Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2009, Article ID 467182, 16 pages
doi:10.1155/2009/467182

Research Article

Cross-Layer Optimal Rate Allocation for Heterogeneous Wireless Multicast

Amr Mohamed[1] and Hussein Alnuweiri[2]

[1] *Department of Computer Engineering, Qatar University, P.O. Box 2317, Doha, Qatar*
[2] *Department of Electrical Engineering, Texas A&M University at Qatar, P.O. Box 23874, Doha, Qatar*

Correspondence should be addressed to Amr Mohamed, amrm@ece.ubc.ca

Received 30 March 2008; Revised 5 December 2008; Accepted 12 January 2009

Recommended by Ekram Hossain

Heterogeneous multicast is an efficient communication scheme especially for multimedia applications running over multihop networks. The term heterogeneous refers to the phenomenon when multicast receivers in the same session require service at different rates commensurate with their capabilities. In this paper, we address the problem of resource allocation for a set of heterogeneous multicast sessions over multihop wireless networks. We propose an iterative algorithm that achieves the optimal rates for a set of heterogeneous multicast sessions such that the aggregate utility for all sessions is maximized. We present the formulation of the multicast resource allocation problem as a nonlinear optimization model and highlight the cross-layer framework that can solve this problem in a distributed ad hoc network environment with asynchronous computations. Our simulations show that the algorithm achieves optimal resource utilization, guarantees fairness among multicast sessions, provides flexibility in allocating rates over different parts of the multicast sessions, and adapts to changing conditions such as dynamic channel capacity and node mobility. Our results show that the proposed algorithm not only provides flexibility in allocating resources across multicast sessions, but also increases the aggregate system utility and improves the overall system throughput by almost 30% compared to homogeneous multicast.

1. Introduction

The one-hop broadcast characteristic of the MAC layer in wireless ad hoc networks has triggered the use of multicast communication scheme as one of the natural strategies that can multiply the overall network throughput with very limited overhead. This is because multicast packets are forwarded once to reach all the multicast members in the neighborhood using a single transmission, and this effect increases even more in multihop ad hoc networks.

Heterogeneous multicast, also called *multirate* multicast, is an efficient mode of data delivery for many multimedia applications, especially those operating in real time such as audio/video teleconferencing and TV broadcasting. In multirate multicast, the receivers of a multicast group are offered service at different rates commensurate with their capabilities (e.g., processing power limitations) or based on their local network conditions (e.g., surrounding wireless link states). Therefore, multirate schemes have a great advantage over unirate multicast (or homogeneous multicast) in adapting to diverse receiver requirements and heterogeneous network conditions.

The simplest way of attaining multirate multicast is by frame dropping. In this approach, intermediate nodes in a multicast tree may drop data frames to lower the rate for the downstream nodes. Another way is by hierarchical encoding or layered streaming which is particularly suitable for audio/video traffic. In this approach, the sender provides data in several layers organized in a hierarchy. Receivers subscribe to the layers cumulatively to provide progressive refinement [1]. This means that the receiver can only choose from a discrete set of data rates on each link. Another method of attaining multirate multicast which is particularly suitable for overlay multicast [2] is stream adaptation through transcoding [3] using intermediate media gateways, thus allowing the receivers to choose their streaming rates from

a broader continuous range. We assume that the network has one or more of these capabilities.

In this paper, we present an optimal resource allocation algorithm for heterogeneous multicast over wireless ad hoc networks. Multirate multicast has a distinct advantage compared to unirate multicast especially for optimal resource allocation. This is because unirate multicast techniques are often unable to efficiently allocate network resources for multicast groups that have some congested group members (receivers). For such multicast groups, unirate multicast techniques tend to allocate rates based on the most congested receivers potentially wasting significant network resources. On the other hand, multirate multicast allows the rate to change for designated tree members to accommodate the congested receivers downstream. Hence, it provides more flexibility in allocating rates across the multicast tree such that the overall network resource utilization is maximized (see the example in Section 2). Our heterogeneous multicast solution has the following key features.

(i) It guarantees optimal resource utilization while providing system-wide fairness for end-to-end multirate multicast flows.

(ii) It guarantees steering the *entire* network toward the optimal point in real time, and hence reacts robustly to network conditions (e.g., mobility and route changes) as they occur.

(iii) It is based on *primal-dual* and *pricing* methods which facilitate the decomposition of the resource optimization problem into subproblems that are easier to solve in a modular structure.

(iv) For network deployment, we design a cross-layer framework that utilizes a measurement-based technique for MAC-layer channel capacity estimation, and a light-weight network HELLO protocol for constructing contention domains to allow for allocating resources across end-to-end multicast sessions.

(v) This cross-layer solution also works in a truly distributed network environment, with limited overhead, and with no synchronization requirements between node calculations.

The problem of resource allocation for unicast flows has been investigated before in [4–6]. In these works, common pricing mechanism has been used whereby each network resource calculates a *price* that represents the relationship between the load on the network resource and the capacity that it can offer. Resource allocation for multirate multicast in wired networks has been studied in [7, 8]. An iterative algorithm based on subgradient techniques [9] has been employed to account for the nondifferentiability of the primal problem. The authors in [2] proposed an overlay strategy for allocating resources over a multirate multicast tree by considering each link as a point-to-point unicast session. Rates are then allocated across each unicast session such that the aggregate utility across all unicast sessions is maximized. The problem of optimal and fair resource allocation has been widely studied in the context of wired networks. Among these studies (e.g., [2, 4, 5, 7, 8]), price-based methods have shown to be effective in achieving a decentralized solution for rate allocation. The location-based contention, time-varying wireless channel characteristics, and multirate multicast in one-hop broadcast wireless medium represent both challenges and opportunities which we addressed in our model.

The remainder of this paper is organized as follows. In Section 2, we explain the terminology used for heterogeneous multicast and formulate the optimization problem. The approach for multirate multicast is presented in Section 3. We present our distributed asynchronous algorithm for heterogeneous multicast in Section 4. We provide the simulation results in Section 5 and finally, we conclude this paper in Section 6.

2. Model and Problem Formulation

2.1. Model and Notations. Table 1 highlights the notations used by the model. We consider a wireless ad hoc network consisting of a set of nodes V spread over a wireless space, each with a specific transmission range and interference range. We exploit the protocol model explained in [10] and leveraged in [6] for wireless packet transmission. In this model, the transmission from node i is successfully received by node j ($i, j \in V$) if (1) the distance between the two nodes is no more than a certain range (i.e., transmission range), and (2) for all other nodes $k \in V$ simultaneously transmitting over the same channel, the distance between j and k is more than a specific range (i.e., the interference range). For some protocols which require acknowledgment from j to i (e.g., IEEE 802.11 MAC), node i is also required to be interference free at the time of sending the acknowledgment.

We model the wireless ad hoc network as a *directed* graph $G = (V, E)$, where E is the set of wireless "virtual" links produced as a result of nodes located within the transmission range of each other. Each wireless link $e \in E$ has two end nodes i and j (i.e., $e = \{i, j\}$). The network is shared by a set of M *end-to-end* multicast groups. Each multicast group m has a unique source node s_m, a set of receivers $R_m = \{r_{m1}, r_{m2}, \ldots\}$, and uses a subset of wireless links E_m and a subset of nodes V_m for either receiving or relaying traffic. Note that $R_m \subseteq V_m$.

We further divide the multicast tree nodes into *gateway nodes* and *relay nodes* as shown in Figure 1. Gateway nodes are the nodes that have rate control capabilities through one of the methods explained before, such as layered transmission, transcoding and frame dropping. Relay nodes on the other hand merely forward data frames without performing any rate change. We use v_i to denote a gateway at node i. If v_i is a member of multicast tree m, hence denoted by v_{mi}, then v_i can control the rate of the downstream nodes.

A fundamental difference between the unicast and multicast cases is the fact that one-hop broadcast may be used to transfer traffic from one sending node to one or more receiving nodes. To capture this notion, the one-hop data transmission from one sending node i to a set of receiving nodes $J \subseteq V_m$ within the multicast flow m along *one* or

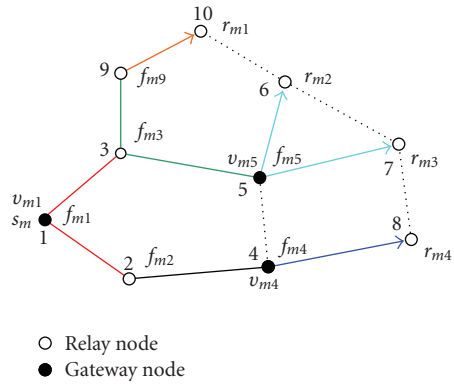


FIGURE 1: Multirate multicast network model.

more wireless links (branches) is referred to as a multicast subflow of m or f_{mi}. Each multicast subflow uses one or more branches b_{mij} from one sending node i to a set of receiving nodes $J \subseteq V_m$, that is, $f_{mi} = \{b_{mij} : \forall b_{mij} = \{i, j\}\ j \in J\}$, with a cardinality K_{mi} equal to the number of branches of f_{mi}. We also define an active wireless link $a_{ij} \in E$ to be the wireless link that carries traffic from at least one multicast group, and $A \subseteq E$ is the set of all active links. Also, a_{ij} refers to the *aggregated* multicast subflow from node i and is represented by the set of active links $a_{ij}\ \forall j \in J$ that are used by one or more multicast subflows $f_{mi}\ \forall m \in M$ simultaneously.

For simplicity, we will assume that a packet is successfully transmitted over a multicast subflow f_{mi} if (1) the packet reaches all receiving nodes J on all the branches b_{mij}; (2) acknowledgments (using the notation of IEEE 802.11 MAC standards) have been transmitted successfully from all these receiving nodes back to the sending node i [11]. Based on this assumption, the protocol model can be extended for multicast subflows as follows: the traffic from two different subflows on a group of active wireless links contend if either the sending node or *any* of the receiving nodes of one subflow are within the interference range of the sending node or *any* of the receiving nodes of the other subflow.

The multicast subtree starting from gateway node v and *ending* at either a terminal node or another gateway node is denoted by T_v. In Figure 1, $T_{v_{m1}}$ starts at node 1 and ends at the set of nodes $\{4, 5, 10\}$, and $T_{v_{m5}}$ starts at node 5 and ends at the set of nodes $\{6, 7\}$. This set of terminal nodes for subtree T_v is denoted by $\Im_v$.

Each multicast subtree T_v has a rate x_v (expressed in bits/s) which is allowed to vary within the rate interval $I_v = [w_v, W_v]$ [5], and I is the set of all such intervals. F_v denotes the set of multicast subflows that belong to subtree T_v. $\Upsilon_m = \{v_{m1}, v_{m2}, \ldots\}$ is the set of all gateway nodes that are members of group m, and Υ is the set of all gateway nodes on all multicast trees $\forall m \in M$. Each multicast group has at least one gateway node (i.e., group source is considered a gateway node) to control the rate to the downstream nodes. We use the notation $\pi_m(v)$ to indicate the parent gateway node of gateway node v by going upstream toward the source of group m (e.g., $v_{m1} = \pi(v_{m4})$). Note that the source node has no parent gateway node (i.e., $\pi(v_{m1}) = \varnothing$).

Also, note that for one multicast group m, any gateway node in the path between the source node and any receiver node may control the transmission by reducing the rate on this path to improve the overall network resource utilization (see the example in Section 2). Therefore, the rate that a gateway node is using for transmission at any given time must be greater than or equal to the maximum rate of all downstream gateway/receiver nodes. For example, in Figure 1, the rate used by v_{m1} for transmission must be greater than or equal to the maximum rate used by any of the gateway nodes v_{m4}, or v_{m5}. This adds a set of new constraints to the resource allocation problem which can be formulated by the following linear inequalities:

$$x_v \le x_{\pi_m(v)} \quad \forall v \in \Upsilon_m : \pi_m(v) \ne \varnothing \quad \forall m \in M, \tag{1}$$

where x_v is the rate used by the gateway node v, and $x_{\pi_m(v)}$ is the rate used by parent gateway of gateway node v across the multicast group m.

To model the contention between the active wireless links, we use a contention domain mechanism [12] by forming a logical contention graph $G_c = (V_c, E_c)$. Each vertex in G_c corresponds to the aggregated multicast subflow a_{ij} which carries the traffic from *one* or *more* subflows simultaneously. A link between two vertices indicates that the traffic on the two aggregated subflows contend with each other. A complete subgraph in G_c is referred to as a *clique*. A *maximal clique* is the clique that is not part of any other clique. This clique represents the maximal set of active wireless links that contend with each other. This means that only one "subflow" within this clique may transmit a packet at a time [6]. Therefore, the sum of the rates over the maximal clique cannot exceed the channel capacity achieved by using a particular scheduling mechanism in the MAC layer (e.g., IEEE 802.11 DCF). The following inequality formulates these set of constraints:

$$\sum_{v:(F_v \cap V_c^q) \ne \varnothing} x_v \le c_q, \quad \forall q \in Q, \tag{2}$$

where q is a maximal clique in the set of all maximal cliques Q, c_q is the achieved channel capacity for clique q based on the scheduling discipline used in the MAC layer, and $V_c^q \subseteq A$ are the set of vertices in G_c that belong to clique q.

Next we present an example to illustrate the above notation and highlight the main difference between unirate and multirate multicast with respect to allocating rates in an ad hoc network. Figure 2 shows an example of an ad hoc network where there are 8 nodes connected through wireless links. The network contains 2 sessions m_1, m_2 where m_2 has a traffic with fixed rate 800 kbps. Session m_1 uses node 1 as the group source, and the receiving nodes are 5, 6, whereas m_2 uses node 7 as the group source, and the receiving node is 8. The aggregated subflows are represented by one node in the contention graph as shown in Figure 2(b). Assume that the channel capacity is 1 Mbps, which means that the aggregate rate for each maximal clique cannot exceed 1 Mbps. In this case, the rate on subflow f_{m4} cannot exceed 200 Kbps because the traffic on that subflow contends with f_{m7} and hence they both exist in the same maximal clique.

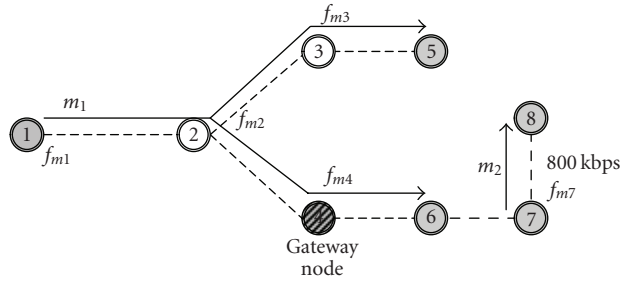
Table 1: Notations used by heterogeneous multicast model.

V_m	Set of ad hoc nodes on a multicast group m
E_m	Set of virtual wireless links used by a multicast group m
s_m	Source node for a multicast group m
R_m	Set of receivers on a multicast group m
v_{mi}	Gateway on node i controlling downstream nodes on multicast group m
f_{mi}	Subflow starting on node i on multicast group m
a_{ij}	Set of active wireless links branching from node i to set of nodes J
$T_{v_{mi}}$	Multicast subtree starting at gateway node v_{mi}
x_v	Rate used by gateway node v
$x_{\pi_m(v)}$	Rate used by parent gateway of gateway node v
q	Maximal clique in the set Q
C_q	Estimated channel capacity for clique q
$\Lambda_m(v)$	Set of all children gateways of gateway node v along multicast group m
p_q	Price for utilizing resources on clique q
p'_v	Price due to forwarding traffic by gateway node v
λ_{vi}	Total price incurred by subflow f_{vi} on all cliques
$\lambda_v(i)$	Accumulated price for subtree T_v at node i
$\pi_v(i)$	The parent node of node i along subtree T_v
B	Configurable time window for rate and price calculations

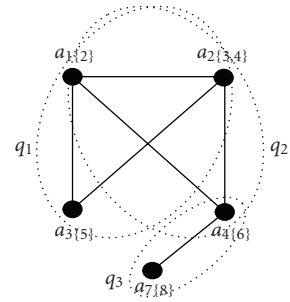
Using unirate multicast, we cannot assign a rate to group m_1 higher than 200 Kbps because one of the receivers in this multicast session is congested. This means that using unirate multicast we allocate the rate based on the most congested receiver. On the other hand, multirate multicast using node 4 as a gateway node can make the rate allocation more efficient because, in this case, the rate used by source node 1 is allowed to exceed 200 Kbps provided that gateway node 4 will adjust this rate to 200 Kbps before forwarding the traffic to the downstream nodes. It can be shown that the rate used by source node 1 can be increased to 333 Kbps in this case.

2.2. Mathematical Formulation. First, we assign a utility function $U_v(x_v)$ for each gateway node on every multicast group $m \in M$ to measure the degree of service satisfaction based on assigning a specific rate x_v to that gateway node. An example of a logarithmic utility function to achieve intersession proportional fairness is given in Section 5. The utility function also serves as a network-wide efficient tool for achieving certain fair allocation behavior (e.g., proportional fairness, max-min fairness) as shown in [4]. The optimization problem is to find the set of rates assigned to all gateway nodes for all multicast groups such that the aggregated utility function of all gateway nodes is maximized. This can be formulated with the following modified set of constraints:

$$\begin{aligned}
\text{(P): maximize } & \sum_{v\in\Upsilon} U_v(x_v) \\
\text{subject to } & \sum_{v:(F_v\cap V_c^q)\neq\varnothing} x_v\Gamma_{qv} \le c_q, \quad \forall q \in Q, \\
& x_v \le x_{\pi_m(v)}, \quad \forall v\in\Upsilon_m : \pi_m(v)\neq\varnothing \quad \forall m\in M, \\
& x_v \in I_v, \quad \forall v \in \Upsilon,
\end{aligned} \tag{3}$$



(a) An example of a multirate multicast ad hoc network



(b) Multicast contention graph $G_c = (V_c, E_c)$

Figure 2: Example for resource allocation in unirate and multirate multicast.

where Γ_{qv} represents the number of multicast subflows which belong to both clique q and the subtree T_v. Throughout the rest of the paper, we will make the following assumptions to facilitate the solution for the *primal* problem (P).

Assumption 1. There exists at least one vector $\widetilde{x} \in I$ such that $\sum_{v\in(F\cap V_c^q)}\widetilde{x}_v \le c_q \ \forall q \in Q$ and $\widetilde{x}_v \in I_v$ (i.e., which means $\sum_{v\in(F\cap V_c^q)}\widetilde{w}_v \le c_q \ \forall q \in Q$).

Assumption 2. On the interval I_m, all the utility functions $U_v \ \forall v \in \Upsilon$ are increasing, strictly concave, and twice continuously differentiable.

Note that if we restrict each multicast group to have only one gateway (source) node, then the constraints in (1) will be eliminated and problem (P) will reduce to unirate multicast.

3. Solution Approach

Solving the resource allocation problem (P) with a centralized approach requires the knowledge of the utility functions and the knowledge of all contention domains and multicast groups, which is impractical. Instead we propose a decentralized scheme that minimizes the coordination between networks nodes and adapts robustly to network changes. The key to our solution is the use of the duality theory [13] which suggests solving the dual problem by introducing additional dual variables called *prices* using the same notation as in [4, 6, 8].

The first step is to define the Lagrangian function $L(x, p, p')$ for the optimization problem (P) as follows:

$$\begin{aligned} L(x,p,p') &= \sum_{v\in\Upsilon} U_v(x_v) + \sum_{q\in Q} p_q(c_q - x_v\Gamma_{qv}) \\ &\quad + \sum_{v\in\Upsilon} p'_v(x_{\pi_m(v)} - x_v) \\ &= \sum_{v\in\Upsilon}\left[U_v(x_v) - x_v\left(\sum_{q\in Q} p_q\Gamma_{qv} + p'_v - \sum_{v'\in\Lambda_m(v)} p'_{v'}\right)\right] \\ &\quad + \sum_{q\in Q} p_q c_q, \end{aligned} \tag{4}$$

where $\Lambda_m(v)$ is the set of all children gateway nodes of node v (if any) along multicast session m. Vectors $p = (p_q \ \forall q \in Q)$ and $p' = (p'_v \ \forall v \in \Upsilon)$ are two vectors of Lagrange multipliers. Γ_{qv} represents the number of multicast subflows that belong to subtree T_v and clique q simultaneously. Again we notice that the first term of (4) is separable in x_v, and this entails

$$\begin{aligned} &\max_{x_{v\in I_v}} \sum_{v\in\Upsilon}\left[U_v(x_v) - x_v\left(\sum_{q\in Q} p_q\Gamma_{qv} + p'_v - \sum_{v'\in\Lambda_m(v)} p'_{v'}\right)\right] \\ &\quad = \sum_{v\in\Upsilon} \max_{x_{v\in I_v}}\left[U_v(x_v) - x_v\left(\sum_{q\in Q} p_q\Gamma_{qv} + p'_v - \sum_{v'\in\Lambda_m(v)} p'_{v'}\right)\right]. \end{aligned} \tag{5}$$

Which means that this objective function can be divided into $|\Upsilon|$ separate subproblems. Each subproblem for subtree T_v can be solved locally if the values of clique prices $p_q \ \forall q : (F_v \cap V_c^q) \neq \varnothing$, gateway forwarding price p'_v, and all children gateway forwarding prices $p'_{v'} \ \forall v' \in \Lambda_m(v)$ are known. The objective function of the dual problem then becomes

$$\begin{aligned} &D(p,p') \\ &= \max_{x_v\in I_v} L(x,p,p') \\ &= \sum_{q\in Q} p_q c_q \\ &\quad + \sum_{v\in\Upsilon} \max_{x_{v\in I_v}}\left[U_v(x_v) - x_v\left(\sum_{q\in Q} p_q\Gamma_{qv} + p'_v - \sum_{v'\in\Lambda_m(v)} p'_{v'}\right)\right], \end{aligned} \tag{6}$$

and the dual problem (D) for the primal problem (P) as explained in [13] can then be defined as follows:

$$\text{(D)}: \min_{\substack{p\ge 0\\ p'\ge 0}} D(p). \tag{7}$$

Equation (7) suggests that to find the optimal rates in a decentralized fashion, we need to find the optimal prices p and p' by solving the constraint-less problem (D). In the following, we will see that p' can be calculated locally at each gateway node and p can also be calculated locally for each contention domain, hence decentralized solution for end-to-end optimal rates is possible as will be discussed later.

3.1. Interpretation of Prices. Consider $P_v(T_v)$ as the profit of the subtree T_v which can be defined as follows:

$$P_v(T_v) = U_v(x_v) - x_v\left(\sum_{q\in Q} p_q\Gamma_{qv} + p'_v - \sum_{v'\in\Lambda_m(v)} p'_{v'}\right). \tag{8}$$

This profit represents the difference between the utility that subtree T_v gains by having rate x_v (i.e., $U_v(x_v)$) minus the summation of prices (denoted by $\widetilde{U}(x_v)$) that this subtree has to pay for gaining such transmission rate, which is defined as

$$\widetilde{U}(x_v) = \sum_{q\in Q} p_q\Gamma_{qv}x_v + p'_v x_v - \sum_{v'\in\Lambda_m(v)} p'_{v'}x_v. \tag{9}$$

This summation of prices is divided into three components:

(i) $\sum_{q\in Q} p_q\Gamma_{qv}x_v$ which can be interpreted as the total price for utilizing resources on all cliques $\forall q \in Q$ such that $F_v \cap V_c^q \neq \varnothing$. In this case, p_q can be interpreted as the price per unit bandwidth consumed at clique q.

(ii) $p'_v x_v$ is the price that subtree T_v must pay to the parent subtree of the same group in order to have traffic with rate x_v forwarded to it. In this case, p'_v is the price per unit bandwidth for forwarding traffic to subtree T_v.

(iii) $\sum_{v'\in\Lambda_m(v)} p'_{v'}x_v$ is the total revenue that subtree T_v gains by forwarding traffic with rate x_v to all children subtrees with each term $p'_{v'}x_v$ indicating the revenue for forwarding traffic to subtree $T_{v'}$ such that $v' \in \Lambda_m(v)$.

Note that at optimality, $p'_v = 0$ if $x_v < x_{\pi_m(v)}$ since p'_v indicates the price when the constraints (1) are violated or the maximum possible rate is used (i.e., $x_v = x_{\pi_m(v)}$). This means that a subtree $T_{v'}$ is not charged for using rate x_v if this rate is *less than* the rate at parent gateway node $\pi_m(v)$.

For p_q we can, similarly, define the price for one subflow $f_{vi} \in T_v$ as the total price for consuming bandwidth on all maximal cliques $q \in Q$ as follows:

$$\lambda_{vi} = \sum_{q:(f_{vi} \in V_c^q) \neq \varnothing} p_q. \tag{10}$$

Moreover, we can also define the aggregated price for subtree T_v as a result of consuming bandwidth on all maximal cliques $q \in Q$ as follows:

$$\lambda_v = \sum_{q:(F_v \cap V_c^q) \neq \varnothing} p_q \Gamma_{qv}. \tag{11}$$

A crucial aspect of our solution is how to calculate the individual subtree clique prices $\lambda_v \ \forall v \in \Upsilon$ in a decentralized way given the prices of the individual maximal cliques $p_q \ \forall q \in Q$. To facilitate presentation, we introduce the following new terms:

(1) $\pi_v(i)$: the parent node of node i along subtree T_v;

(2) $\lambda_v(i)$: the accumulated price for subtree T_v at node i.

Note that, along subtree T_v, there is no parent node for the gateway node v, that is, $\pi_v(v) = \varnothing$, and the accumulated price at the $v\lambda_v(v) = 0$. We can then define the accumulated subtree price recursively as follows:

$$\lambda_v(i) = \frac{\lambda_v(\pi_v(i)) + \lambda_{v\pi_v(i)}}{K_{v\pi_v(i)}} \quad \forall i \in T_v, \tag{12}$$

where $K_{v\pi_v(i)}$ is the cardinality of subflow $f_{v\pi_v(i)}$.

Theorem 1. *If $\Im_v$ defines the set of terminal nodes for subtree T_v, then the subtree clique price can be calculated as follows:*

$$\lambda_v = \sum_{i \in \Im_v} \lambda_v(i). \tag{13}$$

Proof is given in Appendix A.

3.2. Aggregated Subtree Price Calculation. In Section 4 we will explain the iterative method for calculating both clique price p_q (hence subflow price from (10)) and the forwarding price for each gateway node p'_v. In order to calculate the total price defined by (9) at any gateway node v, we need to calculate the accumulated price on each branch recursively using (12) until we hit either a terminal node or another gateway node $v' \in \Lambda_m(v)$. Each gateway node $v' \in \Lambda_m(v)$ subtracts the forwarding price $p'_{v'}$ from the accumulated price to get the net price for the branch leading to that gateway node. Children gateway nodes and terminal nodes which are part of T_v then send the net price value back to node v to calculate the subtree aggregate price per unit bandwidth $\lambda(T_v)$ by simply aggregating all net branch prices and the forwarding price p'_v as follows:

$$\lambda(T_v) = \lambda_v + p'_v - \sum_{v' \in \Lambda_m(v)} p'_{v'}. \tag{14}$$

4. Optimal Resource Allocation for Heterogeneous Wireless Multicast (ORAHWM)

We present a distributed iterative algorithm that solves the primal problem (P) by applying the gradient projection method [13] to the dual problem (D). This implies that the clique prices $p_q(t+1) \ \forall q \in Q$ and forwarding prices $p'_v \ \forall v \in \Upsilon$ are calculated iteratively as follows:

$$\begin{aligned} p_q(t+1) &= \left[p_q(t) - \alpha \frac{\partial D(p(t))}{\partial p_q} \right]^+, \\ p'_v(t+1) &= \left[p'_v(t) - \alpha \frac{\partial D(p'(t))}{\partial p_v} \right]^+, \end{aligned} \tag{15}$$

where $\alpha > 0$ is the gradient step-size. Since $U_v \ \forall v \in \Upsilon$ are concave functions, $D(p, p')$ is continuously differentiable and the gradients for $D(p, p')$ with respect to p and p' are defined as follows:

$$\frac{\partial D(p, p')}{\partial p_q} = c_q - \sum_{v:(F_v \cap V_c^q) \neq \varnothing} x_v(t) \Gamma_{qv}, \quad q \in Q, \tag{16}$$

$$\frac{\partial D(p, p')}{\partial p'_v} = x_{\pi_m(v)}(t) - x_v(t) v, \quad \pi_m(v) \in V_m. \tag{17}$$

Substituting in (15) we get the supply and demand equations for calculating p and p' as follows:

$$p_q(t+1) = \left[p_q(t) - \alpha \left(c_q - \sum_{v:(F_v \cap V_c^q) \neq \varnothing} x_v(t) \Gamma_{qv} \right) \right]^+, \tag{18}$$

$$p'_v(t+1) = \left[p'_v(t) - \alpha \left(x_{\pi_m(v)}(t) - x_v(t) \right) \right]^+. \tag{19}$$

We calculate the subtree aggregate price $\lambda(T_v, t+1)$ defined by (14) at time $(t+1)$ using the clique, and forwarding price values from (18) and (19) as explained in Section 3.2. Finally, the transmission rate used by gateway node v at time $(t+1)$ is calculated as follows:

$$x_v(t+1) = \left[U'_v(\lambda(T_v, t+1)) \right]_{w_v}^{W_v}. \tag{20}$$

In order to prove the convergence of the algorithm described by (15)–(20), we define the following new terms. Let $Y_v = \sum_{q \in Q} \Gamma_{qv}$ indicate the number of subflows in subtree T_v, and $\overline{Y} = \max_{v \in \Upsilon} Y_v + |\Lambda_m(v)| - 1_v$ indicate the maximum summation of subflows in T_v plus number of children gateway nodes in $\Lambda_m(v) \ \forall v \in \Upsilon$, where $1_v = 1$ if $\pi_m(v) \neq \varnothing$, and zero otherwise. Let $\overline{Z} = \max_{q \in Q} \sum_{v \in \Upsilon} \Gamma_{qv}$ be the number of subflows in the most congested clique $q \in Q$, and $\overline{\gamma} = \max_{v \in \Upsilon} \gamma_v$ indicate the upper bound on all $-U''_v(x_v) \ \forall v \in \Upsilon$. We can obtain the following convergence result, the proof of which is in Appendix B.

Theorem 2. *For step-size values of α that satisfy the inequality $0 < \alpha < 2\overline{\gamma}/\overline{Y}\overline{Z}$, starting from any initial rate $x(0)$ ($x_v \in I_v \ \forall v \in \Upsilon$), clique prices $p(0) \geq 0 \ \forall q \in Q$, and forwarding prices $p'(0) \geq 0 \ \forall v \in \Upsilon$, the sequence of vectors $x(t) = (x_v(t), v \in \Upsilon)$ converges to the unique optimal solution of problem (P).*

4.1. Synchronous Versus Asynchronous Computations. Equations (15)–(20) assume that the price and rate iterations are performed at time $t = 1, 2, 3, \ldots$, which implies that the price and rate calculations happen at the same time using a synchronous computation scheme. Such synchronization is however difficult to attain in a distributed network environment where nodes do not have any global synchronization clock. Practically speaking, asynchronism inevitably happens for price and rate calculations at any node because the node may not have the exact current value of the rate, the clique, or the forwarding price. Instead, it receives a sequence of recent values at different time instances. Therefore, the node will use a weighted average of these values in estimating the price or the rate at any given time. For example, for node i to calculate $p_q(t+1)$ from (18), it needs all the rates $x_v(t)\ \forall v : (F_v \cap V_c^q) \neq \varnothing$, at exactly time t. However, because node i may not have the rates at time t, it keeps the rate values at times $(t - B) \leq t' \leq t$, where B is a configurable time window for rate and price calculations. Then, it estimates the rates at time t using the following weighted average:

$$\hat{x}_v(t) = \sum_{t'=t-B}^{t} b_i^q(t', t)\, x_v(t') \quad \text{with} \sum_{t'=t-B}^{t} b_i^q(t', t) = 1. \quad (21)$$

This asynchronous mechanism is general and allows for deploying any estimation policy for the rates or prices. The simulations in Section 5 show that our algorithm attains convergence using some popular update policies such as

(i) *latest instant estimation*: only the last received value for $x_v(\tau)$ for some $\tau \in [t - B, \ldots, t]$, is used to estimate $\hat{x}_v(t)$, that is, $b_i^q(t', t) = 1$ if $t' = \tau$ and 0 otherwise;

(ii) *latest average estimation*: only the average over the latest k received values is used for estimation, that is, $b_i^q(t', t) > 0$ for $t' = \tau - k + 1, \ldots, \tau$ and 0 otherwise.

The details for the asynchronous algorithm for heterogeneous wireless multicast are shown in Algorithm 1. In order to understand the association of this algorithm with the network architecture, we assume that each node i in the network has zero or more multicast subflows $f_{vi}\ \forall v \in \Upsilon$ depending on the traffic passing by this node. Even though the algorithm suggests that the *clique procedure* at clique q can be performed by a designated node from that clique (i.e., clique master), in our simulations we perform the clique procedure at each node i separately for all cliques that have $f_{vi} \cap V_c^q \neq \varnothing$. The *subflow procedure* is performed by each node i that has one or more multicast subflows $f_{vi}\ \forall v \in \Upsilon$ by simply calculating the accumulated prices at the branches of f_{vi} based on the accumulated price at i. Finally, *active gateway nodes* (i.e., gateway nodes that have traffic from one or more multicast groups passing by them) perform the *subtree procedure* by calculating the optimal rate $x_v(t+1)$ based on the aggregated prices for this subtree.

The estimation of the price and rate values at time t (i.e., $\hat{x}_v(t), \hat{p}_q(t), \hat{\lambda}_v(i, t)$ and $\hat{p}_i'(t)$) from the received values at time instances in the range $(t - B) \leq t' \leq t$ may follow any policy such as the *latest instant estimation* or the *latest average estimation* as explained before. The support for these different update policies demonstrates the versatility of our asynchronous algorithm. The following theorem illustrates the convergence of this model (detailed proof is given in [14]).

Theorem 3. *For step-size values of α that are sufficiently small, starting from any initial rate $x(0)$ $(x_m \in I_m\ \forall m \in M)$ and clique prices $p(0) \geq 0\ \forall q \in Q$, every accumulation point $(x^*; p^*)$ of the sequence $(x(t); p(t))$ generated by the asynchronous Algorithm 1 (ORAWHM) is primal-dual optimal.*

4.2. Time-Varying Network Environment. So far, we assume that the cliques achieved capacity and the set of group utility functions are not functions of time (i.e., they do not change with time). However, due to online calculation and subproblem decomposition, it can be shown that our solution will work in the case when these quantities change with time.

For example, the clique capacity may be time-varying depending on the scheduling discipline used at the MAC layer. In this case, (16) will be the same except the current clique capacity $c_p(t)$ is used in place of the constant capacity c_p. For deploying our algorithm in a real network, we account for the time-varying channel capacity by using a bandwidth management mechanism for estimating the channel capacity based on [15]. In general, if the change in the environment parameters is relatively slow, our solution can track the changes in the optimal rates based on changing these quantities with time. This is shown in our experimental evaluation discussed in Section 5.

4.3. Cross-Layer Architecture for ORAHWM. Figure 3 depicts the cross-layer architecture of ORAHWM showing the main procedures and the interaction of ORAHWM with the different layers including MAC, routing, and transport layers. In this architecture, we use the common IEEE 802.11 DCF as the MAC protocol with multicast extensions as presented in [11]. Multicast ad hoc on demand distance vector (MAODV) [16] is used to provide a distributed routing scheme for the multicast sessions. We also use UDP with rate control in the transport layer to minimize the communication overhead through avoiding the excessive feedback packets used by other transport protocols (e.g., TCP).

We use the channel capacity estimator to measure the channel capacity in real time in the MAC layer. For this purpose, we devise a cross-layer mechanism which combines the multicast aware MAC protocol (MMP) [11] with a bandwidth management mechanism for measuring the channel capacity based on [15]. For details about this mechanism, please refer to [17]. We also use the HELLO packets for conveying the clique price information in the routing layer. The information from channel capacity estimator and HELLO packets jointly establish the requirements to calculate the dual gradient for clique prices described by (16). To construct the contention domains, we allow the price information to be broadcasted as part of the HELLO

Clique procedure (by clique q): At times $t \in T_q$
(1) Receive rates $x_v(t')$ from all subtrees T_v where $F_v \cap V_c^q \neq \varnothing$
(2) Update clique price as follows

$$p_q(t+1) = \left[p_q(t) - \alpha\left(c_q - \sum_{v:(F_v \cap V_c^q) \neq \varnothing} \hat{x}_v(t)\Gamma_{qv} \right) \right]^+, \quad \forall q \in Q$$

(3) Send $p_q(t+1)$ to all nodes of group m such that $F_m \cap V_c^q \neq \varnothing$

Subflow procedure (by subflow f_{vi}): At times $t \in T_m^i$
(1) Receive prices $p_q(t')$ from maximal cliques q where $f_{vi} \cap V_c^q \neq \varnothing$
(2) Calculate the subflow price (per hop price) λ_{vi} as follows

$$\lambda_{vi}(t+1) = \sum_{q:(f_{vi} \cap V_c^q) \neq \varnothing} \hat{p}_q(t)$$

(3) Calculate the accumulated price on each branch $b_{vij} \in f_{vi}$

$$\lambda_v(j, t+1) = \frac{\lambda_v(i,t) + \lambda_{vi}(t+1)}{K_{vi}}$$

(4) Forward $\lambda_v(j, t+1)$ to all children subflows of f_{vi}, if no children, send $\lambda_v(j, t+1)$ to v.

Subtree procedure (by gateway v): At times $t \in T_v$
(1) Receive the net prices $\lambda_v(i, t') - p_i'(t')$ from all terminal nodes of T_v (i.e., $\forall i \in \Im_v$), and all children gateway nodes $\forall i \in \Lambda_m(v)$ // (note: $p_i' = 0 \ \forall i \in \Im_v$).
(2) If $\pi_m(v) \neq \varnothing$ // (i.e., $v \neq s_m$)
(i) Receive rate $x_{\pi_m(v)}(t'')$ from parent subtree of T_v
(ii) Calculate the next forwarding price $p_v'(t+1)$ as follows:

$$p_v'(t+1) = \left[p_v'(t) - \alpha\left(\hat{x}_{\pi_m(v)}(t) - x_v(t) \right) \right]^+$$

Else $p_v'(t+1) = 0$
(3) Calculate the next subtree aggregate price $\lambda(T_v, t+1)$ as follows:

$$\lambda(T_v, t+1) = p_v'(t+1) + \sum_{i \in (\Im_m \cup \Lambda_m(v))} \left(\hat{\lambda}_v(i,t) - \hat{p}_i'(t) \right)$$

(4) Calculate the next subtree rate as follows:

$$x_v(t+1) = U_v'^{-1}\left(\lambda(T_v, t+1) \right)$$

(5) Send $x_v(t+1)$ to all cliques q where $F_v \cap V_c^q \neq \varnothing$

Algorithm 1: ORAHWM: asynchronous distributed algorithm.

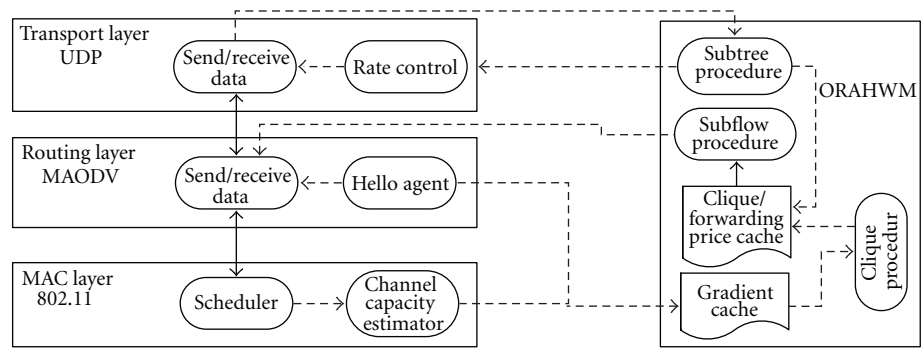


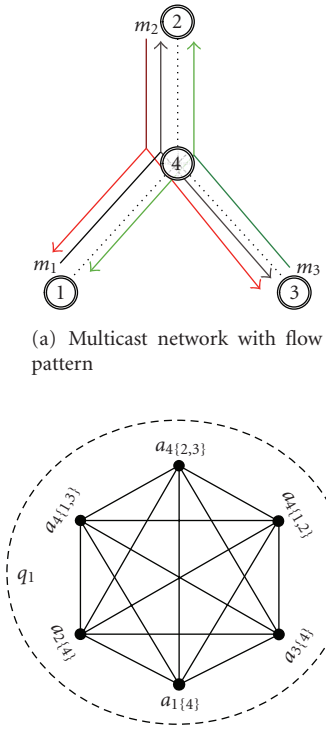
Figure 3: Cross-layer architecture for ORAHWM.

packets to all neighboring nodes within 3 hops away. Such multihop protocol-based scheme in calculating the maximal cliques is proven to work when the interference range is greater than or equal to the transmission range, given that nodes within the same interference range are reachable to each other through multihop communication, as shown in [6]. Feedback packets from the terminal nodes and gateway nodes can be used to convey the rate and accumulated price information which are used by the *subtree procedure* to calculate the dual gradient for forwarding prices described by (17) and the aggregated subtree price in (14) and hence the next subtree rate described by (20).

5. Simulation Results

In all our experiments, we use the utility functions $U_v(x_v) = g_v \ln(x_v)$ $x_v > 0$ for imposing proportional fairness [4] amongst the multicast groups, where g_v is the differentiation gain for gateway node v, that is, $x_v(t) = g_v/\lambda_v(t)$. The default transmission and interference ranges are 250 m and 550 m, respectively. We implemented all the simulations using nanosecond-2 simulator. Unless otherwise stated, we use the *latest instant estimation* for asynchronous calculations.

5.1. Effect of Time-Varying Wireless Channel. In this experiment, we study the effect of time-varying wireless channel on the speed of convergence for our algorithm ORAHWM. we deployed our algorithm in a real network that uses multicast aware IEEE 802.11 DCF MAC scheduler [11] with a bandwidth management mechanism for measuring the channel capacity (i.e., channel capacity estimator in Figure 3) based on [15] and (MAODV) [16] for routing. We take as an example the network in Figure 4 with 3 multicast sessions as shown in Figure 4(a) and the corresponding contention graph as shown in Figure 4(b) . We use equal differentiation gains, that is, $g_v = 1\ \forall v \in M$. However, we start each of these sessions in a different time to test the ability of our algorithm to track network changes. The start times of sessions m_1, m_2, m_3 are $20, 40, 60$ seconds, respectively, and the initial rates $x_v(0)\ \forall v \in M$ are selected from a uniform distribution in the range 50–250 kbps. We have fixed all the other parameters including the step-size, and we measured the rate of each multicast session against time. Figure 5 shows the result using a time-varying channel capacity realized by the MAC scheduler. From this figure, we observe that although the MAC channel capacity (i.e., the basic rate of sending data in IEEE 802.11 DCF) is set to 1 Mbps, the achieved channel capacity changes with time and does not go above 800 Kbps. Nevertheless, our algorithm continuously tracks the change in channel capacity and provides proportional fairness amongst all the multicast sessions based on the current available channel capacity. We also notice that the algorithm spends less than 2 seconds to achieve the optimal rates every time a new multicast session is added, which is deemed reasonable. However, intuitively, this convergence speed is affected by the number of hops that each multicast session spans and hence it may decrease in larger networks as we will see in the following experiments.



(a) Multicast network with flow pattern

(b) Resulting contention graph

Figure 4: Effect of time-varying wireless channel.

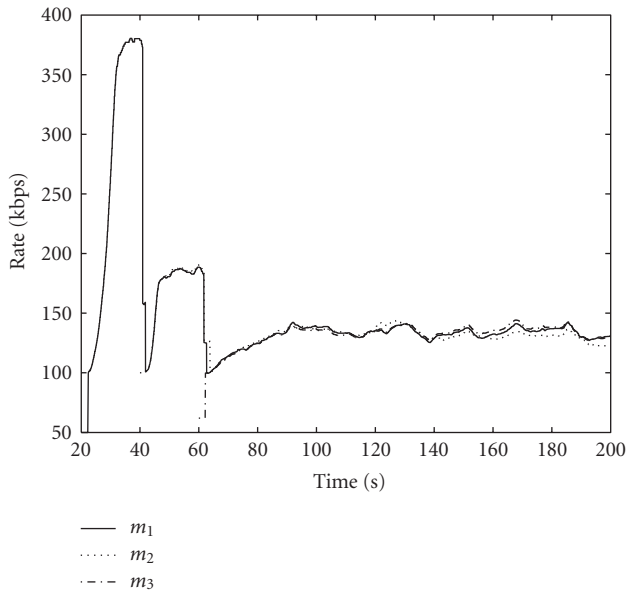


Figure 5: Convergence for time-varying channel capacity using IEEE 802.11 DCF scheduling.

5.2. Convergence in Random Network for Unirate Multicast. In this experiment, we study the convergence behavior of our algorithm ORAHWM with respect to both calculated rate and throughput in a randomly generated wireless network as shown in Figure 6. This network consists of 30 nodes deployed randomly over the $1000 \times 1000\ \text{m}^2$ wireless space.

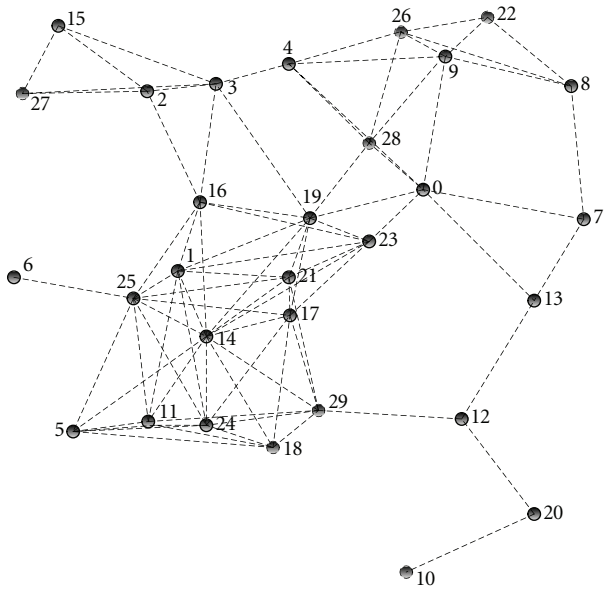


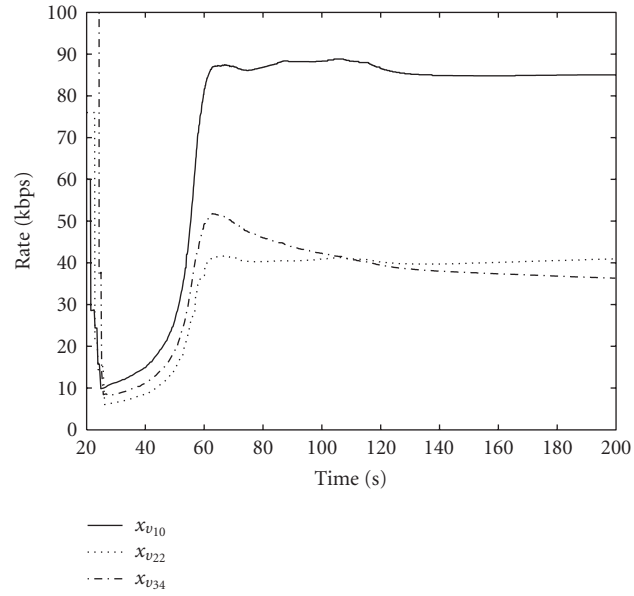
Figure 6: Random wireless network with 30 nodes.

Table 2: Multicast traffic pattern.

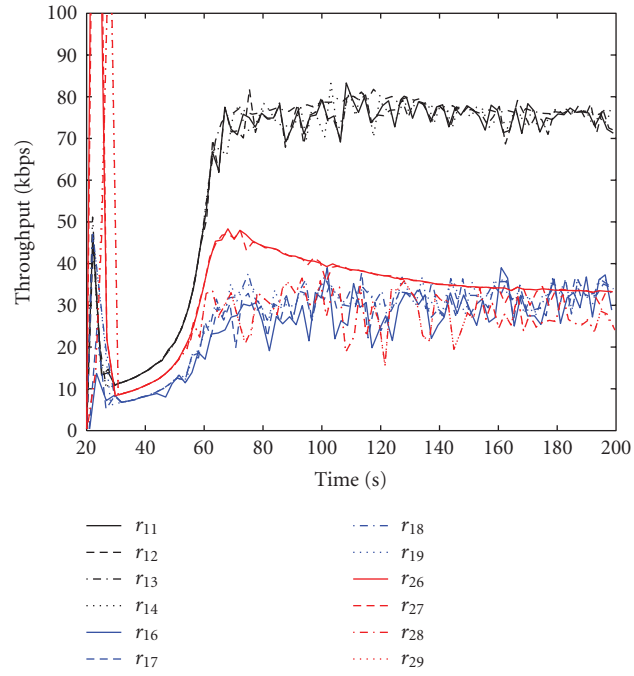
Session	Source/gateway	Receivers
m_1	v_{10}	r_{11}, r_{12}, r_{13}, r_{14}
m_2	v_{22}	r_{16}, r_{17}, r_{18}, r_{19}
m_3	v_{34}	r_{26}, r_{27}, r_{28}, r_{29}

We started 3 multicast sessions m_1, m_2, and m_3 at time 20 seconds, each with one source and gateway node v_{mi} and four receivers as shown in Table 2 using $\alpha = 10^{-8}$. The differentiation gain for all the three sessions is $g_v = 1\ \forall v \in M$).

Figure 7 shows the calculated rates and receiver throughput of each multicast session with time. From these results, we observe that our algorithm attains convergence with satisfactory speed even in relatively large-scale networks. We also observe that the throughput achieved by each receiver on all sessions follows the calculated rates fairly well, which confirms the correctness of the calculated rates. Note that the optimal calculated rates are different for each session depending on the size of the multicast tree and how much resources each session consumes from the total network resources. If this discrimination based on tree topology is undesirable, it can be compensated using different differentiation gains (g_v) on each session, which will be discussed in the next experiment. We also observe that the time spent by the algorithm to achieve full optimality is almost 35 seconds in such large fully distributed network. However, after 5 seconds only, the rates start to approach optimal point gradually. This indicates that, although the algorithm may not have enough time to achieve full optimality especially in large-distributed environments, it will always attempt to approach optimal point and follow network changes concurrently and satisfactorily.



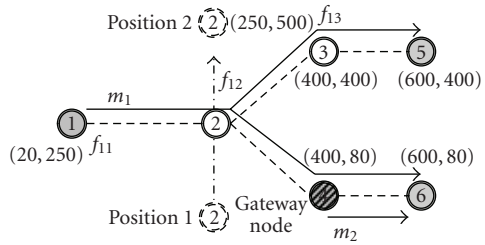
(a) Calculated rates



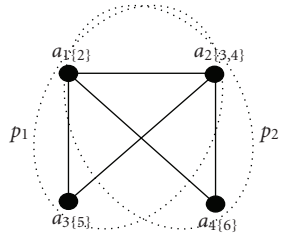
(b) Receiver throughput

Figure 7: Convergence of ORAHWM in large random networks.

5.3. Effect of Changing Differentiation Gains on the Calculated Rates and Aggregate Utility. In these experiments, we study the effect of changing the differentiation gains on the calculated rates for unirate and multirate multicast sessions. We consider the small topology shown in Figure 8. Two sessions m_1 and m_2 are sharing this network with source and receiver nodes as shown in Figure 8. we consider 3 cases where we change the differentiation gain and show the effect on the calculated rates in each case. Case 1 is the unirate multicast where we use one gateway/source node for each



(a) Multicast network topology



(b) Contention graph

FIGURE 8: Effect of changing differentiation gains on the calculated rates and aggregate utility.

multicast group, and we use equal differentiation gains for both sessions (i.e., $g_{v_{11}} = g_{v_{24}} = 5$). For both cases 2 and 3, m_1 gateway node 4 for rate control and the differentiation gain $g_{v_{24}}$ is set to. Case 2 uses differentiation gains $g_{v_{11}} = 3$, $g_{v_{14}} = 2$ whereas case 3 uses $g_{v_{11}} = 4$, $g_{v_{14}} = 1$. In all cases we start both multicast groups at $t = 20$ seconds, we fix all the other parameters including $\alpha = 3 \times 10^{-7}$, and we set the channel capacity for all maximal cliques to 1 Mbps.

Figure 9 shows the calculated rates and the aggregated utility for the 3 cases. We notice that for case 1 (unirate), as expected, our algorithm ORAHWM will discriminate against session m_1 because it uses more wireless links and hence utilizes more network resources. This happens because for unirate, ORAHWM deals with each session as one entity regardless of how large this session is and how many links it uses. Multirate with additional gateway nodes can reduce this effect by providing more flexibility to assign more priority to some parts of the tree which in turn affects the aggregate utility of the entire system. This is depicted by the results in Figures 9(b) and 9(c). We notice that by increasing the differentiation gain for $T_{v_{11}}$, we can increase the aggregate utility (shown in Figure 9(d)). Therefore, assigning differentiation gains to different parts of the multicast trees is a crucial aspect of this algorithm and may call for a mechanism to assign these differentiation gains in real time in order to maximize the overall aggregate utility. For example, multicast subtrees which serve large number of uncongested receivers will be assigned higher differentiation gains, while multicast subtrees with fewer congested receivers will be assigned lower differentiation gains.

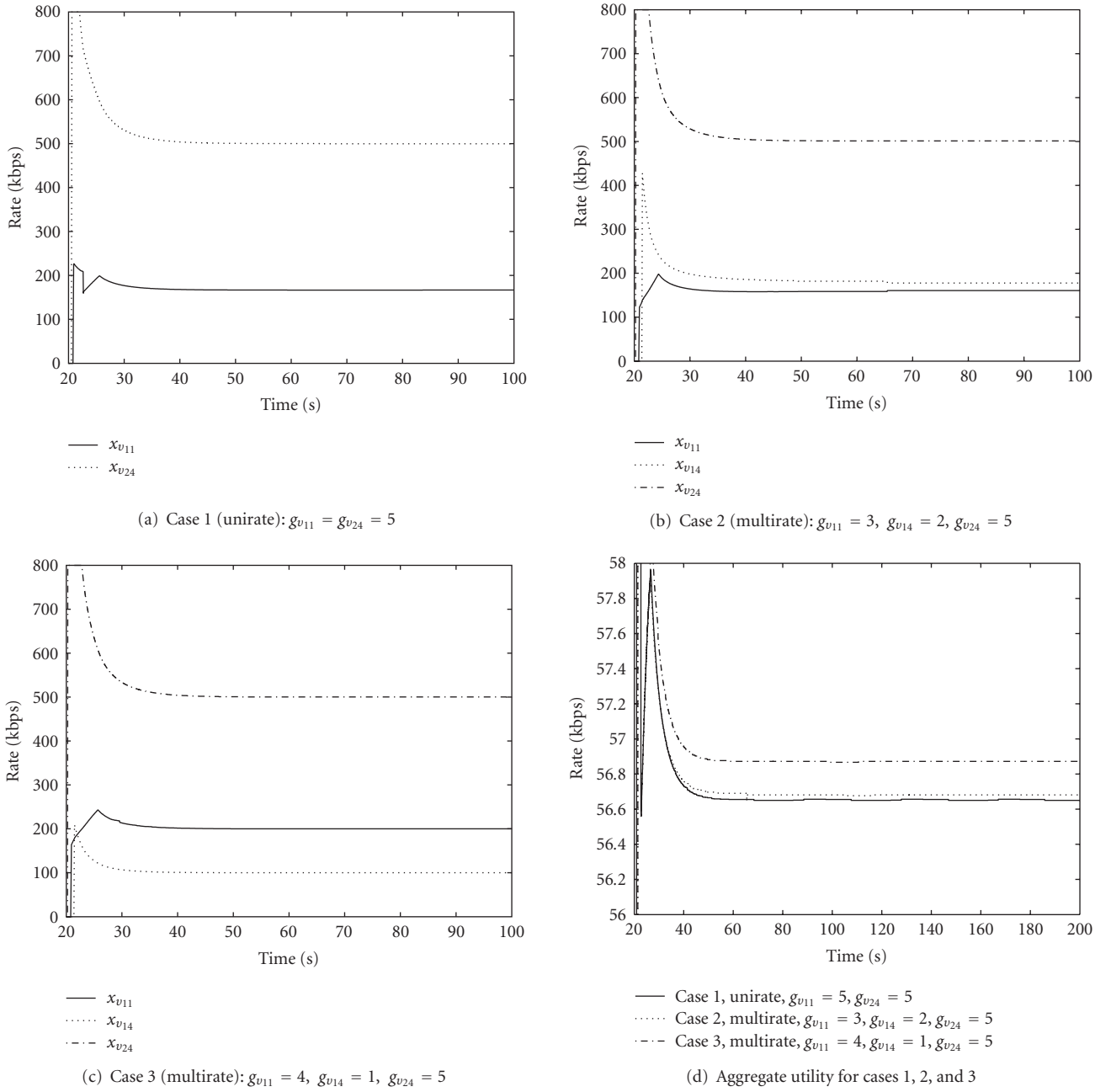
5.4. Effect of Time-Varying Channel and Mobility on the Convergence of ORAHWM. In these experiments, we study the effect of changing network conditions including changing capacity and node mobility on the convergence of our algorithm ORAHWM. We consider the same topology and multicast sessions shown in Figure 8, and we use the $g_{v_{11}} = 4$, $g_{v_{14}} = 1$.

First we study the effect of measuring the real capacity on each clique using the MAC layer channel capacity estimator as explained in Section 4.3. Figure 10 shows the result of using a time-varying channel capacity realized by the MAC scheduler IEEE 802.11 DCF with channel data rate 1 Mbps. From this figure, we observe that although the achieved channel capacity changes with time, our algorithm continuously tracks the change in channel capacity fairly well and provides proportional fairness amongst all the multicast sessions based on the current available channel capacity.

We also study the impact of mobility and route changes on the convergence of our algorithm by generating a mobility pattern where node 2 moves from *position* 1 to *position* 2 as shown in Figure 8(a) with average speed of 3 m/s and pause time 20 seconds. Figure 11 shows the rates calculated by our algorithm with time. The figure shows 3 different regions depending on the change of routes resulting from the node mobility. In region 1, only node 6 is receiving traffic for both multicast sessions. As expected in this case, our algorithm converges to the same rates of case 3 in the previous experiment. As node 2 moves to region 2, the routes for which are shown by Figure 8, both receivers at nodes 5, and 6 become active for session m_1 and the optimal rates converge to the same values, after some transient period, despite the route changes. When node 2 moves to region 3, both the receivers at node 6 and node 4 become inactive for session m_1 and session m_2 can now use the whole channel for its traffic. Therefore, the optimal rate for m_2 in this case is 1 Mbps whereas the capacity is divided amongst the 3 subflows f_{11}, f_{12}, and f_{13} for m_1.

5.5. Effect of Using Multirate on the Total Throughput for Multicast Flows. In this experiment we study the effect of using gateway nodes for rate control as part of a multicast group. Consider Figure 12 which shows two multicast groups m_1 and m_2 sharing an ad hoc network on 11 nodes as shown in Figure 12. m_1 uses gateway/source node 1 (i.e., v_{11}), and has 3 receivers, namely, r_7, r_8, and r_9 while m_2 uses gateway/source node 6 (i.e., v_{26}) and has two receivers, namely, r_{10} and r_{11}. Here, to study the impact of using multirate multicast we consider two cases. Case 1 is the unirate multicast with equal differentiation gains for both multicast groups (i.e., $g_{v_{11}} = g_{v_{26}} = 3$). For case 2, m_1 uses an additional gateway node at 4 (i.e., v_{14}) for rate control. In this case, we set $g_{v_{11}} = 2$, $g_{v_{14}} = 1$ so the total differentiation gain is similar to case 1, and we set $g_{v_{26}} = 3$.

Figures 13 and 14 show the calculated rates and receiver throughput for cases 1 and 2, respectively. We notice that in each case convergence is attained, and the throughput achieved by all receivers on each group tracks the calculated rates appropriately. Comparing the two figures, we notice the effect of using gateway node v_{14} for m_1 which lowers the optimal rate on the subtree $T_{v_{14}}$ (i.e., $x_{v_{14}}$) allowing the



(a) Case 1 (unirate): $g_{v_{11}} = g_{v_{24}} = 5$

(b) Case 2 (multirate): $g_{v_{11}} = 3,\ g_{v_{14}} = 2, g_{v_{24}} = 5$

(c) Case 3 (multirate): $g_{v_{11}} = 4,\ g_{v_{14}} = 1,\ g_{v_{24}} = 5$

(d) Aggregate utility for cases 1, 2, and 3

Figure 9: Effect of changing differentiation gains on the calculated rates and aggregate utility.

other rates $x_{v_{11}}$ and $x_{v_{26}}$ to increase drastically. This happens because we set the differentiation gain $g_{v_{14}} = 1$ giving this subtree lower priority based on our knowledge that this subtree has only one receiver (r_9), and the surrounding area has traffic load more than for example, $T_{v_{11}}$ and we used v_{14} to give us the flexibility of setting $x_{v_{14}}$ accordingly. Such knowledge can either be communicated between the receivers and gateway nodes or tuned manually by an administrator.

To study the effect of this heterogeneity within m_1 we measure the aggregate utility and the total throughput achieved by each group for cases 1 and 2. Figures 15 and 16 show the results for these measurements. We see from Figure 15 that the aggregate utility achieved for case 2 is better as a result of using gateway node v_{14} because both rates $x_{v_{11}}$ and $x_{v_{26}}$ increased significantly by reducing $x_{v_{14}}$. This increase in rates caused the overall throughput achieved by both multicast groups to increase drastically (i.e., ≈30%) as shown in Figure 16.

6. Concluding Remarks

In this paper we have presented the resource optimization algorithm for the case of multirate multicast (ORAHWM)

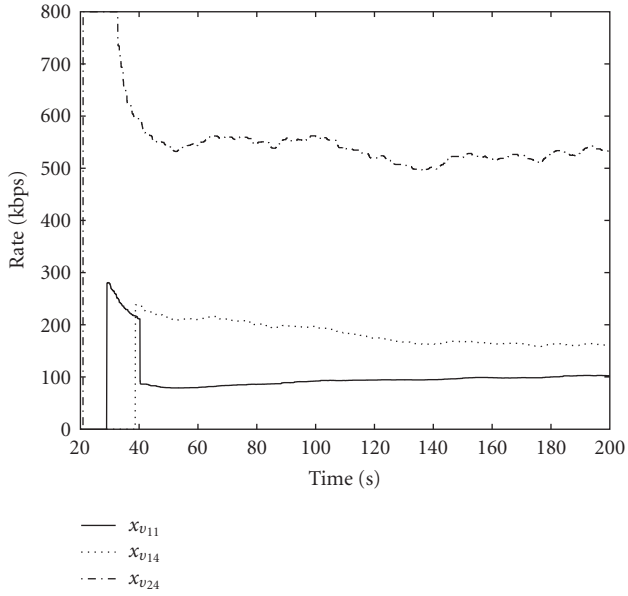


Figure 10: Calculated rates with dynamic capacity: $g_{v11} = 4$, $g_{v14} = 1$, $g_{v24} = 5$.

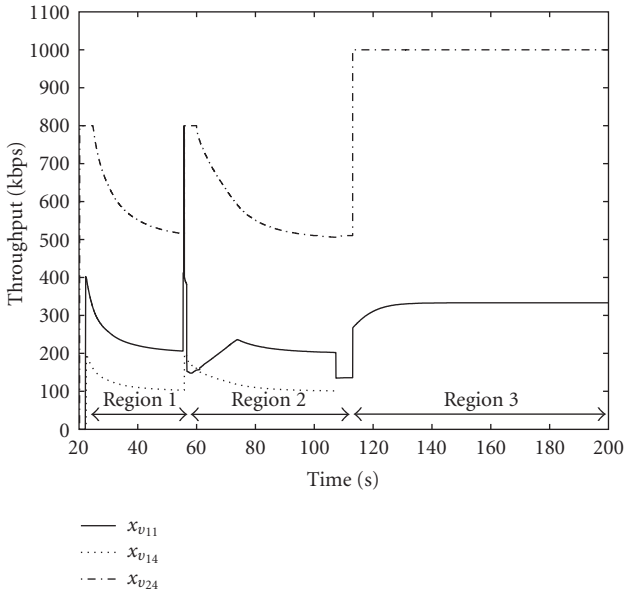


Figure 11: Calculated rates with mobility: $g_{v11} = 4$, $g_{v14} = 1$, $g_{v24} = 5$.

over multihop ad hoc networks. We have introduced the notion of gateway nodes used to control the rates for multirate multicast groups and provided the optimization model that realizes the optimal rates used by each gateway node in order to maximize the overall aggregate utility for the entire system. We also discussed the cross-layer architecture that can be used for deploying this algorithm in real networks. We proposed a mechanism for calculating the subtree price based on the branch accumulated price which allows the calculation to occur in a totally distributed and

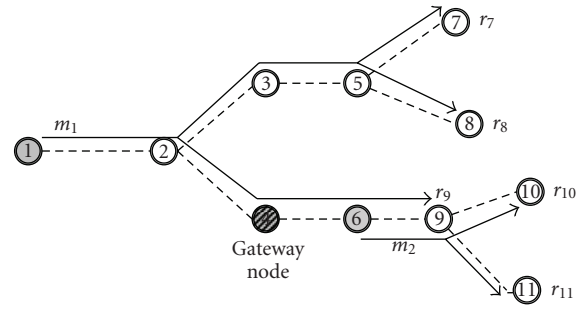
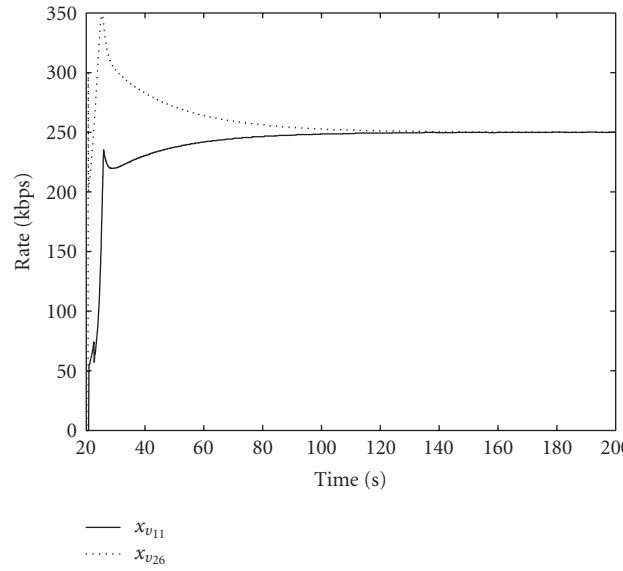
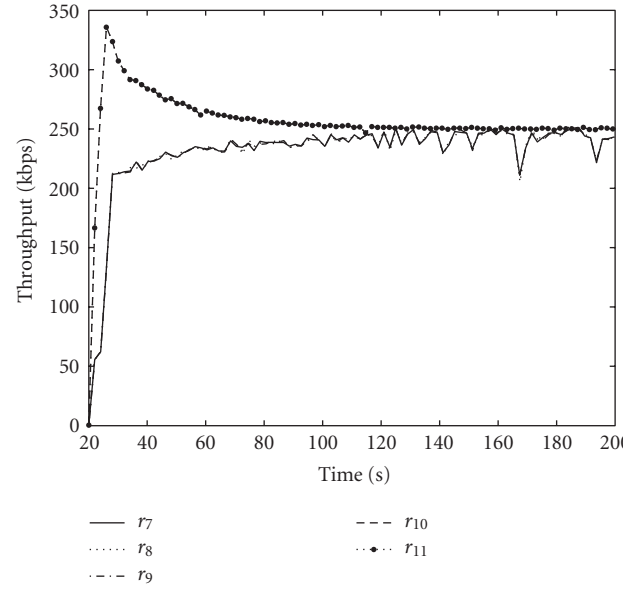


Figure 12: Multirate multicast network topology.

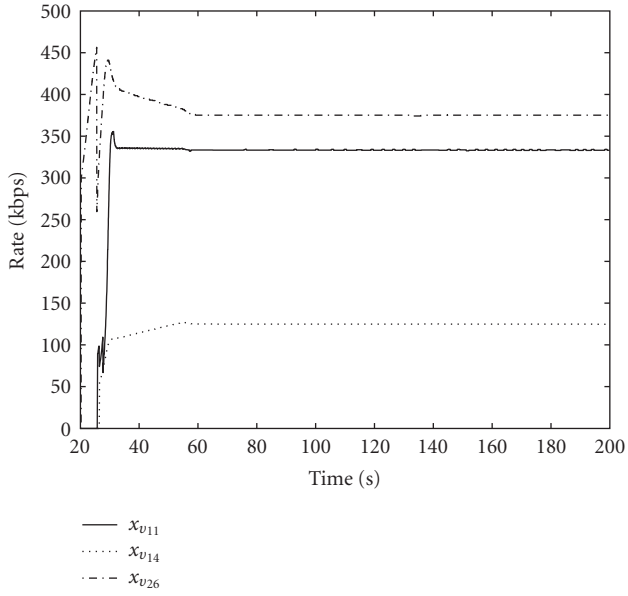


(a) Calculated rates

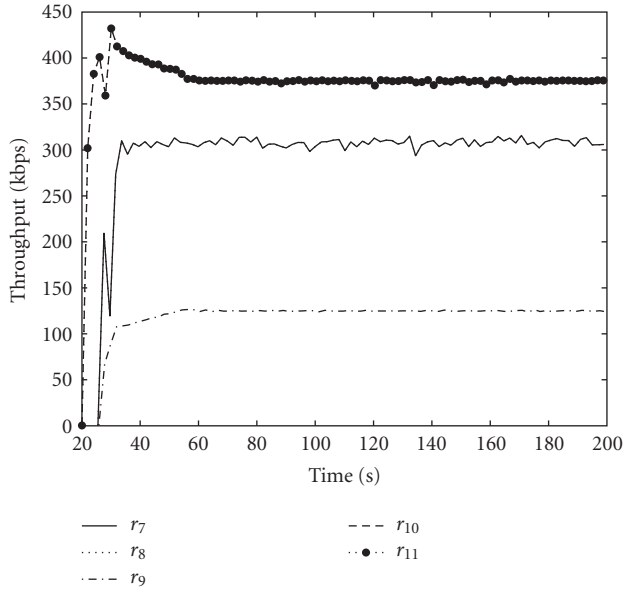


(b) Receiver throughput

Figure 13: Case 1 (unirate): calculated rate and throughput without using gateway node v_{14}.



(a) Calculated rates



(b) Receiver throughput

Figure 14: Case 2 (multirate): calculated rate and throughput using gateway node v_{14} for rate control on m_1.

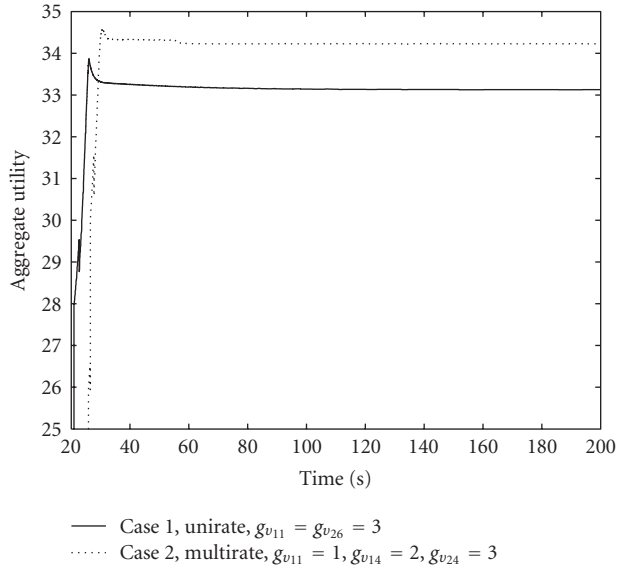


Figure 15: Aggregate utilities for cases 1 (unirate) and 2 (multirate).

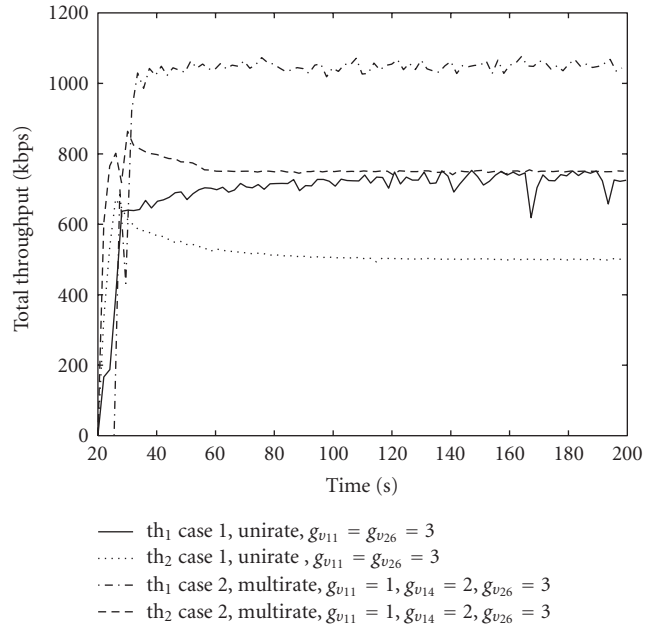


Figure 16: Total throughput for each multicast group for cases 1 and 2: th_1 is total throughput for m_1, th_2 is total throughput for m_2.

asynchronous way. Utilizing the flexibility of using gateway nodes across the multicast trees, ORAHWM is expected to increase the aggregate utility of the system and boost the overall throughput achieved by each multicast group by as high as 30% provided that the differentiation gains are set appropriately.

Appendices

A. Proof of Theorem 1

Proof. Assume that $\Im_v(h)$ is the set of all nodes $i \in T_v$ such that the depth of i is h, and H is the maximum depth of the subtree T_v. Now it is easy to recognize that

$$\lambda_v = \lambda_{vv} + \sum_{i \in \Im_v(1)} \lambda_{vi} + \cdots + \sum_{i \in \Im_v(H-1)} \lambda_{vi}, \tag{A.1}$$

where λ_{vv} is the clique price for subflow $f_{vv} \in F_v$ branching from the gateway node v. Next, we proceed by induction based on H as follows.

(i) For $H = 1$,

$$\sum_{i \in \Im_v} \lambda_v(i) = \sum_{i \in \Im_v(1)} \lambda_v(i) = \frac{K_{vv} \times \lambda_{vv}}{K_{vv}} = \lambda_v. \tag{A.2}$$

(ii) For $H = 2$,

$$\begin{aligned}\sum_{i\in\mathfrak{I}_v(2)}\lambda_v(i) &= \sum_{i\in\mathfrak{I}_v(2)}\frac{\lambda_{mv}/K_{mv}+\lambda_{m\pi_v(i)}}{K_{m\pi_v(i)}}\\ &= \sum_{i\in\mathfrak{I}_v(1)}\frac{\lambda_{vv}/K_{vv}+\lambda_{vi}}{K_{vi}}\times K_{vi}\\ &= \lambda_{vv}+\sum_{i\in\mathfrak{I}_v(1)}\lambda_{vi}=\lambda_v.\end{aligned}\tag{A.3}$$

Notice that if $f_{vi}\notin F_v$, then $\lambda_{vi}=0$.

(iii) Assume that for $H = n-1$,

$$\sum_{i\in\mathfrak{I}_v(n-1)}\lambda_v(i)=\lambda_v=\lambda_{vv}+\cdots+\sum_{i\in\mathfrak{I}_v(n-2)}\lambda_{vi},\tag{A.4}$$

(iv) Hence, for $H = n$,

$$\begin{aligned}\sum_{i\in\mathfrak{I}_v(n)}\lambda_v(i) &= \sum_{i\in\mathfrak{I}_v(n-1)}\frac{\lambda_v(i)+\lambda_{vi}}{K_{vi}}\times K_{vi}\\ &= \lambda_{vv}+\cdots+\sum_{i\in\mathfrak{I}_v(n-1)}\lambda_{vi}=\lambda_v,\end{aligned}\tag{A.5}$$

therefore the result follows. □

B. Proof of Theorem 2

The proof follows the same way as [4, Theorem 1]. We define $\widetilde{\Lambda}$ to be the set of gateway nodes that have $\Lambda_m(v)\neq\varnothing$. Then the vector of forwarding prices p' is defined as $p'=(p'_v,\forall v\in\widetilde{\Lambda})$. First we prove the following lemma.

Lemma 1. *If $\hat{u},\overline{u}=(p,p')$ are any two $(|Q|+|\widetilde{\Lambda}|)\times 1$ feasible system price vectors, that is, $\hat{u},\overline{u}\geq 0$, then based on Assumptions 1 and 2, ∇D satisfies the Lipscitch condition*

$$\left\|\nabla D(\hat{u})-\nabla D(\overline{u})\right\|_2\leq\frac{\overline{YZ}}{\overline{\gamma}}\|\hat{u}-\overline{u}\|_2.\tag{B.1}$$

Proof. From (15), we have $\nabla D=\widehat{C}-\widehat{\Gamma}x$, where $\widehat{C}$ is the $(|Q|+|\widetilde{\Lambda}|)\times 1$ capacity vector with $\hat{c}_i=0\ \forall i\in\widetilde{\Lambda}$, and $\widehat{\Gamma}$ is the $(|Q|+|\widetilde{\Lambda}|)\times|\Upsilon|$ constraints matrix.

Let $(\partial x/\partial u)(u)$ denote the $|\Upsilon|\times(|Q|+|\widetilde{\Lambda}|)$ matrix whose (i,j) element $(\partial x_i/\partial u_j)(u)$ is

$$\frac{\partial x_i}{\partial u_j}(u)=\begin{cases}\dfrac{\widehat{\Gamma}_{ji}}{U_i''(x_i(u))}, & \text{if } U_i'(W_i)\leq u_j\leq U_i'(w_i),\\ 0, & \text{o.w.}\end{cases}\tag{B.2}$$

If we define $\beta_i(u)$ as follows:

$$\beta_i(u)=\begin{cases}-\dfrac{1}{U_i''(x_i(u))}, & \text{if } U_i'(W_i)\leq u_j\leq U_i'(w_i),\\ 0, & \text{o.w,}\end{cases}\tag{B.3}$$

then $(\partial x_i/\partial u_j)(u)$ in matrix form can be written as

$$\left[\frac{\partial x_i}{\partial u_j}(u)\right]=-B(u)\,\widehat{\Gamma}^T,\tag{B.4}$$

where $B(u)=\mathrm{Diag}(\beta_i(u);\ i\in\Upsilon)$ is the diagonal matrix with diagonal elements $\beta_i(u)$. Hence,

$$\nabla^2 D=-\widehat{\Gamma}\left[\frac{\partial x_i}{\partial u_j}(u)\right]=\widehat{\Gamma}\,B(u)\,\widehat{\Gamma}^T.\tag{B.5}$$

Now from [18, Proposition A.25(e)] and knowing that $\nabla^2 D=\widehat{\Gamma}\,B(p)\,\widehat{\Gamma}^T$ is symmetric (i.e., $\|\widehat{\Gamma}\,B(p)\,\widehat{\Gamma}^T\|_1=\|\widehat{\Gamma}\,B(p)\,\widehat{\Gamma}^T\|_\infty$), then we have

$$\begin{aligned}\left\|\widehat{\Gamma}\,B(u)\,\widehat{\Gamma}^T\right\|_2 &\leq \left\|\widehat{\Gamma}\,B(u)\,\widehat{\Gamma}^T\right\|_\infty\\ &= \max_j\sum_{j'}\left[\widehat{\Gamma}B(u)\widehat{\Gamma}^T\right]_{jj'}\\ &= \max_j\sum_{j'}\sum_i\beta_i(u)\widehat{\Gamma}_{ji}\widehat{\Gamma}_{j'i}\\ &= \max_j\sum_i\beta_i(u)\widehat{\Gamma}_{ji}\sum_{j'}\widehat{\Gamma}_{j'i},\end{aligned}\tag{B.6}$$

where $\sum_{j'}\widehat{\Gamma}_{j'i}$ represents the sum of subflows in each maximal clique $j'\ \forall j'\ :\ (F_i\cap V_c^{j'}\neq\varnothing)$ plus the number of children gateway nodes for each subtree T_i, which is by definition $\leq\overline{Y}$. Then we have

$$\left\|\widehat{\Gamma}\,B(u)\,\widehat{\Gamma}^T\right\|_2\leq\frac{\overline{YZ}}{\overline{\gamma}}.\tag{B.7}$$

From [19, Theorem 9.19] we have for (B.7)

$$\left\|\nabla D(\hat{u})-\nabla D(\overline{u})\right\|_2\leq\frac{\overline{YZ}}{\overline{\gamma}}\left\|\hat{u}-\overline{u}\right\|_2,\tag{B.8}$$

hence the result follows. □

Proof. (Theorem 2) from Lemma 1, the dual objective function D is lower bounded and ∇D is Lipschitz. Then, limit point u^* of the sequence $\{u(t)\}$ generated by the gradient projection algorithm for the dual problem is dual optimal provided that $0<\alpha<2\overline{\gamma}/\overline{YZ}$ (see [18, Proposition 3.4]).

Let $\{u(t)\}$ be a subsequence converging to u^*. Since $U_i'(x_i)$ is defined on a compact interval $[w_i,W_i]$, it is continuous and one-to-one (because of the strict concavity of $U_i(x_i)$). Thus, its inverse is continuous (see [19, Theorem 4.17]) and hence from (20), $x(u)$ is continuous. Therefore, $\lim_{t\to\infty}x(t)=x(u^*)$ and that proves the result of Theorem 2. □

Acknowledgment

This research was supported by a collaborative R&D grant from the Qatar National Research Fund Project no. NPRP 26-6-7-10.

References

[1] X. Li, S. Paul, and M. Ammar, "Layered video multicast with retransmissions (LVMR): evaluation of hierarchical rate control," in *Proceedings of the 17th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '98)*, vol. 3, pp. 1062–1072, San Francisco, Calif, USA, March-April 1998.

[2] Y. Cui, Y. Xue, and K. Nahrstedt, "Optimal resource allocation in overlay multicast," in *Proceedings of the 11th International Conference on Network Protocols (ICNP '03)*, pp. 71–81, Atlanta, Ga, USA, November 2003.

[3] E. Amir, S. McCanne, and R. Katz, "An active service framework and its application to real-time multimedia transcoding," *Computer Communication Review*, vol. 28, no. 4, pp. 178–189, 1998.

[4] S. H. Low and D. E. Lapsley, "Optimization flow control—part I: basic algorithm and convergence," *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 861–874, 1999.

[5] R. J. La and V. Anantharam, "Utility-based rate control in the internet for elastic traffic," *IEEE/ACM Transactions on Networking*, vol. 10, no. 2, pp. 272–286, 2002.

[6] Y. Xue, B. Li, and K. Nahrstedt, "Optimal resource allocation in wireless ad hoc networks: a price-based approach," Tech. Rep. UIUCDCS-R-2004-2505, University of Illinois at Urbana-Champaign, Urbana, Ill, USA, June 2004.

[7] K. Kar, S. Sarkar, and L. Tassiulas, "Optimization based rate control for multirate multicast sessions," in *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '01)*, vol. 1, pp. 123–132, Anchorage, Alaska, USA, April 2001.

[8] K. Kar, S. Sarkar, and L. Tassiulas, "A scalable low-overhead rate control algorithm for multirate multicast sessions," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 8, pp. 1541–1557, 2002.

[9] N. Z. Shor, *Minimization Methods for Non-Differentiable Functions*, Springer, Berlin, Germany, 1985.

[10] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, 2000.

[11] H. Gossain, N. Nandiraju, K. Anand, and D. P. Agrawal, "Supporting MAC layer multicast in IEEE 802.11 based MANETs: issues and solutions," in *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN '04)*, pp. 172–179, Tampa, Fla, USA, November 2004.

[12] H. Luo, S. Lu, and V. Bharghavan, "A new model for packet scheduling in multihop wireless networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 76–86, Boston, Mass, USA, August 2000.

[13] D. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, Mass, USA, 1999.

[14] A. Mohamed and H. Alnuweiri, "Optimal resource allocation for homogeneous wireless multicast," Tech. Rep., University of British Columbia, Vancouver, Canada, 2006, http://www.ece.ubc.ca/~amrm/tr/orawm2006_TR.pdf.

[15] S. H. Shah, K. Chen, and K. Nahrstedt, "Dynamic bandwidth management in single-hop ad hoc wireless networks," *Mobile Networks and Applications*, vol. 10, no. 1, pp. 199–217, 2005.

[16] E. M. Royer and C. E. Perkins, "Multicast operation of the ad-hoc on-demand distance vector routing protocol," in *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM '99)*, pp. 207–218, Seattle, Wash, USA, August 1999.

[17] A. Mohamed and H. Alnuweiri, "Cross-layer optimization framework for rate allocation in wireless multicast," in *Proceedings of IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS '06)*, pp. 1–10, Vancouver, Canada, October 2006.

[18] D. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Computation*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1989.

[19] W. Rudin, *Principles of Mathematical Analysis*, McGraw-Hill, New York, NY, USA, 1976.