_{\tilde{x}_s}(\cdot,\cdot,\cdot)$.
- *SA algorithm for shadow price updating*:
$$\lambda_{(i,j)}(n+1) = \left[\lambda_{(i,j)}(n) - \epsilon_n(\hat{L}_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)))\right]_0^{\infty}, \tag{15}$$
where $\hat{L}_{\lambda_{(i,j)}}(\cdot,\cdot,\cdot)$ is an estimator of $L_{\lambda_{(i,j)}}(\cdot,\cdot,\cdot)$.
- The persistence probability updating rule remains the same as (13).

B. Probability One Convergence Of Stochastic Rate Control Algorithm

Next, we examine in detail the models for stochastic perturbations. Let $\{\mathcal{F}_n\}$ be a sequence of $\sigma-$algebras generated by $\{(\tilde{\mathbf{x}}(i), \lambda(i)), \forall\ i \le n\}$. For convenience, we use $E_n[\cdot] = E[\cdot|\mathcal{F}_n]$ to denote the conditional expectation. We have the following models on $\hat{L}_{\tilde{x}_s}(\cdot,\cdot,\cdot)$ and $\hat{L}_{\lambda_{(i,j)}}(\cdot,\cdot,\cdot)$.

1) *Stochastic gradient* $\hat{L}_{\tilde{x}_s}(\cdot,\cdot,\cdot)$: Observe that
$$\hat{L}_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)) = L_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)) + \alpha_s(n) + \zeta_s(n),$$
where
$$\alpha_s(n) \triangleq E_n\left[\hat{L}_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n))\right] - L_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)), \tag{16}$$
i.e., $\alpha_s(n)$ is the biased random error of $L_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n))$, and
$$\zeta_s(n) \triangleq \hat{L}_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)) - E_n\left[\hat{L}_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n))\right]. \tag{17}$$
Note that $\zeta_s(n)$ is a martingale difference noise since $E_n[\zeta_s(n)] = 0$.

2) *Stochastic gradient* $\hat{L}_{\lambda_{(i,j)}}(\cdot,\cdot,\cdot)$: Observe that
$$\hat{L}_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)) = L_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n))) + \beta_{(i,j)}(n) + \xi_{(i,j)}(n),$$
where
$$\beta_{(i,j)}(n) \triangleq E_n\left[\hat{L}_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n))\right] - L_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)), \tag{18}$$
i.e., $\beta_{(i,j)}(n)$ is the biased random error of $L_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n))$, and
$$\xi_{(i,j)}(n) \triangleq \hat{L}_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)) - E_n\left[\hat{L}_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n))\right]. \tag{19}$$
Again, $\xi_{(i,j)}(n)$ is a martingale difference noise.

To examine the convergence of the stochastic primal-dual algorithm, we impose the following standard assumptions:

A1. *We assume that the estimators of the gradients are based on the measurements in each iteration only.*

A2. *Condition on the step size:* $\epsilon_n > 0$, $\epsilon_n \to 0$, $\sum_n \epsilon_n \to \infty$ *and* $\sum_n \epsilon_n^2 < \infty$.

A3. *Condition on the biased error:* $\sum_n \epsilon_n |\alpha_s(n)| < \infty, \forall\ s$ *and* $\sum_n \epsilon_n |\beta_{(i,j)}(n)| < \infty, \forall\ (i,j)$.

A4. *Condition on the martingale difference noise:* $\sup_n E_n[\zeta_s(n)^2] < \infty, \forall\ s$, *and* $\sup_n E_n[\xi_{(i,j)}(n)^2] < \infty, \forall\ (i,j)$.

We have the following proposition:

Proposition 4.1: Under Conditions **A1** – **A4**, the iterates $\{(\mathbf{x}(n), \lambda(n), \mathbf{p}(n)), n = 1, 2, \ldots\}$, generated by stochastic approximation algorithms (14), (15) and (13), converge with probability one to the optimal solutions of Problem Ξ.

Sketch of the proof: The proof consists of two steps. First, using the stochastic Lyapunov Stability Theorem, we establish that the iterates generated by (14), (15) and (13) return to a neighborhood of the optimal points infinitely often, i.e., the neighborhood of the optimal points is recurrent. Then, we show that the recurrent iterates eventually reside in an arbitrary small neighborhood of the optimal points, and this is proved by using "local analysis". The complete proof can be found in [15].

Remarks: We note that Condition **A1** implies that the noise is independent across the iterations, Condition **A2** is a standard technical condition in stochastic approximation for proving probability one convergence, and that Condition **A3** essentially requires that the biased term is asymptomatically diminishing. When the step size ϵ_n does not go to zero (which occurs often in on-line applications) or the biased term $\beta_{(i,j)}(n)$ does not diminish (which may be the case in some practical systems), we cannot hope to get probability one convergence to the equilibrium point. Nevertheless, we expect that the iterates would converge in distribution to some random variable "close" to the equilibrium point. It should be cautioned that even the expectation of the limiting distribution would not be the equilibrium point if $\beta_{(i,j)}(n)$ does not go to zero.

We use the following example to illustrate how to characterize sufficient conditions for the almost sure convergence of stochastic gradient algorithms.

Example 1: We assume that the exponential marking technique is used to feedback the price information to the source nodes. More specifically, in the exponential marking algorithm, every link (i,j) marks a packet independently with probability $1 - \exp\left(-\frac{\lambda_{(i,j)}}{\sum_{s\in\mathcal{S}(i,j)}\exp(\tilde{x}_s)}\right)$. Therefore, the overall non-marking probability is given as follows
$$q_s = \exp\left(-\sum_{(i,j)\in\mathcal{L}(s)} \frac{\lambda_{(i,j)}}{\sum_{s\in\mathcal{S}(i,j)}\exp(\tilde{x}_s)}\right)$$
To estimate of the overall price, source s sends N_n packets during round n and counts the non-marked packets. For

example, if K packets have been counted, then the estimation of the overall price can be $\log(\hat{q_s})$ where $\hat{q_s} = K/N_n$. Therefore,

$$\hat{L}_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)) = \dot{U}_s'(\tilde{x}_s(n)) + \exp(\tilde{x}_s(n))\log(\hat{q_s}). \tag{20}$$

By the definition of (16), we have that

$$\alpha_s(n) = \exp(\tilde{x}_s(n))\left(E_n[\log(\hat{q_s})] - \log(q_s)\right)$$

Note that K is a Binomial random variable with distribution $B(N_n, q)$. When N_n is sufficiently large, it follows that $\hat{q_s} \sim \mathcal{N}(q_s, q_s(1-q_s)/N_n)$ and $\hat{q_s} \in [q_s - c/\sqrt{N_n}, q_s + c/\sqrt{N_n}]$ with high probability, where c is a positive constant. That is, the estimation bias of the price information can be upper-bounded as

$$|\alpha_s(n)| \leq \tilde{M}_s |E[\log(\hat{q_s})] - \log(q_s)| \leq \frac{c'}{\sqrt{N_n}} \tag{21}$$

for large N_n, where c' is some positive constant.

To ensure the convergence of the primal-dual algorithm, from Condition **A3**, it suffices to have that

$$\sum_n \frac{\epsilon_n}{\sqrt{N_n}} < \infty \tag{22}$$

For example, when $\epsilon_n = 1/n$, $N_n \sim O(\log^4(n))$ would satisfy (22).

Next, we verify that the variance condition, i.e., **A4** is satisfied for $\zeta_s(n)$. By (17) and (20), we find that

$$\begin{aligned} E_n[\zeta_s(n)^2] &= E_n\left[\exp(\tilde{x}_s(n))\left(\log(\hat{q}) - E_n[\log(\hat{q})]\right)\right]^2 \\ &\leq \tilde{M}_s^2 E_n[\log(\hat{q})]^2 \\ &\leq \tilde{M}_s^2 E_n[\log(q+c)]^2 \ \forall\ N_n >> 0. \end{aligned}$$

Similar studies can be carried out for $\beta_{(i,j)}(n)$ and $\xi_{(i,j)}(n)$ in (18) and (19).

Worth noting is that (22) indicates that it suffices to have the measurement window size grows at the rate of $\log^4(n)$.

C. Rate of Convergence

Building on the convergence (w.p.1) of the stochastic primal-dual algorithm, we next examine the rate of convergence, which reveals the advantage of the proposed stochastic algorithm. Roughly speaking, the rate of convergence is concerned with the asymptotic behavior of normalized errors about the optimal points. Recall that a general constrained form of the primal-dual algorithm is given as follows:

$$\begin{aligned} &\begin{bmatrix} \tilde{x}_s(n+1) \\ \lambda_{(i,j)}(n+1) \end{bmatrix} \\ = &\begin{bmatrix} \tilde{x}_s(n) \\ \lambda_{(i,j)}(n) \end{bmatrix} + \epsilon_n \begin{bmatrix} L_{\tilde{x}_s}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)) \\ -L_{\lambda_{(i,j)}}(\tilde{\mathbf{x}}(n), \mathbf{p}(n), \lambda(n)) \end{bmatrix} \\ &+\epsilon_n \begin{bmatrix} \alpha_s(n) + \zeta_s(n) \\ \beta_{(i,j)}(n) + \xi_{(i,j)}(n) \end{bmatrix} + \epsilon_n \begin{bmatrix} Z_n^{\tilde{x}_s} \\ Z_n^{\lambda_{(i,j)}} \end{bmatrix}, \end{aligned} \tag{23}$$

where $\epsilon_n Z_n^{\tilde{x}_s}$ and $\epsilon_n Z_n^{\lambda_{(i,j)}}$ are the reflection terms which "force" $\tilde{x}_s$ and $\lambda_{(i,j)}$ to reside inside the sets $(-\infty, \tilde{M}_s]$ and $[0, \infty)$. As is standard in the study on the rate of convergence, we assume that the iterates generated by the stochastic primal-dual algorithm have entered in a small neighborhood of an optimal solution $(\tilde{\mathbf{x}}^*, \lambda^*)$.

To characterize the asymptotic properties, we define $U_{\tilde{\mathbf{x}}}(n) \triangleq (\tilde{\mathbf{x}}(n) - \tilde{\mathbf{x}}^*)/\sqrt{\epsilon_n}$ and $U_\lambda(n) \triangleq (\lambda(n) - \lambda^*)/\sqrt{\epsilon_n}$, and we construct $U^n(t)$ to be the piecewise constant interpolation of $U(n) = \{U_{\tilde{\mathbf{x}}}(n), U_\lambda(n)\}$, i.e., $U^n(t) = U_{n+i}$, for $t \in [t_{n+i} - t_n, t_{n+i+1} - t_n)$, where $t_n \triangleq \sum_{i=0}^{n-1} \epsilon_n$.

A5. *Let* $\theta(n) \triangleq (\tilde{\mathbf{x}}(n), \lambda(n))$ *and* $\phi_n \triangleq (\zeta(n), \xi(n))$. *Suppose for any given small* $\rho > 0$, *there exists a positive definite symmetric matrix* $\Sigma = \sigma\sigma'$ *such that*

$$E_n[\phi_n \phi_n^T - \Sigma] I\{|\theta(n) - \theta^*| \leq \rho\} \to 0$$

as $n \to \infty$.

Define

$$A \triangleq \begin{bmatrix} L_{\tilde{x}\tilde{x}}(\tilde{\mathbf{x}}^*, \mathbf{p}^*, \lambda^*) & L_{\lambda\tilde{x}}(\tilde{\mathbf{x}}^*, \mathbf{p}^*, \lambda^*) \\ -L_{\lambda\tilde{x}}(\tilde{\mathbf{x}}^*, \mathbf{p}^*, \lambda^*) & 0 \end{bmatrix}.$$

A6. *Let* $\epsilon_n = 1/n$; *and assume* $A + I/2$ *is a Hurwitz matrix*. Note that it can be easily shown that the real parts of the eigenvalues of A are all non-positive (cf. page 449 in [16]).

We have the following proposition.

Proposition 4.2: a) Under Conditions **A1** and **A3** – **A6**, $U^n(\cdot)$ converges weakly to the solution (denoted as U) to the Skorohod problem

$$\begin{pmatrix} dU_{\tilde{x}} \\ dU_\lambda \end{pmatrix} = \left(A + \frac{I}{2}\right)\begin{pmatrix} U_{\tilde{x}} \\ U_\lambda \end{pmatrix} dt + \sigma dw(t) + \begin{pmatrix} dz_{\tilde{x}} \\ dz_\lambda \end{pmatrix},$$

b) If $(\tilde{\mathbf{x}}^*, \lambda^*)$ is an interior point in the constraint set, the limiting process U is a stationary Gaussian diffusion process, and $U(n)$ converges in distribution to a normally distributed random variable with mean zero and covariance Σ.

c) If $(\tilde{\mathbf{x}}^*, \lambda^*)$ is on the boundary of the constraint set, then the limiting process U is a stationary reflected linear diffusion process.

Proposition 4.2 can be proved by appealing to a combination of tools used in the proofs of Theorem 5.1 in [17] and Theorem 2.1 in Chapter 6 in [18]. Roughly, we can expand, via a truncated Taylor series, the interpolated process $U^n(t)$ around the chosen saddle point $(\tilde{\mathbf{x}}^*, \lambda^*)$. Then, the main new step is to show that the tightness of $U^n(t)$. To this end, we can follow part 3 in the proof of Theorem 2.1 in Chapter 6 in [18] to establish that the biased term in the interpolated process diminishes asymptotically. Then, the rest follows from the proof of Theorem 5.1 in [17].

Remarks: Proposition 4.2 reveals that for the constrained case, the limiting process of the interpolated process for the normalized iterates depends on the specific structure of the Skorohod problem in the study, which is defined as [18]

$$dX = HXdt + \sigma dw + dz, \tag{24}$$

where H is Hurwitz (i.e., the real parts of the eigenvalues of H are all negative), $w(t)$ is a standard Wiener process and $z(\cdot)$ is the reflection term. In general, the limit process is a stationary reflected linear diffusion process, not necessarily the standard Gaussian diffusion process. The limit process would be Gaussian only if there is no reflection term, which may occur, for instance, when all the link constraints in Problem $\mathbf{P}$ are active at the optimal point.

From (24), the rate of convergence depends heavily on the smallest eigenvalue of $\left(A + \frac{I}{2}\right)$. The more negative the smallest eigenvalue is, the fast the rate of convergence would be. Intuitively speaking, the reflection terms would help increase the speed of convergence, which unfortunately cannot be characterized exactly. As noted in [17], one cannot readily compute the stationary covariance matrix for reflected diffusion process, and we have to resort to simulations to get some understanding of the effect of the constraints on the asymptotic variances. Furthermore, the covariance matrix of the limit process gives a measure of the spread at the equilibrium point, and is typically "smaller" than the unconstrained case [17].

V. Numerical Results

In this section, we investigate the convergence performance of the proposed primal-dual algorithm. Specifically, we consider a simple network scenario depicted in Fig. 1. Note that there are four nodes (A, B, C and D), four links and four flows, where flow 1 is from node A to node B, flow 2 from B and to C, flow 3 from A to C via B, and flow 4 from C to D via B. The utility function in Problem Ξ is taken to be the logarithm utility function where $\kappa = 1$, and $M_s = 1, \forall\ s$.

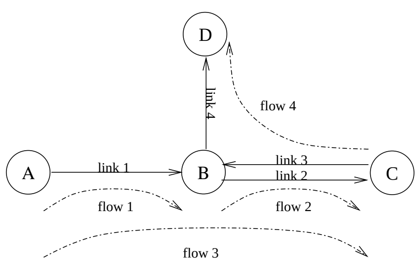

Fig. 1. A simple network topology.

The result obtained by the proposed primal-dual algorithm, together with the theoretical optimal solution, are presented in Table I. It can be seen that the result obtained from the primal-dual algorithm is very close to the optimal solutions.

TABLE I
COMPARISON BETWEEN THE RESULT OF THE PROPOSED PRIMAL-DUAL ALGORITHM AND THE THEORETICAL OPTIMAL SOLUTION.

Link probabilities	$p_{(A,B)}$	$p_{(B,C)}$	$p_{(C,B)}$	$p_{(B,D)}$
Primal-dual algorithm	0.6655	0.3840	0.2009	0.0391
Optimal solution	0.6457	0.3750	0.2152	0.0443
Flow rate	x_1	x_2	x_3	x_4
Primal-dual algorithm	0.2065	0.2065	0.0974	0.0385
Optimal solution	0.1962	0.1962	0.0981	0.0443

VI. Conclusion

We have studied joint flow control and MAC design in multi-hop wireless networks with random access. Particularly, we formulate rate control therein as a network utility maximization problem where the link constraints are given in terms of the persistence probabilities. We then use the Lagrangian dual decomposition method to devise a distributed primal-dual algorithm for joint flow control and MAC design. Our focus is then on the convergence properties of this proposed algorithm under noisy feedback information. We have shown that the proposed primal-dual algorithm converges almost surely to the optimal solutions provided that the estimator is asymptotically unbiased. Our findings on the rate of convergence reveal that in general the limit process of the interpolated process, corresponding to the normalized iterate sequence generated from the primal-dual algorithm, is a reflected linear diffusion process, not necessarily the Gaussian diffusion process.

We believe that the studies we initiated here on convergence properties of stochastic algorithms in the network utility maximization framework, scratch only the tip of the iceberg. There are still many questions remaining open for different utility maximization techniques, and we are currently investigating these issues along this avenue.

References

[1] F. P. Kelly, A. Maulloo, and D. K. H. Tan, "Rate control for communication networks: Shadow price, proportional fairness and stability," *Journal of Operational Research Society*, pp. 237–252, 1998.
[2] X. Wang and K. Kar, "Cross-layer rate control for end-to-end proportional fairness in wireless networks with random access," in *Proc. MOBIHOC'05*, 2005.
[3] J. Lee, M. Chiang, and A. Calderbank, "Jointly optimal congestion and medium access control in ad hoc wireless networks," in *Proc. VTC'05*, 2005.
[4] M. Chiang, "To layer or not to layer: balancing transport and physical layers in wireless multihop networks," in *Proc. INFOCOM'04*, 2004.
[5] L. Xiao, M. Johansson, and S. Boyd, "Simultaneous routing and resource allocation via dual decomposition," *IEEE Transactions on Communications*, 2004.
[6] M. Johansson and L. Xiao, "Cross-layer optimization of wireless networks using nonlinear column generation."
[7] X. Lin and N. Shroff, "The impact of imperfect scheduling on cross-layer rate control in multihop wireless networks," in *Proc. IEEE INFOCOM'05*, 2005.
[8] L. Chen, S. Low, and J. Doyle, "Joint congestion control and media access control design for ad hoc wireless networks," in *Proceedings of INFOCOM*, 2005.
[9] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle, "Jointly optimal congestion control, routing, and scheduling for wireless ad hoc networks," in *Proc. IEEE INFOCOM'06*, 2006.
[10] M. Mehyar, D. Spanos, and S. Low, "Optimization flow control with estimation error," in *Proc. INFOCOM'04*, 2004.
[11] T. Bonald and L. Massoulie, "Impact of fairness on internet performance," in *Proc. Sigmetrics*, 2001.
[12] F. P. Kelly, "Fairness and stability of end-to-end congestion control," *European Journal of Control*, pp. 159–176, 2003.
[13] D. P. Bertsekas and R. Gallager, *Data Networks*. New Jersey: Prentice Hall, 1992.
[14] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE Trans. on Networking*, vol. 8, pp. 556–566, nov. 2000.
[15] J. Zhang and D. Zheng, "A Stochastic Primal-Dual Algorithm for Joint Flow Control and MAC Design in Multi-hop Wireless Networks," *Technical Report, Department of Electrical Engineering, ASU*, 2006.
[16] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1995.
[17] R. Buche and H. J. Kushner, "Rate of convergence for constrained stochastic approximation algorithms," *SIAM Journal on Control and Optimization*, vol. 40, pp. 1011–1041, 2001.
[18] H. Kushner and G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. Springer, 2003.