

Collaborative routing, scheduling and frequency assignment for wireless Ad Hoc networks using spectrum-agile radios

Xin Wang · J. J. Garcia-Luna-Aceves

Published online: 10 August 2010

© The Author(s) 2010. This article is published with open access at Springerlink.com

Abstract We present the CROWN (Collaborative ROuting, scheduling and frequency assignment for Wireless ad hoc Networks) scheme. CROWN is a cross-layer optimization approach for spectrum-agile nodes to adjust their spectrum allocation and transmission scheduling according to the underlying traffic demands. Instead of choosing the optimal route based on predetermined transmission scheduling and frequency assignment results, CROWN incorporates the efficiency of the underlying frequency assignment and scheduling information into the routing metric calculation, so that the route with the maximal joint spatial and frequency reuse is selected. Simulation results show that CROWN efficiently exploits the frequency diversity and spatial reuse features of spectrum-agile radios.

Keywords MAC · Routing · Ad-hoc networks · Spectrum-agile radios · Cross-layer design

A short version of this paper has been presented at the 17th IEEE International Conference on Computer Communication Networks (ICCCN 2008), St. Thomas US Virgin Islands, August 3–7, 2008.

X. Wang (✉) · J. J. Garcia-Luna-Aceves
Computer Engineering Department, University of California,
Santa Cruz, Santa Cruz, CA 95064, USA
e-mail: wangxin@soe.ucsc.edu

J. J. Garcia-Luna-Aceves
e-mail: jj@soe.ucsc.edu

J. J. Garcia-Luna-Aceves
Palo Alto Research Center (PARC), 3333 Coyote Hill Road,
Palo Alto, CA 94304, USA

1 Introduction

In traditional wireless ad hoc networks, each radio operates on a predetermined wireless spectrum. The fixed spectrum allocation rule makes the available spectrum cannot be extensively exploited regarding the number of contending users and the amount of traffic demands. Compared with the fixed allocation schemes, the dynamic spectrum allocation largely reduces spectrum wastage and co-channel interferences. In this paper, we propose a distributed approach to increase the spatial and frequency reuse of wireless ad hoc networks using spectrum-agile radios.

Frequency assignment, routing and scheduling are dependent on each other in wireless networks, and the input of any component is partially decided by the outputs of the other two component, e.g. (1) transmission scheduling: frequency assignment decides whether two links are interfere with each other, and route selection decides which links will be used for transmissions; (2) frequency assignment: different transmission scheduling will generate different interference link set, it further decides which links need to be assigned with non-overlapping spectrums; good route selection balance the traffic throughout the network so that the frequency diversity can be utilized to the largest extent; (3) route selection: frequency assignment decides the network topology, which influences the route selection results directly; since routing control packets are transmitted as the data packets at the MAC layer, the transmission scheduling decides how the routing information is propagated throughout the network.

However, the correlations between these three components are at different timescales. We note that frequency assignment and scheduling are formed based on the two-hop information, while route selection are made based on the end-to-end information between the traffic source and

destination. So frequency assignment and scheduling are coupled with each other more tightly at small timescales (a few packet transmissions), while route selection interacts with the other two components at large timescales (hundreds of packet transmissions).

Based on the above discussions, to fully leverage spatial and frequency diversities in ad hoc networks, frequency assignment, routing and scheduling must be solved as a joint, and this poses several challenges. Among the questions that must be answered we have: How should the available spectrum be allocated according to the traffic demands? What makes a transmission scheduling efficient in terms of both frequency diversity and spatial reuse? How should the MAC and network layers interact to exploit frequency diversity and spatial reuse at both layers?

In this paper, we propose the Collaborative **RO**uting scheduling and frequency assignment scheme for **Wireless ad hoc Networks (CROWN)** to address the above problems. We introduce a heuristic approach to dynamically adjust spectrum allocation according to the traffic demands and achieve fairness across different links. We propose a unified metric, which we call *transmission fraction*, to evaluate the efficiency of the joint spectrum allocation and link scheduling in terms of spatial and frequency reuse. We incorporate the efficiency of the underlying frequency assignment and scheduling information into the routing metric calculation so that the route with the maximal joint spatial and frequency reuse is selected. The rest of the paper is organized as follows. Section 2 describes the related work. Section 3 introduces the details of CROWN (Collaborative **RO**uting, scheduling and frequency assignment for **Wireless ad hoc Networks**). Section 4 numerically analyzes the approximate MAC layer throughput and the complexity of routing protocol. Section 5 evaluates the performance of CROWN through simulations. Section 6 concludes the paper.

2 Related work

Traditionally the channel assignment, routing and MAC layer scheduling has been treated independently in wireless networks. In this section, we briefly reviewed the existing MAC and routing protocols, as well as joint routing and scheduling works for wireless networks. More detailed review can be found in [1–3].

2.1 MAC protocols in wireless networks

Wireless MAC protocols can be classified into contention-based MAC, contention-free MAC and hybrid MAC. In contention-based MAC protocols, each node either detects the transmission collision (collision-detection) or tries to

avoid the transmission collision through random back-offs (collision-avoidance). Based on its understanding of the channel status, each node contends for the channel access in a distributed fashion. Contention-based MAC can be classified into four categories:

1. No coordination: nodes transmit at will when they have data to send (e.g. ALOHA).
2. Career sensing: nodes listen to the channel before they transmit a data packet (e.g. CSMA).
3. Career sensing and collision detection: nodes listen before and during transmission, and stop if a collision is heard when transmitting (e.g. CSMA/CD).
4. Collision avoidance: A handshake is typically used to determine the node that can send a data packet (e.g. IEEE 802.11, CSMA/CA).

Contention-based MAC protocols may experience throughput degradation at high traffic load and due to their best effort nature, they can not provide Quality-of-Service (Qos) support to real-time applications.

For contention-free MAC protocols, a set of timetables for individual nodes or links is prearranged. Each node/links can only transmit in their assigned time/frequency slots, so that the transmissions from these nodes/links are collision-free within the effective range of the transmissions. Dynamic transmission scheduling protocols can exploit spatial reuse of the wireless channel and have higher channel utilization than static scheduling approaches, e.g. TDMA. Based on whether the schedule scheme needs the topology information, the scheduling-based channel access can be further divided into topology dependent/independent scheduling. In topology dependent scheduling, global topology information is required to form the correct channel scheduling. Arikian [4] has shown that the problem of establishing an optimal interference-free schedule where the optimal is considered in term of throughput, is NP complete. Chlamtac [5] first proposed a topology-transparent scheduling algorithm for wireless ad hoc networks. It uses polynomials over a Galois field to assign time slots, which guarantees each node can transmit successfully at least once in a frame. This approach can provide a minimum performance guarantee for each node. It just needs the information of overall number of nodes in the network and the number of neighbors of each node. The frame length is also much smaller than the classic TDMA approach. Konstantinos [6] proposed probabilistic policy to increase the system throughput under various traffic loads. Ju [7] proposed an approach based on code theory to optimize the performance of Chlamtac's algorithm in terms of minimum throughput and maximum delay. However, Carlos [8] has shown that the throughput of topology-transparent scheduling is at most the same with the slotted ALOHA.

Hybrid MAC is proposed to take the advantages of contention-based channel access and topology-dependent scheduling. Nodes first use the contention-based channel access to exchange the neighbor information which is needed to form the correct channel scheduling or reserve the time slots in the scheduling-based transmission period. The examples of hybrid channel access MAC protocol are NAMA [9] and CATA [10].

2.2 Routing protocols in wireless networks

A considerable body of literature has addressed research on routing and architecture of wireless networks. Reviews and performance comparisons of wireless routing protocols have been presented in many earlier publications [11–15].

According to routing strategy, wireless routing protocols can be classified into proactive (or, table-driven) and reactive (or, on-demand) routing protocols. Proactive routing protocols share a common feature, i.e., background routing information exchange regardless of communication requests. The protocols have many desirable properties especially for applications including real time communications and QoS guarantees, such as low latency route access and alternate QoS path support and monitoring. Many proactive routing protocols have been proposed for efficiency and scalability. The Fisheye State Routing (FSR) described in [16, 17] is a simple, efficient link state type routing protocol which maintains a topology map at each node and propagates link state updates. The main differences between FSR and conventional link-state routing protocols are the ways in which routing information is disseminated. First, FSR exchanges the entire link state information only with neighbors instead of flooding it over the network. The link state table is maintained up-to-date based on the information received from neighbors. Second, the link state exchange is periodical instead of event-triggered, which avoids frequent link state updates caused by link breaks in an environment with unreliable wireless links and mobility. Optimized Link State Routing Protocol (OLSR) [18] is a link state routing protocol. It periodically exchanges topology information with other nodes in the network. The protocol uses Multi-Point Relays (MPRs) to reduce the number of superfluous broadcast packet retransmissions and also to reduce the size of the link-state update packets, leading to efficient flooding of control messages in the network.

On-Demand Routing is a popular routing category for wireless routing protocols. The design follows the idea that each node tries to reduce routing overhead by only sending routing packets when a communication is awaiting. Examples include Ad hoc On demand Distance Vector Routing (AODV) [19], Associativity-Based Routing (ABR) [20], Dynamic Source Routing (DSR) [21], Lightweight

Mobile Routing (LMR) [22] and Temporally-Ordered Routing Algorithms (TORA) [23]. Among the many proposed protocols, AODV and DSR have been extensively evaluated in the literature

On-demand algorithms typically have a route discovery phase. Query packets are flooded into the network by the sources in search of a path. The phase completes when a route is found or all the possible outgoing paths from the source are searched. There are different approaches for discovering routes in on-demand algorithms. In AODV, upon receiving a query, the transit nodes learn the path to the source (called backward learning) and enter the route in the forwarding table. The intended destination eventually receives the query and can thus respond using the path traced by the query.

An alternate scheme for tracing on demand paths is DSR. DSR uses source routing, i.e., a source indicates in a data packets header the sequence of intermediate nodes on the routing path. In DSR, the query packet copies in its header the IDs of the intermediate nodes it has traversed. The destination then retrieves the entire path from the query packet, and uses it (via source routing) to respond to the source, providing the source with the path at the same time. Data packets carry the source route in the packet headers.

Due to page limits, we cannot enumerate all the existing wireless routing protocols. More detailed surveys can be found in [1–3].

2.3 Joint routing and scheduling in wireless networks

With the emergence of multi-channel multi-radio networks, more and more research work on joint routing, scheduling and channel assignment is proposed to utilize the channel diversity. Raniwala et al. [24] propose a centralized channel assignment and routing algorithm to obtain a static frequency assignment. Raniwala et al. [25] propose an improved distributed frequency assignment algorithm. Kyasanur et al. [26] propose an interface-assignment strategy where the number of available interfaces is less than the number of available channels. It fixes a channel on one radio and switches channels on other radios. Nodes can communicate with each other through the fixed common radio without requiring specialized coordination algorithms.

Kodialam et al. [27] consider the problem of jointly routing and scheduling transmissions to achieve a given rate vector. They use a simple interference model, which is derived from the CDMA based multi-hop networks to map the scheduling problem to edge coloring problem. They have proven that their solution is within $\frac{2}{3}$ of the optimal solution. Zhang et al. [28] formalize the problem for joint routing and channel switching in wireless mesh

networks and use column generation method to solve the problem. Alicherry et al. [29] formulate the joint frequency assignment and routing problem for infrastructure wireless mesh networks. They aim to maximize the bandwidth allocated to each traffic aggregation point subjected to fairness constraint and propose a constant approximation algorithm for this NP-hard problem. Kodialam et al. [30] develop a network model that characterizes the channel, radio and interference constraint in a fixed broadband wireless network, which provides necessary and sufficient conditions for a feasible frequency assignment and schedule. Meng et al. [31] formulate the joint routing and channel assignment problem based on radio and radio-to-radio link. They introduce a *scheduling graph* and derive a sufficient condition for the feasibility problem of time fraction.

Tam et al. [32] propose a joint multi-channel and multipath control protocol (JMM). JMM coordinates channel usage among slots using a receiver-based channel assignment and schedules transmissions along dual paths. JMM uses a routing metric which explicitly accounts for the disjointness between paths and interference among links to select two maximally disjoint paths. Wu et al. [33] propose a channel cost metric (CCM) which reflects the interference cost and channel diversities. Based on CCM, a distributed joint frequency assignment and routing protocol is proposed.

There are several proposals addressing the joint routing selection and spectrum assignment issues in cognitive radio networks. Cheng et al. [34] propose a propose a Delay motivated On-demand Routing Protocol (DORP). It proposes a novel routing metric, *cumulative delay*, which takes switching delay, backoff delay and queueing delay into account. An on-demand routing protocol based on AODV cooperates with nodes spectrum assignment module to select the routing path and frequency band with minimum cumulative delay. Frequency assignment information is disseminated through the modified routing control messages.

Ma et al. [35] proposes Dynamic Open Spectrum Sharing (DOSS) MAC protocol in which a source node finds candidate routes through standard route discovery procedures. For each candidate route, DOSS finds all feasible frequency assignment combinations and estimates the end-to-end throughput performance. It selects the route and channel assignment that results in the best throughput, and schedules a conflict free channel usage for this route. The source node then broadcasts the decision to all the nodes on the route. However, all these proposals rely on the existing IEEE 802.11 MAC protocol, and do not exploit the spatial-reuse of the system.

From the above summary of prior work, we can observe that the joint routing and scheduling problem has not been

solved for the case of ad hoc networks in which nodes employ spectrum-agile radios.

3 Distributed joint frequency assignment, routing and scheduling

3.1 Problem formulation and assumptions

We assume that there are K radio interfaces at each node, and that all radios operate on unlicensed bands $[F_s, F_e]$. We focus on how to efficiently exploit the available spectrum resources, and do not address the primary user detection problem on licensed bands. We assume that each node is synchronized on slot boundaries and that nodes access the channel based on slotted time boundaries. Each time slot is numbered relative to a consensus starting point. The length of the time slot (t_s) is the minimal unit of the channel-access schedule over the time axis. A time frame is made up of L time slots ($T_f = Lt_s$). We assume the minimum interval by which the spectrum can be divided along the frequency axis is σ . It is constrained by the spectrum-agile radio RF front-end [36]. The available spectrum is allocated in *frequency block* units. A frequency block $(f_0, \Delta f, t_0, \Delta t)$ is a portion of the spectrum $(f_0, f_0 + \Delta f)$ for the time interval $(t_0, t_0 + \Delta t)$.

Since global optimality of the joint routing, scheduling and frequency assignment requires not only a central controller, but also network-wide topology and traffic information [24], which are impossible for wireless ad-hoc networks. We propose a distributed sub-optimal solution. We divide the original problem into two sub-problems: First, we address the transmission scheduling and frequency assignment problem consisting of utilizing the available spectrum in the two-hop neighborhood of each link (i, j) to the largest extent. Second, given a transmission scheduling and frequency assignment of different links, we address the routing problem consisting of deciding which links to use and how to form the correct sequence/ordering of the transmission scheduling across the network.

3.2 Packet queues, scheduler and radio interface interaction

Figure 1 illustrates the interaction between packet queues, schedulers and radio interfaces. In CROWN, there is a specific packet queue for broadcast packets. For unicast transmissions, given that neighbors of a node may be operating on different portions of the spectrum, we choose to schedule packets to each neighbor individually, unicast packet queues are maintained as per-neighbor FIFO queues. Each neighbor is identified with its unique MAC address.

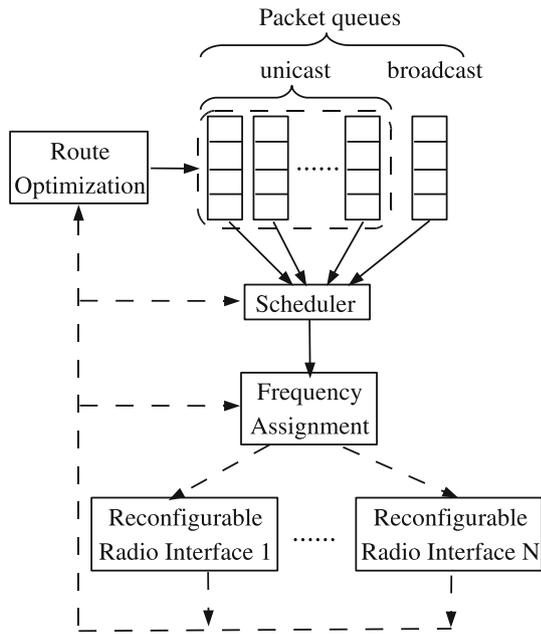


Fig. 1 Packet queues, scheduler and radio interfaces interaction

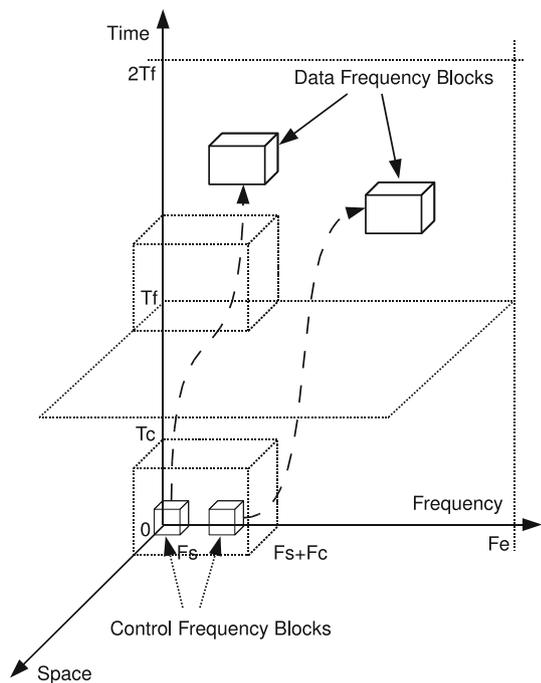


Fig. 2 Control and data frequency block

3.3 Frequency assignment and transmission scheduling strategy

We use a common control frequency block $[F_s, F_s + F_c, 0, T_c]$ in each time frame to exchange the topology information and traffic flow information, as Fig. 2 shows. The lengths of F_c and T_c are chosen that could allow all

traffic flows to broadcast at least once in T_c using the bandwidth F_c .

The reason is that we note there are several problems for MAC protocols without a common control frequency block: (1) It cannot send broadcast transmissions efficiently. The broadcast packets either need to be sent over all channels, or sent during the rendezvous interval on a common channel (e.g. SSCH [37]). The previous approach incurs high overheads, while the latter approach will increase the transmission delay of control packets; (2) When two nodes are not assigned with any common spectrums, even if they may be in the communication range of each other, transmission failures may be mistaken as link breakage, and can adversely affect the performance of higher layer protocols. During time interval $[0, T_c]$ of each time frame, each node switches one of its radio interfaces to the control spectrum $[F_s, F_s + F_c]$. The control frequency block is further divided into mini-frequency blocks of equal size $[0, f_c, 0, t_c]$. Each mini-block is assigned with a unique id MB_{id} . We define the *priority* of node i at frequency block MB_{id} as:

$$prio_i = hash(i \oplus MB_{id}) \tag{1}$$

The node with the highest priority (or the node with the highest id for those with the same priority) is elected to obtain the corresponding frequency block, and it sends a Frequency Allocation Request (FAR) packet. The FAR packet from node i states its: (a) the minimal data rate requirement (r_{ij}) for each link (i, j) that has packets to send, (b) *neighbor information* consisting of its one-hop neighbor list and two-hop neighbor list, and (c) the existing transmission scheduling and frequency assignment result, which is indicated by *frequency allocation table* ($FAT(f, \Delta f, t) = \{(u, v), \dots\}$). It states the spectrum $[f, f + \Delta f]$ is occupied by link set $\{(u, v), \dots\}$ at time slot t .

The size of the mini-frequency block needs to be big enough to send an entire FAR packet (how to estimate the corresponding frequency block size given the traffic demands will be discussed in Sect. 3.5). Based on the information collected in the previous time frame, nodes decide how the data frequency blocks are divided for the next time frame. The detailed approach is discussed in Sects. 3.5 and 3.6.

3.4 Transmission scheduling

Given that frequency diversity is not beneficial for broadcast traffic, in CROWN, a broadcast source sends a request in the control frequency block, which reserves one data frequency block. Each receiving node in the communication range switches one of its radio interfaces to that data frequency block to receive the broadcast packet.

For unicast traffic, CROWN attempts to maximize the frequency(spatial) reuse by assigning different links with different spectrums(time) slots. A unified metric, the *transmission fraction*, is used to evaluate the efficiency of the joint frequency assignment and link scheduling for unicast transmissions. The *total transmission fraction* of link (i, j) (TTF_{ij}) is defined to be the overall spectrum resource that link (i, j) could utilize in its two-hop range, in one time frame and excluding the parts used for control information exchange, that is, frequency block $[F_s + F_c, F_e, T_c, T_f]$. The transmission fraction of link (i, j) (TF_{ij}) is defined as the the maximal proportion of frequency resources (i, j) could obtain through joint frequency assignment and link scheduling.

TF_{ij} is used as the link cost used to compute the *logical distance* (LD_p^{ij}) of path p if link (i, j) . The route that has the minimum logical distance (which means that it can achieve the maximum end-to-end throughput) is selected. Once a path is selected, the corresponding transmission scheduling and frequency assignment that are incorporated in TF_{ij} are also established.

3.5 Data rate adjustment

To correctly divide the available frequency block $[F_s + F_c, F_e, T_c, T_f]$ along the time and frequency axes, the frequency block that each link (i, j) obtains must be guaranteed to satisfy its bandwidth requirements (r_{ij}). Given the available bandwidth $[f_s, f_e] \subseteq [F_s + F_c, F_e]$ and the minimal spectrum division interval σ , the set of all possible spectrum allocations for link (i, j) ($B_{ij} = \{[f_{s1}^{ij}, f_{e1}^{ij}], [f_{s2}^{ij}, f_{e2}^{ij}], \dots, [f_{sm}^{ij}, f_{em}^{ij}]\}$) is obtained, and then the corresponding data rate of each spectrum division is estimated using the following approach.

Let P_k^r denote the received signal power at node r for a signal transmitted by node k (we assume the fixed transmission power allocation for each link). According to Shannon Theorem, the physical-layer channel capacity is

$$C_{ij}^k = B_{ij}^k \times \log_2 \left[1 + \frac{P_i^r}{\sum_j P_j^r + \sigma_r^2} \right] \tag{2}$$

where B_{ij}^k is the bandwidth of spectrum division $[f_{sk}, f_{ek}]$, σ_r^2 is the background or thermal noise power at the front end of the receiver r . B_{ij}^k and σ_r^2 are given values specific to used radio technology. In this paper, we assume the the sum of the interference power follows a lognormal distribution. The log-normal radio model assumes that the logarithmic value of the received signal power is normally distributed with standard deviation around the logarithm of the area mean power. The log-normal radio model that we use in this paper is more realistic than the pathloss model because it allows for random signal power variations.

The motivation for using the log-normal radio model in ad-hoc networks is given in [38–41]. Other channel approximation methods can be incorporated in our calculation as well.

Then we can approximate the physical-layer channel capacity as

$$E[C_{ij}^k] = B_{ij}^k \times \log_2 \left[1 + \frac{P_i^r}{E[\sum_j P_j^r] + \sigma_r^2} \right] \tag{3}$$

where $E[\sum_j P_j^r]$ is the mean of the lognormal distribution.

We denote the maximal achievable data rate of link (i, j) (using spectrum $[F_s + F_c, F_e]$) by R_{\max} . We define a data rate R_{ij}^k for link (i, j) to be *feasible* if it is greater than the minimal data rate requirement of link (i, j) (r_{ij}). We denote the feasible data rate set for link (i, j) by RS_{ij} :

$$RS_{ij} = \{R_{ij}^k \geq r_{ij}, k = 1, \dots, K\} \tag{4}$$

We propose a heuristic approach (Algorithm 1) to compare all possible data rate combinations of links in the two-hop range of (i, j) . We use Jain’s *fairness index* (FI) to choose the data rate combination that maximize the fairness across all links. Let N_{ij}^2 be the total number of links in the two-hop range of (i, j) that have traffic, then

$$FI = \frac{(\sum_k x_{ij}^k)^2}{N_{ij}^2 \sum_k (x_{ij}^k)^2}, \quad \text{with } x_{ij}^k = \frac{R_{ij}^k}{r_{ij}} \tag{5}$$

Through this approach, we obtain the data rate set for all links in the two-hop range of (i, j) . We denote it by $R_{ij}^2 = \{R_1, R_2, \dots, R_{N_{ij}^2}\}$, and the corresponding spectrum bandwidth is $B_{ij}^2 = \{B_1, B_2, \dots, B_{N_{ij}^2}\}$.

Algorithm 1 Data rate adjustment algorithm

```

for each link  $(u, v)$  in the two-hop range of  $(i, j)$  do
    Estimate the feasible data rate set:
     $RS_{uv} = \{R_{uv}^k \geq r_{uv}, k = 1, \dots, K\}$ ;
end for
for each link  $(u, v)$  in the two-hop range of  $(i, j)$  that have traffics
do
    for each data rate  $R_{uv}^k \in RS_{uv}$  do
         $x_{uv}^k = \frac{R_{uv}^k}{r_{uv}}$ ;
    end for
end for
 $\{R_1, R_2, \dots, R_{N_{ij}^2}\} = \arg \max_{k_1, \dots, k_{N_{ij}^2}} \left\{ \frac{(\sum_{uv} x_{uv}^{k_{uv}})^2}{N_{ij}^2 \sum_{uv} (x_{uv}^{k_{uv}})^2} \right\}$ ;

```

3.6 Frequency assignment

After the data rate of each link that has packets to send is obtained, given that the bandwidth required by link (u, v) is B_{uv} , and that the unit of transmission scheduling over time

is t_s , then the data frequency block size for (u, v) is $[0, B_{uv}, 0, t_s]$. The problem of exploiting the frequency diversity and time-reuse in the two-hop range of link (i, j) to the largest extent is equal to placing as many different data frequency blocks as possible into the region of TTF_{ij} . We can map the joint frequency assignment and transmission scheduling problem into a 2D bin-packing problem. We adopt the first-fit decreasing (FFD) strategy to solve this NP-hard problem (FFD is one of the simplest heuristic algorithms for solving the bin packing problem. It requires $\Theta(n \log n)$ time, where n is the number of elements to be packed. It was proved that the bound $11/9 \text{ OPT} + 6/9$ for FFD is tight [42]).

In our approach, the data frequency blocks are sorted in decreasing order of bandwidth requirements, then each link is inserted into the first time slot with sufficient remaining bandwidth, as Fig. 3 shows (the computational complexity of the spectrum allocation algorithm is $\Theta(N_{ij}^2 \log N_{ij}^2)$.) The output of the algorithm includes the amount of bandwidth that link (i, j) could obtain (BW_{ij}), and the average

number of transmissions in the two-hop range of link (i, j) in one time frame (S_{ij}).

$$S_{ij} = \frac{\sum_{(u,v)} T_{uv}}{t_s N_{ij}^2} \tag{6}$$

Given that $TTF_{ij} = R_{\max}(T_f - T_c)$, the transmission fraction of (i, j) is

$$TF_{ij} = \frac{S_{ij} BW_{ij}}{R_{\max}(T_f - T_c)} \tag{7}$$

3.7 Routing

The optimal routing policy for wireless ad hoc networks using spectrum-agile radios needs to jointly consider the throughput and the efficiency of spectrum utilization. To do so, we adapt a proactive distance-vector routing protocol. We use TF_{uv} to replace the link cost information used in the traditional routing distance calculation, and get the *logical distance* (LD_p^{uv}) of path p if link (u, v) is included.

Fig. 3 Spectrum allocation algorithm

```

PSEUDO-CODE OF SPECTRUM ALLOCATION
ALGORITHM

Notations:
Buv: bandwidth requirements for link (u, v);
Ruv: the corresponding data rate of Buv;
RB(t) = [fs, fe]: remaining bandwidth at time slot t;
NRi(t): number of available radios interfaces of node i at time slot t;
FAT(f, Δf, t) = {(u, v), ...}: frequency allocation table, it indicates the spectrum [f, f + Δf] is occupied by link set {(u, v), ...} at time slot t;
I((m, n), (u, v)): indicator function, I((m, n), (u, v)) = 1 if (m, v) or (u, n) is in the interference range of each other;
Tuv: the time that link (u, v) could transmit.
BWuv: the amount of bandwidth link (u, v) could obtain;
Sij: the average number a link could be scheduled in one time frame.

1: procedure INITIALIZATION()
2:   RB(t) ← [Fs + Fc, Fe]; ▷ for every time slot t
3:   Tuv ← 0; ▷ for each link (u, v)
4:   NRi(t) ← K; ▷ for each node i at every time slot t
5:   FAT ← ∅;
6:   /*exclude the existing broadcast transmissions*/
7:   for each broadcast transmission from node u at frequency block [f, Δf, t] do
8:     if (NRu(t) > 0) then
9:       FAT(f, Δf, t) = u;
10:      NRu(t) = NRu(t) - 1;
11:     end if
12:   end for
13: end procedure

14: procedure SPECTRUM_ALLOCATION()
15:   SORT(Bij2 = {B1, B2, ..., BNi2});
16:   /*Sort all links according to the decreasing order
of required bandwidth */
17:   for each link (u, v) ∈ Lij2 do
18:     for each time slot ts ∈ [Tc, Tf] do
19:       if ((NRu(t) > 0) ∧ (NRv(t) > 0)) then
20:         /*There are available radio inter-
faces*/
21:         if (FAT(f, Δf, t) ≠ ∅) then
22:           for each (Δf ≥ Buv) do
23:             /*there are available band-
widths*/
24:             if (∀(m, n) ∈ FAT(f, Δf, t),
25:               I((m, n), (u, v)) == 0)
26:             then
27:               /*(u, v) does not interfere
with any existing transmission schedules*/
28:               FAT(f, Δf, t) =
29:               FAT(f, Δf, t) ∪ (u, v);
30:               NRu(t) = NRu(t) - 1;
31:               NRv(t) = NRv(t) - 1;
32:               Tuv = Tuv + ts;
33:             return;
34:           end if
35:         end for
36:         if ((fe - fs) ≥ Buv) then
37:           /*there are available bandwidths*/
38:           /*Assign links in different fre-
quency spectrums*/
39:           fs = fs + Buv;
40:           FAT(fs, Buv, t) = (u, v);
41:           NRu(t) = NRu(t) - 1;
42:           NRv(t) = NRv(t) - 1;
43:           Tuv = Tuv + ts;
44:         end if
45:       end if
46:     end for
47:   end for
48:   Sij =  $\frac{\sum_{(u,v)} T_{uv}}{t_s N_{ij}^2}$ ;
49: end procedure

```

The route that has the minimum logical distance (i.e., it achieves the maximum end-to-end throughput) is selected. Once a path is selected, the transmission scheduling and frequency assignment incorporated in TF_{uv} are also established.

3.7.1 Logical distance calculation

The *logical distance* (LD) of path p is given by a path function f^p based on the transmission fraction of its consisting links. We define the f^p as:

$$f^p = \min_{(u,v)} \left\{ \frac{1}{TF_{uv}} \right\}, \quad (u,v) \in p \quad (8)$$

Let LD_j^i denote the logical distance from node i to destination j as known by node i . LD_{jk}^i denotes the logical distance LD_j^k from node k , which is a neighbor of node i , to destination j , as reported to node i by node k . FLD_j^i denotes the *feasible logical distance* (FLD) of node i for destination j , which is an estimate of the minimal logical distance maintained for destination j by node i .

Node i maintains a routing entry for each destination j , which includes FLD_j^i , LD_j^i and the successor set chosen for j (denoted by S_j^i). Node i maintains a neighbor table that records the logical distance LD_{jk}^i reported by each node k in its neighbor set N^i for each destination j ; and a link table that reflects the transmission fraction TF_{ik} for each adjacent link (i, k) , $k \in N^i$. The multiple paths computed between node i and destination j are called the *logical shortest multipath*, denoted by LSM_j^i , and it is such that at least one of the paths in it has the minimal logical distance for j . In CROWN, each node maintains up to x LSMs for each destination.

We focus on the operation of node i 's computation of LSMs for a destination j . Provided that each node maintains up to x LSMs for destination j , node i may receive and record x values of LD_{jk}^i from each neighbor k ; node i also reports to its neighbors the logical distances of the x LSMs from itself to destination j , of which the minimal value is also used as the feasible logical distance FLD_j^i of node i . When a node is powered up, FLD is set to ∞ , and all the other entries are set to empty. For destination j we have $LD_j^j = 0$, $FLD_j^j = 0$, and $LD_{jj}^k = 0, \forall k \in N^j$. We also assume that node i knows the transmission fraction of each outgoing link TF_{ik} , $k \in N^i$.

When node i receives an input event at time t , node i behaves in one of three possible ways: (a) Node i remains idle and all distance estimates are left unchanged; (b) node i receives LD_j^k from neighbor k , updates the estimates LD_{jk}^i and leaves all other estimates unchanged; and (c) node i updates $S_j^i(t)$ and $FLD_j^i(t)$ for destination j based on the following equations:

$$S_j^i(t) = \left\{ k \mid LD_{jk}^i(t) < FLD_j^i(t), k \in N^i \right\} \quad (9)$$

and updates its feasible logical distance by

$$FLD_j^i(t) = \min \left(LD_{jk}^i(t), \frac{1}{TF_{ik}} \right) \quad (10)$$

for all LD_j^k reported by each neighbor k and over all neighbors in N^i . Then node i re-computes the logical distance of each LSM maintained for j (up to x LSMs), and sends neighbors updates if any change occurs; otherwise leaves all other estimates unchanged.

The aggregate of the routing entries for destination j maintained at each node forms a directed graph rooted at j , which is a subgraph of network G and is denoted by SG_j . This subgraph includes links $\{l_{i,k} \mid k \in S_j^i \text{ for } \forall i \in V\}$. If routing converges correctly, SG_j is a directed acyclic graph (DAG) in which each node can have multiple successors for node j .

Although multiple SG_j can exist for destination j in a given network, CROWN constructs SG_j in a way that the path with the shortest logical distance for destination j is always maintained (by Eqs. (9) and (10)), and as such makes SG_j an optimal successor graph.

Loop-free routing is also attained by using Eqs. (9) and (10). The proof that this is the case is presented in [43].

3.7.2 Logical distance propagation and deduction

Logical distances are sent in the proactive routing updates messages. The propagation of routing updates messages consists of two phases. During the route discovery stage, routing updates propagate through the network for each destination in order to inform all nodes of possible routes to each destination, regardless of frequency assignments or transmission schedules. Neighbor update messages are transmitted as broadcast packets to accomplish this. After routes are established to destinations, the neighbor update messages that include the future transmission scheduling and frequency assignment information are transmitted as unicast packets to the specific neighbors. This way, multiple neighbor updates can be sent simultaneously over different parts of the spectrum.

The distance vector reporting a path p for destination j by neighbor k is a tuple of $\{j, LD_j^k, TF_{ik}\}$, in which LD_j^k is the logical distance for p , and TF_{ik} is the transmission fraction for adjacent link (i, k) . For each LSM p computed for destination j , besides the logical distance, the raw transmission fraction of adjacent link TF_{ik} must also be maintained, because TF_{ik} is used to verify whether p can be a feasible path when a request to forward traffic arrives.

Assume that the minimal logical distance reported by neighbor k for destination j at node i is LD_{jk}^i , and that the

current feasible logical distance for j at node i is FLD_j^i . Let \circ denote the concatenation of two paths or links. According to Eq. 9, path $l_{i,k} \circ p$ is now considered as a candidate path for j if $\widetilde{LD}_{jk}^i < FLD_j^i$ (i.e., $k \in S_j^i$).

Path $l_{i,k} \circ p$ can be upgraded to a LSM if it has a smaller logical distance than the current feasible logical distance. In mobile scenarios, it may be the case that node i is unable to find a neighbor k that has reported a logical distance that is smaller than the feasible logical distance (FLD_j^i) maintained by node i at the time. CROWN uses diffusing computations to coordinate node i with all upstream nodes that use node i in their LSMs for destination j to update the corresponding logical distance and feasible logical distance (see [43]).

3.7.3 Handling network dynamics

Even though Eqs. (9) and (10) guarantee loop-free LSMs, it may be the case that node i is unable to find a neighbor k that has reported a logical distance that is smaller than the feasible logical distance (FLD_j^i) maintained by node i at the time. In this case, node i must increase FLD_j^i in order to be allowed to choose a successor set, but in a way that all nodes whose LSMs involve node i incorporate that update in their own computations of logical distances and feasible logical distance.

CROWN uses diffusing computations to accomplish the above task by coordinating node i with all upstream nodes that use node i in their LSMs for destination j . Accordingly, nodes running CROWN can be in two states: *ACTIVE* or *PASSIVE*. A node i is in passive state as its successor set S_j^i given by Eq. 9 includes nodes that can provide the optimal path with the shortest logical distance for j . Nodes in passive state behave much like DBF, i.e., they simply use Eq. 10 to compute the shortest logical distance for destination j , without having to establish any coordination with their neighbors.

Node i becomes active when none of its neighboring nodes in S_j^i can provide the optimal path for destination j . At this point, node i must synchronize with its upstream nodes before it is allowed to increase its logical feasible distance in order to obtain a new set of successors. The events that may cause such a state transition are the arrival of a routing message from a neighbor, or the change of the weight or status of an adjacent link. To become active, node i originates a diffusing computation by sending each neighbor a *QUERY* that reports the desired logical distance for destination j . Node i detects the termination of this diffusing computation when it has received a *REPLY* from each neighboring node. After that, node i can be sure that all the upstream nodes using i for destination j either are no longer in the successor graph SG_j , or have incorporated the new logical distance that node i reported in its query, and

as such is free to raise its feasible logical distance to the desired value.

Node i returns to passive state if at least one of the newly obtained successors (by Eq. 9) provides the shortest logical distance for destination j ; otherwise, node i immediately starts another diffusing computation by sending out new queries and continues to stay in active state. The mechanism used by CROWN ensures that node i can return to passive state after a finite number of active phases. After becoming passive, node i sends each neighboring node an *UPDATE* to announce the changes that occur to its routing table, if there is any.

Node i can participate in at most one diffusing computation for destination j at any time, and cannot send new queries for the same destination j during an active phase, until replies are received from each neighboring node. This simplifies the operation and reduces the number of states needed to maintain by nodes in active state. The details of diffusing computations can be found in [44].

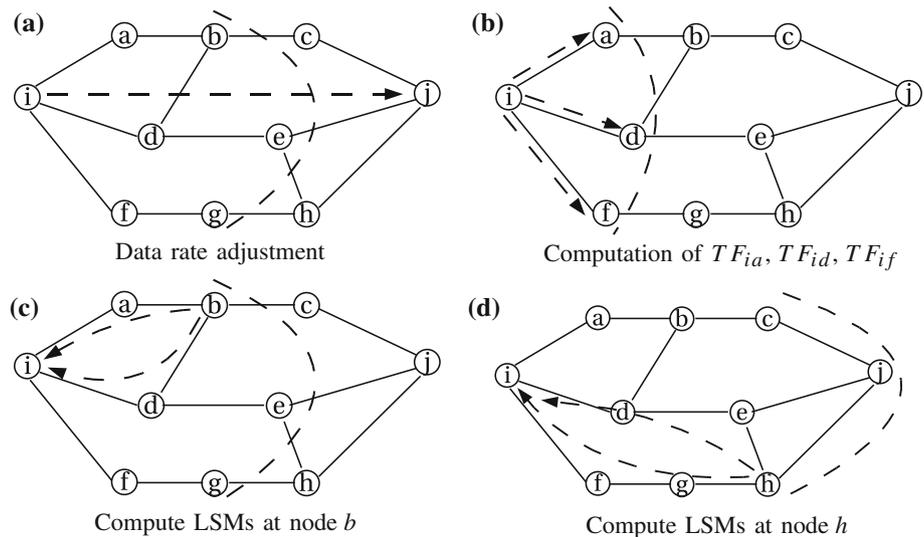
3.8 Example

Figure 4 illustrates how nodes run CROWN to deduce their LSMs for the destination j without knowing global network state. We assume that there is a traffic flow from i to j . Each node is labeled with (LD_j^i, FLD_j^i) , i.e., its shortest logical distance and feasible logical distance for j ; and each link is labeled with the associated transmission fraction.

In Fig. 4(a), based on its bandwidth requirement and the data rate of existing traffic flows in the two-hop range, node i first adjusts the data rate according to Algorithm 1. Then it calculates the transmission fraction for link $\{(i, a), (i, d), (i, f)\}$, and sends the corresponding results through neighbor updates. After receiving the broadcast requests from i on the control frequency, nodes $\{a, d, f\}$ switch one of their radio interfaces to the specified data spectrum to receive the neighbor update. Then nodes $\{a, d, f\}$ calculate the transmission fractions for links $\{(a, b), (d, b), (d, e), (f, g)\}$, and send the neighbor updates to their upstream nodes. In Fig. 4(c), node b selects the optimal path to node i , and nodes $\{b, e, g\}$ calculate the transmission fraction for links $\{(b, c), (b, d), (e, j), (e, h), (g, h)\}$. In Fig. 4(d), node h chooses between paths $(h \rightarrow e \rightarrow d \rightarrow i)$ and $(h \rightarrow g \rightarrow f \rightarrow i)$. This process continues until each node obtains the optimal path to the destination.

Note that a schedule is formed *in sequence* along the routing path from the destination to the source. Descendent nodes exclude the schedule of ascendent nodes, which is indicated in the FAR packets. With this approach, the schedule and frequency assignment along a specific route is compatible, while the schedules among different LSMs may be in conflict. This is why we only allow one LSM to be chosen each time. After the routes are established, the

Fig. 4 Routing example



future neighbor update messages are sent through the existing transmission scheduling and channel assignment as unicast packets, e.g., when i updates the TF_{ia} , it will just send a neighbor update message to a . Other nodes that do not use a in their LSMs to i will not receive the TF_{ia} .

4 Numerical analysis

In this section, we numerically analyze the approximate MAC layer throughput and the complexity of routing layer protocol of CROWN.

4.1 Approximate throughput analysis

To simplify the analysis, we consider a fully-connected network topology with N nodes. All links are bidirectional or symmetrical. Given that CROWN increases the spatial reuse of the system through the distributed link scheduling in the two-hop range, a fully-connected network is the worst case scenario in terms of interference, contention or spatial reuse. Therefore, the throughput of CROWN for a fully-connected network with N nodes is a lower bound of the throughput of CROWN for a general topology where the number of nodes in a two-hop neighborhood is N . The channels are assumed to be error free and have no capture effects. Transmissions overlapped on the same channel at a receiver leads to a collision and no packets involved in it can be received correctly by the receiver.

We assume that traffic arrives at each node according to Poisson process with average arrival rate λ requests per slot. Therefore, the total traffic load is denoted by $G = N\lambda$. We consider variable-length flow and assume that, on the

average, it takes δ slots to send all the data packets in a flow, i.e., the average flow length (AFL) is δ slots. We also assume that the flow length is geometrically distributed, which implies that the probability that a flow ends at the end of a transmission slot is $q = 1/\delta$. We assume all links require the same amount of transmission rate r and the maximal number of parallel transmissions in the data frequency block is $M = \frac{R_{max}}{r}$.

The system can be fully described by one state variable X , the number of communication pairs at time t . When $X = k$, $2k$ nodes are involved in data transmissions, while $N - 2k$ nodes are idle. The maximum number of communication pairs is bounded by the number of nodes and the size of data frequency block: $X \in \{0, 1, \dots, \min(\lfloor \frac{N}{2} \rfloor, M)\}$.

We model the evolvement of the system as a discrete-time Markov chain, where each state of the Markov chain can transit to any state. A transition may occur when new communication pairs are formed or existing transfers end. Let π_k denote the stationary probability that the system is in state k .

For the system is in state k , we denote the probability that n senders end their flows during a frame as

$$T_k^{(n)} = \binom{k}{n} q^n (1 - q)^{k-n} \quad 0 \leq n \leq k \tag{11}$$

We condition on the number of senders ending their flows in a frame (n) to calculate the transition probabilities. For the transition from state k in frame f to state l in frame $f + 1$, at least $\hat{n} = \max(0, k - l)$ nodes will end their flows in frame f . Therefore, $\hat{n} \leq n \leq k$, and $s = l - (k - n)$ new transmission pairs will be formed. We denote the probability that there is i new agreements made when the state is k as $\theta_k^{(i)}$. The transition probability from state k to state l is $P_{kl} = \sum_{n=\hat{n}}^k T_k^{(n)} \theta_k^{(s)}$.

Through solving the global balance equation:

$$\pi_l = \sum_{k=0}^{\min(\lfloor \frac{N}{2} \rfloor, M)} \pi_k P_{kl} \tag{12}$$

with the boundary condition:

$$\sum_{l=0}^{\min(\lfloor \frac{N}{2} \rfloor, M)} \pi_l = 1, \tag{13}$$

we can get the stationary probability π_l .

The average system throughput is

$$S = \sum_{i \in X_i} i \pi_i r \tag{14}$$

We assume each node randomly chooses among M data frequency blocks, then the average number of idle nodes on a data frequency block c when system is in state k is:

$$\bar{n}_c = \lceil \frac{N - 2k}{M} \rceil \tag{15}$$

We can obtain the probability that there is successful reservation as

$$\eta = \bar{n}_c p_a (1 - p_a)^{\bar{n}_c - 1} \tag{16}$$

There can be up to M successful reservations over all data channels:

$$\theta_k^{(i)} = \begin{cases} \binom{M}{i} \eta^i (1 - \eta)^{M-i} & \text{if } 1 \leq i \leq M; \\ 1 - \sum_{i=1}^M \theta_k^{(i)}, & \text{if } i = 0; \\ 0, & \text{otherwise.} \end{cases}$$

We assume the total available bandwidth is $R_{\max} = 54$ Mbps. We set $N = 10$ and vary the traffic arrival rate from 0.1 to 0.9. We also change the bandwidth requirements of traffic flows, and take $r = 3, 6, 16, 27$ Mbps for

example. The throughput comparisons under different AFLs and traffic loads are shown in Figs. 5 and 6. We can find with the increase of the bandwidth requirements, the system throughput does not increase linearly. When r increases from 3 to 16 Mbps, the transmission rate on each data frequency block also increases, but the number of parallel transmission decreases. The decrease of the parallel transmissions not only reduces the control overheads, but also decreases the frequency reuse of the system. The latter dominates when r increases from 16 to 27 Mbps. There is a trade-off between the control overheads and frequency reuse of the system. We can also find CROWN could attain high system throughput even under high traffic loads, excluding the cases that there are not enough frequency reuse in the system ($r = 27$ Mbps).

4.2 Complexity analysis of routing protocol

We model the network as a directed graph $G = \{V, L\}$, where V is the set of nodes and L is the set of links inter-connecting the nodes. N and E are the cardinalities of V and L , i.e., $N = |V|$ and $E = |L|$, respectively. For each node i runs CROWN, the space complexity is $O(xlN_i|N + xN) = O(xlN_i|N)$, where N_i is the number of neighbors of node i , because the main routing table and each neighbor table have $O(N)$ entries, and each entry can keep up to x routes for each destination. The computation complexity of transmission fraction calculation algorithm is $O(D|N_i^2|)$, where N_i^2 is the number of nodes in the two-hop range of i , because in each time slot, the number of links to be scheduled is $O(|N_i^2|)$, and there are D time slots. The total time taken for a node to process distance vectors regarding a particular destination is $O(xD|N_i^2|)$.

Fig. 5 System throughput comparison 1 ($r = 3, 6$ Mbps)

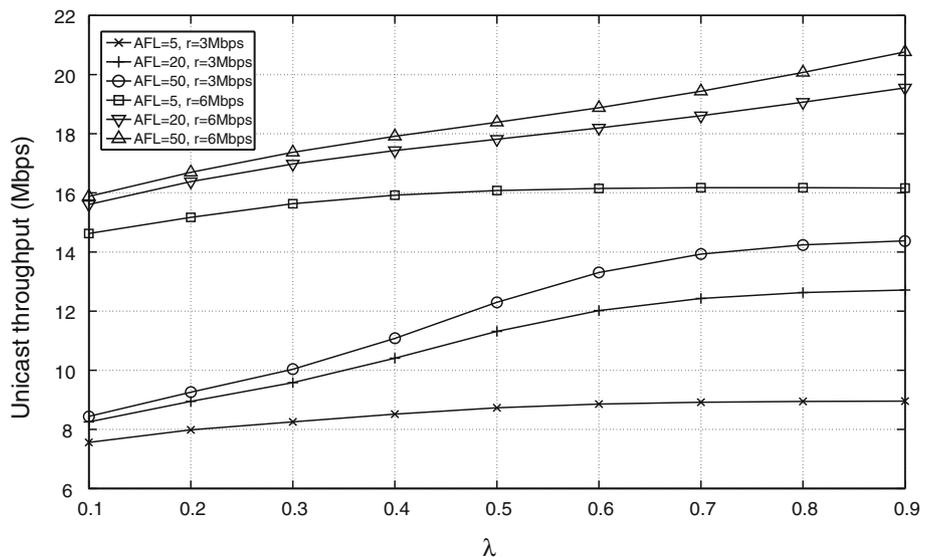
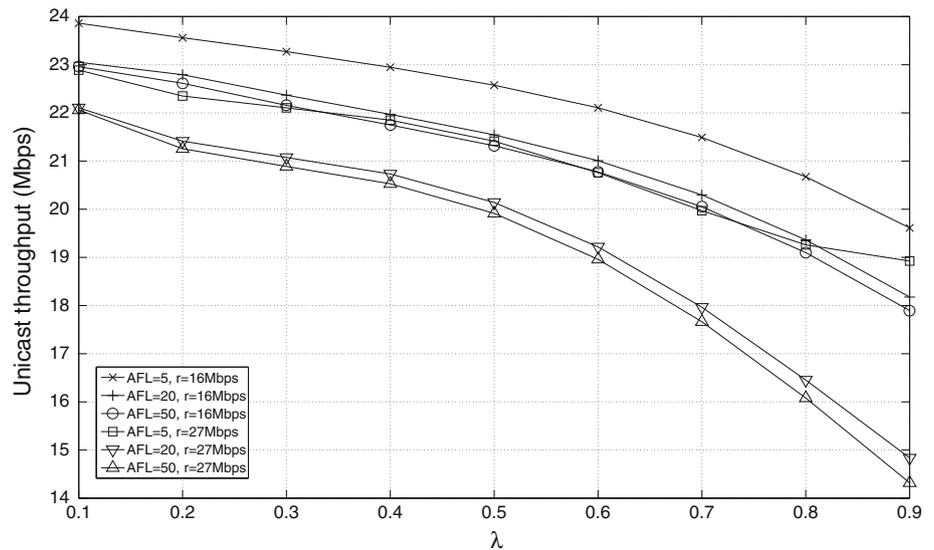


Fig. 6 System throughput comparison 1 ($r = 16, 27$ Mbps)



5 Performance evaluation

We implemented CROWN under Qualnet [45] and compare it with DORP [34]. We select DORP because it is a joint optimization scheme that considers the interaction between on-demand routing and spectrum scheduling, which is most comparable with CROWN. We assume each node has four radio interfaces. The overall available spectrum is 86 MHz (the size of 2.4 ISM band). t_s is 50 ms. A time frame is made up of 100 time slots ($L = 100$). The set of bandwidth interval (σ) includes 5, 10, 15 MHz. The bandwidth requirements of traffic flows are uniformly distributed in 1~10 Mbps. We assume that each 1 MHz spectrum delivers 1.2 Mbps data rate [46]. The packet length used is 1,024 bytes. The duration of the simulation is 100 s. The simulations are repeated with ten different seeds to average the results for each scenario.

The performance gain of CROWN mainly comes from its joint optimization of frequency assignment and distributed link scheduling, as well as from choosing paths based on the efficiency of the underlying transmission scheduling and frequency assignment. We illustrate these performance improvements separately under different scenarios.

5.1 Grid topology

To illustrate the performance gain due to the joint frequency assignment and distributed link scheduling, we first investigate the performance of CROWN under a 6×6 regular grid with static routing of fixed flows. As Fig. 7 shows, the transmission range of each node is T_R , each node is T_R away from each other. The interference range $I_R = \sqrt{2}T_R$. We set up five CBR flows, such that three of

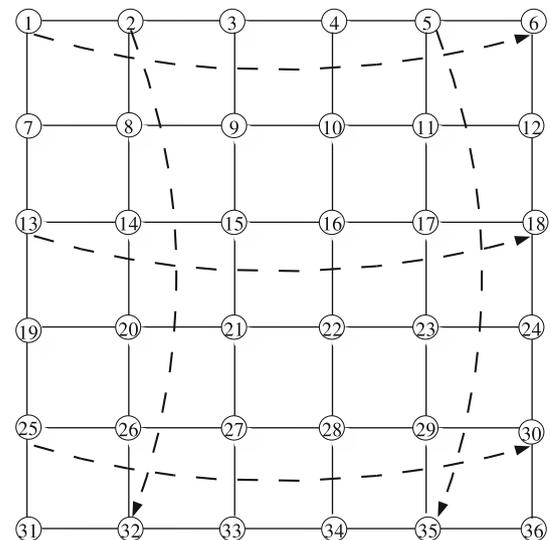


Fig. 7 6×6 regular grid

Table 1 System throughput for grid topology

DORP (5 MHz)	DORP (10 MHz)	DORP (15 MHz)
15.38	13.96	11.27
CROWN (5 MHz)	CROWN (10 MHz)	CROWN (15 MHz)
27.49	25.31	24.12

them have node-disjoint horizontal paths and two of them have node-disjoint vertical paths in the grid, with the vertical path crossing the second and second-to-last hops of the horizontal paths. The system throughput and packet delivery ratio comparison are shown in Table 1 and Table 2. The results indicate that through distributed transmission scheduling and frequency assignment, CROWN improves the system performance significantly.

Table 2 Packet delivery ratio for grid topology

DORP (5 MHz)	DORP (10 MHz)	DORP (15 MHz)
63.28%	58.34%	54.17%
CROWN (5 MHz)	CROWN (10 MHz)	CROWN (15 MHz)
74.39%	67.82%	63.26%

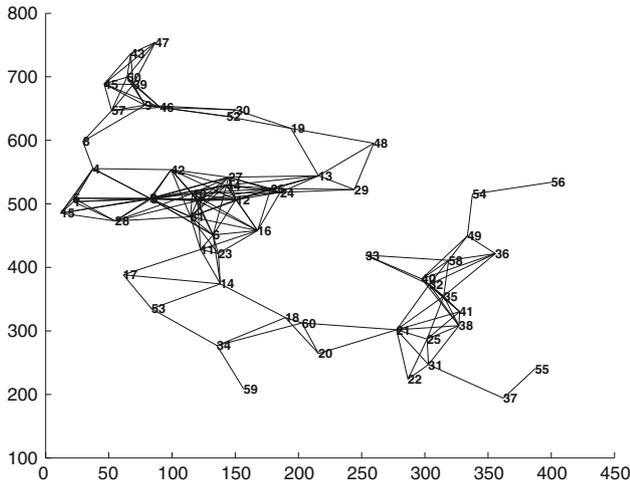


Fig. 8 Random topology

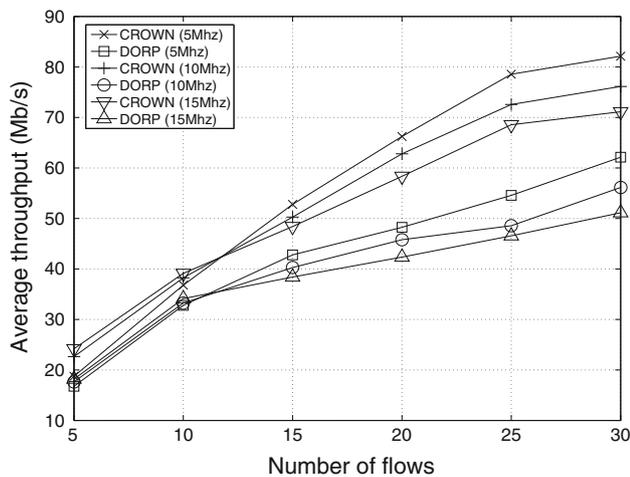


Fig. 9 Random throughput

5.2 Random topology

To illustrate performance improvement due to the joint optimization of MAC and routing, we generated 10 topologies with 60 nodes uniformly distributed across a 800×800 square meters area, as Fig. 8 shows. We varied the number of traffic flows and the minimal bandwidth interval σ . We compare the system throughput, packet delivery ratio and average end-to-end delay in Figs. 9, 10,

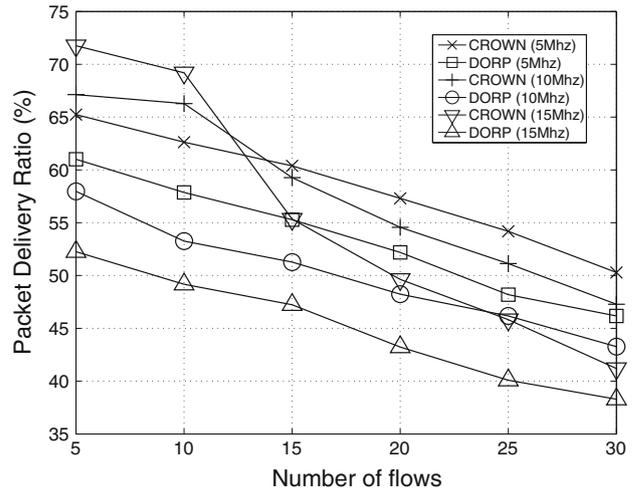


Fig. 10 Packet delivery ratio comparisons

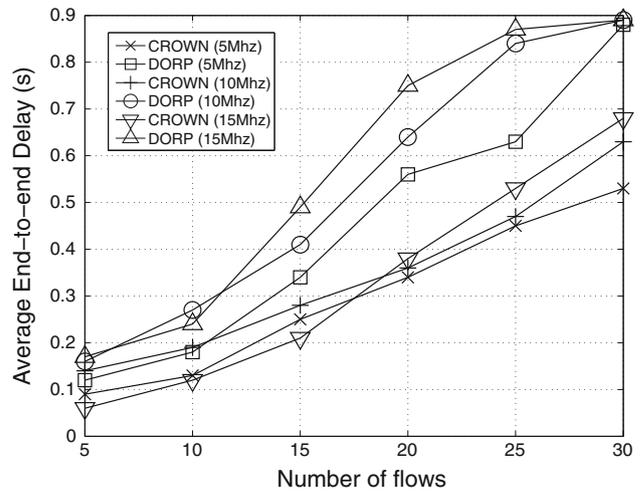


Fig. 11 Average end-to-end delay comparisons

11. As Fig. 9 shows, system throughput decreases with increases of σ , which indicates a reduction of non-overlapping channels. There is a trade-off between the feasible data rates and the frequency reuse of the system. From Fig. 10 to Fig. 11, we observe that compared with DORP, CROWN increases the packet delivery ratio and reduces the average end-to-end delay. From all the simulation results shown in Sects. 5.1 and 5.2, CROWN outperforms DORP significantly. This is because although DORP incorporates the frequency assignment information in the routing control packets, and selects the frequency bands based the path cumulative delay, DORP does not dynamically adjust the data transmission rates to maximize the spectrum utilization. The path cumulative delay estimation is also less optimal compared with CROWN since DORP does not consider the underlying MAC layer transmission scheduling.

6 Conclusion

We proposed a novel distributed link-layer scheduling and routing optimization approach for wireless ad hoc networks using spectrum-agile radios. Each node dynamically adjusts its frequency assignments to satisfy the bandwidth requirements and achieve fairness across different transmission pairs. Routing selection is made based on the efficiency of the underlying link-layer scheduling and frequency assignment schemes. Simulation results show that the proposed approach increases the system performance significantly by load balancing the traffic over different channels and different times.

Acknowledgments This work was partially sponsored by the US Army Research Office under grants W911NF-04-1-0224 and W911NF-05-1-0246, by the National Science Foundation under grant CNS-0435522, and by the Baskin Chair of Computer Engineering

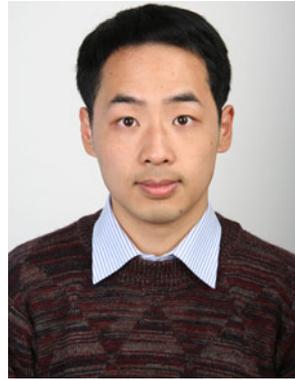
Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

- Kahol, A., Khurana, S., Gupta, S. K. S., & Srimani, P. K. (2001). Adaptive distributed dynamic channel allocation for wireless networks. *Journal of Parallel and Distributed Computing*, 61, 898–914.
- Boukerche, A. (2001). *Algorithms and protocols for wireless, mobile Ad Hoc networks*. London: Wiley Series on Parallel and Distributed Computing.
- Boukerche, A. (2005). *Handbook of algorithms for wireless networking and mobile computing*. London: Chapman and Hall/CRC Computer and Information Science.
- Arikan, E. (1984). Some complexity results about packet radio networks. *IEEE Transactions on Information Theory*, 30(4), 681–685.
- Chlamtac, I., & Farago, A. (1994). Making transmission schedules immune to topology changes in multi-hop packet radio networks. *IEEE/ACM Transactions on Networking*, 2(1), 23–29.
- Oikonomou, K., & Stavrakakis, I. (2004). Analysis of a probabilistic topology-unaware TDMA MAC policy for Ad-Hoc networks. *IEEE JSAC Special Issue on Quality-of-Service Delivery in Variable Topology Networks*, 22(7), 1286–1300.
- Ju, J.-H., & Li, V. O. K. (1998). An optimal topology-transparent scheduling method in multi-hop packet radio networks. *IEEE/ACM Transactions on Networking*, 6(3), 298–306.
- Rentel, C. H. (2006). *Network time synchronization and code-based scheduling for wireless Ad Hoc networks minus 0.4 em*, Ph.D. Thesis, January, Carleton University.
- Bao, L., & Garcia-Luna-Aceves, J. J. (2001). A new approach to channel access scheduling for Ad Hoc networks. In *ACM Seventh Annual International Conference on Mobile Computing and Networking (Mobicom)*.
- Tang, Z., & Garcia-Luna-Aceves, J. (1999). A protocol for topology-dependent transmission scheduling. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, September 21–24.
- Das, S., Perkins, C., & Royer, E. (2000). Performance comparison of two on-demand routing protocols for Ad Hoc networks. In *Proceedings IEEE Conference on Computer Communications (INFOCOM)*, Tel Aviv, Israel, March, pp. 3–12.
- Royer, E. M., & Toh, C.-K. (1999). A review of current routing protocols for Ad Hoc mobile wireless networks.
- Royer, E. M., & Toh, C.-K. (1999). A review of current routing protocols for Ad-Hoc mobile wireless networks. *IEEE Personal Communications Magazine*, pp. 46–55, April.
- Lee, S.-J., Toh, C.-K., & Gerla, M. (1999). Performance evaluation of table-driven and on-demand Ad Hoc routing protocols. In *Proceedings of the IEEE PIMRC*, Osaka, Japan, Sep, pp. 297–301.
- Perkins, C. E. (2001). *AdHoc networking*. Reading: Addison Wesley.
- Iwata, A., Chiang, C.-C., Pei, G., Gerla, M., & Chen, T.-W. (1999). Scalable routing strategies for Ad-hoc wireless networks. *IEEE Journal on Selected Areas in Communications*, pp. 1369–1379, Aug.
- Pei, G., Gerla, M., & Chen, T.-W. (2000). Fisheye state routing: A routing scheme for Ad Hoc wireless networks. In *Proceedings of the IEEE ICC*, New Orleans, LA, Jun.
- Clausen, P. J. T. (2003). *RFC:3626. Optimized link state routing protocol (OLSR)*. Oct.
- Perkins, C., Belding-Royer, E., & Das, S. (2003). *RFC 3561-Ad hoc on-demand distance vector (AODV) Routing*. Jul.
- Toh, C.-K. (1997). Associativity-based routing for Ad Hoc mobile networks. *Wireless Personal Communications Journal, Special Issue on Mobile Networking and Computing Systems, Kluwer Academic Publishers*, 4, 103–139.
- Johnson, D., & Maltz, D. (1996). Dynamic source routing in Ad Hoc wireless networks. *Mobile Computing*, 353.
- Corson, M. S., & Ephremides, A. (1995). A distributed routing algorithm for mobile wireless networks. *ACM/Baltzer Wireless Networks Journal*, 1, Feb.
- Park, V. D., & Corson M. (1997). A highly adaptive distributed routing algorithm for mobile wireless networks. In *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, Kobe, Japan, April, pp. 1405–1413.
- Raniwala, A., & Chker Chiueh, T. (2005). Architecture and algorithms for an Ieee 802.11-based multi-channel wireless mesh network. In *IEEE INFOCOM*, March.
- Raniwala, A., Gopalan, K., & Chiueh, T. C. (2004). Centralized algorithms for multi-channel wireless mesh networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 8(2), 50–65.
- Kyasanur, P., & Vaidya, N. H. (2005). Routing and interface assignment in multi-channel multi-interface wireless networks. In *IEEE WCNC*.
- Kodialam, M., & Nandagopal, T. (2003). Characterizing achievable rates in multi-hop wireless networks: The joint routing and scheduling problem. In *Proceedings of ACM Mobicom*, pp. 42–54.
- Zhang, J., Wu, H., Zhang, Q., & Li, B. (2005). Joint routing and scheduling in multi-radio multi-channel multi-hop wireless networks. In *Proceedings of Broadnets*, pp. 631–640.
- Alicherry, M., Bhatia, R., & Li, L. (2005). Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In *Proceedings of ACM Mobicom*.
- Kodialam, M., & Nandagopal, T. (2005). Characterizing the capacity region in multi-radio, multi-channel wireless mesh networks. In *Proceedings of ACM Mobicom*, pp. 73–87.
- Meng, X., Tan, K., & Zhang, Q. (2006). Joint routing and channel assignment in multi-radio wireless mesh networks. In *Proceedings of IEEE ICC*.
- Tam, W.-H., & Tseng, Y.-C. (2007). Joint multi-channel link layer and multi-path routing design for wireless mesh networks. In *IEEE INFOCOM*.

33. Wu, H., Yang, F., Tan, K., Chen, J., Zhang, Q., & Zhang, Z. (2006). Distributed channel assignment and routing in multi-radio multi-channel multi-hop wireless networks. *Journal on Selected Areas in Communications*, 24, 1972–1983.
34. Cheng, G., Liu, W., Li, Y., & Cheng, W. (2007). Joint on-demand routing and spectrum assignment in cognitive radio networks. In *Proceedings of IEEE ICC*.
35. Ma, L., Han, X., & Shen, C. (2005). Dynamic open spectrum sharing MAC protocol for wireless Ad Hoc networks. In *IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks(Dyspan)*.
36. Cabric, D., Mishra, S., & Brodersen, R. (2004). Implementation issues in spectrum sensing for cognitive radios. In *Proceedings of 38th Asilomar Conference on Signals, Systems and Computers*, Nov, pp. 772–776.
37. Bahl, P., Chandra, R., & Dunagan, J. (2004). “SSCH: Slotted seeded channel hopping for capacity improvement. In *IEEE 802.11 Ad-Hoc Wireless Networks*, in *ACM Mobicom*.
38. Hekmat, R., & Mieghem, P. V. (2005). Interference power sum with log-normal components in Ad-Hoc and sensor networks. In *Proceedings of Third International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt’05)*, pp. 174–182.
39. Hekmat, R., & Mieghem, P. V. (2003). Degree distribution and hopcount in wireless Ad-Hoc networks. In *Proceedings of 11th IEEE International Conference on Networks (ICON 2003)*, Sydney, Australia, pp. 603–609.
40. Hekmat, R., & Mieghem, P. V. (2004). Study of connectivity in wireless Ad-Hoc networks with an improved radio model. In *Proceedings of the 2nd Workshop on Modeling and Optimization in Mobile ad hoc and Wireless Networks (WiOpt04)*, Cambridge, UK, March, pp. 142–151.
41. Bettstetter, C., & Hartmann, C. (2003). Connectivity of wireless multihop networks in a shadow fading environment. In *Proceedings of the 6th ACM international workshop on modeling analysis and simulation of wireless and mobile systems (MSWiM)*, San Diego, CA, USA, Sep, pp. 28–2.
42. Dsa, G. (2007). The tight bound of first fit decreasing bin-packing algorithm Is $FFD(I) \leq (11/9)OPT(I)+6/9$. In *Proceedings of ESCAPE, Springer LNCS 4614*, pp. 1–11.
43. Li, Z., & Garcia-Luna-Aceves, J. J. (2006). A distributed approach for multi-constrained path selection and routing optimization. In *Proceedings of QSHINE’06, Waterloo, Ontario, Canada*, August.
44. Garcia-Lunes-Aceves, J. J. (1993). Loop-free routing using dif-fusing computations. *IEEE/ACM Transactions Network*, 1(1), 130–141.
45. Qualnet Simulator, Scalable Network Technologies, <http://www.scalable-networks.com/>.
46. Proakis, J. (2001). *Digital communications*. NY: McGraw Hill.

Author Biographies



Xin Wang received the B.S. and M.S. degrees in electronic engineering from Tsinghua University, Beijing, China, in 2002 and 2005, respectively. He received Ph.D. degree in computer engineering from University of California, Santa Cruz, U.S. His research interests include performance evaluation and protocol design, cross layer optimization in wireless networks. He is a member of IEEE and ACM.



J. J. Garcia-Luna Aceves has published a book, more than 330 papers, and 24 U.S. patents. He has directed 25 Ph.D. theses and 20 M.S. theses since he joined UCSC in 1993. He is the General Chair of the ACM Mobicom 2008 conference. He has been the General Chair of the IEEE SECON 2005 Conference; Program Co-Chair of ACM MobiHoc 2002 and ACM Mobicom 2000; Chair of the ACM SIG Multimedia; General Chair of ACM Multimedia93 and ACM SIGCOMM88; and Program Chair of IEEE MULTIMEDIA 92, ACM SIGCOMM87, and ACM SIGCOMM86. He has served in the IEEE Internet Technology Award Committee, the IEEE Richard W. Hamming Medal Committee, and the National Research Council Panel on Digitization and Communications Science of the Army Research Laboratory Technical Assessment Board. He has been on the editorial boards of the IEEE/ACM Transactions on Networking, the Multimedia Systems Journal, and the Journal of High Speed Networks. He is a Fellow of the IEEE and is listed in Marquis Who’s Who in America and Who’s Who in The World. He is the co-recipient of Best Paper Awards at SPECTS 2007, IFIP Networking 2007, and IEEE MASS 2005 conferences, and the Best Student Paper Award of the 1998 IEEE International Conference on Systems, Man, and Cybernetics. He received the SRI International Exceptional-Achievement Award in 1985 for his work on multimedia communication and in 1989 for his work on routing algorithms.