Cross-layer Design for Efficient Resource Utilization in WiMedia UWB-based WPANs

RAED AL-ZUBI and MARWAN KRUNZ
Department of Electrical and Computer Engineering. University of Arizona.

Ultra-wideband (UWB) communications has emerged as a promising technology for high data rate wireless personal area networks (WPANs). In this paper, we address two key issues that impact the performance of a multi-hop UWB-based WPAN: throughput and transmission range. Arbitrary selection of routes in such a network may result in reserving an unnecessarily long channel time, and hence low network throughput and high blocking rate for prospective reservations. To remedy this situation, we propose a novel cross-layer resource allocation design. At the core of this design is a routing technique (called RTERU) that uses the allocated channel time as a routing metric. RTERU exploits the dependence of this metric on the multiple-rate capability of an UWB system. We show that selecting the route that consumes the minimum channel time while satisfying a target packet delivery probability over the selected route is an NP-hard problem. Accordingly, RTERU resorts to approximate path selection algorithms (implemented proactively and reactively) to find near-optimal solutions at reasonable computational/communication overhead. We further enhance the performance of RTERU by integrating into its design a packet overhearing capability. Simulations are used to demonstrate the performance of our proposed solutions.

Categories and Subject Descriptors: I.6.0 [**Simulation and Modeling**]: General; G.1.6 [**Numerical Analysis**]: Optimization—*Constrained optimization*; G.2.2 [**Discrete Mathematics**]: Graph Theory—*Graph algorithms*

General Terms: Algorithms, Design, Performance

Additional Key Words and Phrases: Cross-layer Design, OFDM-based UWB, Packet Overhearing, Routing, Slots Reservation

1. INTRODUCTION

UWB has recently emerged as an attractive technology for short range, high data rate wireless communications. Following the FCC's First Report and Order that permitted the deployment of UWB devices [FCC 2002], efforts have been made to exploit the unique features of UWB in various applications, including wireless personal area networks (WPANs), wireless sensor networks, imaging and radar systems, and precision location tracking systems. Several architectures for UWB-based WPANs have been proposed. One widely popular proposal is based on multi-band OFDM. Industry advocates of this system formed an organization called the

Authors' address: R. Al-Zubi and M. Krunz, Department of Electrical and Computer Engineering, University of Arizona, 1230 East Speedway Blvd, Tucson, AZ 85721-0104, {alzubi, krunz}@ece.arizona.edu.

Multi-band OFDM Alliance (MBOA) [Multi-Band OFDM Alliance 2004], which eventually evolved into a large industrial alliance known as WiMedia. WiMedia's UWB specifications have been adopted by the European Computer Manufacturers Association (ECMA) as a basis for an OFDM-based UWB standard [European Computer Manufacturers Association 2008]. This standard, called ECMA-368, is used as a basis for the work in the underlying paper.

ECMA-368 defines 8 data rates, from 53.3 Mbps to 480 Mbps. It uses a TDMA channel access structure, whereby time is divided into 65.536 msec intervals, called *superframes* (see Figure 1). Each superframe is further divided into 256 medium access slots (MASs), which form two parts of the superframe: a beaconing period (BP) and a data transfer period (DTP). The BP is used for control and coordination purposes (e.g., bandwidth reservation, synchronization, device discovery). Each node is required to send a beacon at the beginning of each superframe (i.e., during BP) and must listen to the beacons of its neighbors. Nodes use information elements (IEs) in their beacons to exchange information. Data transmission is done in the DTP using one of two modes: random access and time-based reservation. The latter mode, known as the distributed reservation protocol (DRP), is particularly suitable for real-time streaming applications. According to DRP, devices that want to communicate with each other reserve their MASs from DTP MASs that are not already reserved by neighboring nodes (see [European Computer Manufacturers Association 2008] and [Batra et al. 2004] for more details).

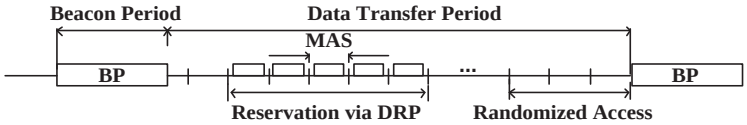


Fig. 1. Superframe structure in ECMA-368.

One of the challenges in UWB-based WPANs is how to maintain high throughput in dense topologies. Most works that considered this problem (e.g., [Liu et al. 2008], [Radunovic and Boudec 2004], [Cai et al. 2008]) assume a fixed exclusive region around the receiver of an active link. No other transmissions can concurrently take place inside this region. The transmission rate over an active link is, thus, adapted according to the interference from nodes outside the exclusion region. One of the key issues in these works is how to determine the optimal size of the exclusion region for an arbitrary network topology.

Another challenge in UWB systems is the short transmission range. To overcome this problem, researchers have recently considered the possibility of multi-hop transmissions, and consequently investigated routing techniques for multi-hop UWB-based WPANs [Gao and Daut 2006]. Research in this area is still in its infancy. Existing routing protocols for UWB (e.g., [Gao and Daut 2006] [Abdrabou and Zhuang 2006]) are based on hopcount and distance metrics. They do not account for resource utilization and throughput enhancement issues.

In the context of a multi-hop UWB WPAN, route selection, rate assignment, and session throughput are all inter-related issues. To see that, consider the example in Figure 2. In this example, there are two possible paths between devices A and

C: $A \rightarrow C$ and $A \rightarrow B \rightarrow C$. Suppose that the traffic demand is 10 Mbps and the packet size is 1 Kbyte. Given these values, the sum of MASs that should be reserved along $A \rightarrow C$ and $A \rightarrow B \rightarrow C$ are 55 and 43, respectively (an explanation of how these numbers are obtained is given in Section 3.1). In this example, the 2-hop path $A \rightarrow B \rightarrow C$ consumes 12 fewer MASs than the direct path $A \rightarrow C$, and is hence a better path. Note that because of device half-duplicity, when path $A \rightarrow B \rightarrow C$ is selected, MASs allocated along the segment $A \rightarrow B$ cannot overlap with those allocated to $B \rightarrow C$.

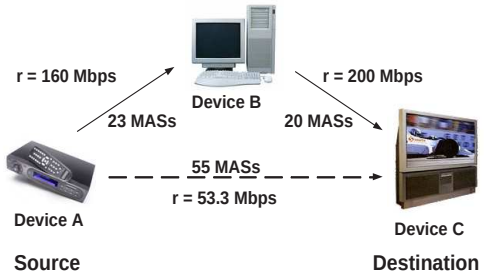


Fig. 2. Example illustrating the inter-dependence of route selection, rate assignment, and session throughput in multi-hop UWB networks.

Motivated by the above, in this paper we propose a new cross-layer design for efficient resource utilization in WiMedia UWB-based WPANs. The structure of this design is shown in Figure 3. At the core of this design is a routing technique for efficient resource utilization (RTERU). RTERU exploits the fact that the higher the link transmission rate, the smaller the number of MASs to be allocated, and hence the higher the number of free MASs that are available for other prospective reservations. At the same time, the transmission rate also impacts the packet error rate (PER) over a link: for a given link quality (i.e., SNR), the higher the transmission rate, the higher the PER. This places an upper limit on the transmission rate that can be used to support a target PER. RTERU is designed to consider both metrics (i.e., number of MASs and PER). As shown in Figure 3, RTERU involves interactions between different layers. It takes as input the traffic demand γ (in bps) and a target end-to-end PER ε, which are provided by the application layer. It also uses link-quality information, represented by PER-vs.-SNR curves, from the link/physical layer. The outputs of RTERU are an end-to-end path between a source and a destination, which is passed to the network layer, the rate assignment along the selected path, which is passed to the link/physical layer, and the number of required MASs over each link along the selected path, which is passed to the DRP protocol in the MAC layer. To further improve its performance, RTERU optionally exploits an important feature of broadcast communications, namely packet overhearing (i.e., packets transmitted from a source to a destination are likely to be overheard by non-intended nodes). This feature is exploited twice. First, during the session admission phase for the purpose of reducing the estimated end-to-end PER. This essentially gives a spatial diversity gain, which is achieved by supporting multiple paths to the destination. Secondly, packet overhearing is exploited during

actual data transmission, where the IDs of overheard packets are announced so as to avoid relaying these packets by subsequent nodes along the path. MASs that are already allocated for relaying these overheard packets can now be released and used for other transmissions.

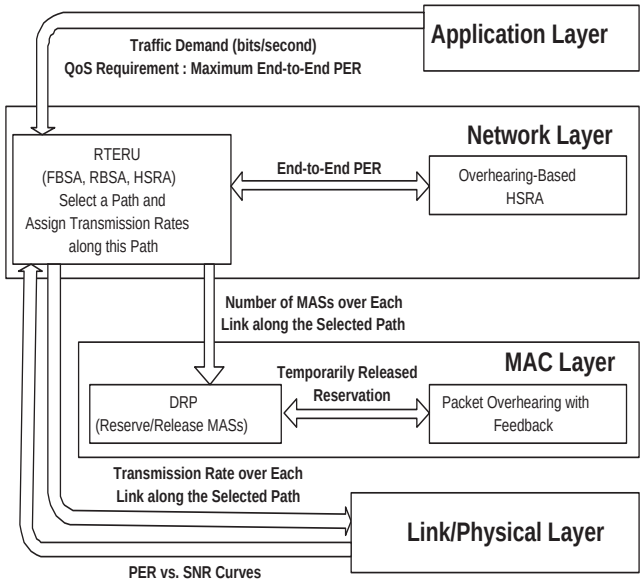


Fig. 3. Framework for cross-layer design in WiMedia UWB-based WPANs.

For each source-destination pair, RTERU aims at selecting a path and a rate assignment that minimize the *sum* of the allocated MASs while at the same time satisfying the target end-to-end PER (ε). We show that this problem is NP-hard. Therefore, RTERU relies on a two-phase approximation. The first phase (path-search) outputs a set of candidate paths that is likely to contain a near-optimal path. By design, the size of this set is small, so it can be inspected with small computational overhead. Examining this set is done in the second phase (rate-assignment phase), in which the transmission rates over each candidate path are assigned such that the total number of required MASs along the path is minimized while simultaneously satisfying the constraint ε. Finally, the best feasible path in the candidate set is selected.

For the path-search phase, we propose two approximate algorithms: flooding-based search algorithm (FBSA) and rate-based search algorithm (RBSA). FBSA is a reactive approach, whereby the source node floods a route request (RREQ) throughout the network. Upon receiving the first RREQ, the destination node waits for a certain number of superframes in anticipation of receiving additional RREQs. Because of the synchronized nature of the network (each RREQ propagates one hop per superframe), RREQs that arrive in different superframes carry information about different-length paths. From these paths, the destination selects its candidate set.

RBSA is a proactive algorithm. For each transmission rate, the source constructs a graph, where the weight of each link in this graph is a function of the link PER and the number of MASs that required to satisfy the traffic demand γ. The source executes Dijkstra's algorithm to find the shortest path with respect to the link weight. The output of RBSA is also a set of candidate paths.

In the rate assignment phase, the source node (in RBSA) or the destination node (in FBSA) performs rate assignment and the selection of a near-optimal feasible path. We show that this phase is equivalent to the *multiple-choice knapsack problem (MCKP)*, which is NP-hard. Therefore, we propose a heuristic solution, called HSRA, for the rate assignment problem.

The rest of the paper is organized as follows. Section 2 presents related work. The problem setup and formulation are provided in Section 3. Section 4 presents RTERU, and Section 5 presents our proposed technique to exploit packet overhearing for the purpose of reducing the end-to-end PER. Section 6 presents the second approach for exploiting packet overhearing during data transmission. In Section 7, we use simulations to evaluate the performance of our proposed cross-layer designs. Concluding remarks are drawn in Section 8.

2. RELATED WORK

Few works have addressed routing in multi-hop UWB-based WPANs. In [Gao and Daut 2006], the authors mainly focused on improving the performance of a common geographic routing approach, which is based on Euclidean distance. According to this approach, a node forwards packets to the neighbor that is geographically closest to the destination. The authors in [Gao and Daut 2006] noted that even if the above simple strategy leads to a min-hop path, it may result in selecting bad links (i.e., links with high PERs). Therefore, they proposed considering the link quality (i.e., PER) in choosing the next hop, so as to achieve a tradeoff between hopcount and link quality. For transmission rates 200 and 480 Mbps, the authors experimentally found the best transmission radii that achieve the tradeoff between hopcount and link quality. They concluded that the position-based greedy routing strategy with carefully selected transmission radius works well in multi-hop UWB-based WPANs. However, they did not provide a specific mechanism for selecting the transmission radius for other transmission rates, and they did not address the rate assignment issue (they assumed a fixed transmission rate over a path, and did not provide a procedure to determine this rate). They also did not account for the relationship among the number of reserved MASs, the end-to-end PER, and the transmission rate. As shown in our work, considering such relationship has a significant impact on the overall network performance.

In [Abdrabou and Zhuang 2006], the authors proposed a position-based QoS routing protocol for UWB networks. Their protocol is based on the greedy perimeter stateless routing (GPSR). Different QoS parameters were considered, including the rate demand, PER, and end-to-end delay. Their simulation results showed that their protocol works well for multi-hop UWB networks. However, the authors did not account for the relationship between the number of reserved MASs, the end-to-end PER, and the transmission rate. Furthermore, none of the aforementioned protocols exploited packet overhearing. Although packet overhearing was used in

several previous works (e.g., [Hsu et al. 2006], [Biswas and Morris 2005], [Katti et al. 2008]), none of these works accounted for this feature during rate assignment or utilized the reserved channel time of overheard packets. Specifically, in [Hsu et al. 2006], packet overhearing was employed to reduce the overhead of route discovery, where routing information may be obtained in advance by analyzing overheard packets. The authors in [Biswas and Morris 2005] exploited packet overhearing to improve the performance of an opportunistic routing protocol. According to this protocol, one of the nodes that overheard the transmitted packet is chosen to forward the packet. The work in [Katti et al. 2008] is based on the fact that even when no node receives (overhears) a packet correctly, any given bit is likely to be received (overheard) correctly by some nodes. Nodes that overhear a transmitted packet are allowed to forward parts of this packet, allowing the final destination to recover the original packet.

3. PROBLEM SETUP AND FORMULATION

3.1 Preliminaries

As mentioned before, RTERU exploits the dependence among the multi-rate capability of an OFDM-based UWB system, the number of required MASs for a reservation, and the PER. Therefore, it is worth clarifying such dependence. A given traffic demand γ (in bps) must first be packetized before being transported. let κ be the payload portion of a packet (in bytes), ν the number of MASs that should be reserved in a superframe, and ξ the number of packets corresponding to the demand γ that should be sent per superframe. Then,

$$\xi = \left\lceil \frac{\gamma \lambda_{sp}}{8\kappa} \right\rceil \tag{1}$$

where $\lambda_{sp} = 65.536$ msec is the superframe interval. Let λ_f be the amount of time needed to transmit these ξ packets. Then,

$$\nu = \left\lceil \frac{\lambda_f}{\lambda_{ms}} \right\rceil \quad \text{slots} \tag{2}$$

where $\lambda_{ms} = 256$ μsec is the MAS duration. Next, we explain how λ_f is impacted by the transmission rate r. Note that $\lambda_f = \xi(\lambda_p + \text{SIPS})$ seconds, where SIPS $= 10$ μsec is the short inter-packet spacing and λ_p is a packet transmission time. This λ_p is given by $\lambda_p = \lambda_m + \lambda_h + \lambda_u$ seconds, where $\lambda_m = 5.625$ μsec is the preamble interval (used for synchronization, carrier-offset recovery, and channel estimation), $\lambda_h = 3.75$ μsec is the header interval, and λ_u is the transmission time of a PHY service data unit. Note that λ_m and λ_h are fixed, and only λ_u varies with r. As given in the ECMA-368 standard, λ_u can be expressed as:

$$\lambda_u = 6 \times \left\lceil \frac{8\kappa + 38}{\varrho} \right\rceil \times \lambda_{sy} \quad \text{sec} \tag{3}$$

where $\lambda_{sy} = 0.3125$ μsec is the symbol interval and ϱ is the number of information bits per 6 OFDM symbols, which depends on r, as shown in Table I.

The above straightforward analysis allows us to express ν as a function of r for various values of γ and κ, as shown in Figure 4. It can be seen that as γ increases, ν becomes more sensitive to r. Furthermore, for given γ and r (e.g., $\gamma = 10$ Mbps

Table I. Rate-Dependent Modulation and Coding Parameters in ECMA-368.

r Mbps	Modulation type	Coding rate	ϱ
53.3	QPSK	1/3	100
80	QPSK	1/2	150
106.7	QPSK	1/3	200
160	QPSK	1/2	300
200	QPSK	5/8	375
320	DCM	1/2	600
400	DCM	5/8	750
480	DCM	3/4	900

and $r = 480$ Mbps), ν expectedly decreases with κ, but in a sub-linear fashion. This can be explained by noting the nonlinear relationship between λ_f and κ.

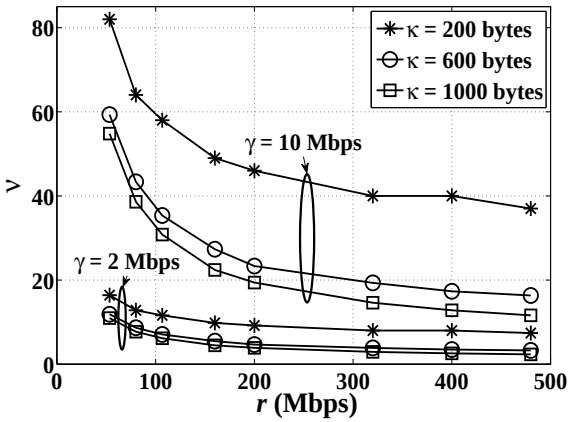


Fig. 4. ν versus r at different γ and κ.

Next, we explore the relationship between r and the PER. Devices can estimate the probability of correct packet/bit delivery based on the received SNR or using historical data of the number of packets or bits sent and received over a link. This information can then be used to obtain the PER-vs.-SNR curves [Stojmenovic et al. 2005]. In [Kuruvila et al. 2004], to reduce the computation time, the authors derived a reasonably accurate yet simple approximation of such curves. In our work, we assume that the PER-vs.-SNR curves are provided by the physical layer. For our simulation purposes, to generate these curves, we extract the BER from the BER-vs.-SNR curves given in [Nowak et al. 2008] (one curve for each transmission rate), and calculate the PER as:

$$\text{PER} = 1 - (1 - \text{ BER})^{8\kappa} \tag{4}$$

Note that Equation 4 assumes independence between bits in a packet.

3.2 Problem Formulation

Consider an UWB WPAN. Its topology is represented by a graph $\mathcal{G}(\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ is the set of nodes and $\mathcal{L}$ is the set of links. A link ℓ exists between two nodes

if these nodes can communicate directly at the lowest transmission rate r_1. Given the rate demand γ of a requested reservation and given a set of transmission rates $\mathcal{R} = \{r_1, r_2, \ldots, r_M\}$, where $M = |\mathcal{R}|$ and $r_1 \leq r_2 \leq \ldots \leq r_M$, each link $\ell \in \mathcal{L}$ is associated with two sets of parameters:

— $n_\ell(r)$: minimum number of time slots that are required to establish the requested reservation over link ℓ at a transmission rate r, $\forall\ r \in \mathcal{R}$.

— $e_\ell(r, \mathrm{SNR}_\ell)$: PER over link ℓ when this link is operated at a transmission rate r and the received SNR is SNR_ℓ, $\forall\ r \in \mathcal{R}$.

Let $n(p) \stackrel{\text{def}}{=} \sum_{\ell \in p} n_\ell(r)$ be the sum of required MASs along a path p. Given a source S and a destination D, a PER constraint ε, and γ, the problem is to find a path p^* and a rate assignment along p^* such that:

(i) $e(p^*) \leq \varepsilon$, where $e(p^*) = 1 - \prod_{\ell \in p^*}(1 - e_\ell(r, \mathrm{SNR}_\ell))$ is the end-to-end PER over path p^*.

(ii) $n(p^*)$ is minimized over all feasible paths satisfying (i).

In this paper, we use the terms *feasibility condition* and *optimization metric* to refer to (i) and (ii), respectively. Constraint (i) is a basic requirement for streaming applications. Note that other QoS requirements (i.e., rate demand and end-to-end delay jitter requirements) are being met. Specifically, the rate demand is met by reserving the required MASs in each superframe. Because of the TDMA structure, the end-to-end delay jitter (i.e., the maximum allowable deviation from the nominal inter-packet time) experienced by a flow over a route cannot exceed the duration of one superframe (~ 65 msec), which meets the jitter requirement of typical streaming applications.

Remark 3.1. In Constraint (i), we assume that the PERs over various links are independent. This assumption can be justified as follows. The PER over a link is determined by the channel quality at the receiver of that link, which depends on the SNR and transmission rate. In other words, packet errors depend on the receiver location. Two different receivers (including those of adjacent links along a path) experience independent fading, and hence their PERs are independent. Therefore, at a node, the PERs over incoming and outgoing links are independent.

For the optimization metric, we try to minimize the total sum of slots along a path, despite possible MASs reuse along such a path. The reason is explained in Figure 5. In this figure, there is a path p between nodes A and E (i.e., $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$). For a given rate assignment, the allocated MASs over each link of this path are shown in the figure. The link $D \rightarrow E$ can reserve the same MASs reserved by $A \rightarrow B$. However, it is misleading to set $n(p) = 2 + 5 + 3 = 10$ slots, because link $D \rightarrow E$ actually prevents other links (e.g., $F \rightarrow G$) from using the same two slots that are reserved for $D \rightarrow E$. In order to consider the actual impact of path reservation, the optimization metric should be calculated without considering slot reuse (i.e., $n(p) = 12$ slots). In other words, despite the potential for slot reuse, from a network-wide perspective reused slots do not come for free, as they potentially impact the slot assignment in other contention regions.

Note that the metric used in the feasibility condition is a multiplicative metric. However, it can be transformed into an additive metric through a logarithmic

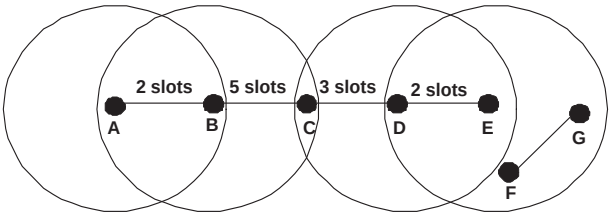


Fig. 5. Example illustrating why slot reuse is not considered in the optimization metric.

transformation: $\sum_{\ell \in p^*} \log(\frac{1}{1-e_\ell(r,\mathrm{SNR}_\ell)}) \leq \log(\frac{1}{1-\varepsilon})$. With this transformation and under a given transmission rate, the problem reduces to minimizing an additive metric (i.e., number of slots) subject to a constraint on another additive metric. This is exactly the *restricted shortest path* (RSP) problem, which is known to be NP-hard [Ahuja et al. 1993]. With the inclusion of a rate assignment per link, our problem is actually more general than RSP, so it is also NP-hard. Therefore, approximate solutions with suboptimal performance are needed. Such solutions must exhibit reasonable computational/communication overhead.

4. RTERU

RTERU consists of two main phases: path search and rate assignment. In the first phase, a set of candidate paths is determined. The size of this set has to be small enough such that its paths can be examined quickly during the second phase. In the second phase, transmission rates are assigned over various links along a given path such that the total sum of slots along the path is minimized while at the same time the target end-to-end PER is satisfied. Finally, a near-optimal path p^* is selected from the candidate set.

4.1 Path-Search Phase

An exact solution to our problem requires examining every path between S and D. Because the number of paths grows exponentially with the size of the network, this method is not practical. In the path-search phase, we aim at finding a near-optimal solution at a low computational cost. To this end, we propose two path-search algorithms: flooding-based search algorithm (FBSA) and rate-based search algorithm (RBSA). FBSA is intended for reactive routing (similar to AODV), whereas RBSA is intended for a proactive implementation.

4.1.1 *FBSA.* Node S starts FBSA by broadcasting an RREQ as an IE in a beacon. The RREQ includes the source ID, the destination ID, ε, γ, and the packet size κ. Every intermediate node U that receives the RREQ should first update the route sequence of the received path by adding its ID, and then include the SNR value over the last link of the updated sequence. U then rebroadcasts the updated RREQ via a beacon during the beacon period of the upcoming superframe. If there exists a direct (1-hop) path between S and U, U first checks if the SNR over at least one link of the indirect path between S and D is less than that of the direct path. If so, U discards the received RREQ. The rational behind this mechanism can be clarified by the following observation.

Observation 4.1. Let p be the direct path from S to D, and let q be a multi-hop path from S to D. If the SNR over at least one link ℓ in q is less than the SNR over p, then the sum of the required MASs along q must be greater than the number of MASs along p. In this case, there is no advantage in rebroadcasting the received RREQ at the intermediate node U.

Further reduction in the flooding overhead can be achieved by preventing U from rebroadcasting the RREQ if the RREQ has previously U.

Following the receipt of the first RREQ, node D waits for up to K superframes to accumulate more RREQs, where K is a controllable parameter. RREQs received during the mth superframe, $m = 1, \ldots, K$, will contain information about m-hop paths from S to D. Let ψ_m be the set that contains such m-hop paths. In the worst case, the size of ψ_m grows exponentially with $|\mathcal{N}|$. If the topology is dense and K is large, node D will have to choose from a large set of paths. To reduce the computational cost, for $m = 1, \ldots, K$, D randomly selects a smaller subset $\omega_m \subset \psi_m$ from which it determines the best feasible path g_m^* among all $|\omega_m|$ paths. The computation of g_m^* is done during the rate assignment phase. This g_m^* is added to the candidate set $\mathcal{J}$.

After computing ω_m, node D needs to decide whether to terminate the algorithm or continue to wait for the $(m+1)$th superframe. If $\frac{n(g_m^*)}{n_\ell(r_M)} - m > 0$, then D continues to wait. Otherwise, $\frac{n(g_m^*)}{n_\ell(r_M)} - m = 0$, and D terminates the algorithm. This criterion can be clarified by the following observation.

Observation 4.2. Let p be a path between S and D. For a given load demand γ, the number of allocated MASs along every link $\ell \in p$ is $\geq n_\ell(r_M)$. Note that $n_{\ell_1}(r_M) = n_{\ell_2}(r_M) = \tilde{n}(r_M)$ for any two links ℓ_1, $\ell_2 \in p$. Then, p consists of at most $\frac{n(p)}{\tilde{n}(r_M)}$ links. Furthermore, if $\exists$ another path q between S and D such that $n(q) \leq n(p)$, then q also consists of at most $\frac{n(p)}{\tilde{n}(r_M)}$ hops.

During superframe m, node D selects g_m^* as the best path in the set ω_m. Consider path g_{m+1}^*, which is selected in superframe $m+1$. According to Observation 2, if $n(g_{m+1}^*) \leq n(g_m^*)$, then the hopcount of g_{m+1}^* cannot be greater than $\frac{n(g_m^*)}{\tilde{n}(r_M)}$ hops. Therefore, during superframe m, node D knows that m and $\frac{n(g_m^*)}{\tilde{n}(r_M)}$ are the current and maximum hopcounts of g_{m+1}^*, respectively. Then, if $\frac{n(g_m^*)}{\tilde{n}(r_M)} - m > 0$, D will continue to wait for superframe $m+1$. Otherwise, $\frac{n(g_m^*)}{\tilde{n}(r_M)} - m = 0$, and D terminates the algorithm.

Eventually, D terminates the algorithm , after waiting for up to K superframes, and selects the optimal path p^* from $\mathcal{J}$. This selection is sent back to the source S. A pseudo-code of FBSA is shown in Algorithm 1.

Complexity: FBSA has an execution complexity of $\mathcal{O}(|\omega_m| K C_{\text{HSRA}})$, where C_{HSRA} is the computational complexity of the rate-assignment phase. Note that FBSA is distributively executed over K superframes.

Result 4.3. (Bounds on the Path Returned by FBSA): For a source-destination pair, let $h_{\min}$ and h_{opt} be the hopcounts of the min-hop and optimal paths, respectively. Also, let N_{opt} and N_{FBSA} be the sums of required MASs along the

optimal path and the one returned by FBSA, respectively. Then, $N_{\text{opt}} \leq N_{\text{FBSA}} \leq \frac{h_{\min}}{h_{\text{opt}}} \frac{n_\ell(r_1)}{n_\ell(r_M)} N_{\text{opt}}$.

PROOF. Assume the destination waits for K superframes. Then, the destination selects the path in $\mathcal{J}$ that has the minimum number of MASs N_{FBSA} (i.e., $N_{\text{FBSA}} = \min\{n(g_m^*), m = 1, 2, \ldots, K\}$). We know that $h_{\text{opt}} n_\ell(r_M) \leq N_{\text{opt}} \leq h_{\text{opt}} n_\ell(r_1)$, $N_{\text{FBSA}} \leq n(g_1^*) \leq h_{\min} n_\ell(r_1)$, and $N_{\text{opt}} \leq N_{\text{FBSA}}$. Therefore, $h_{\text{opt}} n_\ell(r_M) \leq N_{\text{opt}} \leq N_{\text{FBSA}} \leq h_{\min} n_\ell(r_1)$. This inequality can be rewritten as $h_{\text{opt}} n_\ell(r_M) \leq N_{\text{opt}} \leq N_{\text{FBSA}} \leq \frac{h_{\min} n_\ell(r_1)}{N_{\text{opt}}} N_{\text{opt}}$. Also, if we replace $\frac{h_{\min} n_\ell(r_1)}{N_{\text{opt}}}$ with $\frac{h_{\min} n_\ell(r_1)}{h_{\text{opt}} n_\ell(r_M)}$, then the inequality is still valid. Therefore, $N_{\text{opt}} \leq N_{\text{FBSA}} \leq \frac{h_{\min}}{h_{\text{opt}}} \frac{n_\ell(r_1)}{n_\ell(r_M)} N_{\text{opt}}$. □

Result 4.3 shows that FBSA is a provable approximation, i.e., there is a quantifiable performance gap between this approximate solution and the optimal one.

Algorithm 1 FBSA (executed at the destination D)

Input:
ψ_m % set of paths received at D in the mth superframe

Output:
The feasible path p^* that requires the minimum number of MASs, or failure if no feasible path can be found

Initialization:
$\mathcal{J} = \emptyset$ % set that will include candidate feasible paths from which p^* will be % selected
decision = continue % wait for another superframe

while (decision == continue)
- Randomly select a subset ω_m from ψ_m, where the size of ω_m is a design parameter
- Using Algorithm 3, find the feasible path g_m^* that requires the minimum number of MASs among the paths in ω_m
- Add g_m^* to $\mathcal{J}$
- **if** $\frac{n(g_m^*)}{n_\ell(r_M)} - m > 0$
  decision = continue
  Wait one more superframe
  $m = m + 1$
  Get new ψ_m during the new superframe
  else
  decision = stop
  end

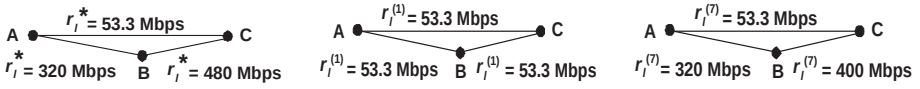
end

Among the paths in $\mathcal{J}$, select path p^* that requires the minimum number of MASs

if p^* is found
return p^*
else
return "no feasible path found"
end

4.1.2 *RBSA.* In contrast to the reactive approach used in FBSA, RBSA follows a proactive approach, in which nodes exchange link-state information using beacons. The source node S initiates RBSA by constructing a set of graphs $\Omega = \{G_1, G_2, \ldots, G_M\}$, where M is the number of transmission rates. Each graph $G_i \in \Omega$, $i = 1, 2, \ldots, M$, consists of the same set of nodes $\mathcal{N}$ and the set of links $\mathcal{L}$. The transmission rates over the links in G_i are not allowed to exceed r_i. For $\ell \in \mathcal{L}$, let r_ℓ^* be the maximum transmission rate that can be used over link ℓ while satisfying the target PER (ε). Then, the transmission rate over link $\ell \in \mathcal{L}$ in G_i is set to $r_\ell^{(i)} = \min\{r_i, r_\ell^*\}$. Figure 6 illustrates the process of assigning link rates in G_i. Then, each link ℓ in G_i is weighted by $w_\ell^{(i)}$, which is given by the following Lagrangian function (with Lagrangian multiplier equals 1):

$$w_\ell^{(i)} = n_\ell(r_\ell^{(i)}) + \log(\frac{1}{1 - e_\ell(r_\ell^{(i)}, \mathrm{SNR}_\ell)}). \tag{5}$$



(a) Original graph.

(b) Graph G_1 ($r_1 = 53.3$ Mbps).

(c) Graph G_7 ($r_7 = 400$ Mbps).

Fig. 6. Example illustrating the process of constructing the set of graphs Ω in RBSA. Note that $r_\ell^{(i)} = \min\{r_i, r_\ell^*\}$.

Recall that we use the logarithmic function to transform the multiplicative metric in the feasibility condition to an additive metric.

Now, for each graph G_i, Dijkstra's algorithm is used to find the shortest $S \rightarrow D$ path q_i^* with respect to $w_\ell^{(i)}$. The set $\mathcal{P} = \{q_1^*, q_2^*, \ldots, q_M^*\}$ is then examined during the rate-assignment phase. Finally, S selects a near-optimal path p^* from this set. A pseudocode for RBSA is shown in Algorithm 2.

The rationale behind RBSA can be explained by considering the two extreme cases: G_1 and G_M. For G_1, the first term $n_\ell(r_\ell^{(1)})$ in the right-hand side of (5) is the largest among $\{n_\ell(r_\ell^{(i)}) : i = 1, \ldots, M\}$ for all links ℓ. Furthermore, all links ℓ in G_1 will have the same $n_\ell(r_\ell^{(1)})$ value. Similarly, the second term $\log(\frac{1}{1-e_\ell(r_\ell^{(1)}, \mathrm{SNR}_\ell)})$ is the smallest among $\{\log(\frac{1}{1-e_\ell(r_\ell^{(i)}, \mathrm{SNR}_\ell)}) : i = 1, \ldots, M\}$, this value varies from one link ℓ to another in the graph G_1, since it depends on SNR_ℓ. Note that when Dijkstra's algorithm is executed, $n_\ell(r_\ell^{(1)})$ is the dominant factor when selecting a path among paths of different hopcounts. This is because $n_\ell(r_\ell^{(1)}) \gg \log(\frac{1}{1-e_\ell(r_\ell^{(1)}, \mathrm{SNR}_\ell)})$. However, in case of paths that have the same hopcount, the second term becomes dominant. Accordingly, q_1^* is the most feasible min-hop path (i.e, the min-hop path that has the lowest end-to-end PER).

Now, consider the case of G_M. The first term $n_\ell(r_\ell^{(M)})$ in the right-hand side of (5) is the smallest among $\{n_\ell(r_\ell^{(i)}) : i = 1, \ldots, M\}$ for all links ℓ. Similarly, the

Algorithm 2 RBSA (executed at the source S)

Input:

· $\mathcal{G}(\mathcal{N}, \mathcal{L})$
· $\mathcal{R} = \{r_1, r_2, \ldots, r_M\}$ % set of transmission rates
· S and D % source and destination nodes
· $n_\ell(r)$ and $e_\ell(r, \text{SNR}_\ell)$ for all $\ell \in \mathcal{L}$ and $r \in \mathcal{R}$ % the number of MASs and the PER over link ℓ at transmission rate r, respectively

Output:

The feasible path p^* between S and D that requires the minimum number of MASs or failure if no feasible path can be found

Initialization:

$\mathcal{P} = \emptyset$ % set that will include candidate feasible paths from which p^* will % be selected

for all $r_i \in \mathcal{R}$
  for all $\ell \in \mathcal{L}$
    • Compute $r_\ell^{(i)} = \min\{r_i, r_\ell^*\}$ % the transmission rate that will be assigned % over each link ℓ in G_i
    • Use $r_\ell^{(i)}$ in equation (5) to compute $w_\ell^{(i)}$
    • Assign $w_\ell^{(i)}$ to link ℓ in graph G_i
  end
    • For graph G_i, use Dijkstra's algorithm to find the shortest path q_i^* with respect to $w_\ell^{(i)}$ between S and D
    • Record q_i^* in $\mathcal{P}$
end

Among the paths in $\mathcal{P}$, select path p^* that requires the minimum number of MASs

if p^* is found
  return p^*
else
  return "no feasible path found"
end

second term $\log(\frac{1}{1-e_\ell(r_\ell^{(M)},\text{SNR}_\ell)})$ is the largest among $\{\log(\frac{1}{1-e_\ell(r_\ell^{(i)},\text{SNR}_\ell)}) : i = 1, \ldots, M\}$. Therefore, q_M^* is the optimal path in terms of the number of MASs (if q_M^* is feasible, then it is the optimal solution). In that sense, as r_i increases, the probability of satisfying the feasibility condition decreases while the probability of satisfying the optimization metric increases. RBSA tries to find a compromise between the two extreme cases.

Complexity: RBSA has a computational complexity of $\mathcal{O}(M(|\mathcal{N}|\log(|\mathcal{N}|) + |\mathcal{L}| + C_{\text{HSRA}}))$, where $|\mathcal{N}|\log(|\mathcal{N}|) + |\mathcal{L}|$ is the computational complexity of Dijkstra's algorithm and C_{HSRA} is the computational complexity of the rate-assignment phase.

Observation 4.4. If q_i^* is infeasible, then q_j^* is also infeasible for $i \leq j$, since $e(q_i^*) \leq e(q_j^*)$.

The above observation can be used to further reduce the computational complexity of RBSA by allowing it to sequentially determine q_i^*, $i = 1, 2, \ldots, M$.

Bounds on the Path Returned by RBSA: Following similar analysis to the

one used in determining the bounds for FBSA (see Section 4.1.1), it can be shown that $N_{\text{opt}} \leq N_{\text{RBSA}} \leq \frac{h_{\min}}{h_{\text{opt}}} \frac{n_\ell(r_1)}{n_\ell(r_M)} N_{\text{opt}}$, where N_{RBSA} is the number of MASs along the path returned by RBSA (other parameters are defined in Section 4.1.1).

4.2 Rate Assignment Phase

Definition 4.5. Rate-Assignment Problem: Consider an h-hop path between S and D. Let R_j be the set of transmission rates that can be assigned to the jth link on that path. Each rate $r_x \in R_j$ corresponds to an optimization metric $n_j(r_x)$ and a feasibility metric $\log(\frac{1}{1-e_j(r_x,\text{SNR}_j)})$. The rate-assignment problem is to choose one transmission rate for each link such that the sum of the values of optimization metric is minimized while the sum of feasibility metric does not exceed a predefined value.

The rate-assignment problem is related to the well-known Multiple-Choice Knapsack problem (MCKP).

Definition 4.6. Multiple-Choice Knapsack Problem (MCKP) [Pisinger 1995]: Given y classes $C_1, C_2, \ldots, C_y$ of items to pack a knapsack. Each item $i \in C_j$ has a profit f_{ij} and a weight w_{ij}. The problem is to choose one item from each class such that the profit sum is maximized while the weight sum does not exceed a predefined value.

If we let the classes of items in MCKP be the sets of transmission rates, the knapsack in MCKP be a path, and the profit and weight of each item in MCKP be the optimization and feasibility metrics of each transmission rate, respectively, then an instance of MCKP can be converted into an equivalent instance of the rate-assignment problem.

Because MCKP is NP-hard [Dudzinski and Walukiewicz 1987], the rate-assignment problem is also NP-hard. Accordingly, we propose a heuristic solution for the rate-assignment problem (HSRA). HSRA starts by assigning the highest possible transmission rate over each hop of a given path p in order to minimize the total number of MASs along p. However, a high transmission rate over a link means a high PER, which may lead to an end-to-end PER larger than ε. Therefore, until the feasibility condition is satisfied, HSRA gradually reduces the end-to-end PER such that the corresponding increase in the total number of MASs along the path is maintained as small as possible.

The operational details of HSRA are as follows. As mentioned above, the first step in HSRA is to select r_M for each link in p. If this rate assignment satisfies the feasibility condition, then it is optimal. Otherwise, the algorithm examines h different rate assignments, each being created by replacing r_M over one link by r_{M-1}. For example, if the first rate assignment over a 3-hop path is $\{r_8, r_8, r_8\}$ and it is infeasible assignment, then HSRA will examine three other rate assignments: $\{r_7, r_8, r_8\}$, $\{r_8, r_7, r_8\}$, and $\{r_8, r_8, r_7\}$. The best feasible one among them will be selected. If none of them is feasible, the second step is initiated with a new rate assignment that is created by replacing r_M by r_{M-1} over only the link that has the highest PER under the first rate assignment. For example, under the first rate assignment $\{r_8, r_8, r_8\}$, if the third link has the highest PER, then the second step will be initiated with $\{r_8, r_8, r_7\}$. In other words, the second step of HSRA is

to reduce the transmission rate (or PER) over the "worst" link of the given path (i.e., the link that has the highest PER among all links of the given path). We believe this is a logical approach because the transmission rate over the worst link has a significant impact on the end-to-end PER. This procedure continues until the feasibility condition is satisfied. A pseudocode for HSRA is shown in Algorithm 3.

Algorithm 3 HSRA

Input:

· Path p with h links
· $\mathcal{R} = \{r_1, r_2, \ldots, r_M\}$
· $n_\ell(r)$ and $e_\ell(r, \mathrm{SNR}_\ell)$ for all $\ell \in p$ and $r \in \mathcal{R}$
· ε

Output:

The feasible rate assignment C that results in minimum number of MASs along path p or failure if no feasible rate assignment can be found

Initialization:

$C = [r_M, r_M, \ldots, r_M]$, $|C|$=h % assign the maximum transmission rate r_M over each link of p
$n(p) = \infty$ % set the number of MASs along p to ∞

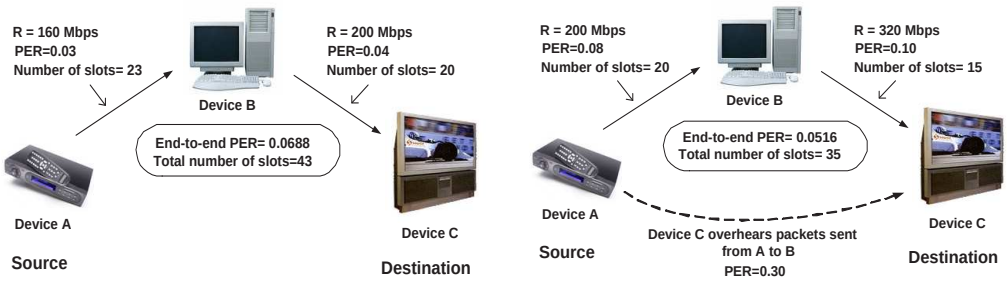
for step = 1 to Mh
if C is feasible % C results in end-to-end PER less than ε
• return C % this is the best feasible rate assignment
• break
end
for $i = 1$ to h % examine h rate assignments that are constructed by reducing the % transmission rates in C over each link of p one at a time
• $C_o = C$ % record a copy of C
• $C(i)$ = next highest transmission rate % e.g., r_M is replaced by r_{M-1}
• $n(p)^{(C)}$ = total number of MASs along p at rate assignment C
• **if** C is feasible and $n(p)^{(C)} < n(p)$
$C^* = C$ % this is the best feasible rate assignment
$n(p) = n(p)^{(C)}$ % this is the minimum number of MASs along p
end
• $C = C_o$ % return a copy of C that is stored in C_o
end
if C^* is found
• return C^* % this is the best feasible rate assignment
• break
end
% construct a new rate assignment for the second step, as follows:
• $C(\ell^*)$ = next highest transmission rate over the link ℓ^*, where $\ell^* \in p$ is the link that has the highest PER at C. If the transmission rate over the link ℓ^* equals r_1, then ℓ^* is the link that has the next highest PER
end
return "no feasible rate-assignment found"

Complexity: The worst-case computational complexity of HSRA occurs when both of the following events take place: (1) Assigning the minimum rate (i.e., r_1) over each link in path p is the only feasible rate-assignment, and (2) $e_{\ell_i}(r_1) \geq e_{\ell_{i+1}}(r_M)$

for every two adjacent links ℓ_i and ℓ_{i+1}, where $i = 1, 2, \ldots, h-1$. In this case, the algorithm will run Mh steps. In each step, it examines h rate assignments. Therefore, the worst-case computational complexity of HSRA is $\mathcal{O}(Mh^2)$.

5. OVERHEARING-BASED HSRA

An important feature of broadcast communications is that packets transmitted from a source to a destination are likely to be overheard by other non-intended nodes. In this section, we exploit this feature to enhance the performance of HSRA. Our proposed technique for exploiting packet overhearing is based on the fact that packet overhearing increases the packet delivery rate over a given path by supporting multiple paths to the destination. Such spatial diversity gain allows the use of higher transmission rates along the given path, leading to a decrease in the total number of required MASs, and hence an increase in the number of admitted flows. To illustrate, consider the example in Figures 7 (a)-(b). In Figure 7 (a), rate assignment over each link on the path $A \rightarrow B \rightarrow C$ is done ignoring the fact that the packet sent from A to B is likely to be overheard by C. Using the equation of end-to-end PER used in section 3.2, the end-to-end PER is 0.0688 and the total number of required MASs along the path is 43 slots. However, as shown in Figure 7 (b), by considering packet overhearing, there are two possible paths to deliver the packet from A to C: $A \rightarrow B \rightarrow C$ and $A \rightarrow C$. Assume that the PER over link $A \rightarrow C$ at rate $R = 160$ Mbps (the rate used over link $A \rightarrow B$) is 0.10. Then, the end-to-end PER (not shown in the figure) will decrease to 0.0069. This reduction in PER can be exploited to increase the transmission rate over the links along the path $A \rightarrow B \rightarrow C$ while at the same time satisfying the target end-to-end PER of 0.08. Accordingly, the total number of required time slots along the path is reduced to 35 slots (see Figure 7 (b)).



(a) Without Exploiting Packet Overhearing. (b) With Exploiting Packet Overhearing.

Fig. 7. Example that illustrates the exploitation of packet overhearing to improve resource utilization.

To explain how we obtained the end-to-end PER in the above example, consider a 3-hop path between a source S and a destination D, as shown in Figure 8. In this case, considering packet overhearing results in 4 possible paths: $S \rightarrow A \rightarrow D$, $S \rightarrow A \rightarrow B \rightarrow D$, $S \rightarrow B \rightarrow D$, and $S \rightarrow D$. Let $\mathrm{PER_{SD}}$ and $\mathrm{E_{XY}}$ respectively denote

the end-to-end PER and the event that a packet is successfully delivered over a link $X \rightarrow Y$. Then, $\mathrm{PER_{SD}} = 1 - \Pr[(\mathrm{E_{SA}} \cap \mathrm{E_{AD}}) \cup (\mathrm{E_{SA}} \cap \mathrm{E_{AB}} \cap \mathrm{E_{BD}}) \cup (\mathrm{E_{SB}} \cap \mathrm{E_{BD}}) \cup (\mathrm{E_{SD}})]$. Note that the union terms in this expression represent the events of successful packet delivery over the various paths, e.g., $(\mathrm{E_{SA}} \cap \mathrm{E_{AD}})$ represents the event that a packet is successfully delivered over the path $S \rightarrow A \rightarrow D$. According to the set theory, evaluating $\mathrm{PER_{SD}}$ requires that the dependency between these events should be accounted for. This dependency can be easily handled by understanding the physical meaning of the various combinations. As an example, the term $\Pr[(\mathrm{E_{SA}} \cap \mathrm{E_{AD}}) \cap (\mathrm{E_{SA}} \cap \mathrm{E_{AB}} \cap \mathrm{E_{BD}})]$ reflects the probability of receiving the packet over $S \rightarrow A \rightarrow D$ and $S \rightarrow A \rightarrow B \rightarrow D$, which means a successful reception over all links of the two paths. Therefore, $\Pr[(\mathrm{E_{SA}} \cap \mathrm{E_{AD}}) \cap (\mathrm{E_{SA}} \cap \mathrm{E_{AB}} \cap \mathrm{E_{BD}})] = \Pr[\mathrm{E_{SA}} \cap \mathrm{E_{AD}} \cap \mathrm{E_{AB}} \cap \mathrm{E_{BD}}]$. Other combinations (e.g., $(\mathrm{E_{SA}} \cap \mathrm{E_{AD}}) \cap (\mathrm{E_{SA}} \cap \mathrm{E_{AB}} \cap \mathrm{E_{BD}}) \cap (\mathrm{E_{SB}} \cap \mathrm{E_{BD}})$) can be recursively computed using previously calculated combinations (e.g., $\Pr[(\mathrm{E_{SA}} \cap \mathrm{E_{AD}}) \cap (\mathrm{E_{SA}} \cap \mathrm{E_{AB}} \cap \mathrm{E_{BD}}) \cap (\mathrm{E_{SB}} \cap \mathrm{E_{BD}})] = \Pr[\mathrm{E_{SA}} \cap \mathrm{E_{AD}} \cap \mathrm{E_{AB}} \cap \mathrm{E_{BD}} \cap \mathrm{E_{SB}}]$). In general, computing the probability of a union of events is an NP-hard problem [Ball and Provan 1988]. Several approximate solutions were proposed to solve this problem (e.g., [Ball and Provan 1988], [Frigessi and Vercellis 1985]), which can be used to determine an estimate of $\mathrm{PER_{SD}}$. In practical WPANs, the path lengths are limited to a few hops, which allows calculating the exact solution with manageable processing overhead using the exact algorithm presented in [Miller 1968].

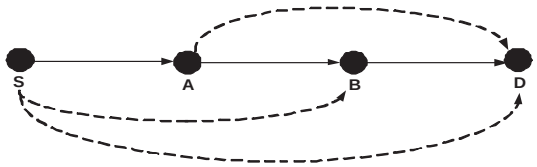


Fig. 8. Example illustrating packet overhearing in a 3-hop path.

Implementation: Our proposed technique for exploiting packet overhearing can be integrated into any given routing scheme (e.g., min-hop and shortest-distance). Specifically, to integrate our packet overhearing technique into RTERU, we modify the calculation of the end-to-end PER done by HSRA. Also, our technique requires that once a data packet is received by a node, the packet will not be discarded even if the target of the packet is not this node.

6. PACKET OVERHEARING WITH FEEDBACK

We now want to present a new technique to exploit the reserved channel time (i.e., MASs) of overheard packets. In this technique, the final destination node D of a given h-hop path p announces the IDs of the packets that overheard by itself during a given superframe m to avoid unnecessary relaying of these packets in superframe $m+1$. During superframe $m+1$, nodes are allowed to utilize the reserved relaying time of the overheard packets by using the prioritized contention access (PCA) protocol [European Computer Manufacturers Association 2008]. To illustrate, consider the example in Figure 9. In this example, assume that after applying our RTERU mechanism along with our technique that discussed in Section 5, the selected path between the source A and destination C is $A \rightarrow B \rightarrow C$. Further,

assume that the reservations for the links $A \rightarrow B$ and $B \rightarrow C$ are as shown in the figure and the number of transmitted packets during each reservation is 5 packets. As indicated in the figure, during the first superframe of the session, A sends the maximum possible number of packets during $A \rightarrow B$ reservation. In the second superframe, C announces the IDs of overheard packets by using IE in its beacon. This announcement should be considered by node B to avoid retransmitting the overheard packets and allow other nodes (e.g., D and E) to utilize (if they need) the relaying time of these packets for other communications, e.g., transmitting a text file from a computer to a printer.

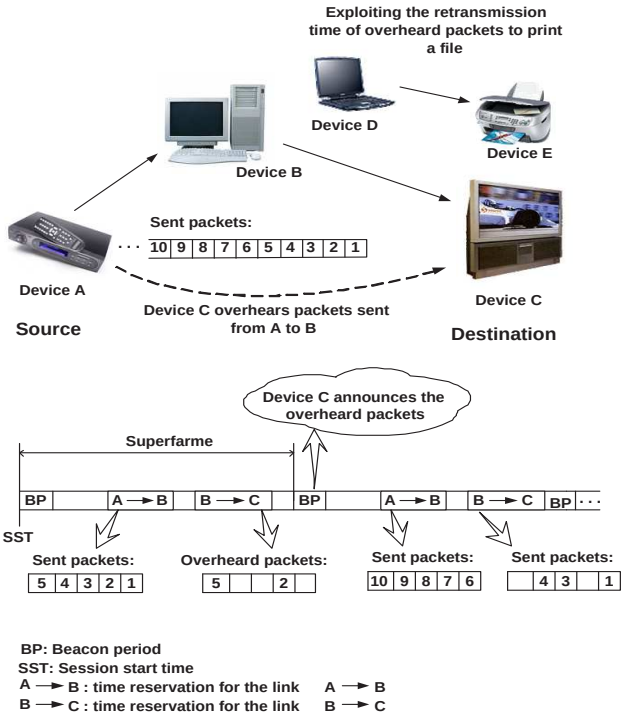


Fig. 9. Example illustrating the idea of exploiting packet overhearing with feedback.

Implementation: To allow the destination of a given path to overhear packets transmitted along the path, the destination needs to switch to the receive mode during the MASs that reserved along this path. Also, any announcement of overheard packets by the destination of the path should be considered by the nodes that receive it. If the node participates in the path, it should avoid relaying the announced overheard packets. Otherwise, the node has the choice to use the reserved relaying time of overheard packets. Note that the node needs the reserved relaying time of the announced overheard packets if there is no sufficient MASs available.

Remark 6.1. In the aforementioned technique, we do not consider packets overheard by the intermediate nodes of a path p, i.e., we only consider packets overheard by the final destination. The reason behind this can be explained as follows. If an

intermediate node of path p forwards the packets without considering the packets overheard by the next intermediate node of path p, this will result in multiple paths to the destination, which is required by our technique discussed in Section 5.

7. PERFORMANCE EVALUATION

7.1 Simulation Setup

In this section, we study the performance of RTERU and contrast it with three routing techniques: min-hop, shortest-distance, and LOAD [Kim et al. 2007]. The reason behind choosing these protocols for the comparison can be explained as follows. In our work, we mainly focus on the path selection component of the routing solution (not on the route discovery component). This is because the beaconing process defined by ECMA-368 is fundamental to any route discovery mechanism used in WPANs, where the path length is limited to a few hops. Therefore, our work aims at developing a new path selection algorithm and compare it with commonly used path selection schemes (e.g., min-hop and shortest-distance). These schemes are commonly implemented by several well known routing protocols (e.g., AODV and DSR). Note that the min-hop and shortest-distance schemes choose arbitrarily among the different paths of the same length, ignoring link quality, thus not necessarily finding the best routes. One version of AODV (called LOAD) was proposed to consider link quality in selecting the routes. Its routing metric is composed of the hop count (HC) and the number of weak links (WL) along the path. Specifically, the path cost is the tuple (WL, HC). Paths are ordered lexicographically according to this tuple. A path with cost (WL_1, HC_1) is said to be better than a path with cost (WL_2, HC_2) if $WL_1 < WL_2$, or if $WL_1 = WL_2$ and $HC_1 \leq HC_2$. A weak link is a link whose quality (i.e., received SNR) is less than a given threshold. Our results are based on simulation experiments conducted using CSIM programs (CSIM is a C-based process-oriented discrete-event simulation package [CSI]). The determination of interference and noise is done according to the physical (SINR) model. We consider a multi-band UWB WPAN, where N nodes are uniformly placed within 20 x 20 meter^2 field (Unless stated otherwise). This size is representative of realistic deployment scenarios (e.g., indoor offices, apartments, etc.). Nodes are randomly paired. For a given source-destination pair, the session length is randomly selected in the range $[0, 60]$ seconds. Once the session terminates, a new session is immediately initiated with a newly selected duration. For all sessions, the traffic load γ (in bps) of a reservation is a controllable parameter and is taken to be the same for all sessions. For simplicity, data packets are assumed to be of a fixed size (1 KB). Other parameter values used in the simulation are given in Table II. These values correspond to realistic hardware settings [Batra et al. 2004] [European Computer Manufacturers Association 2008].

7.2 Results

We mainly focus on five performance metrics: (1) network throughput (i.e., goodput), (2) PER, (3) Jain's fairness index (i.e., throughput fairness) (5) blocking rate, and (5) *deficiency*. To explain these metrics, we first clarify the procedure for establishing a session between two nodes. The source node starts by checking the available channel time, i.e., unreserved MASs in the superframe. If adequate

Table II. Parameters Used in the Simulation.

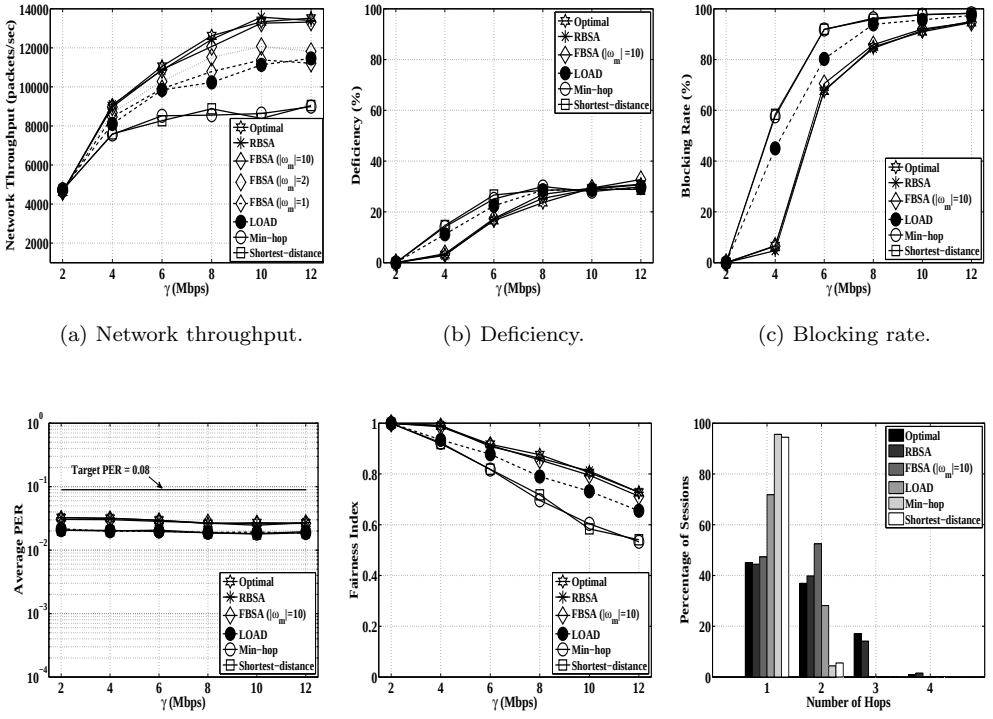
Transmission rates	53.3-480 Mbps
Average transmission power	−10.3 dBm
Transmitter antenna gain	0 dBi
Receiver antenna gain	0 dBi
Path loss factor	2
Receiver noise figure	6.6 dB
Hardware-related loss	2.5 dB

channel time is available to support the given traffic load (in bits/superframe), the node proceeds with the reservation. Otherwise, the request may or may not be blocked, depending on the type of the application. In our simulation, we consider two types of applications: elastic and non-elastic applications. If the application is "elastic", the session will be established using whatever channel time is available (but not to exceed the required demand), and the unsatisfied traffic load is captured via the deficiency metric. On the other hand, if the application is non-elastic, the request will be blocked. According to the above discussed procedure, we calculate the blocking rate as the ratio between the number of blocked sessions and the total number of generated sessions. Deficiency is calculated as the ratio between the unsatisfied load and the total offered load for elastic traffic.

Remark 7.1. Due to space limitations, some simulation results are included in the Online Supplement of TOMACS.

Without Packet Overhearing: Figures 10 (a)-(e) depict the performance of various routing schemes without implementing our proposed techniques to exploit packet overhearing. Besides RBSA, FBSA, LOAD, min-hop, and shortest-distance schemes, we also provide the "optimal" performance of the joint routing and rate selection, obtained using exhaustive search. The computational time for this optimal solution was controlled by considering the following factors in the search process: the chosen network size (i.e., $N = 20$), the area size that limits the path lengths to a few hops , and the number of transmission rates that were determined by the standard (i.e., 8 rates). Since there is no rate assignment mechanism in LOAD, min-hop, and shortest-distance schemes, we let them use exhaustive search to find the best rate assignment along the selected path. For FBSA, we consider several values of $|\omega_m|$. Figures 10 (a)-(e) show that both RBSA and FBSA achieve high performance relative to the other compared schemes. On average, RBSA and FBSA improve the network throughput by 17% and 33% compared with LOAD and (min-hop and shortest-distance), respectively. In terms of fairness, both RBSA and FBSA provide higher fairness index (more requests are admitted) than the other compared schemes. This is because both RBSA and FBSA consider the rate assignment during the route selection to minimize the required number of MASs for a reservation, which results in more admitted requests, higher throughput, and lower deficiency. In contrast, LOAD, min-hop, and shortest-distance do not consider the rate assignment during the path selection. This may result in selecting a path that requires assigning low transmission rates, hence more reservation time, high blocking rate, and low throughput. Note that, LOAD mitigates this issue by considering the link quality in its routing metric. The performance gain achieved by RBSA and FBSA increases with γ, which supports the trend in Figure 4. Compared with

the optimal solution, both RBSA and FBSA are very close to the optimal solution (within 3% on average). However, the performance of FBSA depends on the choice of $|\omega_m|$. The higher the value of $|\omega_m|$, the higher the number of examined routes, and hence the higher the likelihood returning the optimal solution.



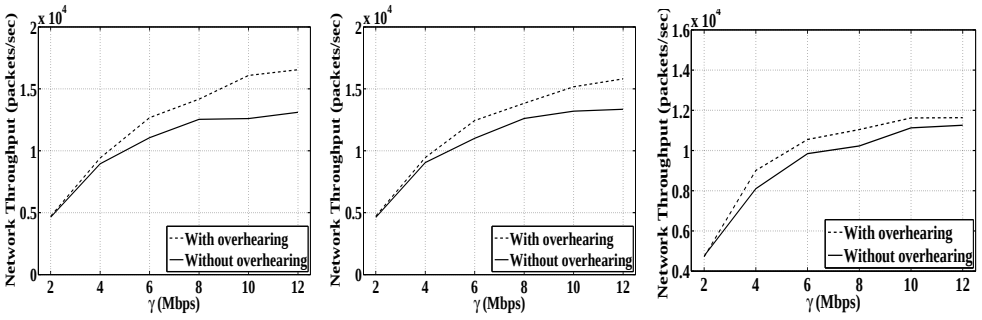
(a) Network throughput. (b) Deficiency. (c) Blocking rate.

(d) Average PER. (e) Throughput fairness. (f) Histogram of path length ($\gamma = 2$ Mbps).

Fig. 10. Performance of various routing techniques versus traffic load γ ($N = 20$, elastic traffic, without exploiting packet overhearing).

With Packet Overhearing: We now want to show that our proposed technique for exploiting packet overhearing, which is discussed in Section 5, is general. It can be integrated into any routing scheme to improve network throughput. Note that we do not aim at comparing different routing techniques. In Figures 11 (a)-(c), we study the effect of integrating our technique into FBSA, RBSA, and LOAD. As shown in these figures, this integration achieves an improvement in the network throughput (on average, 18% by FBSA, 15% by RBSA, and 9% by LOAD). As discussed before, this improvement in the network throughput is attributed to the fact that packet overhearing results in multiple paths for the transmitted packets to be received by the destination, which leads to increase the reliability of the communication channel. Then, according to our proposed technique, the increase in the reliability can be exploited by increasing the transmission rates along the

path, which results in more admitted requests and higher network throughput. Note that the low throughput improvement in the case of LOAD protocol is due to the fact that LOAD prefers the direct paths more than FBSA or RBSA, see Figure 10 (f). Accordingly, with the existence of direct paths, packet overhearing rarely takes place.

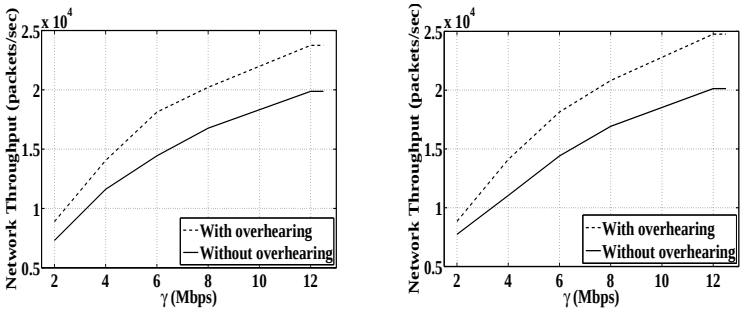


(a) FBSA. (b) RBSA. (c) LOAD.

Fig. 11. Performance of various routing techniques under employing our proposed technique for exploiting packet overhearing for the purpose of reducing the end-to-end PER (N = 20, Area 20×20 meter^2, elastic traffic).

In Figures 12 (a)-(b), we study the effect of integrating our technique for exploit packet overhearing, which is discussed in Section 5, into min-hop and shortest-distance schemes. As mentioned before, we consider a WPAN, where N nodes are uniformly placed within 20 x 20 meter^2 field. This size is representative of realistic deployment scenarios (e.g., indoor offices, apartments, etc.). Figure 10 (f) shows that, in such scenarios, the min-hop and shortest-distance schemes result in mostly single-hop paths. In this case, packet overhearing rarely takes place, and hence showing the effect of integrating our techniques into min-hop and shortest-distance schemes is not possible. Therefore, for these schemes we used a network of 40 nodes, which were uniformly placed within 40 x 40 meter^2 field. These settings increase the chances of producing multi-hop paths, hence showing the effect of integrating our techniques into min-hop and shortest-distance schemes.

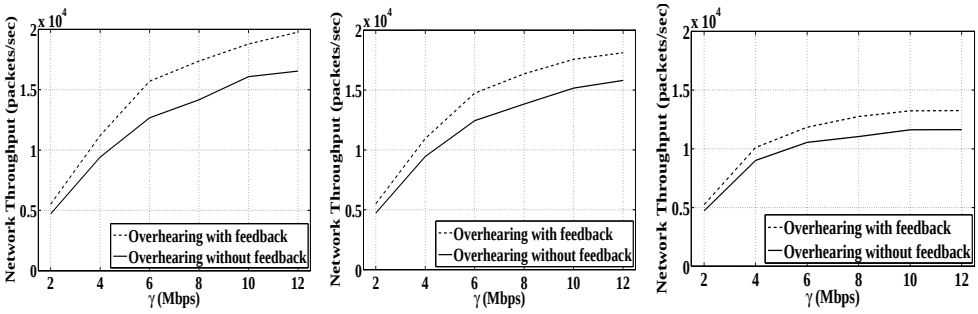
Packet Overhearing with Feedback: We now want to study the performance of our proposed technique for utilizing the reserved channel time of the overheard packets (as discussed in Section 6). Actually, we propose this technique as a complementary step for our proposed technique discussed in Section 5. Therefore, in Figures 13 (a)-(c) and 14 (a)-(b), we use the term "Overhearing without feedback" to refer to our technique that discussed in Section 5 and the term "Overhearing with feedback" to refer to the both techniques that discussed in Sections 5 and 6. In Figures 13 (a)-(c) and 14 (a)-(b), we study the effect of integrating "Overhearing with feedback" into different routing techniques: FBSA, RBSA, LOAD, min-hop, and shortest-distance. Note that due to the same reasons that discussed before we use the same simulation settings that used in Figures 11 (a)-(c) and 12 (a)-(b). In this experiment, we use 10 additional source-destination pairs that compete via



(a) Min-hop. (b) Shortest-distance.

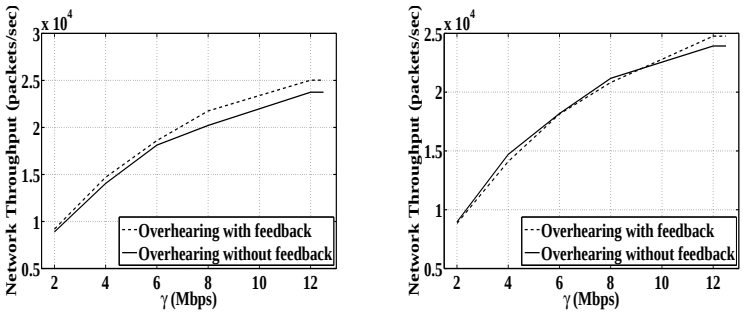
Fig. 12. Performance of various routing techniques under employing our proposed technique for exploiting packet overhearing for the purpose of reducing the end-to-end PER ($N = 40$, Area 40×40 meter^2, elastic traffic).

PCA protocol to use the released transmission time of the overheard packets. The results in Figures 13 (a)-(c) show that integrating "Overhearing with feedback" into FBSA, RBSA, and LOAD achieves an improvement in the overall network throughput (on average, 19% by FBSA and 17% by RBSA, and 13% by LOAD). Figures 14 (a)-(b) shows that integrating "Overhearing with feedback" into min-hop or shortest-distance schemes does not achieve a significant improvement. This is because min-hop and shortest-distance schemes try to minimize the hopcount and the end-to-end distance, respectively. Therefore, this may lead to maximize the distance traveled by each hop, which is likely to minimize the probability for the transmitted packets over a hop to be overheard by the nodes of the next hops. Accordingly, overheard packets rarely take place.



(a) FBSA. (b) RBSA. (c) LOAD.

Fig. 13. Performance of various routing techniques under employing our proposed technique for exploiting the reserved relaying time of overheard packets ($N = 20$, Area 20×20 meter^2, elastic traffic).



(a) Min-hop. (b) Shortest-distance.

Fig. 14. Performance of various routing techniques under employing our proposed technique for exploiting the reserved relaying time of overheard packets ($N = 40$, Area 40×40 meter^2, elastic traffic).

8. CONCLUSION

In this paper, we proposed a new cross-layer resource allocation design for WiMedia UWB-based WPANs. In our design, for each source-destination pair, we aimed at selecting the route and rate assignment that minimize the sum of the required MASs along the path while satisfying a target end-to-end PER. This allows establishing more reservations (i.e., high network throughput). Furthermore, to improve the performance of our design we proposed two techniques to exploit an important feature of broadcast communications, namely packet overhearing (i.e., packets transmitted from a source to destination are likely to be overheard by non-intended nodes). Our simulation results showed that our design achieved better performance than other routing schemes: min-hop, shortest-distance, and LOAD.

It is worth clarifying the implications of our technique on latency and power consumption. Because of the TDMA structure of the underlying system, the end-to-end delay-jitter experienced by a flow (i.e., the maximum allowable deviation from the nominal inter-packet time) cannot exceed the duration of one superframe (~ 65 msec), which meets the jitter requirement of typical streaming applications. In terms of power consumption, RTERU is less efficient than min-hop, shortest-distance, and LOAD routing schemes, for two reasons. First, packet overhearing forces devices to stay in the receive mode, which is known to consume more power than the idle mode. Second, RTERU often selects longer paths than the min-hop, shortest-distance, and LOAD schemes. Even with its higher power consumption, RTERU is still an efficient technique in networks where power consumption is not so critical (e.g., WPANs that include many devices that are powered by main supplies, including TVs, PCs, printers, DVD players, etc.).

ACKNOWLEDGMENTS

The authors would like to thank Prof. Leo Lopes for his useful discussion and valuable comments.

REFERENCES

Mesquite Software Incorporation, http://www.mesquite.com.

Abdrabou, A. and Zhuang, W. 2006. A position-based QoS routing scheme for UWB mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications 24,* 4 (April), 850–856.

Ahuja, R., Magnanti, T., and Orlin, J. 1993. *Network Flows: Theory, Algorithm, and Applications*. Prentice Hall, Inc.

Ball, M. and Provan, J. 1988. Disjoint products and efficient computation of reliability. *Operations Research 36,* 5 (October), 703–715.

Batra, A., Balakrishnan, J., Aiello, G. R., Foester, J. R., and Dabak, A. 2004. Design of a multiband OFDM system for realistic UWB channel environments. *IEEE Transactions on Microwave Theory and Techniques 52,* 9 (September), 2123–2138.

Biswas, S. and Morris, R. 2005. ExOR: Opportunistic multi-hop routing for wireless networks. In *Proceedings of the ACM Annual Conference of the Special Interest Group on Data Communication (SIGCOMM)*.

Cai, J., Liu, K., Shen, X., Mark, J., and Todd, T. 2008. Power allocation and scheduling for ultra-wideband wireless network. *IEEE Transactions on Vehicular Technology 57,* 2 (March), 1103–1112.

Dudzinski, K. and Walukiewicz, S. 1987. Exact methods for the knapsack problem and its generalizations. *European Journal of Operational Research 28,* 1, 3–21.

European Computer Manufacturers Association 2008. *ECMA-368 (3rd Edition): High Rate Wideband PHY and MAC Standard*. European Computer Manufacturers Association.

FCC 2002. *First Report and Order: In the Matter of Revision of Part 15 of the Commissions Rules Regarding Ultra-Wideband Transmission Systems*. FCC.

Frigessi, A. and Vercellis, C. 1985. An analysis of Monte Carlo algorithms for counting problems. *Calcolo 22,* 4 (October), 413–428.

Gao, H. and Daut, D. 2006. Position-based greedy stateless routing for multihop WPANs based on a realistic OFDM UWB physical layer. In *Proceedings of the IEEE Wireless Communications, Networking, and Mobile Computing International Conference (WiCOM)*.

Hsu, C., Wu, J., and Wang, S. 2006. Support of efficient route discovery in mobile ad hoc access networks. *IEICE transactions on communications E89-B,* 4, 1252–1262.

Katti, S., Katabi, D., Balakrishnan, H., and Medard, M. 2008. Symbol-level network coding for wireless mesh networks. In *Proceedings of the ACM Annual Conference of the Special Interest Group on Data Communication (SIGCOMM)*.

Kim, K., Park, S., Montenegro, G., Yoo, S., and Kushalnagar, N. 2007. 6LoWPAN ad hoc on-demand distance vector routing (LOAD). *IETF Internet Draft, draft-daniel-6lowpan-load-adhocrouting-03*.

Kuruvila, J., Nayak, A., and Stojmenovic, I. 2004. Hop count optimal position based packet routing algorithms for ad hoc wireless networks with a realistic physical layer. In *Proceedings of the IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*.

Liu, K., Cai, L., and Shen, X. 2008. Exclusive-region based scheduling algorithms for UWB WPAN. *IEEE Transactions on Wireless Communications 7,* 3 (March), 933–942.

Miller, G. D. 1968. Programming techniques: An algorithm for the probability of the union of a large number of events. *Communications of the ACM 11,* 9 (September), 630–631.

Multi-Band OFDM Alliance 2004. *Multi-Band OFDM Physical Layer Proposal for IEEE Task Group 3a*. Multi-Band OFDM Alliance.

Nowak, S., Hundt, O., and Kays, R. 2008. Joint efficiency and performance enhancement of multiband OFDM ultra-wideband (WiMedia) systems by application of LDPC codes. In *Proceedings of the IEEE International Symposium on Consumer Electronics (ISCE)*.

Pisinger, D. 1995. Minimal algorithm for the multiple-choice knapsack problem. *European Journal of Operational Research 83,* 2, 394–410.

Radunovic, B. and Boudec, J. 2004. Optimal power control, scheduling, and routing in UWB networks. *IEEE Journal on Selected Areas in Communications 22,* 7 (September), 1252–1270.

Stojmenovic, I., Nayak, A., and Kuruvila, J. 2005. Design guidelines for routing protocols in ad hoc and sensor networks with a realistic physical layer. *IEEE Communications Magazine 43,* 3 (March), 101–106.