On Wireless Network Scheduling with Intersession Network Coding

Chih-Chun Wang
Center for Wireless Systems and Applications
School of Electrical & Computer Engineering
Purdue University, West Lafayette, Indiana, USA
Email: chihw@purdue.edu

Ness B. Shroff
Departments of ECE and CSE
The Ohio State University, Columbus, Ohio, USA
Email: shroff@ece.osu.edu

***Abstract*—Cross-layer optimization including congestion control, routing, and scheduling has shown dramatic throughput improvement over layered designs for wireless networks. In parallel, the paradigm-shifting network coding has empirically demonstrated substantial throughput improvement when coding operations are permitted at intermediate nodes and packets from different sessions are mixed. Designing network codes and the associated flow in network coding presents new challenges for cross-layer optimization for wireless multi-hop networks. This work shows that with a new flow-based characterization of *pairwise intersession network coding*, a joint optimal scheduling and rate-control algorithm can be implemented distributively. Optimal scheduling is computationally expensive to achieve even in a purely routing-based (without network coding) paradigm, let alone with network coding. Thus, in this paper, the impact of imperfect scheduling is studied, which shows that pairwise intersession network coding can improve the throughput of routing-based solutions regardless of whether perfect/imperfect scheduling is used. Both the deterministic and stochastic packet arrivals and departures are considered. This work shows for the first time a striking resemblance between pairwise intersession network coding and routing, and thus advocates extensions of routing-based wisdoms to their network coding counterpart.**

***Index Terms*—Network coding, pairwise intersession network coding, imperfect scheduling, cross-layer optimization, congestion control.**

I. Introduction

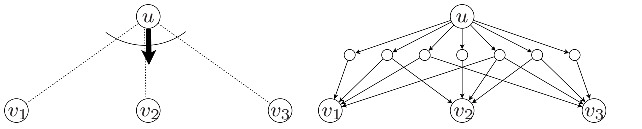
The interference-heavy nature of wireless media presents a great challenge for designing high-throughput wireless multi-hop networks. Recently, many techniques have been developed for enhancing the throughput of wireless multi-hop networks, among which at least two techniques demonstrate promising improvements. The first method is built around the existing "routing" (non-network-coding) concepts and focuses on cross-layer design that considers jointly route-selection, node-scheduling and rate-control by converting the networking problem to a corresponding utility maximization problem. Significant throughput improvements have been demonstrated and a rich literature has been developed (see [14], [15] and the reference therein). The second method is network coding, which allows intermediate nodes to perform both relaying operations as in the classic routing paradigm and new coding operations suggested by the information-theoretic development. Advantages of network coding are shown both theoretically [1], [8], [10], [12] and empirically [2], [3], [9], [21]. Network coding can be further classified into two different sub-categories: intrasession and intersession network coding, the former of which focuses on a single multicast session and coding is performed on packets from the same session. The latter considers multiple coexisting sessions and coding is performed on packets across different sessions.

Intrasession network coding is well-understood as its performance is characterized by the minimum max-flow value between each source-sink pairs of a multicast session. More explicitly, if a unicast rate x can be supported between all source-destination pairs (s, t_i), where t_i is the i-th destination, by multi-path routing and by assuming no other destinations $t_j \neq t_i$ are sharing the network, then a multicast rate x can be supported between $(s, \{t_i\}_i)$ by network coding. This flow-based characterization leads to a natural extension of the cross-layer optimization framework to intrasession network coding, including throughput, utility, and energy optimization as in [16], [19], [26], [27], [28], [29] Its multicast nature makes intrasession network coding an appealing technique for content distribution [6], [25]. Unfortunately, for the most frequent scenario in which only *unicast sessions* are present, the benefits of intrasession network coding vanish.

Intersession network coding provides performance improvement even when only unicast sessions are present. Although its benefit is clearly demonstrated in the butterfly structure both theoretically [12] and empirically [9], the much needed characterization of intersession network coding is less understood and many properties/problems unique in intersession network coding have been discovered [4], [11], [13]. The absence of a fundamental understanding leads to suboptimal achievability results, including analysis for practical systems [21], butterfly-based construction [22], and its associated backlog rate-control algorithms [5], [7], [18].

In [23], [24], a new flow-based necessary and sufficient condition is established for *pairwise intersession network coding* (PINC) that allows intersession network coding between pairs of coexisting sessions. With a form similar to the max-flow theorem for multi-path routing and for intrasession network coding, the new characterization of PINC prompts tighter integration of cross-layer optimization and network coding. This work provides the first optimal solution combining both PINC and cross-layer optimization. The following questions

978-1-4244-2247-0/08/$25.00 ©2008 IEEE.



(a) Wireless Broadcast (b) Wireline Counterpart

Fig. 1. Modelling the wireless mulitcast advantage

will be answered in this work. How much is the optimal throughput improvement when PINC is used? How should one design distributed scheduling and routing protocols for PINC without much complexity overhead? What is the impact of PINC in terms of the stability of the system? It is known that routing is resilient to imperfect scheduling [14]. Does PINC also achieve a higher throughput by sacrificing the resiliency to imperfect scheduling?

II. Background & Preliminary Results

A. Analytical Framework for Wireless Multi-hop Networks

An important feature of wireless network is the broadcast nature of wireless media, which is termed the wireless multicast advantage (WMA). The WMA can be modelled as follows (see [28] for further details). For each node u with k neighbors $\{v_1, \cdots, v_k\}$, introduce $2^k - 1$ auxiliary nodes such that each auxiliary node corresponds to a non-empty element of the powerset of $\{v_1, \cdots, v_k\}$. Add $2^k - 1$ directed edges connecting u and each of the auxiliary nodes. For each auxiliary node, add directed edges from the auxiliary node to each node in the corresponding subset of neighbors. Fig. 1 illustrates a node with three neighbors. Hence, $2^3 - 1 = 7$ auxiliary nodes are added and connected to the corresponding neighbors following the powerset mapping. In a wireless network, every time a packet is about to be sent, the sender u chooses the target receiver(s) of the packet and determines the appropriate power level of the signal in order to reach the target receiver(s), which is equivalent to associating different "required power levels" to links connecting u and auxiliary nodes. Node u then decides which auxiliary link to use. Therefore, designing a wireless transmission scheme that exploits the advantages of the WMA is equivalent to designing a good routing/scheduling algorithm on its wireline counterpart *with the additional node-exclusive scheduling constraints that auxiliary nodes corresponding to the same u cannot be active simultaneously.* This framework takes into account the WMA and maps the wireless scheduling problem to a wireline scheduling problem while the underlying interference model for the former is absorbed as scheduling constraints for the latter problem.

B. Pairwise Intersession Network Coding

Consider directed acyclic network $G = (V, E)$, in which each edge is able to support one packet per unit time without transmission delay and high rate links are modelled as parallel edges. A pair of unicast sessions (s_1, t_1) and (s_2, t_2) would like to transmit two packets X_1 and X_2 (one for each session) simultaneously within a unit time. Pairwise intersession network coding is allowed and packets X_1 and X_2 can be mixed. The existence of a PINC solution that supports simultaneous transmission of X_1 and X_2 is characterized as follows [23]. We use $P_{u,v}$ to represent a path connecting nodes u and v.

Theorem 1: A PINC solution exists if and only if one of the following two conditions holds.

- Condition 1: There exist two edge-disjoint paths P_{s_1,t_1} and P_{s_2,t_2}.
- Condition 2: There exist six paths grouped into two sets $\{P_{s_1,t_1}, P_{s_2,t_1}, P_{s_2,t_2}\}$ and $\{Q_{s_2,t_2}, Q_{s_1,t_2}, Q_{s_1,t_1}\}$ such that for all $e \in E$,

$$1_{\{e \in P_{s_1,t_1}\}} + 1_{\{e \in P_{s_2,t_1}\}} + 1_{\{e \in P_{s_2,t_2}\}} \leq 2 \qquad (1)$$

$$1_{\{e \in Q_{s_2,t_2}\}} + 1_{\{e \in Q_{s_1,t_2}\}} + 1_{\{e \in Q_{s_1,t_1}\}} \leq 2, \qquad (2)$$

where $1_{\{\cdot\}}$ is the indicator function.

Theorem 1 shows that as the existence of routing solutions is equivalent to finding edge-disjoint paths, the existence of PINC solutions is equivalent to finding paths with *controlled edge overlap.* The subgraph G' induced by any six paths satisfying (1) and (2) will be referred as a pairwise intersession coding configuration (PICC). Theorem 1 can be generalized from two unicast sessions to two multicast sessions [24]. All results in this paper can be generalized from unicast sessions to multicast sessions provided the multicast version of Theorem 1 is used.

III. Optimal Congestion Control for PINC

We model a wireless network by its corresponding stationary counterpart denoted by $G = (V, E)$ where V is the set of network nodes plus auxiliary nodes and E is the edge set. Consider slotted transmission, a scheduling policy Θ is a collection of active edges and the associated power levels. Under a given interference model, we use r_e^Θ to denote the rate that can be supported on edge e under the scheduling policy Θ, and we often use $\mathbf{r}^\Theta$ for the collective rate vector. Let $\boldsymbol{\Theta}$ denote the collection of all policies and let $\mathcal{R} \triangleq \{\mathbf{r}^\Theta : \forall \Theta \in \boldsymbol{\Theta}\}$ denote the corresponding rates. Any rate vector $\mathbf{r} \in \mathrm{Co}(\mathcal{R})$, the convex hull of $\mathcal{R}$, can be achieved via time sharing. There are N different unicast sessions using the network to send data from source s_i to destination t_i where $i = 1, \cdots, N$. The utility function $U_i(x)$ for each session is strictly concave and monotonically increasing, where x is the end-to-end data rate for the session.

Theorem 1 says that utility optimization for multiple unicast sessions using only PINC can be cast as follows.

$$\max_{\mathbf{x} \geq \mathbf{0}, \mathbf{r} \in \mathrm{Co}(\mathcal{R})} \sum_{i=1}^{N} U_i \left(\sum_{k=1}^{|\mathcal{P}_i|} x_i^k + \sum_{j:j \neq i} \sum_{l=1}^{|\mathcal{PICC}_{ij}|} x_{ij}^l \right) \qquad (3)$$

$$\text{subject to} \quad \sum_{i=1}^{N} \sum_{k=1}^{|\mathcal{P}_i|} H_i^k(e) x_i^k + \sum_{(i,j):i \neq j} \sum_{l=1}^{|\mathcal{PICC}_{ij}|} \frac{H_{ij}^l(e) x_{ij}^l}{2} \leq r_e, \forall e \in E \qquad (4)$$

$$x_{ij}^l = x_{ji}^l, \quad \forall (i,j) : i < j, \forall l \qquad (5)$$

where $\mathcal{P}_i$ is the collection of paths from s_i to t_i along which packets will be routed without any coding operations and x_i^k is the rate assigned for the k-th path. $\mathcal{PICC}_{ij}$ is the collection of PICCs between sessions i and j on which intersession network coding will be performed and x_{ij}^l is the packet rate of source s_i that will be network coded using l-th PICC of $\mathcal{PICC}_{ij}$. Without loss of generality, we further assume the indice of $\mathcal{PICC}_{ij}$ and of $\mathcal{PICC}_{ji}$ are consistent. Namely, for all l, the l-th PICC of $\mathcal{PICC}_{ij}$ is also the l-th PICC of $\mathcal{PICC}_{ji}$. Since in PINC, packets from s_i and s_j are coded bijectively with each other, the system requires the equal-rate constraint (5).

In (4), $H_i^k(e)$ is the indicator function whether the k-th path in $\mathcal{P}_i$ uses edge e. $H_{ij}^l(e)$ is the indicator function whether the l-th PICC in $\mathcal{PICC}_{ij}$ uses edge e. PINC ensures that the two packet flows (with rates x_{ij}^l and x_{ji}^l respectively) jointly use only the max rate $\max(H_{ij}^l(e)x_{ij}^l, H_{ji}^l(e)x_{ji}^l)$ instead of the sum rate $H_{ij}^l(e)x_{ij}^l + H_{ji}^l(e)x_{ji}^l$. By the fact that the indicator function is symmetric by definition, i.e. $H_{ij}^l(e) = H_{ji}^l(e)$, and by the equal-rate constraint in (5), we have

$$\max(H_{ij}^l(e)x_{ij}^l, H_{ji}^l(e)x_{ji}^l) = \frac{H_{ij}^l(e)x_{ij}^l + H_{ji}^l(e)x_{ji}^l}{2}. \quad (6)$$

Summing over rates contributed by multi-path routing and by multi-PICC network coding leads to the capacity constraint (4). The non-negative rate vector $\mathbf{x}$, including all x_i^k and x_{ij}^l, is the subject of rate control and the edge rate vector $\mathbf{r} \in \mathrm{Co}(\mathcal{R})$ is the subject of optimal scheduling and time-sharing.

The optimal solution of (3–5) can be achieved in a decoupled way by solving its dual problem via the gradient method.

Algorithm A:

Rate Update For each source s_i, update its rate vector $\mathbf{x}_i[\tau] = \{x_i^k[\tau], x_{ij}^l[\tau] : \forall k, j, l\}$ for the τ-th time slot by

$$\begin{aligned}\mathbf{x}_i[\tau] = \arg\max_{\mathbf{x}_i \geq \mathbf{0}} U_i &\left(\sum_{k=1}^{|\mathcal{P}(i)|} x_i^k + \sum_{j:j\neq i}\sum_{l=1}^{|\mathcal{PICC}_{ij}|} x_{ij}^l\right)\\ - \sum_{e\in E} q_e[\tau] &\left(\sum_{k=1}^{|\mathcal{P}_i|} H_i^k(e)x_i^k + \sum_{j:j\neq i}\sum_{l=1}^{|\mathcal{PICC}_{ij}|} \frac{H_{ij}^l(e)x_{ij}^l}{2}\right)\\ - \sum_{l=1}^{|\mathcal{PICC}_{ij}|} &\left(\sum_{j:j>i} q_{ij}^l[\tau]x_{ij}^l - \sum_{j:j<i} q_{ji}^l[\tau]x_{ij}^l\right)\\ -\alpha_i &\left(\sum_{k=1}^{|\mathcal{P}_i|}\left(x_i^k - y_i^k\right)^2 + \sum_{j:j\neq i}\sum_{l=1}^{|\mathcal{PICC}_{ij}|}\left(x_{ij}^l - y_{ij}^l\right)^2\right),\end{aligned} \quad (7)$$

where $q_e[\tau]$ and $q_{ij}^l[\tau]$ are dual variables at the τ-th time slot, generally termed the *queue lengths*, whose values are feedback to s_i. α_i are small constants and $\mathbf{y}_i = \{y_i^k, y_{ij}^l : \forall i, j, k, l\}$ are auxiliary variables of the proximal method in order to eliminate oscillating behaviors. Periodically, $\mathbf{y}_i$ is set to $\mathbf{x}_i[\tau]$ and the iteration continues using the new $\mathbf{y}_i$.

Scheduling Update The network selects the optimal scheduling policy for the τ-th time slot by

$$\mathbf{r}[\tau] = \arg\max_{\mathbf{r}\in\mathcal{R}} \sum_{e\in E} q_e[\tau] r_e.$$

Queue-length Update Each link e updates its dual variable $q_e[\tau+1]$ according to the following equation.

$$\begin{aligned}q_e[\tau+1] = \Bigg[q_e[\tau] + \beta_e &\Bigg(\sum_{i=1}^{N}\sum_{k=1}^{|\mathcal{P}_i|} H_i^k(e)x_i^k[\tau]\\ &+ \sum_{(i,j):i\neq j}\sum_{l=1}^{|\mathcal{PICC}_{ij}|}\frac{H_{ij}^l(e)x_{ij}^l[\tau]}{2} - r_e[\tau]\Bigg)\Bigg]^+,\end{aligned}$$

where $[\cdot]^+ \triangleq \max(\cdot, 0)$ is the projection operator and β_e is a small step size for the gradient method.

Balance Update Each destination t_i updates the dual variable $q_{ij}^l[\tau+1]$ for all $j > i$. q_{ij}^l accounts the difference between packet rates of sources i and j that use the same PICC.

$$q_{ij}^l[\tau+1] = q_{ij}^l[\tau] + \beta_i\left(x_{ij}^l[\tau] - x_{ji}^l[\tau]\right), \forall j : j > i, \forall l,$$

where β_i is a small step size for the gradient method.

The four different parts of Algorithm A are coupled implicitly via the queue lengths and the balance information at the destinations. One important observation is that with PINC, only the rate and the balance updates, performed at the sources s_i and destinations d_i, differ from its routing counterpart. The scheduling and queue-length updates remain identical. The impact of PINC on rate-control and scheduling is thus minimal and confined only in sources and destinations. We then have the following convergence result from standard results on proximal algorithms.

Proposition 1: If the update period of the proximal variable $\mathbf{y}_i \leftarrow \mathbf{x}_i[\tau]$ is sufficiently large and the step sizes β_e and β_i are sufficiently small, Algorithm A converges to the optimal solution of (3–5).

In addition to utility maximization, the proposed decoupled cross-layer optimization algorithm also prompts the following notion of stability.

Definition 1: A system load $\{w_i : i = 1, \cdots, N\}$ can be stabilized by Algorithm A if there exists a non-negative vector $\mathbf{w} = \{w_i^k, w_{ij}^l : \forall i, j, k, l\}$ such that

$$w_i = \sum_{k=1}^{|\mathcal{P}_i|} w_i^k + \sum_{j:j\neq i}\sum_{l=1}^{|\mathcal{PICC}_{ij}|} w_{ij}^l, \forall i \quad (8)$$

$$\text{and } w_{ij}^l = w_{ji}^l, \forall (i,j) : i < j, \forall l. \quad (9)$$

And if we replace the "rate update" in Algorithm A by a fixed rate assignment $\mathbf{x}[\tau] = \mathbf{w}$, then the dual variables $q_e[\tau]$ and $q_{ij}^l[\tau]$ are bounded away when τ tends to infinity.

Let Λ denote a set of system loads $\Lambda = \{\{w_i\}_i\}$ such that for any $\{w_i\}_i \in \Lambda$, there exists a non-negative vector $\mathbf{w} = \{w_i^k, w_{ij}^l : \forall i, j, k, l\}$ satisfying (8) and (9), a rate vector $\mathbf{r} \in \mathrm{Co}(\mathcal{R})$, and jointly $\mathbf{w}$ and $\mathbf{r}$ satisfy

$$\sum_{i=1}^{N}\sum_{k=1}^{|\mathcal{P}_i|} H_i^k(e)w_i^k + \sum_{(i,j):i\neq j}\sum_{l=1}^{|\mathcal{PICC}_{ij}|}\frac{H_{ij}^l(e)w_{ij}^l}{2} \leq r_e, \forall e \in E.$$

We then have the following stability result.

Proposition 2: With sufficiently small step sizes β_e and β_i, any system load $\{w_i\}_i$ that is in the interior of Λ can be stabilized by the optimal rate-control and scheduling Algorithm A.

IV. PINC with Imperfect Scheduling

Similar to its routing-counterpart, it is computationally expensive to find the optimal scheduling decision in Algorithm A. Depending on different interference models, finding the optimal scheduling $\mathbf{r}$ that maximizes $\sum_e q_e[\tau]r_e$ is NP-hard in many cases and requires centralized implementation. In practice, we would often have to resort to imperfect scheduling schemes that select the rate vector $\mathbf{r}[\tau]$ that achieves γ fraction of the maximum value. Namely,

$$\sum_{e\in E} q_e[\tau]r_e[\tau] \geq \gamma \max_{\mathbf{r}} \sum_{e\in E} q_e[\tau]r_e, \qquad (10)$$

where γ is a constant in $[0,1]$. With imperfect scheduling ($\gamma < 1$) the tie between Algorithm A and the gradient method for the dual problem is severed and Algorithm A may not converge to any fixed-point solution. Our results show that even with imperfect scheduling, the proposed cross-layer optimization algorithm still shows tractable performance in terms of the stability region.

Proposition 3: With sufficiently small β_e and β_i, any system load $\{w_i\}$ that is in the interior of $\gamma\Lambda$ can be stabilized by the decoupled cross-layer algorithm with imperfect scheduling. Proposition 3 collapses to Proposition 2 when $\gamma = 1$.

A. Networks with Dynamic Arrivals

We also consider the case of dynamic system loads. Consider N classes of users. For all i, users in class i have a common utility function $U_i(x) = \kappa_i \frac{x^{1-\nu}}{1-\nu}$ or $U_i(x) = \kappa_i \log(x)$ where $\kappa_i, \nu > 0$ are predefined system parameters. All users in class i will send packets from s_i to t_i and will use the same routing paths in $\mathcal{P}_i$ and the same PICCs in $\mathcal{PICC}_{ij}$ for transmission. We also assume that users of class i arrive according to a Poisson process with rate λ_i and each user needs to send a file whose size is exponentially distributed with mean $\frac{1}{\mu_i}$. The system load of this network with dynamic arrivals is then defined as $\left\{\left(\frac{\lambda_i}{\mu_i}\right) : \forall i\right\}$. The dynamic nature of this setting prompts a slightly different definition of stability.

Definition 2: A system load $\left\{\left(\frac{\lambda_i}{\mu_i}\right) : \forall i\right\}$ can be stabilized by Algorithm A if the dual variables $q_e[\tau]$ and $q^l_{ij}[\tau]$ are bounded away from infinity for each iteration with probability one.[1]

We then have the following stability result.

Proposition 4 (Stability for Dynamic Systems): With sufficiently small α_i, β_e, and β_i, any system load $\left\{\left(\frac{\lambda_i}{\mu_i}\right) : \forall i\right\}$ that is in the interior of $\gamma\Lambda$ can be stabilized by the decoupled cross-layer algorithm with imperfect scheduling.

Proposition 4 implies that although the instantaneous system load imposed on the network may well exceed the network capacity, as long as the average system load is within γ times the capacity and the optimal rate control in (7) works jointly with an imperfect scheduling scheme, the queue lengths of the network are bounded away from infinity and the system is stable. This stability result that was once known for only routing-based transmission in [14] also holds for PINC. Shifting from routing- to network-coding-based solutions enhances the throughput without sacrificing the associated stability even with imperfect scheduling.

B. Selecting Scheduling Policies from A Reduced Set

It is suggested in [14] that for routing, one can select the imperfect scheduling policy from a small pool of candidates instead of from the general policy set $\mathbf{\Theta}$, which reduces further the complexity of scheduling update. The same complexity reduction method can be applied to the PINC-based solution as well. To be more explicit, consider a reduced set of policy candidates with K different imperfect scheduling policies θ_a, $a = 1, \cdots, K$ such that each corresponds to a rate vector $\mathbf{r}^{\theta_a}$ and when the queue lengths are $\{q^a_e : \forall e\}$, policy θ_a is a γ_a-approximation policy such that

$$\sum_{e\in E} q^a_e r^{\theta_a}_e \geq \gamma_a \max_{\mathbf{r}} \sum_{e\in E} q^a_e r_e. \qquad (11)$$

Then if the following condition holds in the τ-th unit time

$$\max_{a=1,\cdots,K} \sum_{e\in E} q_e[\tau] r^{\theta_a}_e$$
$$\geq \gamma \min_{a=1,\cdots,K} \left(\sum_{e\in E} [q_e[\tau] - q^a_e]^+ r^{\max}_e + \frac{\sum_{e\in E} q^a_e r^{\theta_a}_e}{\gamma_a} \right),$$

for some γ, where $r^{\max}_e$ is the maximum possible rate along edge e, then policy θ^*_a that maximizes the left-hand side is a γ-approximation of the optimal scheduling policy with weights $q_e[\tau]$ on each edge.

C. The Node Exclusive Interference Model

The above stability results of cross-layer optimization take into account general interference models by the achievable rate region of all scheduling policies. For the special case of node-exclusive mode such that the data rate of each link is fixed at c_e and each node can only send to or receive from one other node at any time, the objective function of optimal scheduling is equivalent to

$$\max_{\mathbf{r}} \sum_{e\in E} q_e[t]r_e = \max_{\mathcal{M}} \sum_{e\in E} q_e[t]c_e 1_{\{e\in\mathcal{M}\}} = \max_{\mathcal{M}} \sum_{e\in\mathcal{M}} q_e[t]c_e,$$

where $\mathcal{M}$ is a matching of the underlying graph G. Finding the optimal scheduling thus becomes the maximum weighted matching problem. It is known that the greedy maximal matching is a $1/2$-approximation of the maximum weighted matching solution. Hence if the scheduling update in Algorithm A is replaced by a greedy maximal matching, then stability region is at least $\frac{1}{2}\Lambda$ and any system load within $\frac{1}{2}\Lambda$ can be stabilized.

[1] In contrast with Definition 1 where the rate update rule is modified for a static system load, for a dynamic system load, the optimal rate update is am important ingredient of Algorithm A and has to remain intact. Only the scheduling update will be changed to imperfect scheduling as in (10).

In the node exclusive interference model, one can also obtain inner and outer bounds of PINC stability region Λ by Edmonds' matching polytope theorem. For any $\mathbf{w} = \{w_i^k, w_{ij}^l : \forall i, j, k, l\}$, consider the following condition:

$$\forall v, \quad \sum_{e: e \in E(v)} \frac{1}{c_e} \left(\sum_{i=1}^{N} \sum_{k=1}^{|\mathcal{P}_i|} H_i^k(e) w_i^k + \sum_{(i,j): i \neq j} \sum_{l=1}^{|\mathcal{PICC}_{ij}|} \frac{H_{ij}^l(e) w_{ij}^l}{2} \right) \leq 1, \quad (12)$$

where $E(v)$ contains all edges adjacent to v counting both incoming and outgoing edges. Let

$$\Psi = \{\{w_i\}_i : \exists \mathbf{w} \text{ satisfying (8), (9), and (12)}\}. \quad (13)$$

One can prove that $\frac{2}{3}\Psi \subseteq \Lambda \subseteq \Psi$ for PINC. As suggested in the layered routing-based solutions [20], [30], the lower bound $\frac{2}{3}\Psi$, stated without specifying the underlying scheduling scheme, can be used as a rate-region for the layered approach of PINC rate-optimization, which demonstrates the benefit of the proposed cross-layer PINC scheme since the cross-layer region Λ is larger than the upper bound $\frac{2}{3}\Psi$ of the layered approach. It is worth mentioning that with imperfect scheduling, both layered and cross-layer approaches suffer the same amount of performance degradation (in the worst case, the stability region is reduced by half with greedy maximal matching) and the throughput advantages of cross-layer versus layered approaches remain.

D. Distributed Implementation for the Node Exclusive Case

The flow-based characterization in Theorem 1 also enables other distributed 1/2-approximation of the perfect scheduling with different degrees of distributiveness for the node exclusive model. The first one relies on the *locally heaviest edge* [17] and the second one is built around *maximal matching*. For the following, we say a node is *matched* if one of its incoming or outgoing edges is scheduled.

1) Greedy Algorithm Based on Locally Heaviest Edge: Each unmatched node v scans its unmatched neighbors and records the corresponding edge weights q_e for its unmatched neighbors. v then sends a matching (scheduling) request along the edge with the largest weight (among all edges connecting its unmatched neighbors). For any edge e, if both its end nodes choose to send a matching request along e, we put e in the current schedule. Repeat the scan/record/request/schedule iteration until a maximal weighted matching is found. [17] guarantees that a $\frac{1}{2}$-approximation matching is found. This *locally greedy scheme* requires only local comparison instead of a global search as that is used in the greedy maximal matching. As a result, its implementation admits much higher degrees of distributiveness and tighter pipelining for faster convergence.

2) Fully Distributed Maximal Matching Scheduling: In the locally greedy maximal matching scheme, each unmatched node sends out one scheduling request in a time slot and waiting for a possible match. One can accelerate the convergence by sending out matching requests along all edges connecting its unmatched neighbors. For any edge $e = uv$, if both its end nodes choose to send a matching request along e and neither u nor v is matched by other edges, we put e in the current schedule and mark both u and v matched, which prevents any other $e' = uw$ or $e' = vw$ being scheduled simultaneously. Some randomization and local coordination is necessary in this maximal matching method. By sacrificing the chance of selecting a locally heaviest weighted edge, the convergence rate is improved. To ensure its $\frac{1}{2}$-approximation capability, we need to modify the update rules and the underlying mechanism of collecting dual variable information. A detailed description is as follows.

Additional Q_v Update For every node $v \in V$,

$$Q_v[\tau] = \sum_{e: e \in E(v)} q_e[\tau]. \quad (14)$$

Rate Update At each source s_i, update its rate vector $\mathbf{x}_i[\tau] = \{x_i^k[\tau], x_{ij}^l[\tau] : \forall k, j, l\}$ for the τ-th time slot by

$$\begin{aligned} \mathbf{x}_i[\tau] = \arg\max_{\mathbf{x}_i \geq \mathbf{0}} U_i & \left(\sum_{k=1}^{|\mathcal{P}(i)|} x_i^k + \sum_{j: j \neq i} \sum_{l=1}^{|\mathcal{PICC}_{ij}|} x_{ij}^l \right) \\ - \sum_{v \in V} Q_v[\tau] & \sum_{e: e \in E(v)} \frac{1}{c_e} \left(\sum_{k=1}^{|\mathcal{P}_i|} H_i^k(e) x_i^k \right. \\ & \left. + \sum_{j: j \neq i} \sum_{l=1}^{|\mathcal{PICC}_{ij}|} \frac{H_{ij}^l(e) x_{ij}^l}{2} \right) \\ - & \sum_{l=1}^{|\mathcal{PICC}_{ij}|} \left(\sum_{j: j > i} q_{ij}^l[\tau] x_{ij}^l - \sum_{j: j < i} q_{ji}^l[\tau] x_{ij}^l \right) \\ - \alpha_i & \left(\sum_{k=1}^{|\mathcal{P}_i|} \left(x_i^k - y_i^k \right)^2 + \sum_{j: j \neq i} \sum_{l=1}^{|\mathcal{PICC}_{ij}|} \left(x_{ij}^l - y_{ij}^l \right)^2 \right). \end{aligned}$$

Then *each user of class i* will sent at rate $\mathbf{x}_i[\tau]$. If there are $n_i[\tau]$ users of class i, then the total traffic for the i-th class of users is $n_i[\tau]\mathbf{x}_i[\tau]$.

Imperfect Scheduling Update Choose the *maximal matching* of the *subgraph induced by edges with* $q_e[\tau] \geq 1$. Let $\mathcal{M}[\tau]$ denote the selected maximal matching in the τ-th time slot.

Queue-length Update For each edge $e \in E$, update a *virtual queue length* $q_e[\tau+1]$ by

$$\begin{aligned} q_e[\tau+1] = \Bigg[q_e[\tau] + \frac{\beta_e}{c_e} \sum_{i=1}^{N} n_i[\tau] & \left(\sum_{k=1}^{|\mathcal{P}_i|} H_i^k(e) x_i^k[\tau] \right. \\ \left. + \sum_{j: j \neq i} \sum_{l=1}^{|\mathcal{PICC}_{ij}|} \frac{H_{ij}^l(e) x_{ij}^l[\tau]}{2} \right) & - \beta_e 1_{\{e \in \mathcal{M}[\tau]\}} \Bigg]^+ . \end{aligned}$$

Balance Update For each t_i, update $q_{ij}^l[\tau+1]$ for all $j > i$ and for all l:

$$q_{ij}^l[\tau+1] = q_{ij}^l[\tau] + \beta_i \left(x_{ij}^l[\tau] n_i[\tau] - x_{ji}^l[\tau] n_j[\tau] \right). \quad (15)$$

Proposition 5: With sufficiently small step sizes $\alpha_i, \beta_e, \beta_i$ and with the node exclusive interference model, any system load $\left\{\left(\frac{\lambda_i}{\mu_i}\right) : \forall i\right\}$ that is in the interior of 0.5Λ can be stabilized by the above maximal-matching-based algorithm.

The proof of this proposition is sketched as follows. Following the same approach as in [14], one can construct a fictitious system with $|V|$ edges mimicking the stability region upper bound Ψ, namely, each "v edge" satisfies the "rate-constraint"

$$\sum_{e:e\in E(v)} \frac{1}{c_e}\left(\sum_{i=1}^{N}\sum_{k=1}^{|\mathcal{P}_i|} H_i^k(e)w_i^k + \sum_{(i,j):i\neq j}\sum_{l=1}^{|\mathcal{PICC}_{ij}|} \frac{H_{ij}^l(e)w_{ij}^l}{2}\right) \leq r_e.$$

The modified virtual queue length $q_e[\tau]$ now corresponds to the cost of scheduling link e in terms of the cumulative time-wise delay instead of the cumulative packet-wise delay. We can then show that any maximal matching for the original graph focusing on non-trivial edges (with significant time-delay $q_e[\tau] > 1$) is asymptotically a 1/2-approximation scheduling in the new "v-edge" system.

V. Conclusion

We have considered the wireless scheduling problem with pairwise intersession network coding (PINC). Our results have proven that in a wireless multi-hop network, the throughput advantage of PINC can be achieved without sacrificing the stability conditions. Similar to its routing counterpart, a decoupled algorithm has been derived and we have shown that the only new component necessary for PINC is the *balance update* performed at the receiver. Following this new formulation, the impact of imperfect scheduling on PINC-based rate-control algorithm has been studied. With a γ-approximation imperfect scheduling scheme, at least γ fraction of the stability region of PINC can be retained for wireless networks with both static and dynamic packet arrival time. With a node-exclusive interference model, fully distributed scheduling schemes have been derived based either on the locally heaviest edge or on maximal matching.

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.

[2] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading structure for randomness in wireless opportunistic routing," in *Proc. SIGCOMM 2007*. Kyoto, Japan, Aug. 2007.

[3] P. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proc. 41th Annual Allerton Conf. on Comm., Contr., and Computing*. Monticello, IL, Oct. 2003.

[4] R. Dougherty, C. Freiling, and K. Zeger, "Insufficiency of linear coding in network information flow," *IEEE Trans. Inform. Theory*, vol. 51, no. 8, pp. 2745–2759, August 2005.

[5] A. Eryilmaz and D. Lun, "Control for inter-session network coding," in *Proc. Workshop on Network Coding, Theory & Applications*, Jan. 2007.

[6] N. Harvey, K. Jain, L. Lau, C. Nair, and Y. Wu, "Conservative network coding," in *Proc. 44th Annual Allerton Conf. on Comm., Contr., and Computing*. Monticello, IL, Sept. 2006.

[7] T. Ho, Y. Chang, and K. Han, "On constructive network coding for multiple unicasts," in *Proc. 44th Annual Allerton Conf. on Comm., Contr., and Computing*. Monticello, IL, Oct. 2006.

[8] T. Ho, M. Médard, R. Koetter, D. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Trans. Inform. Theory*, vol. 52, no. 10, pp. 4413–4430, April 2006.

[9] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "XORs in the air: Practical wireless network," in *Proc. ACM sigcom*, 2006.

[10] R. Koetter and M. Médard, "An algebraic approach to network coding," *IEEE/ACM Trans. Networking*, vol. 11, no. 5, pp. 782–795, October 2003.

[11] A. Lehman, "Network coding," Ph.D. dissertation, MIT, 2005.

[12] S.-Y. R. Li, R. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inform. Theory*, vol. 49, no. 2, pp. 371–381, Feb. 2003.

[13] Z. Li and B. Li, "Network coding: the case of multiple unicast sessions," in *Proc. 42nd Annual Allerton Conf. on Comm., Contr., and Computing*. Monticello, Illinois, USA, Sept. 2004.

[14] X. Lin and N. Shroff, "The impact of imperfect scheduling on cross-layer congestion control in wireless networks," *IEEE Trans. Networking*, vol. 14, no. 2, pp. 302–315, April 2006.

[15] ——, "Utility maximization for communication networks with multi-path routing," *IEEE Trans. Automat. Control*, vol. 51, no. 5, pp. 766–781, May 2006.

[16] D. Lun, N. Ratnakar, M. Médard, R. Koetter, D. Karger, T. Ho, E. Ahmed, and F. Zhao, "Minimum-cost multicast over coded packet networks," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2608–2623, June 2006.

[17] R. Preis, "Linear time 1/2-approximation algorithm for maximum weighted matching in general graphs," ser. Lecture Notes in Computer Science, vol. 1563/1999. Springer Berlin / Heidelberg, 2001, pp. 259–269.

[18] J. Price and T. Javidi, "Network coding for resource redistribution in a unicast network," in *Proc. 45th Annual Allerton Conf. on Comm., Contr., and Computing*. Monticello, IL, Sept. 2007.

[19] Y. Sagduyu and A. Ephremides, "On joint MAC and network coding in wireless ad hoc networks," *IEEE Trans. Inform. Theory*, vol. 53, no. 10, October 2007.

[20] S. Sarkar and L. Tassiulas, "End-to-end bandwidth guarantees through fair local spectrum share in wireless ad hoc networks," in *Proc. IEEE Conf. Decision and Contr.* Maui, HI, USA, Dec. 2003, pp. 564–569.

[21] S. Sengupta, S. Rayanchu, and S. Banerjee, "An analysis of wireless network coding form unicast sessions: the case for coding aw3are routing," in *Proc. INFOCOM 2008*. Phoenix, AZ, USA, April 2008.

[22] D. Traskov, N. Ratnakar, D. Lun, R. Koetter, and M. Médard, "Network coding for multiple unicasts: An approach based on linear optimization," in *Proc. Int'l Symp. Inform. Theory*. Seattle, USA, July 2006.

[23] C.-C. Wang and N. Shroff, "Beyond the butterfly — a graph-theoretic characterization of the feasibility of network coding with two simple unicast sessions," in *Proc. Int'l Symp. Inform. Theory*. Nice, France, June 2007.

[24] ——, "Intersession network coding for two simple multicast sessions," in *Proc. 45th Annual Allerton Conf. on Comm., Contr., and Computing*. Monticello, IL, Sept. 2007.

[25] M. Wang and B. Li, "r^2: Random push with random network coding in live peer-to-peer streaming," *IEEE J. Selected Areas of Commun.*, vol. 25, no. 9, pp. 1655–1666, Dec. 2007.

[26] Y. Wu, M. Chiang, and S.-Y. Kung, "Distributed utility maximization for network coding based multicasting: A critical cut approach," in *Proc. IEEE NetCod*. Boston, Massachusetts, April 2006.

[27] Y. Wu, P. Chou, and S.-Y. Kung, "Minimum-energy multicast in mobile ad hoc networks using network coding," *IEEE Trans. Commun.*, vol. 53, no. 11, pp. 1906–1918, Nov. 2005.

[28] Y. Wu, P. Chou, Q. Zhang, K. Jain, W. Zhu, and S.-Y. Kung, "Network planning in wireless ad hoc networks: a cross-layer approach," *IEEE J. Selected Areas Commun.*, vol. 23, no. 1, pp. 136–150, Jan. 2005.

[29] Y. Wu and S.-Y. Kung, "Distributed utility maximization for network coding based multicasting: A shortest path approach," *IEEE J. Selected Areas in Commun.*, vol. 24, pp. 1475–1488, August 2006.

[30] Y. Yi and S. Shakkottai, "Hop-by-hop congestion control over a wireless multi-hop network," in *Proc. IEEE INFOCOM*. Hong Kong, March 2004, pp. 2548–2558.