Decentralized Throughput Maximizing Policies for Deadline-Constrained Wireless Networks

Rahul Singh[1] and P. R. Kumar[1]

Abstract— We consider multi-hop wireless networks serving multiple flows in which only packets that meet hard end-to-end deadline constraints are useful, i.e., if a packet is not delivered to its destination node by its deadline, it is dropped from the network. We design decentralized scheduling policies for such multi-hop networks that attain the maximum throughput of useful packets. The resulting policy is decentralized in the sense that in order to make a transmission decision, a node only needs to know the "time-till-deadline" of the packets that are currently present at that node, and not the state of the entire network. The key to obtaining an easy-to-implement and highly decentralized policy is to replace the hard constraint on the number of simultaneous packet transmissions that can take place on the outgoing links of a node, by a time-average constraint on the number of transmissions. The policy thus obtained is guaranteed to provide maximum throughput. Analysis can be extended to the case of time-varying channel conditions in a straightforward manner.

Simulations showing significant improvement over existing policies for deadline based scheduling, such as Earliest Deadline First, and supporting the theory, are presented.

I. INTRODUCTION

The focus of this paper is on multi-hop networks with per-packet end-to-end deadlines. Scheduling policies such as the back-pressure algorithm [1], which are designed to achieve maximum throughput for traditional multi-hop wireless networks, provide guarantees only on average end-to-end delays, not per-packet delays. Due to the unreliable nature of the wireless medium, the per-packet delay along sample paths can become arbitrarily large. In-fact the back pressure algorithm or the MaxWeight scheduler has been shown to optimize the average end-to-end network delay only in the heavy-traffic regime [2]. However since the average delay grows roughly as $\frac{1}{1-\text{traffic intensity}}$, the delay becomes unbounded as traffic intensity approaches 1. This is not desirable for applications such as cyber-physical systems where control-loops are closed over networks and system stability is sensitive to delays.

In this paper, we consider multi-hop wireless networks serving various flows, in which if a packet is not delivered to its destination by its deadline, it is dropped from the network and not counted in the throughput. Since the wireless channel is unreliable, the outcome of packet

transmissions is modeled as random processes. We suppose that each node can transmit multiple packets on its out-links. To incorporate an average power constraint, which presents itself as an average rate constraint on the out-going communication channels linked to that node, we impose an upper bound on the average number of transmissions that can be made by a node in the network, a quantity which is allowed to depend on the individual node. Furthermore we assume that a node can transmit and receive packets on multiple channels, a property which can be achieved via various techniques such as TDMA, CDMA, OFDMA [3]–[5] which enable multiple access. We remark that such an assumption allows the network to make full use of the available resource-sharing disciplines.

The throughput of a flow is then the average number of packets delivered to its destination node per unit time, and our goal is to design decentralized scheduling policies that maximize the total throughput of the network.

We use the scalarization principle [6], and consider a weighted throughput of all flows, whose maximum value will be one point on the Pareto frontier of the rate region. We pose the problem of obtaining a (weighted) throughput-maximizing scheduling policy as a Markov Decision Process (MDP).

Our approach to solving this problem is via the Lagrangian dual of this MDP. The Lagrange multipliers associated with the rate-constraints are interpreted as prices paid by a packet to a node for transmitting its packet. This renders our approach very different from that leading to the backpressure policy, where the Lagrange multipliers are associated with flows and not packets, and are therefore completely different, corresponding to queue lengths.

The resulting overall MDP decomposes conveniently into a "unit-packet unit flow" MDP (Section VI). This makes possible a decentralized packet-by-packet solution, where not just flows or nodes but individual packets can be individually optimized with respect to their treatment by nodes. A node only needs to know the remaining lifetime till deadline of each packet that is present at the node and makes a decison on transmitting the packet based on that. Moreover the packet level MDP does not suffer from a severe curse of dimensionality since its state space is only the cardinality of the number of nodes multiplied by the relative deadline bound on the packet, and is thus relatively easily solved. Thus, introduction of Lagrange multipliers, specifically prices for

attempting packet transmissions, gives rise to a tractable and easy to implement decentralized scheduling policy. These Lagrange multipliers are shown to be computable in a decentralized online fashion.

The key to our results is the long-term average constraint on the number of concurrent packet transmissions by a node. This can regarded as a long-term average relaxation of a more stringent hard constraint on the number of concurrent transmissions that a node may make at any time. This relaxation is in the same spirit as Whittle's relaxation for multi-armed bandits [7], [8], where the constraint on the number of arms that can be simultaneously pulled is relaxed to a constraint on the average number of arms that can be pulled. Our relaxation therefore also has an asymptotic optimality property in the same manner that Whittle's relaxation has. In Whittle's case it is asymptotically optimal as the number of arms goes to infinity. Our relaxation is asymptotically optimal as the link capacities are scaled proportionally across the entire network, and results from the fact that on an average, not much is gained in terms of the total reward collection, by relaxing the hard constraint to an average constraint [9]. The analysis can be carried out using the theory of large deviations for Markov processes, and it can be shown that the sub-optimality gap between the relaxed policy, and the optimal policy, is of the order $\frac{1}{\sqrt{\text{Network Capacity}}}$. Extensive simulations have been performed which show that the policy thus obtained drastically out-performs existing policies for deadline constrained networks, such as Earliest Deadline First.

The system model considered excludes a key aspect of wireless networks, namely wireless interference. It could be incorporated by not allowing interfering links to schedule packet transmissions simultaneously. This would require the co ordinated effort of all the nodes in order to decide the optimal set of non-interfering links to be activated, and is the subject of future works.

II. SYSTEM MODEL

We consider wireless networks in which the data-packets have a hard deadline constraint on the time at which they are delivered to their destination nodes in order to be regarded as useful and counted in the throughput. The network comprises of several nodes $1, 2, \ldots, V$ that are connected via wireless links. The wireless network is described by a directed graph in which there is a directed edge $i \to j$ if node i can transmit packets to node j.

We assume that time is discrete, and evolves over discrete time slots numbered $1, 2, \ldots$. One time-slot is the time taken to attempt a packet transmission over any link in the wireless network. The network is shared by F flows $f_1, f_2, \ldots, f_F$. Each flow f_i has a pre-determined route comprised of a sequence of links that connect the
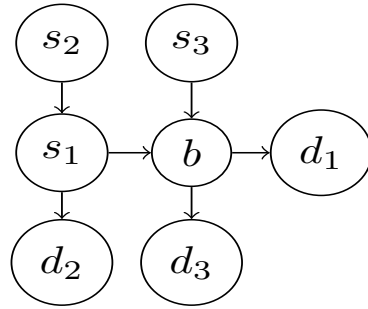

Fig. 1. Multi-hop network serving three flows f_1 on the route $s_1 \to b \to d_1$, f_2 on the route $s_2 \to s_1 \to d_2$, and f_3 on the route $s_3 \to b \to d_3$.

source node of the flow f_i to its destination node; see Figure 1.

The wireless channel between any two nodes is allowed to be random. If a packet of any flow f is attempted on the wireless link l, then the transmission is successful with probability p_l. The outcomes of packet transmission attempts are independent across links and time-slots.

Each node v has an average rate-constraint M_v, which is the maximum number of packet transmission attempts per unit time that it can make. We will use the notation $l \in v$ to mean that link l is in the set of out-links from node v, and thus v can use the link l for transmitting packets. Let $u_f^l(t)$ be the random variable which denotes the number of packets of flow f that are attempted for transmission on link l at time t. The rate constraints are given by,

$$\limsup_{T \to \infty} \mathbb{E} \left(\frac{\sum_{t=1}^{T} \sum_f \sum_{l \in v} u_f^l(t)}{T} \right) \le M_v, \quad (1)$$
$$\forall v \in \{1, 2, \ldots, V\}.$$

Note that we allow a node to transmit and receive packets simultaneously over several outgoing links, which is possible via various techniques such as TDMA, OFDMA, CDMA etc. [3]–[5], and hence M_v can be larger than 1. The rate-constraint is due to the fact that wireless nodes have power constraints, which in turn induces constraints on communication.

Each packet that is generated by the network has a "relative-deadline", and if the packet is not delivered to its destination within this deadline, it is dropped from the network and will not be transmitted in future time-slots. More precisely, if a packet has a relative-deadline δ, and is generated at the beginning of time-slot τ, then either it is delivered to its destination node by time-slot $\tau + \delta$, or it is discarded from the network.

We assume that the relative deadlines of the packets are i.i.d. and bounded by a fixed Δ, where the distribution of the relative deadlines depends on the flow that the packet belongs to. The relative deadline becomes known to the network as soon as the packet is generated.

Our analysis can be extended in a straight-forward manner to consider the case when the relative-deadline of a packet is an arbitrary stochastic process that is adapted, though we eschew that here for brevity. When so chosen as an adapted stochastic process, some very useful models can be captured. For example, suppose we have a video context, and we have a frame buffer at the receiver, then the relative deadline is equal to the "remaining play time" left in the frame buffer since we don't want the buffer emptied. In that case Relative Deadline $= -$ (Elapsed time since the Last time that Destination Buffer was empty, i.e., the current age of the "busy epoch") $+$ (Number of packets that arrived at the Destination since then)$\times$ (Time to play one packet). Note that in this case the deadline process depends on the policy being used.

We assume that the inter-arrival times of packets at different source nodes for each flow are governed by renewal processes having finite means. The throughput attained by a flow f under a policy is defined to be,

$$q_f := \liminf_{T \to \infty} \mathbb{E} \left(\frac{\sum_{t=1}^{T} d_f(t)}{T} \right), \qquad (2)$$

where the random variable $d_f(t)$ is $= 1$ if a packet of flow f is delivered to its destination at time t, and is $= 0$ otherwise, and expectation is under the policy that is applied. A throughput vector $\mathbf{q}$ that can be achieved via some scheduling policy will be called an "achievable throughput vector", the set of all achievable throughput vectors constitutes the "rate-region", and a scheduling policy that achieves the rate-region is said to be throughput-optimal.

Of course, all the above definitions depend on the random process which decides the relative-deadlines of the packets, and so therefore does the rate-region. Thus we might call such networks as "deadline-constrained networks".

Vectors will be in bold font, and by $\mathbb{R}_+^N$ we refer to N dimensional vectors which are non-negative component wise.

III. Previous Works

References [10]–[12] consider a network model in which multiple flows share a single-hop network, and all the packets across every flow have the same relative deadline. Restricting consideration to a periodic arrival process with relative deadline equal to the period at any given time at most, only one packet belonging to each flow is present in the network. References [13], [14] consider a similar one-hop network model and characterize the throughput maximizing policy. Clearly the single-hop model is restrictive.

Reference [15] considers the challenging problem of scheduling deadline-constrained packets over a multi-hop network, but the proposed policies are not shown to have any provable guarantees on the resulting throughput. To the best of the author's knowledge, [16] is the only work which provides a provable sub-optimal policy for deadline-constrained networks, though it only concerns wired networks. Moreover the policies proposed in [16] guarantee only a fraction

$$\frac{1}{\text{length of the longest route in network}}$$

of the maximum possible throughput, i.e., only a small fraction of the capacity region.

Though the MaxWeight policy [2] has been popular among researchers in designing multi-hop scheduling-routing policies such as the Back-Pressure [1], or decentralized CSMA wireless scheduling policies [17], [18], it should be noted that an application of the MaxWeight principle to design multi-hop scheduling policies for wireless networks results in the number of possible switch vectors available to the scheduler [2] growing roughly exponentially in the quantities: total number of flows, sum of the relative deadlines of flows, and the sum of the route lengths of flows. While some of the previous works on deadline-constrained scheduling [10]–[12] have been successful in designing deadline-constrained throughput optimal "debt-based" policies for single-hop networks, any generalization to the case of multi-hop wireless network seems difficult.

Joint routing scheduling policies designed for multi hop networks, such as the Backpressure [1] are throughput optimal, but perform poorly with respect to the end-to-end delay [19]–[22]. References [23] and [24] perform a fluctuation analysis of the "packet starvation" that occurs in deadline constrained networks, while [25]–[27] develop a framework for service regularity for networks shared by multiple flows.

Thus, presently the knowledge of provably optimal scheduling policies for deadline-constrained multi-hop wireless networks is severely limited. Any optimal scheduling policy needs to take into account how much time each packet has spent in the network, as well as the channel reliabilities of the links that the packets have to traverse in order to reach the destination node.

IV. Characterizing the Rate Region

The rate-region of the network (defined in Section II) will be denoted by $\mathbf{\Lambda}$. We note that in order to characterize it, it is sufficient to characterize the set of Pareto-optimal vectors $\mathbf{q} \in \mathbf{\Lambda}$, defined as

$$\left\{ \mathbf{q} : \exists \boldsymbol{\alpha} \in \mathbb{R}_+^N \text{ such that } \mathbf{q} \in \arg\max_{\mathbf{y} \in \Lambda} \sum_f \alpha_f y_f \right\},$$

since $\mathbf{\Lambda}$ is simply its closed convex hull. Thus the problem of obtaining the set $\mathbf{\Lambda}$ is equivalent to that of finding scheduling policies which maximize a non-negatively weighted sum of throughputs.

A. Constrained MDP Formulation

The problem of maximizing a non-negatively weighted sum of throughputs subject to rate-constraints can be posed as a Constrained Markov Decision Process (CMDP) [28]. The system state can be described as follows. At time t, each packet in the network belonging to flow f is described by the two tuple (l, s), where l is the link at which the packet is present, and s is the time-to-go till its deadline. The state of the network is then simply given by the states of each packet present in the network.

Since the number of packets in the network at any time is bounded (because the relative deadlines are bounded only a bounded number of packets can hope to be served in this time period and others can be safely dropped), the system state $x(t)$ takes on finitely many values. A scheduling policy π has to choose, for each time t, at each node, which packets to transmit from the set of packets available to it. The probability distribution of the system state at time $t+1$, $x(t+1)$, depends only on the system state at time t, $x(t)$ and the action chosen at time t by the policy π. The problem of maximizing the throughput subject to node-capacity constraints (1) is posed as a Constrained Markov Decision Process, where a reward of α_f is received when a packet of flow f is delivered to its destination.

Thus a policy maximizing the network throughput solves the following optimization problem:

$$\max_{\pi} \liminf_{T \to \infty} \frac{1}{T} \mathbb{E} \left(\sum_f \sum_{t=1}^{T} \alpha_i d_f(t) \right), \text{ such that}$$

(Primal MDP)

$$\limsup_{T \to \infty} \frac{1}{T} \mathbb{E} \left(\sum_{t=1}^{T} \sum_f \sum_{l \in v} u_f^l(t) \right) \le M_v, \forall v \in \{1, 2, \ldots, V\}.$$

(3)

We note that the above CMDP, parameterized by the vector $\boldsymbol{\alpha} := (\alpha_1, \ldots, \alpha_F)$ is solved by a Stationary Randomized Policy ([28]). Since the state-space of the network, and the number of link-capacity constraints (1) are finite, it follows that there is a finite set $\{\pi_1, \pi_2, \ldots, \pi_M\}$ of Stationary Randomized Policies such that for each value of $\boldsymbol{\alpha}$, there is a policy that belongs to this set and solves the CMDP (3) ([28]). Let $\gamma_1, \gamma_2, \ldots, \gamma_M$ be the vectors of throughputs associated with the policies $\pi_1, \pi_2, \ldots, \pi_M$. We then have the following characterization of $\boldsymbol{\Lambda}$.

Lemma 1:

$$\boldsymbol{\Lambda} = \left\{ \mathbf{q} : \mathbf{q} = \sum_{i=1}^{M} \gamma_i c_i, c_i \ge 0, \sum_i c_i \le 1 \right\}.$$

Note that the number of Stationary Markov policies is exponentially large in the parameter:
maximum possible number of packets in the network $\times$ maximum path length of the flows $\times$ maximum possible relative deadline.

Hence using Lemma 1 to compute $\boldsymbol{\Lambda}$ is out of the question. Thus we seek to design low-complexity decentralized scheduling policies that achieve the region $\boldsymbol{\Lambda}$.

V. THE DUAL MDP

We write the Lagrangian for the Primal MDP (3) as,

$$\mathcal{L}(\pi, \hat{\boldsymbol{\lambda}}) =$$

$$\liminf_{T \to \infty} \frac{1}{T} \mathbb{E} \left(\sum_f \sum_t \alpha_f d_f(t) \right)$$

$$+ \sum_v \hat{\lambda}_v \left(\liminf_{T \to \infty} \frac{1}{T} \mathbb{E} \left(\sum_{t=1}^{T} \sum_f \sum_{l \in v} [-u_f^l(t)] \right) \right) \quad (4)$$

$$+ \sum_v \hat{\lambda}_v M_v.$$

The corresponding dual is,

$$D(\hat{\boldsymbol{\lambda}}) = \max_{\pi} \mathcal{L}(\pi, \hat{\boldsymbol{\lambda}}). \quad (5)$$

Next we develop a useful upper-bound on $D(\hat{\boldsymbol{\lambda}})$, the proof of which follows from the sub-addivity of $\liminf$ operation and super-additivity of $\limsup$ operation.

Lemma 2:

$$D(\hat{\boldsymbol{\lambda}}) \le \sum_f V_f(\hat{\boldsymbol{\lambda}}) + \sum_v \hat{\lambda}_v M_v, \quad (6)$$

where

$$V_f(\hat{\boldsymbol{\lambda}}) :=$$

$$\max_{\pi_f} \limsup_{T \to \infty} \frac{1}{T} \mathbb{E} \left(\sum_t \left(\alpha_f d_f(t) + \sum_v \sum_{l \in v} \hat{\lambda}_v [-u_f^l(t)] \right) \right),$$

and π_f is a policy that schedules packets of flow f.

VI. THE SINGLE FLOW, SINGLE PACKET PROBLEM

Consider the expression in the r.h.s. of (6). We note that the introduction of the vector $\hat{\boldsymbol{\lambda}}$ while considering the dual problem (5) decouples the problem (3) into F number of *single-flow problems*, wherein if a packet of flow f is present at node v at time t, and it is attempted on a link l that belongs to the set of out going links of node v, then flow f is charged a price of amount λ_l. Thus $\hat{\lambda}_v$ can be interpreted as the price that the node v charges for each use of any of its out-links.

We note that the above Lagrange multipliers are very different from the Lagrange multipliers employed in deriving the Backpressure policy. There the Lagrange multipliers correspond to queue lengths, whereas here, as we will show below, they are the price paid by a packet to a node for the privilege of being transmitted. It is this difference that will allow us to decompose the overall problem on packet-by-packet basis. It is this that will allow us to obtain a decentralized policy that is not only decentralized across flows, but also over nodes, and in fact over packets, with not even any coupling between packets at the same node.

We will first solve these F single-flow problems. Then we will show that the policy that implements the solution to the single-flow problem for each flow solves the relaxed problem. Denote by π_f a policy for flow f. The policy π_f knows the value of the node prices $\hat{\lambda}_v$ of all the nodes v that lie on the route of flow f, and the state of all the packets in the network that belong to flow f, and has to make a decision whether or not to schedule a packet transmission for flow f at a link l, but does not need to keep track of the state of other flows.

The single-flow problem, parametrized by the vector of prices $\hat{\boldsymbol{\lambda}}$ is to find the policy π_f that maximizes,

$$V_f(\hat{\boldsymbol{\lambda}}) = \max_{\pi_f} \limsup_{T\to\infty} \frac{1}{T}\mathbb{E} \sum_{j=1}^{N_f(t)} \left(\sum_t \Big(\alpha_f d_f^j(t) + \right.$$
$$\left. \sum_v \sum_{l\in v} \hat{\lambda}_v \left[-u_f^{l,j}(t) \right] \Big) \right), \quad (7)$$

where $u_f^{l,j}(t)$ assumes the value 1 if the j-th packet belonging to the flow f is served at link l in time-slot t, and 0 otherwise, $d_f^j(t)$ assumes the value 1 if the j-th packet of flow f is delivered to its destination at time t, and $N_f(t)$ is the number of packets for flow f that arrive into the network by time t. Also note that $u_f^{l,j}(t)$ can assume the value 1 only if the j-th packet of flow f is present at link l at time t.

Note that while introducing the single-flow problem, we have reduced it to a *single-flow single-packet problem*, i.e. the total reward earned under the application of a policy π_f is the sum of the rewards it earns from each packet. *Thus the policy that solves the single-slow problem makes a decision whether or not to schedule a packet for flow f depending only on the state of that single packet!* Thus the introduction of node-prices $\hat{\boldsymbol{\lambda}}$ not only decouples the original problem into F separate problems, but further separates the problem for a single flow into that involving only a particular packet of that flow.

Next we describe and solve the *single-packet-single-flow problem*. We note that the state of a packet is much smaller than the state of the entire network. The state of a packet only consists of the node it is at, and its time till deadline. Therefore the Markov Decision Process for the single packet problem is much more tractable than the MDP for the overall system. To make the discussion simpler, since we are dealing with a single-flow, we will assume that the nodes have been re-labelled so that the source node is labelled as 1, while the destination node is $L+1$. With this notation in place, the link between nodes i and $i+1$ is labelled as l_i (so that l_i is the i-th link on the route). The single flow under consideration has to be served using the route $l_1, l_2, \ldots, l_L$. A single packet for this flow is generated at time $t=1$ at the source node 1, and needs to be delivered to the node $L+1$ by the time $D+1$. The wireless link (channel) l has a channel reliability p_l. Thus the packet has a relative

deadline of D time-slots, and has to traverse L hops in order to reach the destination.

If the packet is delivered in a time-slot $t \in \{1, 2, \ldots, D+1\}$, then a reward of α is received and the packet leaves the system. However if the packet is not delivered by the time $D+1$, it is removed from the system without generating a reward of α units. If the packet is present at node v and it is attempted, then it is charged an instantaneous price of $\hat{\lambda}_v$. The total reward accrued by a packet is any rewards it obtains when it reaches its destination on time, minus all the prices it paid at all the links along its path to the destination.

The policy π_f has to decide at each time whether or not to schedule the packet's transmission, so as to maximize the total expected reward earned in the time $\{1, 2, \ldots, D+1\}$.

The single-packet single-flow problem is clearly a finite-state stochastic dynamic programming problem wherein the state of the packet assumes values in

$$\{(l_i, s) : 0 \le i \le L \text{ and } L - i \le s \le D - i\},$$

where l_i is the link at which the packet is present and s is the "time remaining till deadline".

Let $V^{\hat{\boldsymbol{\lambda}}}(l,s)$ be the maximum expected reward that can be earned by the packet starting in state (l, s). The function $V^{\hat{\boldsymbol{\lambda}}}(l,s)$ is not to be confused with the function $V_f(\hat{\boldsymbol{\lambda}})$ defined for the single-flow problem in (7). The super scipt shows the dependence of the value function $V^{\hat{\boldsymbol{\lambda}}}(l,s)$ on the price vector $\hat{\boldsymbol{\lambda}}$. At times, the superscript $\hat{\boldsymbol{\lambda}}$ will be omitted to make the notation simpler.

Clearly $V(l_i, L-i) = 0$ for $i = 1, 2, \ldots, L$, since a packet at link l_i needs at least $L-i$ time-slots to reach the destination node, but since the time-to-go till deadline is $L-i$, it is dropped from the network. Using the principle of optimality, the values $V(l, s)$ are computed as,

$$V(l_i, s) = \max\{V(l_i, s-1),$$
$$-\lambda_{l_i} + p_{l_i}V(l_{i+1}, s-1) + (1 - p_{l_i})V(l_i, s-1)\},$$
$$(8)$$

where the second expression within the braces corresponds to the rewards associated with scheduling the packet transmission in state (l_i, s). The first expression is the reward under the choice of not scheduling, and the optimal action is the one that achieves the maximum on the rhs. The computations in (8) are performed for all states in the order

$(l_L, 1)), (l_L, 2), \ldots, (l_{L-1}, 2), \ldots$, and thus the optimal actions and value function are obtained in LD steps.

Let $\phi_f(l, s) = 0$ if the first term in the rhs of (8) is strictly larger, and $= 1$ if the second term is strictly larger. For values of states (l, s) in which both the terms in the rhs of (8) are equal, set $\phi_f(l, s)$ to be any value in the set $[0, 1]$. Let $\boldsymbol{\phi}_f$ be the vector consisting of the values $\phi_f(l, s)$. Denote by $\pi_f(\hat{\boldsymbol{\lambda}}, \boldsymbol{\phi}_f)$ the policy which transmits the packet of flow f with a probability $\phi_f(l, s)$ when the packet is in state (l, s). By construction, $\pi_f(\hat{\boldsymbol{\lambda}}, \boldsymbol{\phi}_f)$ solves the single-flow-single-packet problem.

Thus the optimal policy $\pi_f(\hat{\boldsymbol{\lambda}}, \phi_f)$ can be parameterized by the vector ϕ_f. Since the total rewards earned in the Single Flow Problem (7) is the sum of rewards earned from each packet, we have,

Lemma 3: Policies $\pi_f(\hat{\boldsymbol{\lambda}}, \phi_{\boldsymbol{f}})$ solve the Single Flow Problem (7).

It should be noted that the optimal policy depends on the price vector $\hat{\boldsymbol{\lambda}}$. The key to the dramatic decomposition of an originally very complex problem of optimizing the behavior of the entire network lies in the fundamental nature of the constraint: It is an average constraint on number of packets. One can regard this as a relaxation of a hard constraint on the number of concurrent packet transmissions allowed to a node, a relaxation that has an asymptotic optimality property. To see this, an analogy with the multi-arm bandit problem where one is allowed up to n simultaneous pulls per play is revealing. For this problem, there is no Gittins Index; however Whittle [8] has shown that if this constraint is relaxed to an average constraint of no more than n pulls, then one can obtain a decomposition. Moreover this constraint is optimal in the limit as the number of arms of different types and arm pulls are increased in proportion to infinity. In our case too, as the link capacities are scaled proportionally across the entire network we get asymptotic optimality.

VII. SOLUTION OF PRIMAL MDP

Given price vector $\hat{\boldsymbol{\lambda}}$ and vector $\phi = \{\phi_{\boldsymbol{f}}, f = 1, 2, \ldots, F\}$, let us denote by $\pi(\hat{\boldsymbol{\lambda}}, \phi)$ the policy that jointly follows the rule $\pi_f(\hat{\boldsymbol{\lambda}}, \phi_{\boldsymbol{f}})$ for each flow f, where $\pi_f(\hat{\boldsymbol{\lambda}}, \phi_{\boldsymbol{f}})$ is as defined in Section VI. First, we show that there exist values of the vectors $\hat{\boldsymbol{\lambda}}, \phi$ such that the policy $\pi_f(\hat{\boldsymbol{\lambda}}, \phi_{\boldsymbol{f}})$ solves the Primal MDP.

Lemma 4: $D(\hat{\boldsymbol{\lambda}}) = \mathcal{L}(\pi(\hat{\boldsymbol{\lambda}}, \phi), \hat{\boldsymbol{\lambda}})$.

Proof: From the definition of the dual function (5) and the function V_f (7), it follows that,

$$\mathcal{L}(\pi, \hat{\boldsymbol{\lambda}}) \leq D(\hat{\boldsymbol{\lambda}}) \leq \sum_f V_f + \sum_v \hat{\lambda}_v M_v + \sum_v \hat{\lambda}_v M_v,$$

for any policy π. However since the policies $\pi(\hat{\boldsymbol{\lambda}}, \phi)$ are Stationary Randomized Markov policies (all the $\liminf$s and $\limsup$s in the definition of its Lagrangian and also the corresponding rewards associated with the single-flow problem change to $\lim$), it follows from Lemma 3 that $\sum_f V_f$ is the value of $\mathcal{L}(\pi, \hat{\boldsymbol{\lambda}})$ evaluated at the policies $\pi(\hat{\boldsymbol{\lambda}}, \phi)$, and thus the inequality above becomes an equality,

$$\mathcal{L}(\pi(\hat{\boldsymbol{\lambda}}, \phi), \hat{\boldsymbol{\lambda}}) = D(\hat{\boldsymbol{\lambda}}) = \sum_f V_f + \sum_v \hat{\lambda}_v M_v. \quad (9)$$

■

Theorem 1: Consider the Primal MDP (3) and its associated dual problem defined in (5). There exists a price-vector $\hat{\boldsymbol{\lambda}}^\star$, and vectors $\phi_{\boldsymbol{f}}^\star, f = 1, 2, \ldots, F$, such that $(\pi(\hat{\boldsymbol{\lambda}}^\star, \phi^\star), \hat{\boldsymbol{\lambda}}^\star)$ is an optimal primal-dual pair, and thus the policy $\pi(\hat{\boldsymbol{\lambda}}^\star, \phi^\star)$ solves the relaxed problem.

Proof: We use the ergodic control approach developed in [28]–[30], particularly useful for constrained MDPs, wherein an average-cost MDP can be viewed as a linear programming problem. More precisely, the infinite-horizon average-reward problem can be posed as that of optimizing a linear cost function over a convex set after one considers occupation measures of the combined state-cum-control Markov process. If $\nu^\pi(\cdot)$ is the occupation measure induced under the policy π, the problem reduces to maximizing

$$\sum_{\text{state,control}} \text{reward(state,control)}\nu^\pi(\text{state,control}).$$

We further note that the constraints in the Primal MDP are linear functions of $\nu(\cdot)$, and hence the set over which optimization is to be carried out is convex.

Thus if we show that there exists a policy such that the constraints in the Primal MDP (3) hold with strict inequality ">" rather than "$\geq$", then the proof of the Lemma would follow from Slater's condition [31]. But then the policy which never schedules any packet regardless of the value of the system state satisfies the constraints with strict > (the parameters M_v are assumed to be > 0, i.e. the total capacity of all the out-links associated with each node v is assumed to be > 0.)

■

We note that the policy $\pi(\hat{\boldsymbol{\lambda}}^\star, \phi^\star)$ is decentralized, and each node only needs the knowledge of the time-till-deadlines of the packets it has.

A. Obtaining $\hat{\boldsymbol{\lambda}}^\star, \phi^\star$

The policy obtained above is decentralized and easy to implement; one has only to obtain the price vector $\hat{\boldsymbol{\lambda}}^\star$ that solves the dual problem (5). Next we provide an iterative algorithm to find $\hat{\boldsymbol{\lambda}}^\star$ that can be implemented in a decentralized manner, i.e., the nodes need only local information of the total link usage on all of its out-links. Thus the nodes need not communicate amongst themselves to obtain global information about the state of the network in order to derive $\hat{\boldsymbol{\lambda}}^\star$, which rules out involvement of any communication over heads.

Since the dual function $D(\hat{\boldsymbol{\lambda}})$ is convex, we can use the sub-gradient method [32] to iteratively find $\hat{\boldsymbol{\lambda}}^\star$. From Lemma 4, we have $D(\hat{\boldsymbol{\lambda}}) = \mathcal{L}(\pi(\hat{\boldsymbol{\lambda}}, \phi), \hat{\boldsymbol{\lambda}}) + \sum_v \hat{\lambda}_v M_v$, where the value of the Lagrangian $\mathcal{L}(\pi(\hat{\boldsymbol{\lambda}}, \phi), \hat{\boldsymbol{\lambda}})$ does not depend on the probability with which the packet is scheduled for values of the states in which the active (scheduling) and passive (not scheduling) actions yield equal rewards. Thus in order to find the value of dual function $D(\hat{\boldsymbol{\lambda}})$, it suffices to fix the randomizing probability to be 0 for any value of the state in which the active and passive actions yield equal rewards. We can choose as sub-gradients the gradient associated with the policy $\pi(\hat{\boldsymbol{\lambda}}, \phi)$, i.e., the quantities $\frac{\partial \mathcal{L}(\pi(\hat{\boldsymbol{\lambda}}, \phi), \hat{\boldsymbol{\lambda}})}{\partial \hat{\lambda}_v}, v = 1, 2, \ldots, V$. Next we provide the expression for the gradients $\frac{\partial \mathcal{L}(\pi(\hat{\boldsymbol{\lambda}}, \phi), \hat{\boldsymbol{\lambda}})}{\partial \hat{\lambda}_v}, v = 1, 2, \ldots, V$, which helps us in

interpreting it as congestion at the nodes, and the node-prices as a means to control the congestion.

From the description of the Single Flow Problem (7), it follows that the total cost a packet pays due to the link-price $\hat{\lambda}_v$ is equal to $\hat{\lambda}_v$ times the total number of times that it uses an out-link at node v. Thus letting $\tilde{\tau}(f, v)$ to be a random variable that has the same distribution as the total link-usage per-unit time-slot by packets of flow f at node v, we have from (6),

$$\frac{\partial D}{\partial \hat{\lambda}_v} = M_v - \mathbb{E}_{\pi(\hat{\boldsymbol{\lambda}}, \boldsymbol{\phi})} \sum_f \tilde{\tau}(f, v). \qquad (10)$$

The quantity $\mathbb{E}_{\pi(\hat{\boldsymbol{\lambda}}, \boldsymbol{\phi})} \sum_f \tilde{\tau}(f, v)$ is the total link-usage at node v, and thus measures the congestion at node v. The price $\hat{\lambda}_v$ is thus a means to prevent congestion at node v.

The iterations are given by

$$\hat{\boldsymbol{\lambda}}^{k+1} = \hat{\boldsymbol{\lambda}}^k - \alpha_k \boldsymbol{g}^k, \qquad (11)$$

where $\boldsymbol{g}^k$ is the sub-gradient evaluated at $\hat{\boldsymbol{\lambda}}^k$ as in (10). Since the value of the i-th component of sub-gradient as provided in equation (10) is a local information, meaning that node i can calculate its total link-usage without resorting to any communication with other nodes, these iterations can be performed locally. The
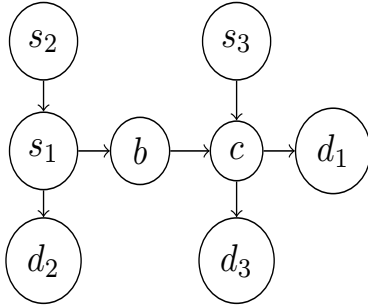

Fig. 2. Multi-hop network serving three flows f_1 on the route $s_1 \to b \to c \to d_1$, f_2 on the route $s_2 \to s_1 \to d_2$, and f_3 on the route $s_3 \to c \to d_3$.

analysis here can be extended to cover the case of time-varying channels so as to incorporate the wireless fading. This is accomplished by appending the state with the state of the wireless channels across the network.

VIII. SIMULATIONS

Apart from the scheduling policy deployed, we note that for a deadline-constrained network, the net throughput depends upon the following factors: relative deadlines of the packets, the random process that decides the number of the packet arrivals in different flows, the channel reliabilities of various links and the node capacities M_v's.

We compare the optimal policy derived in the paper with the policy that implements the EDF scheduling rule at each node in the network. The EDF policy is known to be optimal in the case of single-hop network.

For the network shown in Figure 2, the channel reliabilities of each link are fixed at 1, except for the link $s_1 \to d_2$, which is fixed at 0.5. Packets belonging to flow f_1 have a relative deadline of 3 time slots, while packets belonging to flows f_2, and f_3 have a relative deadline of 2 time slots. Moreover it is assumed that each flow receives 50 packets in each time slot. The node capacities of each node except the node s_1 are assumed to be 100 packets/slot. The throughputs of all the flows are weighted equally, i.e., $\alpha_1 = \alpha_2 = \alpha_3 = 1$. Figure 3 compares the throughput attained by the optimal policy with that attained by the EDF policy as a function of capacity of node s_1.

IX. CONCLUSIONS

We have posed the problem of designing throughput-maximizing scheduling policies for deadline-constrained multi-hop wireless networks as a Constrained Markov Decision Process (CMDP). Though a first look at the complex nature of the problem suggests that the optimal policy should make a scheduling decision based on the entire state of the network, and thus suggests that each node in the network must know the state of the network, we have however solved the dual of the original CMDP and thereby derived an optimal policy that is highly decentralized. In order to implement this optimal policy, each node in the network needs to know only the time-till-deadline of each packet present with it.

Furthermore, the problem of obtaining the optimal Lagrange multipliers or the node prices has also been shown to have a decentralized solution, in which each node monitors its own congestion and updates the prices accordingly. The node-prices have been shown to be a means of congestion control. Simulations show that the optimal policy indeed out performs the Earliest Deadline First policy, comporting with the theory.

REFERENCES

[1] L. Tassiulas and Anthony Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936–1948, Dec 1992.

[2] A. Stolyar, "Max weight scheduling in a generalized switch: State space collapse and workload minimization in heavy traffic," in *Annals of Applied Probability*, vol. 14, no. 1, 2004, pp. 1–53.

[3] S. Zhang and V. K. N. Lau, "Resource allocation for ofdma system with orthogonal relay using rateless code," *IEEE Transactions on Wireless Communications,*, vol. 7, Nov. 2008.

[4] M. Duarte and A. Sabharwal, "Full-duplex wireless communications using off-the-shelf radios: Feasibility and first results," in *Proceedings of ASILOMAR*, 2010, pp. 1558–1562.

[5] J. I. Choi, M. Jain, K. Srinivasan, P. Levis, and S. Katti, "Achieving single channel, full duplex wireless communication," in *ACM MobiCom*, 2010.

[6] C. Hwang and A. Masud, *Multiple Objective Decision Making Methods and Applications*. Springer-Verlag New York, Inc., 1979.

[7] J.C. Gittins, K. Glazebrook, and R. Weber, *Multi-armed Bandit Allocation Indices*. John Wiley & Sons, 2011.

[8] Peter Whittle, "Restless bandits: Activity allocation in a changing world," *Journal of Applied Probability*, vol. 25, pp. 287–298, 1988.
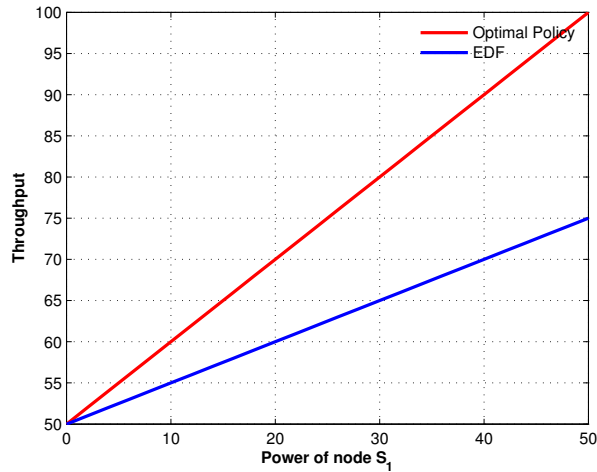
Fig. 3. Plot showing the throughputs attained by the policies for the network of Figure 2 as the node capacity of s_1 is varied. The relative-deadline was fixed at 3 time-slots for flow packets of flow f_1, and at 2 time slots for packets of flows f_2 and f_3. Each flow was assumed to receive 50 packets per time slot. Reliability of the link $s_1 \rightarrow d_2$ was fixed at $.5$, while those of other links were assumed to be 1.

[9] R. R. Weber and G. Weiss, "On an index policy for restless bandits," *Journal of Applied Probability*, vol. 27, no. 3, pp. pp. 637–648, 1990.

[10] I-Hong Hou, V. Borkar, and P.R. Kumar, "A Theory of QoS for Wireless," *IEEE INFOCOM 2009*, pp. 486–494, April 2009.

[11] I-Hong Hou and P.R. Kumar, "Scheduling heterogeneous real-time traffic over fading wireless channels," in *INFOCOM 2010*, March 2010, pp. 1–9.

[12] ——, "Real-time communication over unreliable wireless links: a theory and its applications," *IEEE Wireless Communications,*, vol. 19, no. 1, pp. 48–59, February 2012.

[13] Kyu Seob Kim, Chih-Ping Li and E. Modiano, "Scheduling multi-cast traffic with deadlines in wireless networks," *IEEE INFOCOM 2014*, pp. 2193–2201, April 2014.

[14] K. J. H. Ahmed and S. Bhashyam, "Fair scheduling with deadline guarantees in single-hop networks," in *Sixth International Conference on Communication Systems and Networks (COMSNETS), 2014*, Jan 2014, pp. 1–7.

[15] Ruogu Li and Atilla Eryilmaz, "Scheduling for end-to-end deadline-constrained traffic with reliability requirements in multihop networks," *IEEE/ACM Transactions on Networking*, vol. 20, no. 5, pp. 1649–1662, Oct 2012.

[16] Z. Mao, C.E. Koksal and N.B. Shroff, "Optimal online scheduling with arbitrary hard deadlines in multihop communication networks," *IEEE/ACM Transactions on Networking*, Nov. 2014.

[17] L. Jiang and J. Walrand, "A Distributed CSMA Algorithm for Throughput and Utility Maximization in Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 3, pp. 960–972, June 2010.

[18] Jian Ni, Bo Tan and R. Srikant, "Q-CSMA: Queue-Length Based CSMA/CA Algorithms for Achieving Maximum Throughput and Low Delay in Wireless Networks," *IEEE INFOCOM 2010*, pp. 1–5, March 2010.

[19] J. N. Libin Jiang, Mathieu Leconte, R. Srikant, and J. Walrand, "Fast mixing of parallel glauber dynamics and low-delay csma scheduling." *IEEE Transactions on Information Theory*, vol. 58, no. 10, pp. 6541–6555, Dec 2012.

[20] L. Bui, R. Srikant, and A. Stolyar, "A novel architecture for reduction of delay and queueing structure complexity in the back-pressure algorithm," *Networking, IEEE/ACM Transactions on*, vol. 19, no. 6, pp. 1597–1609, Dec 2011.

[21] G. Gupta and N. Shroff, "Delay analysis for multi-hop wireless networks," in *INFOCOM 2009, IEEE*, April 2009, pp. 2356–2364.

[22] L. Ying, S. Shakkottai, A. Reddy, and S. Liu, "On combining shortest-path and back-pressure routing over multihop wireless networks," *IEEE/ACM Trans. Netw.*, vol. 19, no. 3, pp. 841–854, Jun. 2011.

[23] Rahul Singh, I-Hong Hou and P.R. Kumar, "Fluctuation analysis of debt based policies for wireless networks with hard delay constraints," in *INFOCOM, 2014 Proceedings IEEE*, April 2014, pp. 2400–2408.

[24] ——, "Pathwise performance of debt based policies for wireless networks with hard delay constraints," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, Dec 2013, pp. 7838–7843.

[25] Rahul Singh, Xueying Guo and P.R. Kumar, "Index policies for optimal mean-variance trade-off of inter-delivery times in real-time sensor networks," in *IEEE INFOCOM 2015*, 2015.

[26] Xueying Guo, Rahul Singh, P.R. Kumar and Niu, Zhisheng, "A high reliability asymptotic approach for packet inter-delivery time optimization in cyber-physical systems," in *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '15. ACM, 2015, pp. 197–206. [Online]. Available: http://doi.acm.org/10.1145/2746285.2746305

[27] R. Singh and A. Stolyar, "Maxweight scheduling: Asymptotic behavior of unscaled queue-differentials in heavy traffic," *ACM SIGMETRICS*, 2015.

[28] E. Altman, "Constrained markov decision processes with total cost criteria: Occupation measures and primal lp," *Mathematical Methods of Operations Research*, vol. 43, no. 1, pp. 45–72, 1996.

[29] Alan S. Manne, "Linear programming and sequential decisions," *Management Science*, vol. 6, pp. 259 –267, 1960.

[30] V. S. Borkar, "Control of Markov Chains with Long-Run Average Cost Criterion," *Stochastic Differential Systems, Stochastic Control Theory and Applications*, pp. 57–77, 1988.

[31] Dimitri P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

[32] N. Shor, K. C. Kiwiel, and A. Ruszcaynski, *Minimization Methods for Non-differentiable Functions*. New York, NY, USA: Springer-Verlag New York, Inc., 1985.