On the Benefits of Network Coding in Multi-Channel Wireless Networks

Xinyu Zhang, Baochun Li
Department of Electrical and Computer Engineering
University of Toronto
Email: {xzhang, bli}@eecg.toronto.edu

***Abstract*—Wireless mesh networks have emerged as a favorable infrastructure that promises to unify the existing 802.11 wireless LANs. With multiple orthogonal channels and possibly multiple interfaces on the mesh nodes, such networks can provide broadband access for a large number of wireless clients. However, efficient assignment of channels to the available network interfaces has long been a daunting task for network designers. Existing heuristic and theoretical work unanimously focuses on joint design of channel assignment with the conventional transport/IP/MAC architecture. In this paper, we show that a new paradigm, network coding, is able to further increase the capacity of multi-channel mesh networks. We propose a joint optimization problem that accounts for routing, channel assignment, and network coding, and analyze its potential performance gains over the non-coding schemes. This problem inspires a practical algorithm that naturally combines network coding and routing. We also explore the benefits of network coding for emerging multi-channel wireless networks, including 802.16 and 802.11n, and derive the upper bound for its performance gains over existing channel assignment protocols.**

I. Introduction

Wireless mesh networks are promising to connect wireless devices over a wide area, allowing them to share information with each other and to access the Internet. The backbone of such networks are formed by access points called *mesh routers*, which serve a number of subscribers called *mesh clients*. The capacity of mesh networks can be boosted by allowing the coexistence of multiple orthogonal channels. Recently, it has also been shown that mounting multiple interfaces on one 802.11 device is feasible, and much more cost-efficient than increasing the number of access points [1]. Such trends have triggered a large body of work on designing multi-channel wireless mesh networks (MC-WMN) with multiple interfaces.

Since the number of orthogonal channels are limited (3 in 802.11b/g and 12 in 802.11a), the key problem for MC-WMN is to assign appropriate channels to the interfaces on each mesh node, in order to ensure connectivity of the network, and to reduce the interference between neighboring links with overlapping channels, thereby maximizing the network capacity. Most existing work focused on incorporating the channel/interface design problem into the traditional network architecture, proposing joint design with routing [2]–[4], with topology control [5], with MAC protocols [6], [7], as well as with congestion control [8]. The perspectives of existing problem formulations include both optimization based theoretical studies and heuristic based protocol design. However, most of them are confined within conventional network protocol stacks.

This work was supported in part by LG Electronics, and by Bell Canada through its Bell University Laboratories R&D program.

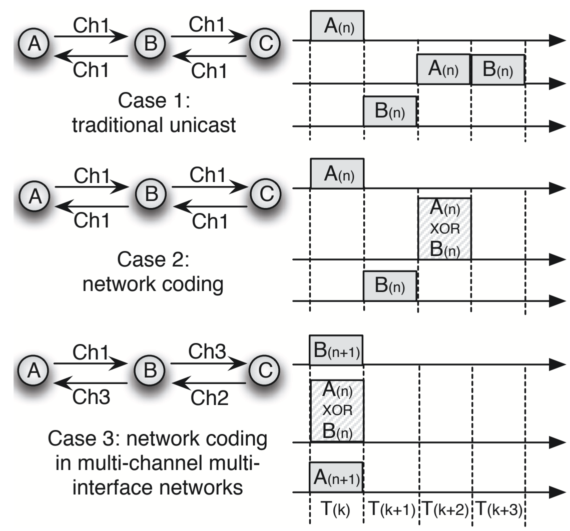

Fig. 1. The basic scenario demonstrating the benefits of network coding in multi-channel multi-interface wireless mesh networks.

In this paper, we add a new dimension — network coding — to the channel assignment problem, and demonstrate the performance gains of network coding in multi-channel multi-interface wireless mesh networks. As a preliminary evaluation, we consider the simplest form of network coding, *i.e.*, coding over GF(2) [9], [10], which allows intermediate relay nodes to opportunistically XOR incoming packets heading towards different next-hops, based on prior knowledge of the decodability at the intended downstream nodes. The encoding nodes broadcast the coded packets to all downstream nodes, thereby reducing the number of transmissions compared with traditional routing. As an intuitive justification of the network coding potential in MC-WMN, consider the scenario in Fig. 1, where node A and C intend to exchange packets with each other through a shared intermediate forwarder B. With traditional routing, 4 time slots are needed to finish the packet exchange. Using XOR network coding, 3 slots are needed. When applying network coding in MC-WMN, however, the amortized transmission time is only 1 slot since the transmission of new packets from A and C can overlap with the broadcasting of previously encoded packets. Although it is possible to achieve the same performance by assigning orthogonal channels to all 4 links, this would require node B to be equipped with 4 interfaces, while network coding only needs 3 interfaces. The saved interface can be assigned another channel and communicate with other nodes, which further increases the network capacity.

Inspired by the above scenario, we wonder: how much benefit can network coding provide in large-scale multi-hop MC-WMN? In particular, when routing and network coding are both taken into account, is there an optimal channel assignment that maximizes the network throughput? It is non-trivial to answer either of these questions. Although assigning diverse channels to neighboring nodes implies less interference in general, it may conversely result in isolated nodes or reduce coding opportunities due to the loss of the broadcast advantage. On the other hand, extensive use of the broadcast channel may cause the channel capacity to be insufficiently utilized, and may even result in idle interfaces. To tackle this fundamental trade-off, we formulate an optimization problem that jointly optimizes **r**outing, **a**ssignment of channels, and **c**oding (henceforth referred to as RAC). Since this problem is NP-hard, we design a simulated annealing [11] based algorithm to approximate the optimal solution. We show that the optimal throughput with network coding can be much higher than non-coding schemes. Even under random channel assignment and using heuristic routing/coding schemes, the advantages of network coding are still notable.

The benefits of network coding in MC-WMN can be naturally extended to other state-of-the-art multi-channel networks. Specifically, we consider 802.16 [12], which is based on orthogonal frequency division multiple access (OFDMA), and 802.11n [13], which is based on multiple-input multiple-output (MIMO) communications. Our analysis demonstrates that network coding is able to improve the efficiency of channel allocation in such networks and increase the data rate. As far as we know, this is the first paper that explores the advantage of network coding in such multi-channel wireless networks.

In summary, the main contributions of this paper are as follows: (1) We formulate the problem of network coding in multi-channel multi-interface 802.11 mesh networks, and analyze its performance gains compared with non-coding schemes; (2) We design a distributed algorithm that improves the performance of MC-WMN by integrating network coding into traditional routing and channel assignment schemes; (3) We explore the implication of network coding in emerging multi-channel wireless networks, including 802.16 OFDMA systems and 802.11n MIMO systems.

The remainder of the paper is organized as follows. In Sec. II, we present a literature review of related work, including the channel assignment problems and the application of network coding to wireless networks. In Sec. III, we formulate the problem of joint routing, channel assignment, and network coding as a mixed-integer optimization problem and derive a heuristic algorithm to solve this problem. The optimal solution is compared against existing optimization based approaches to design MC-WMN. Sec. IV presents a practical distributed protocol that exploits network coding in MC-WMN. Then we continue to extend the algorithm to other multi-channel wireless networks and analyze its performance in Sec. V. Finally, Sec. VI concludes the paper.

II. Related Work

Multi-channel wireless mesh networks have been extensively explored in literature. Along this line of research, the work most closely related with ours is the joint routing and channel assignment scheme aiming at maximizing the aggregate network throughput when multiple concurrent unicast sessions are running [2], [5], [7], [14]. Since the problem is NP-hard in general [15], Alicherry *et al.* [2] approached it using a linear programming formulation, and used a centralized approximation algorithm to derive the feasible solution. Optimization based problem formulation has also been adopted by Das *et al.* [16]. With the addition of network coding, however, the problem becomes much more complicated, due to the inter-dependence between routing, channel assignment and network coding. Various heuristic algorithms have also been proposed to design distributed routing protocols that are promising for practical implementation [3], [4], [7], [15], [17]. These algorithms attempted to take into account the specific properties of MC-WMNs using new routing metrics. Our work can be complemented by such algorithms since even simply augmenting network coding onto any routing protocol for MC-WMN can boost network performance without any additional cost (Sec. III).

Network coding has been a promising information theoretic approach to improve the performance of wireless networks. By allowing encoding operations on intermediate forwarders, it has been shown that even the simplest form of network coding, the XOR (as shown in case 2 of Fig. 1), can considerably improve the UDP unicast throughput [9], [10]. Katti *et al.* [10] implemented XOR network coding on an 802.11 single-channel mesh network, which reduces the number of transmissions by allowing the encoding nodes to broadcast the coded packets, rather than sending them separately to each downstream. Following such system implementation, Sengupta *et al.* [18] formulated an optimization framework that jointly optimizes routing and XOR network coding. In particular, they identified several possible elemental topologies that may create coding opportunities, and incorporated the coding topologies into a multi-commodity flow problem. The key message of [18] is that the joint design of coding and routing, rather than the separate solution in [10], may significantly enlarge the benefits of network coding. Our work differs from [18] in that we consider the constraints imposed by the channel assignment problem, which results in a mixed-integer programming problem. In addition, we observe the fundamental trade-off between diversifying the channels of interfering links and unifying the channels of links that may create coding opportunities, which has not been explored in any previous work.

III. The RAC Optimization Problem

In this section, we present the system architecture and network models that impose constraints on the RAC optimization, *i.e.*, the joint routing, channel assignment and network coding problem in mesh networks. We solve the problem using simulated annealing, and analyze the performance gains of network coding over non-coding schemes.

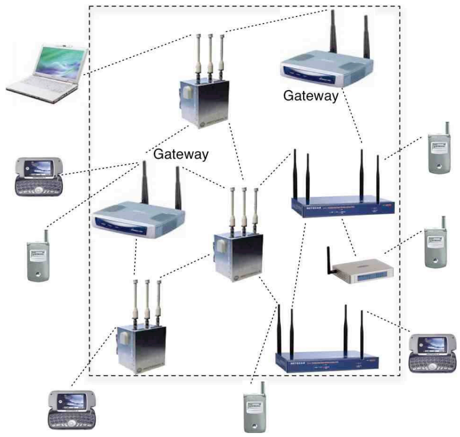

Fig. 2. The infrastructure of multi-channel multi-interface wireless mesh networks.

A. System Models

Wireless mesh networks typically consist of a static backbone and a number of mobile clients. The backbone includes gateways that can directly access the Internet, and mesh routers that serve as intermediate forwarders for each other and for the mesh clients. This paper mainly focuses on the scenarios where mesh routers and gateways are legacy 802.11a/b/g access points equipped with a diverse number of interfaces, as shown in Fig. 2. To fully utilize the available radio resources, the number of channels must not be fewer than the maximum number of interfaces mounted on a mesh node.

In particular, we are concerned with the unicast transmissions between any two nodes in the static backbone network (the interior of the rectangular region in Fig. 2). Due to the channel switching delay, it is infeasible to switch the interface from one channel to another on a per-packet basis [19]. Therefore, we adopt a fixed channel assignment, *i.e.*, the channels assigned to each interface on each node are fixed at the flow level. We further assume the set of unicast pairs remains stable over a large time-scale, and thus allowing for time to re-assign the channels to all interfaces when the flow structure changes.

B. Problem Formulation

Given a set of traffic demands, the foci of the RAC problem is to maximize the aggregate throughput, subject to coding constraints, routing constraints, MAC layer scheduling constraints, and the channel/interface constraints. Before modeling each of these constraints, we list the relevant variables and notations in Table. I.

The optimization objective. One simple objective function is to maximize the total throughput of all concurrent sessions, *i.e.*: $\max \ \sum_s \lambda_s$. However, such an optimization objective may unfairly allocate more bandwidth to the sessions that tend to have higher throughput, while those less competitive sessions might be starved. In view of this, we adopt an objective that satisfies the same proportion of the throughput demanded by each session:

$$\max \quad \gamma \tag{1}$$

$$\text{subject to:} \quad \lambda_s = \gamma \cdot d_s, \forall s \in \xi. \tag{2}$$

TABLE I
LIST OF VARIABLES AND NOTATIONS

V	The set of nodes in the target network
E	The set of directed links in the network
$G(V,E)$	The topology graph formed by all nodes and edges
ξ	The set of unicast sessions
ω	The set of non-overlapping channels
λ_s	The end-to-end throughput of the unicast session s
(i,j)	The directed link (edge) with transmitter i and receiver j
$U^c_{(i,k)}(s)$	The flow rate of session s routed over link (i,k) through channel c
$b^c_{jik}(s,s')$	The rate of the traffic that is coded on node i and broadcasted to node j and k, which are the next-hops for session s and s', respectively.
M^c_i	The total rate of traffic belonging to channel c, which interferes node i, or is transmitted/received by i
C_c	The capacity of channel c
$I(i)$	The set of nodes interfering with node i
B^c_i	The amount of traffic that belongs to channel c and that is sent or received by node i
N^c_i	The number of interfaces on node i tuned to channel c
R_i	The number of interfaces on node i
$\alpha(i)$	The number of unassigned interfaces on node i
$c \in i$	Channel c is used by node i
$c \in (i,j)$	Channel c is used by edge (i,j)

where d_s denotes the traffic demand of session s and γ is the throughput for routing all sessions' demands.

The coding constraint. In general, the XOR coding opportunities exist in two types of scenarios: the *information exchange paradigm* and the *opportunistic listening paradigm*. In both paradigms, the encoding node is the shared relay for different sessions. An XOR encoding happens only if the encoding node estimates that the corresponding next-hops have all but one of the packets that are encoded, and the missing one is exactly what each next-hop is intended to obtain by decoding. Specifically, in the former case, each next-hop decodes the encoded packet by XORing it with a packet that it previously sent and cached, which is best illustrated in case 3 of Fig. 1. In the latter case, the encoded packet is XORed with a packet that the next-hop has *overheard* (Fig. 3). In a multichannel system, the overhearing opportunities are scarce, since neighboring nodes tend to use orthogonal frequency bands unless they serve for the same session. Therefore, we only formulate the information exchange paradigm in this paper. The quantitative study of the opportunistic listening paradigm is left as future work.

Assume two consecutive links (j,i) and (i,k) belong to one of the paths for session s, and links (k,i) and (i,j) belong to one of the paths for session s', then a coding opportunity can be created at node i by XORing packets from both sessions, and then broadcasting the coded packets to i and k. It is straightforward to see that the encoding and broadcasting rate at node i is limited by the minimum incoming rate of the two sessions, flowing through (j,i) and (k,i), respectively. Thus we have:

$$b^c_{jik}(s,s') \le \sum_{q\in\omega} U^q_{(k,i)}(s'), \forall s \in \xi, s' \in \xi, (j,i) \in E, (k,i) \in E, c \in \omega \tag{3}$$

In addition, the broadcast is symmetric, thus:

$$b^c_{jik}(s,s') = b^c_{kij}(s',s) \tag{4}$$

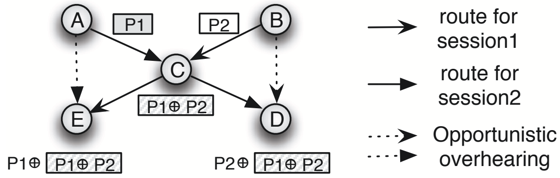

Fig. 3. A typical scenario where network coding opportunity exist for nodes that are overhearing packets.

The routing constraint. The routing constraint requires flow conservation at the intermediate forwarders for each session, except the corresponding source and destination nodes, *i.e.*,

$$\sum_{s'\in\xi}\sum_{j,k}\sum_{c\in\omega} b^c_{jik}(s,s') + \sum_{k}\sum_{c\in\omega} U^c_{(i,k)}(s)$$
$$-[\sum_{s'\in\xi}\sum_{j,k}\sum_{c\in\omega} b^c_{jki}(s,s') + \sum_{k}\sum_{c\in\omega} U^c_{(k,i)}(s)] = \pi_i(s),$$
$$\forall i \in V, s\in\xi, (j,i)\in E, (i,k)\in E, (k,i)\in E \quad (5)$$

where $U^c_{(k,i)}(s) \geq 0$, and

$$\pi_i(s) = \begin{cases} \lambda_s & \text{if } i = S_s, \\ -\lambda_s & \text{if } i = T_s, \\ 0 & \text{otherwise.} \end{cases}$$

For intermediate forwarders, the incoming/outgoing flows may either be unicast flows, or broadcast flows originating from encoding nodes. Note that the above constraint allows for multipath routing. For single-path routing, the flow of a session on a specific link is either 0 or equal to the throughput of the session, *i.e.*, $\forall s \in \xi, (k,i)\in E$,

$$\frac{\sum_{s'\in\xi}\sum_j\sum_{c\in\omega} b^c_{jki}(s,s') + \sum_k\sum_c U^c_{(k,i)}(s)}{\lambda_s} \in \{0,1\} \quad (6)$$

The scheduling constraint. The scheduling constraint models the interference among competing nodes belonging to the same channel. For single-channel unicast MAC protocols, it is known that a sufficient condition for scheduling is [2], [20]:

$$f_{ij} + \sum\nolimits_{(k,l)} f_{kl} \leq C$$

where f_{ij} denotes the flow rate on link (i,j), and (k,l) is an arbitrary link that interferes with (i,j). C is the channel capacity that is shared by (i,j) and (k,l). For the case with network coding, both the unicast and broadcast flows have to be taken into account. Specifically, we adopt a receiver based interference model:

$$0.5\sum_{s\in\xi}\sum_{s'\in\xi}\sum_{(j,k)} b^c_{jik}(s,s') + 0.5\sum_{m\in I(i)}\sum_{s\in\xi}\sum_{s'\in\xi}\sum_{j,k} b^c_{jmk}(s,s')$$
$$+\sum_{s\in\xi}\sum_{k} U^c_{(i,k)}(s) + \sum_{m\in I(i)}\sum_{s\in\xi}\sum_{k} U^c_{(m,k)}(s) \leq C_c, \forall c\in\omega,$$
$$i\in V, (j,i)\in E, (j,m)\in E, (i,k)\in E, (m,k)\in E, \quad (7)$$

Note that due to the symmetric constraint (4), the rate $b^c_{jik}(s,s')$ has to be halved to avoid repeated count. It is shown that such a sufficient condition may result in a constant approximation to a feasible network flow problem [20]. In any case, this does not compromise our results, as we evaluate the performance gain of the RAC problem over conventional schemes using the same ideal scheduling model.

The channel/interface constraint. Since the number of interfaces on each node are limited, the number of different channels used by a node must be less than or equal to the number of available interfaces, *i.e.*,

$$\sum\nolimits_{c\in\omega} N^c_i \leq R_i, N^c_i \in \{0,1\}, \forall i \in V \quad (8)$$

In addition, we have to relate the number of different channels with the properties of the flows associated with each node. Observe that the total amount of incoming and outgoing traffic of node i and channel c is:

$$B^c_i = 0.5\sum_{s\in\xi}\sum_{s'\in\xi}\sum_{j,k} b^c_{jik}(s,s') + 0.5\sum_{s\in\xi}\sum_{s'\in\xi}\sum_{(k,j)} b^c_{ikj}(s,s')$$
$$+\sum_{s\in\xi}\sum_{k} U^c_{(i,k)}(s) \quad (9)$$

This amount of traffic must not exceed the capacity of channel c. Moreover, if it is non-zero, then node i must be using channel c. Therefore,

$$\frac{B^c_i}{C_c} \leq N^c_i, N^c_i \in \{0,1\}, \forall i\in V, c\in\omega \quad (10)$$

Putting everything together. Consequently, the problem of joint routing, channel assignment and network coding becomes a mixed-integer linear program, with the objective of (1), subject to constraint (2), (3), (4), (5), (7), (8), (10), plus the non-negative constraint for all variables. This is a mixed-integer program and is NP-hard in general [15]. Thus only small scale problems like the scenario in Fig. 1 can be solved using legacy optimization package. To evaluate the coding aware channel assignment for large networks, we propose the use of *simulated annealing* [11] to approximate the optimal solution.

C. Simulated Annealing Based Solution

Simulated annealing has proven be able to approximate the global optimum of a function that is constrained within a search space with a large number of dimensions. The basic idea is to iteratively search for a better solution, while occasionally accepting less favorable solutions to avoid trapping into a local optimum. In what follows, we provide more details on its application to the RAC solution.

1) Iterative update: In simulated annealing, each feasible variable set is called a *configuration*. The algorithm first creates an initial configuration, and then iteratively updates the solution by producing *neighboring configurations* based on the *current configuration*. To solve an combinatorial optimization problem using simulated annealing, the key step is to design appropriate initialization and update algorithms. In our problem, the initial condition is generated using **Algorithm 1**, which minimizes the diversity of channels in order to maximize network connectivity. Note that when searching for coding opportunities, each node has to inspect all sessions that are routed through it, identify the pairs that form the information exchange paradigm, and choose the one that maximizes the broadcast rate. Following the initial configuration, neighboring configurations are iteratively generated subject to all the constraints in the RAC problem. Each neighboring configuration should have the potential (but not always) to achieve a better objective than the current one. To achieve this, we use the heuristic in **Algorithm 2**. When generating a new configuration, the algorithm favors those

Algorithm 1 Initialization for the simulated annealing based optimization

for all $i \in V$ **do**
  for all interface ϵ on node i **do**
    Assign ϵ-th channel to interface ϵ
  end for
end for
for all session $s \in \xi$ **do**
  Find all paths from S_s to T_s with the hop-count metric
end for
for all $i \in V$ **do**
  Identify coding opportunities by inspecting the previous and next-hop of existing flows
end for
repeat
  Augment Δ unit of flow on each path
until All channels along all paths are satiated

Algorithm 2 Generate a neighboring configuration from the current one

Sort all edges according to their utility ratio
for all (i, j) from highest utility to lowest utility **do**
  Assign a channel to (i, j) using the randomized channel assignment scheme in **Algorithm 3**
end for
for all session $s \in \xi$ **do**
  Find all paths from S_s to T_s, with the hop-count metric.
end for
for all $i \in V$ **do**
  Search for coding opportunities.
end for
repeat
  for all session $s \in \xi$ **do**
    Augment Δ unit of flow on each path
  end for
until the channels along all paths are satiated

Algorithm 3 Randomized channel assignment

for all $i \in V$ **do**
  if $\alpha(i) \geq 1$ and $\alpha(k) \geq 1$ **then**
    randomly select a channel c for (i, k), $c \in \omega, c \notin i, c \notin k$
  else if $\alpha(i) < 1$ and $\alpha(k) < 1$ **then**
    if i and k already have shared channels **then**
      randomly select a shared channel for (i, k)
    else
      remove (i, k) from $G(V, E)$
    end if
  else if $\alpha(i) < 1$ and $\alpha(k) \geq 1$ **then**
    randomly select $c \in i$ and assign it to (i, k)
  else
    randomly select $c \in k$ and assign it to (i, k)
  end if
end for

edges with high *utility ratio* in current configuration. The utility ratio implies how important an edge is with respect to the overall network performance. Edges that are heavily used, and are subject to intensive interferences should be put to a higher priority, and be assigned diverse channels. The remaining edges have higher probability of using existing assigned channels on the interfaces of their head/tails. Specifically, we define the utility ratio of edge (i, j) as:

$$\mathcal{U}_{ij} = \chi_{ij} \cdot \eta_{ij} + \mathcal{U}_{ij}(0) \tag{11}$$

where η_{ij} denotes the number of sessions that are routed through edge (i, j) in current configuration. χ_{ij} is the interference intensity of the edge in current configuration, defined as the maximum normalized traffic through all nodes that may interfere with the transmission on (i, j), *i.e.*,

$$\chi_{ij} = \max_{k \in I(j)} \frac{M_k^c}{C_c}, c \in (i, j) \tag{12}$$

$\mathcal{U}_{ij}(0)$ denotes the utility ratio of edge (i, j) after the initial configuration. Before the initial configuration, all utility ratios are set to 0. Note that the definition of utility ratio essentially takes coding into account, since the benefit of XOR coding lies in reducing the interferences among neighboring edges with shared channels.

The annealing process. Given the initialization algorithm and the neighbor generation algorithm, the simulated annealing algorithm attempts to search for the optimal solution in an iterative manner. In each iteration, the algorithm generates a new configuration and determines whether it should be *accepted*. If it achieves a better objective value, then the new configuration is accepted and is set as *current*. Otherwise the new configuration replaces the *current* one with probability p. To ensure asymptotical approximation to the optimal value, the *acceptance probability* p is chosen as [11]:

$$p = e^{\frac{\lambda(\mathcal{C}_i) - \lambda(\mathcal{C})}{\tau}} \tag{13}$$

where $\lambda(x)$ denotes the aggregate network throughput under configuration x; τ is the *temperature*, a control parameter that regulates the diminishing of the acceptance probability for non-favorable configurations. τ is initialized to τ_0 (a value larger than the expected λ) and multiply-decreased in each iteration, so that the acceptance probability approximates zero as the annealing algorithm approximates the optimal solution. In summary, the annealing algorithm for the RAC problem is described in **Algorithm 4**. Note the algorithm involves another two control parameters, θ and T_m. θ is the multiply-decrease factor used to update the temperature τ. The empirical value of θ is between 0.85 and 1. T_m is the minimum temperature that dictates the termination of the algorithm.

D. Performance Evaluation

In this section, we compare the performance of the above joint routing, channel assignment and coding scheme with the non-coding schemes. Specifically, we compare with the centralized optimization algorithm in [2] (referred to as the *ABL algorithm*), which attempts to achieve optimal throughput

Algorithm 4 The simulated anneal algorithm for the RAC problem

Generate the initial configuration $\mathcal{I}$ using **Algorithm 1**. $\mathcal{C} = \mathcal{I}$
repeat
  $i = i + 1$.
  Generate a new configuration $\mathcal{C}_i$.
  if $(\lambda(\mathcal{C}_i) > \lambda(\mathcal{C}))$ **then**
    $\mathcal{C} = \mathcal{C}_i$
  else if $p(\mathcal{C}_i) > \text{random}[0, 1)$ **then**
    $\mathcal{C} = \mathcal{C}_i$
  end if
  $\tau = \theta \cdot \tau$
until $\tau < T_m$

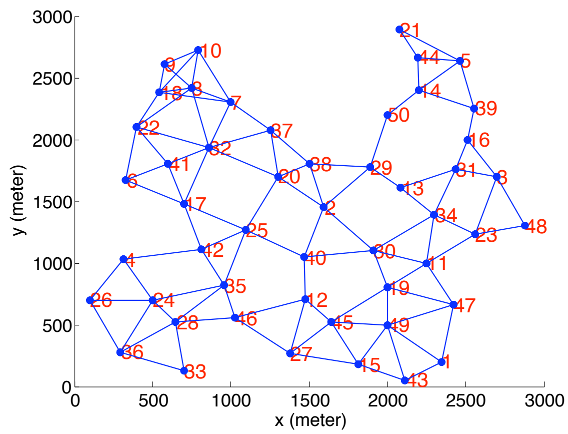

Fig. 4. The mesh network topology used for evaluation. Two nodes are connected with a line if they falls in the transmission range of each other, no matter if they are sharing the same channel.

by jointly optimize routing and channel assignment. The ABL algorithm approximates the optimal throughput using a relaxation of the original joint optimization problem. Here we only compare with the original unrelaxed problem, which imposes an upper bound on ABL. We use simulated annealing to solve the unrelaxed mixed-integer problem, in a similar manner to RAC, except that network coding is never applied.

We adopt a random mesh topology consisting of 50 nodes (Fig. 4) for evaluation. Each node is equipped with a random number of interfaces, ranging from 1 to 3. The performance metric is the aggregate network throughput (normalized by capacity), as a function of the number of concurrent sessions with randomly selected source-destination pairs. For simplicity, we assume the capacity of all channels are the same.

First of all, we demonstrate the convergence of the simulated anneal algorithm in a typical scenario, with 3 orthogonal channels and 20 unicast sessions (Fig. 5). Initially, the algorithm keeps oscillating, attempting to walk out of local optimums. As the temperature is reduced, it gradually approximates a stable solution. The convergence speed depends on the control parameters. Our empirical settings for the parameters are: $\theta = 0.87, T_m = 10^{-3}$; τ_0 is set to ten times of the total capacity of all channels.

We then vary the number of concurrent sessions, and run the

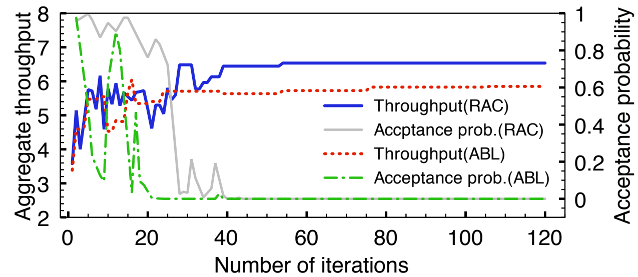

Fig. 5. The convergence of the simulated annealing algorithm. Capacity of each channel $= 10^4, \Delta = 10$. The y axis denotes the aggregate throughput normalized by capacity.

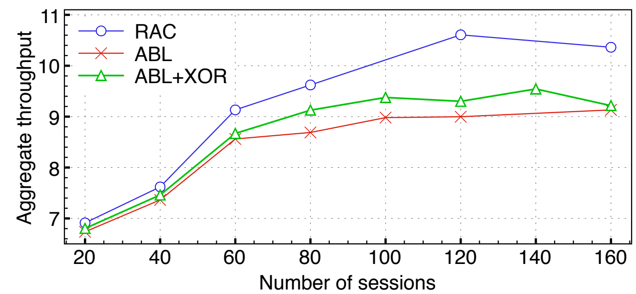

Fig. 6. The aggregate network throughput achieved by RAC and the ABL optimization.

simulated annealing algorithm for RAC and ABL, respectively. We plot the results in Fig. 6. Since the advantage of network coding comes with no additional cost, the RAC consistently performs better than ABL. The aggregate throughput of RAC can be 21% higher than ABL, with an average of 11%. Beside ABL, we also illustrate a heuristic that simply searches for coding opportunities in the optimal ABL flows (denoted ABL+XOR). Without the joint optimization of coding and channel assignment, this scheme results in lower aggregate throughput than RAC, although it indeed improves upon the non-coding scheme.

The total number of available channels also has a large impact on the aggregate network throughput, for both RAC and ABL. In Fig. 7, we fix the traffic demand to 80 sessions and vary the number of channels. We observe that the throughput has a steep increase in the beginning. However, it does not improve further with the addition of more channels. This is because adding excessive channels reduces the number of paths for each session, and may even result in network partition.

IV. Coding Directed Routing

So far, we have analyzed the optimal performance of the joint routing, channel assignment, and network coding scheme, computed by centralized optimization algorithms. In this section, we design and evaluate a practical algorithm that integrate network coding into existing routing and channel assignment schemes for MC-WMNs, which is called *Coding Directed Routing* (CDR).

The pioneering work, COPE [10], has implemented XOR coding, but only based on prescribed routes. The theoretical study in [18] has established the potential advantage of jointly optimizing routing and coding. However, the optimization

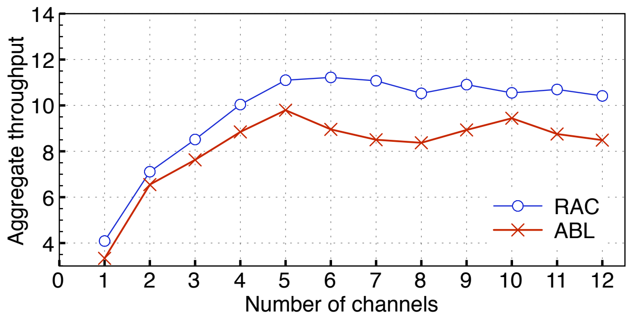

Fig. 7. Influence of the number of channels on network capacity.

problem therein is NP-hard in general, and only small-scale instances can be centrally solved.

Algorithm 5 The Coding Directed Routing (CDR) algorithm

Step1: Run the route discovery procedure.
Step2: Re-routing.
current = S_s
loop
  Add *current* to *Path*
  if *current* == T_s **then**
    return
  end if
  if *current* has T_s in its potential next-hop **then**
    Add T_s to $Path$
    return
  end if
  for all i in $D($*current*$)$ **do**
    for all j in $D(i)$ **do**
      if (*current*, i, j) has a coding opportunity with an existing session **then**
        Add (*current*, i, j) as a new coding opportunity
      end if
    end for
  end for
  Search for the coding opportunity (*current*, i, j) with the smallest dist(j) (distance to the destination).
  if dist(j) − dist(*shortest-next-hop*) < 2 **then**
    next-hop = i, *next-two-hop* = j, *current* = j
    Add *next-hop* to *Path*
  else
    next-hop = *shortest-next-hop*, *current* = *next-hop*
  end if
end loop

As a contrast, CDR is a distributed algorithm that jointly considers routing and network coding. Instead of finding the next-hop as in conventional single-path routing schemes, CDR selects the *next two hops*, taking into account the possible coding opportunities that are created owning to the selection. Specifically, CDR involves two steps. The first step involves a route discovery phase similar to traditional routing algorithms (*e.g.*, the link-quality aware ETX routing scheme [21]), where each node finds its distance to the destination. After that, the source node transmits a probing packet to all its neighbors. A

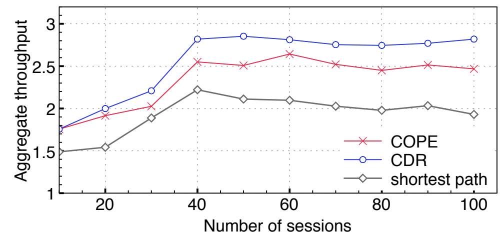

Fig. 8. The performance of CDR, compared with COPE and shortest path routing.

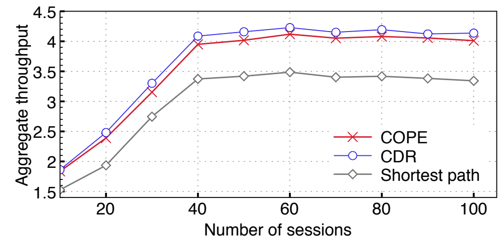

Fig. 9. The performance of CDR with 3 channels.

neighbor continues forwarding the probe if it is closer to the destination than its predecessor. Such neighbors are called the *potential next-hops* of their predecessor, denoted as $D(i)$ for predecessor i. The second step is a re-routing process, which is the key of CDR. In particular, the source node searches for a coding aware path by inspecting the coding opportunities in potential next-hops and the corresponding next-two-hops. The triple (current, next-hop, next-two-hop) with a coding opportunity on next-hop, and with the shortest distance to the destination will be eventually chosen. To avoid *deviating* too long from the shortest path, a next-two-hop is chosen only if its distance is no more than two hops compared with the next-two-hop node on the shortest path (called *shortest-next-two-hop*). We formally describe CDR in **Algorithm 5**.

Note that during the re-routing phase, the routes of existing sessions remain the same, thus the algorithm can be integrated with existing distributed and real-time route-discovery schemes. In addition, although CDR adopts hop-count distance as the path metric, it can naturally extends to other metrics such as link-quality based metric [21]. In these cases, the threshold for determining the deviation (currently set to 2 hops as in **Algorithm 5**) should be estimated as two times of the average link-distance along the shortest path. Also note that the above description is based on the single-channel case, but can be easily extended to a multi-channel network with prescribed channel assignment.

To evaluate the performance of CDR, we compare it with the COPE scheme, as well as the traditional shortest-path routing scheme. We simulated CDR and COPE under the network models described in Sec. III, *i.e.*, the routing, MAC, channel/interface, and coding models. Fig. 8 plots the aggregate network throughput of the above schemes, as a function of the number of concurrent sessions under the single-channel constraint. Although CDR is essentially a greedy algorithm that tries to locally maximize the coding opportunities, it achieves up to 12% throughput gain over COPE, and up to 40% gain over traditional routing. The best performance is seen when a large number of sessions are running concurrently. This is because CDR promises to reduce interferences, especially when the network is highly congested. For the multi-channel case, we compare CDR with COPE under an arbitrary channel assignment scheme. Specifically, we assume nodes in the network are equipped with 1 to 3 interfaces, and 3 channels are used concurrently. We use the randomized channel assignment scheme in **Algorithm 3** and evaluate the throughput when different number of concurrent sessions are running. For each set of sessions, the algorithm is repeated for 50 times. The averaged results are shown in Fig. 9. Our observation is that with diversified channels, less coding opportunities are seen, and the maximal performance gain over traditional routing is reduced to 21%. For the same reason, the gain over COPE becomes marginal. We remark that the performance of CDR for the multi-channel case can be improved by incorporating the channel-aware routing metric (see, *e.g.*, [22]), instead of the hop-count metric. Development of the channel-aware metric is complementary to our work and is omitted due to space constraint.

V. Extension to Other Multi-channel Networks

Beside the legacy 802.11 based multi-channel mesh infrastructure, network coding can be incorporated into other state-of-the-art multi-channel wireless networks. In this section, we investigate the scenarios where network coding can be applied to boost the capacity of 802.16 and 802.11n wireless networks, and analyze the potential performance gains over existing schemes.

A. Coding Aware Channel Assignment in 802.16 OFDMA networks

The IEEE 802.16 standard [12] proposes infrastructure support to mobile wireless devices for last-mile Internet access. It is built atop the OFDMA (Orthogonal Frequency Division Multiple Access) physical layer. With OFDMA, the prescribed frequency band is divided into multiple orthogonal sub-channels (up to 2048 in the 802.16 standard). Each mobile client is allocated a number of sub-channels by the base station. The maximum link layer throughput of a client is proportional to the number of sub-channels allocated to it, as well as the PHY capacity of each sub-channel.

To understand the benefits of network coding in 802.16 OFDMA systems, consider the scenario in Fig. 10, where two mobile clients are exchanging information with each other. Traditional channel assignment schemes attempt to assign orthogonal set of sub-channels to the two uplinks and two down-links [12]. When network coding is taken into account, however, the uplink packets can be XORed and broadcast to both clients, which decodes the information in a similar manner to the information exchange paradigm in Fig. 1. Consequently, the two downlinks can be assigned the same set of sub-channels, saving one set of sub-channels to serve for other clients.

The channel-saving advantage of network coding is even more promising for large-scale scenarios where several hun-

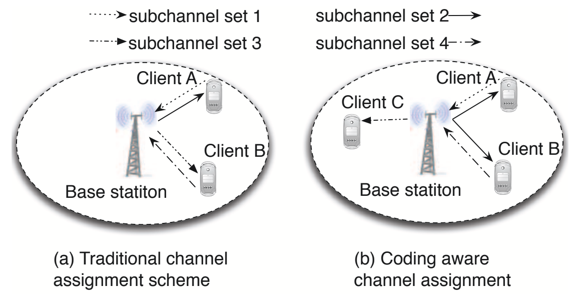

Fig. 10. The application of network coding to 802.16 OFDMA systems.

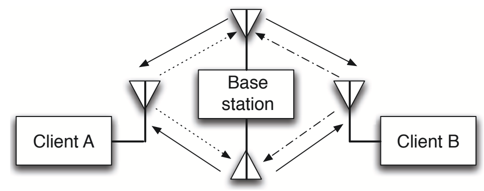

Fig. 11. The application of network coding to 802.11n MIMO systems.

dreds clients are subscribed to the base station. In an ideal case where every client is interested in exchanging information with another one via the base station, it is straightforward that to achieve the same capacity, the current 802.16 standard requires $\frac{4}{3}$ times more channels than coding aware channel assignment. With the same number of sub-channels, the coding aware channel assignment is able to achieve $\frac{4}{3}$ higher data rate than the current standard. In other words, the performance gain of network coding can be up to $\frac{4}{3}$ when compared with the traditional scheme.

B. Application to 802.11n MIMO networks

IEEE 802.11n [13] is a recently developed standard that extends the well established 802.11a/b/g WLAN specification by adding MIMO (multiple-input multiple-output). Using multiple antennas, MIMO can achieve two separate objectives via different physical layer coding schemes. The first is *interference suppression*. With multiple antennas, the receiver is able to isolate and decode multiple packet streams from different sources, as long as the total number of transmitting antennas is less than or equal to the total number of receiver antennas [23]. With this capability, MIMO can essentially be abstracted as a multi-channel system, although in reality all the antennas are using the same frequency band. The second objective is *SINR improvement*, which increases link capacity by combining multiple copies of the same signal from multiple antennas. Existing protocols achieve either objectives under different scenarios [23], [24]. In this section, we show that network coding can naturally combine both objectives and further improve the network capacity.

We consider a practical scenario where the base station in the WLAN is equipped with multiple antennas, while the mobile clients are portable wireless devices with a single antenna. And again, we consider the case when two clients cannot communicate directly, but intend to exchange information with each other through the base station, as shown in Fig. 11. With the interference suppression advantage, client A and client B

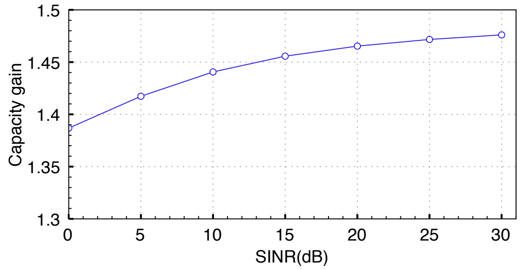

Fig. 12. The performance gain of network coding in 802.11n MIMO system.

are able to transmit concurrently, without any collisions. Since both clients are single-antenna receivers, the base station can only forward packets to one of them each time. However, using two antennas concurrently (the SINR improvement advantage), the base station can provide a downlink rate that is much higher than the uplink rate. Without loss of generality, we assume the three nodes are using the same transmission power, and the uplink SINR is ρ. Then the average downlink capacity is given by [23]: $C_d \approx \log_2(1+k\cdot\rho)$, where k is the number of antennas at the base station (2 in our case). The uplink capacity is given by: $C_u \approx \log_2(1+\rho)$. To exchange one unit of information, the required transmission time is:

$$\frac{1}{\frac{1}{C_u}+\frac{2}{C_d}} \tag{14}$$

In contrast, when using network coding, the packets from client A and client B are coded and broadcast, thus the required transmission time is:

$$\frac{1}{\frac{1}{C_u}+\frac{1}{C_d}} \tag{15}$$

The capacity improvement is:

$$\frac{\frac{1}{C_u}+\frac{2}{C_d}}{\frac{1}{C_u}+\frac{1}{C_d}} = 1+\frac{1}{\frac{\log_2(1+k\cdot\rho)}{\log_2(1+\rho)}+1} \tag{16}$$

It is straightforward to see that as the SINR approximate infinity, the performance gain of network coding is bounded by 1.5. For realistic SINR values in 802.11n systems and 2 antennas on the base station, the gain ranges from 1.3 to 1.48 (Fig. 12), which is still notable.

VI. Conclusion

In this paper, we analyzed the performance gains of network coding in multi-channel wireless networks. For the case of traditional 802.11 mesh networks with multiple interfaces, we derived the potential throughput gain when routing, channel assignment and network coding are jointly optimized. We then design a decentralized algorithm that naturally combines network coding with traditional routing, under arbitrary channel assignment schemes. In addition, we identified the potential scenarios where network coding can boost the capacity of emerging multi-channel networks, including 802.16 OFDMA and 802.11n MIMO based systems, and analyzed the performance bounds of network coding in such systems. As future work, one interesting problem would be to investigate more complex coding schemes, such as the randomized network coding over GF(2^8), and XOR coding with opportunistic listening. We are also planning to investigate the benefits of network coding in multi-hop OFDMA and MIMO networks.

References

[1] P. Bahl, A. Adya, J. Padhye, and A. Walman, "Reconsidering Wireless Systems With Multiple Radios," *SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 5, 2004.

[2] M. Alicherry, R. Bhatia, and L. Li, "Joint Channel Assignment and Routing for Throughput Optimization in Multi-radio Wireless Mesh Networks," in *Proc. of ACM MobiCom*, 2005.

[3] H. Wi, F. Yang, K. Tan, J. Chen, Q. Zhang, and Z. Zhang, "Distributed Channel Assignment and Routing in Multiradio Multichannel Multihop Wireless Networks," in *IEEE JSAC*, vol. 24, no. 11, Nov 2006.

[4] P. Kyasanur and N. Vaidya, "Routing and Interface Assignment in Multi-channel Multi-interface Wireless Networks," in *Proc. of IEEE WCNC*, 2005.

[5] J. Tang, G. Xue, and W. Zhang, "Interference-Aware Topology Control and QoS Routing in Multi-channel Wireless Mesh Networks," in *Proc. of ACM MobiHoc*, 2005.

[6] A. H. M. Rad and V. W. Wong, "Joint Channel Allocation, Interface Assignment and MAC Design for Multi-Channel Wireless Mesh Networks," in *Proc. of IEEE INFOCOM*, 2007.

[7] X. Lin and S. Rasool, "A Distributed Joint Channel-Assignment, Scheduling and Routing Algorithm for Multi-Channel Ad Hoc Wireless Networks," in *Proc. of IEEE INFOCOM*, 2007.

[8] A. H. M. Rad and V. W. Wong, "Joint Optimal Channel Assignment and Congestion Control in Multi-Channel Wireless Mesh Networks," in *Proc. of IEEE ICC*, 2006.

[9] Y. Wu, P. A. Chou, and S.-Y. Kung, "Information Exchange in Wireless Networks with Network Coding and Physical-Layer Broadcast," in *Proc. of CISS*, 2005.

[10] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in The Air: Practical Wireless Network Coding," in *Proc. of ACM SIGCOMM*, July 2006.

[11] P. J. M. Laarhoven and E. H. L. Aarts, Eds., *Simulated Annealing: Theory and Applications*. Kluwer Academic Publishers, 1987.

[12] C. Eklund, R. B. Marks, K. L. Stanwood, and S. Wang, "IEEE Standard 802.16: A Technical Overview of the WirelessMAN^{TM}. Air Interface for Broadband Wireless Access," *IEEE Communication Magzine*, vol. 40, no. 6, 2002.

[13] Y. Xiao, "IEEE 802.11n: Enhancements for Higher Throughput in Wireless LANs," *IEEE Wireless Communications*, vol. 12, no. 6, 2005.

[14] X. Meng, K. Tan, and Q. Zhang, "Joint Routing and Channel Assignment in Multi-Radio Wireless Mesh Networks," in *Proc. of IEEE ICC*, 2006.

[15] M. Shin, S. Lee, and Y. Kim, "Distributed Channel Assignment for Multi-radio Wireless Networks," in *Proc. of IEEE MASS*, 2006.

[16] A. K. Das, H. M. K. Alazemi, R. Vijayakumar, and S. Roy, "Optimization Models for Fixed Channel Assignment in Wireless Mesh Networks with Multiple Radios," in *Proc. of IEEE SECON*, 2005.

[17] K. N. Ramachandran, E. M. Belding, K. C. Almeroth, and M. M. Buddhikot, "Interference-Aware Channel Assignment in Multi-Radio Wireless Mesh Networks," in *Proc. of IEEE INFOCOM*, 2006.

[18] S. Sengupta, S. Rayanchu, and S. Banerjee, "An Analysis of Wireless Network Coding for Unicast Sessions: The Case for Coding-Aware Routing," in *Proc. of IEEE INFOCOM*, 2007.

[19] A. Raniwala and T. Chiueh, "Architecture and Algorithms for an IEEE 802.11-Based Multi-Channel Wireless Mesh Network," in *Proc. of IEEE INFOCOM*, 2006.

[20] V. S. A. Kumar, M. V. Marathe, S. Parthasarathy, and A. Srinivasan, "Algorithmic Aspects of Capacity in Wireless Networks," *SIGMETRICS Perform. Eval. Rev.*, vol. 33, no. 1, 2005.

[21] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris, "A High-throughput Path Metric for Multi-hop Wireless Routing," in *Proc. of ACM MobiCom*, 2003.

[22] R. Draves, J. Padhye, and B. Zill, "Routing in Multi-radio, Multi-hop Wireless Mesh Networks," in *Proc. of ACM MobiCom*, 2004.

[23] K. Sundaresan, R. Sivakumar, M. A. Ingram, and T.-Y. Chang, "Medium Access Control in Ad Hoc Networks with MIMO Links: Optimization Considerations and Algorithms," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, 2004.

[24] K. Sundaresan and R. Sivakumar, "Routing in Ad-hoc Networks with MIMO Links," in *Proc. of IEEE ICNP*, 2005.