1-1-2000

Efficient Multicast in Heterogeneous Networks of Workstations

Ran Libeskind-Hadas
Harvey Mudd College

Jeff R.K. Hartline '01
Harvey Mudd College

Efficient Multicast in Heterogeneous Networks of Workstations*

Ran Libeskind-Hadas Jeffrey Hartline

Department of Computer Science
Harvey Mudd College
Claremont, CA 91711
E-mail contact: hadas@cs.hmc.edu

Abstract

This paper studies the problem of efficient multicast in heterogeneous networks of workstations (HNOWs) using a parameterized communication model [3]. This model associates a sending overhead and a receiving overhead with each node as well as a network latency parameter. The problem of finding optimal multicasts in this model is known to be NP-complete in the strong sense. Nevertheless, we show that for two different properties that arise in typical HNOWs, provably near-optimal and optimal solutions, respectively, can be found in polynomial time. Specifically, we show the following two results: When the ratios of receiving overhead to sending overhead among the nodes is bounded by constants, solutions within a bounded ratio of optimal can be found in time $O(n \log n)$. Secondly, if the number of distinct types of workstations is fixed then optimal solutions can be found in polynomial time. These results provide a practical means of finding optimal and provably near-optimal multicast schedules in a large class of frequently occurring heterogeneous networks of workstations.

1 Introduction

Networks of workstations (NOWs) have been shown to provide an inexpensive alternative to massively parallel processors (MPPs) [1, 6, 15]. An important problem in both MPPs and NOWs is that of efficient support for multicast communication, which is used in a wide variety of both parallel applications and system-level operations. In many cases, only point-to-point communication is supported by the system and multicast communication must be implemented as a collection of point-to-point messages [2]. Given a source and destination nodes, the *optimal multicast problem* is that of finding a schedule of point-to-point message transmissions that minimizes the elapsed time until all the destination nodes have received the message. Efficient algorithms are known for the optimal multicast problem in a number of communication models including the *one-port model* [11], the *postal model* [4], the *LogP model* [8], and extensions of these models [14]. These results are for *homogeneous networks* in which all nodes are assumed to have identical communication latency parameters.

In a NOW, the constituent workstations, networking devices, and communication protocols may be heterogeneous, resulting in varying computation and communication speeds [2, 5]. Such networks are called *heterogeneous networks of workstations (HNOWs)*. The presence of heterogeneity significantly complicates the problem of finding optimal multicast schedules. Banikazemi et al. [2] and Hall et al. [9] have independently proposed a model of heterogeneous communication, henceforth referred to as the *heterogeneous node model*, in which each node x has an associated *message initiation cost*, $c(x)$. In this model node x incurs its message initiation cost to send the message to any destination node y. Thus, at time $c(x)$, node y receives the message and may begin sending the message to another destination node, incurring its message initiation cost $c(y)$. Concurrently, node x may send the message to another node, again incurring its message initiation cost $c(x)$.

While Banikazemi et al. [2] showed experimentally that a polynomial time greedy algorithm generally finds near-optimal multicast schedules in the heterogeneous node model, Hall et al. [9] showed that the problem of finding optimal multicasts in this model, as

*This work was supported by the National Science Foundation under grant CCR-9900491. Parts of this work were completed while the first author was a visitor at the Department of Computer Science, Technion, Israel Institute of Technology and at the Department of Computer Science, University of Pittsburgh.

0-7695-0771-9/00 $10.00 © 2000 IEEE

well as several other related models, is NP-complete. Libeskind-Hadas et al. proved that the greedy algorithm finds schedules which are within a factor of two of optimal and that this bound can be further improved under certain conditions [13].

In related work, Bhat et al. have proposed an alternative model that accounts for heterogeneity in both the nodes and the network [5]. This model is particularly well-suited for wide-area networks where network latencies over "long haul" links may be very different from those within a local area network. Itkis et al. have studied multicasting in a model in which all nodes are identical but a node may select one of several different communication "services" each time it sends a message, where each service has an associated and price [10].

Banikazemi et al. have recently observed that more complex parameterized communication models are required to accurately characterize the performance of heterogeneous NOWs [3]. They have proposed a new model, henceforth referred to as the *heterogeneous receive-send model*, which associates both a *sending overhead* and a *receiving overhead* with each node. The sending overhead is the time incurred by the node upon sending the message while the receiving overhead is the time incurred by the node upon receiving the message. While a node incurs the sending or receiving overhead, it cannot perform other communication operations. In addition, a global parameter L specifies the network latency incurred in sending the message between any two nodes. Thus, the network is assumed to comprise a heterogeneous collection of nodes interconnected using a single type of network[1]. Banikazemi et al. have verified the accuracy of this model using a heterogeneous NOW testbed.

Figure 1 shows an example of two different schedules for the same instance of the multicast problem in this model. In this example "fast" nodes have sending and receiving overheads of 1 while "slow" nodes have sending overheads of 2 and receiving overheads of 3. The source is a slow node and there are three fast and one slow destination nodes. The network latency, L, is 1. The number in brackets next to each node indicates the time at which the node receives the message in the schedule. For example, in the schedule in Figure 1(a), the source node first sends the message to a fast node. This first transmission incurs a sending overhead of 2 at the source node, followed by the network latency of 1 to transmit the message, followed a receiving overhead of 1 at the destination. Thus, this fast node receives the message at time 4. After the source node has incurred the sending overhead of 2, it immediately begins sending the message to another fast node. Thus, at time 2 the source node again incurs its sending overhead of 2, the message takes 1 unit of time to reach the second fast node, and that node incurs a receiving overhead of 1. Thus, the second fast node receives the message from the source at time 6. Similarly, the fast node that receives the message at time 4 sends the message to a fast node and then to a slow node. The fast child receives the message at time $4 + 1 + 1 + 1 = 7$ and the slow child receives the message at time $5 + 1 + 1 + 3 = 10$.

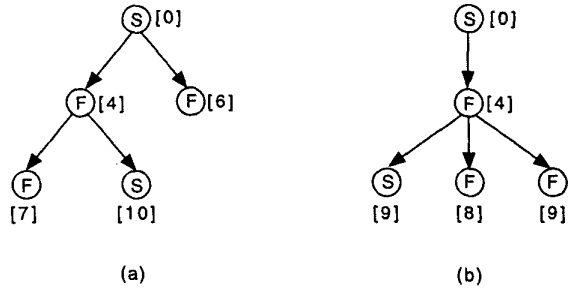

Figure 1: Two schedules for a multicast from a slow node to three fast destinations and one slow destination. Fast nodes have sending and receiving overheads of 1 while slow nodes have sending overhead of 2 and receiving overhead of 3. The network latency is 1. Numbers in brackets next to each node denote time of receipt of the message. (a) A schedule completing at time 10. (b) A schedule completing at time 9.

In this paper we show that although the optimal multicast problem for the heterogeneous receive-send model is NP-complete in the strong sense, a polynomial time approximation algorithm exists for HNOWs that arise under a condition that is likely to be satisfied in virtually all cases in practice. In addition, optimal solutions can be found in polynomial time for another class of HNOWs that frequently arise in practice.

The first result addresses HNOWs in which the ratio of the receiving overhead to the sending overhead in each node is upper- and lower-bounded by constants. More precisely, let the *receive-send* ratio of a node denote its receiving overhead divided by its sending overhead. Let $\alpha_{\min}$ and $\alpha_{\max}$ be constants for a given network such that all receive-send ratios in the network are between $\alpha_{\min}$ and $\alpha_{\max}$. Let n denote the

[1]The model in [3] considers both fixed and message-length dependent components for the overheads and network latency. For a multicast with any given message length, we may combine the fixed and message-length dependent components as is done here.

number of nodes participating in a given multicast and let OPTR denote the completion time of an optimal schedule for the multicast. We show that an $O(n \log n)$ greedy algorithm finds a schedule whose completion time is at most $C \times \text{OPTR} + \beta$ where C is a constant computed from $\alpha_{\min}$ and $\alpha_{\max}$ and β is the difference of the maximum and minimum receiving overheads of the destination nodes.

This result therefore provides a polynomial time approximation algorithm under the condition that the receive-send ratios are upper- and lower-bounded by constants. The receive-send ratios depend on factors such as the hardware capabilities of the nodes, the protocols employed, and the length of the message being sent. Recently reported benchmarks have found receive-send ratios in the range of 1.05 to 1.85 [3, 7]. Although the ratio-bound of the proposed approximation algorithm is not known to be tight, the existence of an approximation algorithm provides a theoretical basis for using the greedy algorithm. In addition, this result suggests that the search for tighter bounds or other approximation algorithms is a worthwhile endeavor.

The second result addresses HNOWs with limited heterogeneity; networks that comprise an arbitrary number of workstations but a limited number of distinct types of workstations. In many cases, the number of distinct types of workstations remains constant even as the size of the network scales. We show that for a network involving n nodes of k distinct types, an optimal multicast schedule can be found in time $O(n^{2k})$. In fact, in time $O(n^{2k})$ a table of optimal schedules for all possible multicasts can be constructed. Thus, for small k it may be desirable to precompute this table so that optimal schedules can later be found in constant time.

The remainder of this paper is organized as follows. In Section 2 we give definitions and results that are used throughout the paper. In Section 3 we analyze a greedy approximation algorithm for the multicast problem. In Section 4 we describe and analyze a polynomial time algorithm for the case of limited heterogeneity. We conclude in Section 5 with some directions for future research.

2 Preliminaries

We consider the *heterogeneous receive-send communication model* which comprises the following set of parameters:

- A network *latency*, L, incurred in communicating a message from one node to another.
- A *sending overhead*, $o_{\text{send}}(p)$, defined as the time incurred by node p to send a message. During this time node p cannot perform other communication operations.
- A *receiving overhead*, $o_{\text{receive}}(p)$, defined as the time incurred by node p to receive a message. During this time node p cannot perform other communication operations.

We note that the sending and receiving overheads may depend on the length of the message being sent. Thus, these values are computed for each node for the length of the given multicast message. We assume that all of the above parameters are measured in the same time units and have positive integer values.

A *multicast set* is a set $S = \{p_0, \ldots, p_n\}$ where p_0 is the source of the multicast and the remaining elements are destination nodes. We assume that the sending and receiving overheads are directly correlated to the speed of a node so that for any two nodes $p, q \in S$, $o_{\text{send}}(p) < o_{\text{send}}(q)$ if and only if $o_{\text{receive}}(p) < o_{\text{receive}}(q)$. We henceforth assume that $p_1, \ldots, p_n$ are indexed in order of non-decreasing overhead so that $o_{\text{send}}(p_i) \leq o_{\text{send}}(p_{i+1})$ and $o_{\text{receive}}(p_i) \leq o_{\text{receive}}(p_{i+1})$ for $1 \leq i < n$.

For a given multicast set S, a *multicast schedule* is a directed tree T with one vertex for each node in S. The root of T corresponds to the source node and all remaining vertices correspond to destination nodes. Henceforth, we use "node" and "vertex" interchangeably. Each non-root vertex v has exactly one incoming edge representing transmission of the message to v. Each non-leaf vertex v has one or more outgoing edges corresponding to transmissions of the message from v to other destination vertices. These edges are ordered from left to right to indicate the order, from first to last, in which v transmits the message to its children. Alternatively, we say that a list $(w_1, \ldots, w_\ell)$ is the *delivery ordered* list of children of v if v sends the message to vertex w_i before sending to vertex w_{i+1}, $1 \leq i < \ell$.

The *delivery time* for a non-root vertex v in schedule T, denoted $d_T(v)$, is the time at which the message is delivered to vertex v. The *reception time* for vertex v in schedule T, denoted $r_T(v)$, is equal to $d_T(v) + o_{\text{receive}}(v)$; the time at which vertex v has completed incurring its receiving overhead. Although d_T and r_T are related, it will be convenient in our analysis to distinguish between these two functions. Define the *delivery completion time* to be $D_T = \max_{v \in T} d_T(v)$ and the *reception completion time* to be $R_T = \max_{v \in T} r_T(v)$. The objective of the optimal

multicast problem in this model is to find a schedule T for multicast set S such that the reception completion time, R_T, is minimized. The optimal multicast problem for this model, under the assumptions above, is easily shown to NP-complete in the strong sense. The interested reader is referred to [12] for details.

By definition, in any schedule T the root has the message at time 0, so $r_T(p_0) = 0$. For each non-leaf vertex v with the delivery ordered list of children $(w_1, \ldots, w_\ell)$, $d_T(w_i) \geq r_T(v) + i \times o_{\text{send}}(v) + L$, $1 \leq i \leq \ell$. Without loss of generality, we assume that $d_T(w_i) = r_T(v) + i \times o_{\text{send}}(v) + L$, $1 \leq i \leq \ell$, since otherwise the idle time can be removed without increasing the delivery time of any vertex. We say that v *completes delivery* to its i^{th} child at time $r_T(v) + i \times o_{\text{send}}(v) + L$.

Next, we consider a simple greedy algorithm for constructing multicast schedules similar to the algorithm proposed for the heterogeneous node model [2, 9]. Let S be a multicast set $\{p_0, p_1, \ldots, p_n\}$. Recall that by convention p_0 is the source and that the destinations are indexed in non-decreasing order of overhead.

```
Let T be the tree with a single node p_0.
for i = 1 to n
  begin
    Find a vertex p ∈ T that can complete delivery
      as early as possible.
    Let p send the message to p_i, thereby inserting p_i
      into T.
  end
return T
```

Lemma 1 *The running time of the greedy algorithm for a multicast set of n nodes is $O(n \log n)$.*

Proof: The algorithm requires that the n destination nodes first be sorted in non-decreasing order of overheads. This can be done in $O(n \log n)$ time. The nodes in schedule T can be maintained in a priority queue in which the key associated with each element in the priority queue is the next earliest delivery time of the message. Initially, the source node p_0 is inserted into an empty priority queue with the key equal to $o_{\text{send}}(p_0) + L$ since this is the first possible delivery time for the message. At each iteration i of the algorithm, the node p with the smallest key is removed from the priority queue. Let C denote the value of the key for node p. Node p_i is now inserted into the priority queue with key equal to $C + o_{\text{receive}}(p_i) + o_{\text{send}}(p_i) + L$. Next, node p is reinserted into the priority queue with key equal to $C + o_{\text{send}}(p)$. The process is repeated n times. By using a heap to implement the priority queue, each deletion and pair of insertions performed per iteration can be accomplished in $O(\log n)$ time. Thus the total running time is $O(n \log n)$. □

A multicast schedule T is said be *layered* if for every pair of non-root nodes $u, v \in T$ if $o_{\text{send}}(u) < o_{\text{send}}(v)$ (or equivalently $o_{\text{receive}}(u) < o_{\text{receive}}(v)$) then $d_T(u) \leq d_T(v)$. Note that by definition, every schedule produced by the greedy algorithm is layered. The following lemma and its corollary show that the greedy algorithm finds a schedule with minimum delivery completion time among all layered schedules.

Lemma 2 *Let $S = \{p_0, \ldots, p_n\}$ and $S' = \{p'_0, \ldots, p'_n\}$ be two multicast sets with sources p_0 and p'_0, respectively, with destination nodes appearing in non-decreasing order of overhead. Assume that $o_{send}(p_i) \leq o_{send}(p'_i)$ and $o_{receive}(p_i) \leq o_{receive}(p'_i)$ for $0 \leq i \leq n$. Let T denote the schedule for S found by the greedy algorithm and let T' denote any layered schedule for S'. Then $D_T \leq D_{T'}$.*

Proof: Since two nodes with the identical overhead parameters can be interchanged without affecting delivery times in the schedule, without loss of generality if $o_{\text{send}}(p_i) = o_{\text{send}}(p_j)$ and $i < j$, then $d_T(p_i) \leq d_T(p_j)$. Similarly, if $o_{\text{send}}(p'_i) = o_{\text{send}}(p'_j)$ and $i < j$, then $d_{T'}(p'_i) \leq d_{T'}(p'_j)$. Then, since T and T' are layered, $d_T(p_i) \leq d_T(p_{i+1})$ and $d_{T'}(p'_i) \leq d_{T'}(p'_{i+1})$, $1 \leq i < n$.

By way of contradiction, assume that $D_{T'} < D_T$. Since T and T' are layered, $D_{T'} = d_{T'}(p'_n)$ and $D_T = d_T(p_n)$ and thus $d_{T'}(p'_n) < d_T(p_n)$. Let $1 \leq j \leq n$ be the smallest index such that $d_{T'}(p'_j) < d_T(p_j)$. Then $d_T(p_i) \leq d_{T'}(p'_i)$ for all i such that $1 \leq i \leq j-1$ and $d_{T'}(p'_j) < d_T(p_i)$ for all i such that $j \leq i \leq n$. Therefore, in schedule T, $p_0, \ldots, p_{j-1}$ collectively complete at most $j-1$ message deliveries by time $d_{T'}(p'_j)$.

Let k denote the number of message transmissions in T' collectively completed by nodes $p'_0, \ldots, p'_{j-1}$ by time $d_{T'}(p'_j)$. Since T' is layered, the message is delivered to p'_j from some p'_i, $0 \leq i \leq j-1$. Therefore, nodes $p'_0, \ldots, p'_{j-1}$ collectively complete at least j message transmissions by time $d_{T'}(p'_j)$ and thus $k \geq j$. However, since $o_{\text{send}}(p_i) \leq o_{\text{send}}(p'_i)$ and $o_{\text{receive}}(p_i) \leq o_{\text{receive}}(p'_i)$, $0 \leq i \leq j-1$, and $d_T(p_i) \leq d_{T'}(p'_i)$, $1 \leq i \leq j-1$, nodes $p_0, \ldots, p_{j-1}$ in T collectively have at least k points at which message transmissions can be completed before time $d_{T'}(p'_j)$. Since the greedy algorithm performs message deliveries as early as possible, $p_0, \ldots, p_{j-1}$ collectively complete at least $k \geq j$ message transmissions by time $d_{T'}(p'_j)$, contradicting the observation above that these nodes complete at most $j-1$ message deliveries by time $d_{T'}(p'_j)$. □

Corollary 1 *For any multicast set S, the schedule produced by the greedy algorithm has the minimum D_T over all layered schedules T.*

Proof: Follows immediately from Lemma 2 by letting $S = S'$. □

3 A Bound for the Greedy Algorithm

In this section we show that the greedy algorithm is an approximation algorithm for the multicast problem when the receive-send ratios are constant bounded. In particular, for a multicast set $S = \{p_0, p_1, \ldots, p_n\}$ let $\alpha_i = \frac{o_{\text{receive}}(p_i)}{o_{\text{send}}(p_i)}$, $0 \leq i \leq n$. Let $\alpha_{\max} = \max_{0 \leq i \leq n} \alpha_i$ and let $\alpha_{\min} = \min_{0 \leq i \leq n} \alpha_i$. Let $\beta = \max_{1 \leq i \leq n} o_{\text{receive}}(p_i) - \min_{1 \leq i \leq n} o_{\text{receive}}(p_i)$. Finally, let OPTR denote the reception completion time of an optimal schedule. We show that the greedy algorithm constructs a schedule with reception completion time of less than $2\frac{\lceil \alpha_{\max} \rceil}{\alpha_{\min}}\text{OPTR} + \beta$. As a special case, note that if the sending overhead is equal to the receiving overhead in each node then $\alpha_{\max} = \alpha_{\min} = 1$ and the bound becomes $2 \times \text{OPTR} + \beta$. The following lemma will be used to establish this bound.

Lemma 3 *Let $S = \{p_0, p_1, \ldots, p_n\}$ be a multicast set and assume that there exists a positive integer C such that $\frac{o_{receive}(p_i)}{o_{send}(p_i)} = C$ for all i, $0 \leq i \leq n$. Let T be a schedule for S and let u, v be two non-root nodes in T such that $d_T(u) < d_T(v)$ and $o_{send}(u) = \ell \times o_{send}(v)$ for some positive integer $\ell \geq 2$. Then there exists a schedule T' satisfying the following properties:*

1. *$d_{T'}(u) > d_{T'}(v)$.*
2. *$d_T(w) = d_{T'}(w)$ for all $w \in S$ such that w is not a descendant of u or v in T.*
3. *$D_{T'} \leq D_T$.*

Proof: Let $\{u_1, \ldots, u_x\}$ and $\{v_1, \ldots, v_y\}$ denote the delivery ordered list of children of nodes u and v, respectively, in schedule T. Let $t_i = (C+i) \times \ell - C - 1$, $1 \leq i \leq x$. Schedule T' is constructed from T as follows: Nodes u and v are exchanged. The delivery ordered list of children of v becomes

$$v_1, \ldots, v_{t_1}, u_1, v_{t_1+2}, \ldots, v_{t_2}, u_2, \ldots, u_i, v_{t_i+2}, \ldots, v_{t_{i+1}}, u_{i+1}, \ldots, u_x, v_{t_x+2}, \ldots, v_y.$$

The delivery ordered list of children of u becomes

$$v_{t_1+1}, v_{t_2+1}, \ldots, v_{t_i+1} \ldots, v_{t_x+1}.$$

In the special case that v is a child of u, and thus $v = u_i$ for some i, $1 \leq i \leq x$, the construction is altered so that in T' v sends to u in place of sending the message to u_i.

The first two properties of the lemma are satisfied by construction of T'. We now show that the third property, $D_{T'} \leq D_T$, is satisfied. By construction, $d_{T'}(v) = d_T(u)$. We begin by showing that $d_{T'}(u_i) = d_T(u_i)$, $1 \leq i \leq x$. Observe that

$$\begin{aligned} d_T(u_i) &= d_T(u) + o_{\text{receive}}(u) + i \times o_{\text{send}}(u) + L \\ &= d_T(u) + C \times o_{\text{send}}(u) + i \times \ell \times o_{\text{send}}(v) + L \\ &= d_T(u) + C \times \ell \times o_{\text{send}}(v) + i \times \ell \times o_{\text{send}}(v) + L \\ &= d_T(u) + (C+i) \times \ell \times o_{\text{send}}(v) + L \end{aligned}$$

while

$$\begin{aligned} d_{T'}(u_i) &= d_{T'}(v) + o_{\text{receive}}(v) + (t_i + 1) \times o_{\text{send}}(v) + L \\ &= d_T(u) + C \times o_{\text{send}}(v) + \\ &\quad [(C+i) \times \ell - C - 1 + 1] \times o_{\text{send}}(v) + L \\ &= d_T(u) + (C+i) \times \ell \times o_{\text{send}}(v) + L. \end{aligned}$$

Thus, $d_{T'}(u_i) = d_T(u_i)$, $1 \leq i \leq x$.

Next we show that $d_{T'}(u) = d_T(v)$. If u is not an ancestor of v in T then $d_{T'}(u) = d_T(v)$ by construction of T'. If u is the parent of v in T then $v = u_i$ for some i, $1 \leq i \leq x$. From the above analysis, $d_{T'}(u_i) = d_T(u_i) = d_T(v)$. Since by construction of T', u is inserted in place of u_i in this case, we have $d_{T'}(u) = d_T(v)$. Finally, if v is a descendant of u but not the child of u in T, then v is a descendant of some u_i in T, $1 \leq i \leq x$. Since $d_{T'}(u_i) = d_T(u_i)$, it again follows that $d_{T'}(u) = d_T(v)$.

Next we show that $d_{T'}(v_{t_i+1}) = d_T(v_{t_i+1})$, $1 \leq i \leq x$. Observe that

$$\begin{aligned} d_T(v_{t_i+1}) &= d_T(v) + o_{\text{receive}}(v) + (t_i + 1) \times o_{\text{send}}(v) + L \\ &= d_T(v) + C \times o_{\text{send}}(v) + \\ &\quad [(C+i) \times \ell - C - 1 + 1] \times o_{\text{send}}(v) + L \\ &= d_T(v) + (C+i) \times \ell \times o_{\text{send}}(v) + L \end{aligned}$$

while

$$\begin{aligned} d_{T'}(v_{t_i+1}) &= d_{T'}(u) + o_{\text{receive}}(u) + i \times o_{\text{send}}(u) + L \\ &= d_T(v) + C \times o_{\text{send}}(u) + i \times \ell \times o_{\text{send}}(v) + L \\ &= d_T(v) + C \times \ell \times o_{\text{send}}(v) + i \times \ell \times o_{\text{send}}(v) + L \\ &= d_T(v) + (C+i) \times \ell \times o_{\text{send}}(v) + L. \end{aligned}$$

Thus, $d_{T'}(v_{t_i+1}) = d_T(v_{t_i+1})$, $1 \leq i \leq x$.

Finally, we show that $d_{T'}(v_i) \leq d_T(v_i)$, $1 \leq i \leq y$. If $i = t_j + 1$ for some j, $1 \leq j \leq x$, then $d_{T'}(v_i) =$

$d_T(v_i)$ from the argument above. For all remaining v_i, note that v_i is the i^{th} child of v in T and is the i^{th} child of v in T'. Since $d_{T'}(v) < d_T(v)$, it follows immediately that $d_{T'}(v_i) < d_T(v_i)$ for all remaining v_i.

Thus, $d_{T'}(u) = d_T(v)$, $d_{T'}(v) = d_T(u)$, and the delivery time of every child of u or v is no larger in T' than in T. Therefore, the delivery time of every descendant of u or v is no larger in T' than in T. Since the delivery times of all other nodes are unaltered by this transformation, it follows that $D_{T'} \leq D_T$. □

Theorem 1 *Let $S = \{p_0, p_1, \ldots, p_n\}$ be a multicast set. Let $\alpha_i = \frac{o_{receive}(p_i)}{o_{send}(p_i)}$, $0 \leq i \leq n$. Let $\alpha_{\max} = \max_{0 \leq i \leq n} \alpha_i$, let $\alpha_{\min} = \min_{0 \leq i \leq n} \alpha_i$, and let $\beta = \max_{1 \leq i \leq n} o_{receive}(p_i) - \min_{1 \leq i \leq n} o_{receive}(p_i)$. Let OPTR denote the minimum reception completion time over all schedules for S. The greedy algorithm constructs a schedule with reception completion time less than $2\frac{\lceil \alpha_{\max} \rceil}{\alpha_{\min}} OPTR + \beta$.*

Proof: Let S' be the multicast set constructed as follows: For each $p_i \in S$, introduce a corresponding p'_i in set S' such that $o_{\text{send}}(p'_i) = 2^k$ for the smallest integer value k such that $2^k \geq o_{\text{send}}(p_i)$. Let $o_{\text{receive}}(p'_i) = \lceil \alpha_{\max} \rceil \times o_{\text{send}}(p'_i)$. Note that $\frac{o_{\text{send}}(p'_i)}{o_{\text{send}}(p_i)} < 2$. Similarly,

$$\frac{o_{\text{receive}}(p'_i)}{o_{\text{receive}}(p_i)} < 2 \times \frac{\lceil \alpha_{\max} \rceil}{\alpha_i} \leq 2 \times \frac{\lceil \alpha_{\max} \rceil}{\alpha_{\min}}.$$

Let OPTR′ denote the minimum reception completion time over all schedules for S'. Let OPTD and OPTD′ denote the minimum delivery completion times over all schedules for S and S', respectively. Similarly, let GREEDYR and GREEDYR′ denote the reception completion times for schedules constructed by the greedy algorithm for S and S', respectively. Let GREEDYD and GREEDYD′ denote the delivery completion times for schedules constructed by the greedy algorithm for S and S', respectively.

Since the sending or receiving overhead of a node in S' is less than $2 \times \frac{\lceil \alpha_{\max} \rceil}{\alpha_{\min}}$ times larger than the sending or receiving overhead, respectively, of the corresponding node in S, it follows that

$$\text{OPTR}' < 2 \times \frac{\lceil \alpha_{\max} \rceil}{\alpha_{\min}} \times \text{OPTR}. \quad (1)$$

Next, since the reception time for a node is the sum of its delivery time and its receiving overhead

$$\text{OPTD}' + \min_{1 \leq i \leq n} o_{\text{receive}}(p_i) \leq \text{OPTR}'. \quad (2)$$

Combining (1) and (2) we have

$$\text{OPTD}' < 2 \times \frac{\lceil \alpha_{\max} \rceil}{\alpha_{\min}} \times \text{OPTR} - \min_{1 \leq i \leq n} o_{\text{receive}}(p_i). \quad (3)$$

Next, in S', $\frac{o_{\text{receive}}(p'_i)}{o_{\text{send}}(p'_i)} = \lceil \alpha_{\max} \rceil$ for all i, $0 \leq i \leq n$. In addition, if $o_{\text{send}}(u') > o_{\text{send}}(v')$ for $u', v' \in S'$ then $o_{\text{send}}(u') = \ell \times o_{\text{send}}(v')$ for $\ell = 2^k$ where k is an integer greater than 0. Therefore, Lemma 3 can be applied to any pair of nodes u', v' in a schedule for S' such that the delivery time of u' is less than that of v' but the sending overhead of u' is greater than that of v'. In particular, we may begin with a schedule for S' with delivery completion time OPTD′. If p'_1 does not have the minimum delivery time in this schedule, Lemma 3 is applied to swap p'_1 and a node of minimum delivery time. This process is performed sequentially, as necessary, for $p'_1, p'_2, \ldots, p'_n$. Since by Lemma 3 the transformation affects only the subtrees rooted at the exchanged nodes, when the transformation is applied so that p'_i has the i^{th} smallest delivery time, the delivery times of $p'_1, \ldots, p'_{i-1}$ remain unchanged. Thus, after at most n applications of the transformation, the schedule with delivery completion time OPTD′ is transformed into a layered schedule for S'. Since each application of the transformation does not increase the delivery completion time, the delivery completion time of the resulting layered schedule remains OPTD′. Thus, by Corollary 1,

$$\text{GREEDYD}' = \text{OPTD}'. \quad (4)$$

Since $o_{\text{send}}(p_i) \leq o_{\text{send}}(p'_i)$ and $o_{\text{receive}}(p_i) \leq o_{\text{receive}}(p'_i)$, $0 \leq i \leq n$, Lemma 2 implies that

$$\text{GREEDYD} \leq \text{GREEDYD}'. \quad (5)$$

Finally, since the reception time for a node is the sum of its delivery time and its receiving overhead,

$$\text{GREEDYR} \leq \text{GREEDYD} + \max_{1 \leq i \leq n} o_{\text{receive}}(p_i). \quad (6)$$

Combining equations (3) through (6) we have,

$$\begin{aligned} \text{GREEDYR} &< 2 \times \frac{\lceil \alpha_{\max} \rceil}{\alpha_{\min}} \times \text{OPTR} + \\ &\quad \max_{1 \leq i \leq n} o_{\text{receive}}(p_i) - \min_{1 \leq i \leq n} o_{\text{receive}}(p_i) \\ &= 2 \times \frac{\lceil \alpha_{\max} \rceil}{\alpha_{\min}} \times \text{OPTR} + \beta. \end{aligned}$$

□

Finally, we note that a slight modification should be made to the aforementioned greedy algorithm when

used in practice. Since the greedy algorithm constructs a layered schedule, "fast" nodes take delivery of the message before "slow" nodes in the schedule. While this property is evidently desirable for internal nodes in the schedule tree, it is not desirable for the leaf nodes. In other words, in order to minimize reception completion time a leaf node with small receiving overhead should not take delivery of the message before a leaf node with larger receiving overhead. Thus, once the greedy algorithm completes construction of the schedule, reversing the order of the leaf nodes will not increase the reception completion time and may decrease it.

4 A Polynomial Time Algorithm for Limited Heterogeneity

We now show that for any fixed constant k, the optimal multicast problem for n nodes of k distinct types can be solved in time $O(n^{2k})$. Let $\tau(s, i_1, \ldots, i_k)$ represent the minimum reception completion time for a multicast from a source of type s, $1 \leq s \leq k$ to i_j nodes of type j, $1 \leq j \leq k$. Let $S(i)$ and $R(i)$ denote the sending and receiving overheads, respectively, for a node of type i, $1 \leq i \leq k$. Our algorithm is based on the following lemma.

Lemma 4 *For every $1 \leq s \leq k$, $i_j \geq 0$, $1 \leq j \leq k$,*

$$\tau(s, 0, 0, \ldots, 0) = 0$$

$$\tau(s, i_1, \ldots, i_k) = \min_{1 \leq \ell \leq k, 0 \leq y_1 \leq i_1, \ldots, 0 \leq y_\ell \leq i_\ell - 1, \ldots, 0 \leq y_k \leq i_k} \max\{\tau(\ell, y_1, \ldots, y_\ell, \ldots, y_k) + S(s) + L + R(\ell), \tau(s, i_1 - y_1, \ldots, i_\ell - 1 - y_\ell, \ldots, i_k - y_k) + S(s)\}$$

Proof: The first equation is by definition. For the second equation, in any schedule in which a source node of type s sends to i_j nodes of type j, $1 \leq j \leq k$, the source node's first transmission is sent to some node of type ℓ, $1 \leq \ell \leq k$. This node of type ℓ is the root of a subtree containing $0 \leq y_j \leq i_j$ nodes of type j for each $1 \leq j \leq k$ with the exception of type ℓ, for which $y_\ell \leq i_\ell - 1$ since the selected node of type ℓ is one of the i_ℓ destination nodes for the original multicast. Node ℓ receives the message at time $S(s)+L+R(\ell)$ and an optimal schedule for the portion of the multicast rooted at ℓ completes after an additional $\tau(\ell, y_1, \ldots, y_\ell, \ldots, y_k)$ units of time. Once the source node has transmitted the message to its first destination, the source node can continue transmitting the message to the remaining destinations. The source node begins performing these remaining transmissions at time $S(s)$ and, by definition, the optimal solution for these remaining destinations takes time $\tau(s, i_1 - y_1, \ldots, i_\ell - 1 - y_\ell, \ldots, i_k - y_k)$. Therefore, the maximum of $\tau(\ell, y_1, \ldots, y_\ell, \ldots, y_k) + S(s) + L + R(\ell)$ and $\tau(s, i_1 - y_1, \ldots, i_\ell - 1 - y_\ell, \ldots, i_k - y_k) + S(s)$ is the actual completion time of the multicast. Finally, by computing the minimum over all possible values of $\ell, y_1, \ldots, y_k$, the completion time of an optimal schedule is found. □

Theorem 2 *For any constant k, given n nodes of k types an optimal multicast schedule can be found in time $O(n^{2k})$.*

Proof: The algorithm applies dynamic programming to the relation in Lemma 4. Specifically, let $n_1, n_2, \ldots, n_k$ represent the number of nodes of each of the k types. The algorithm now operates as follows:

```
for s = 1 to k set τ(s, 0, 0, ..., 0) = 0
 for s = 1 to k
  for i_1 = 0 to n_1
   for i_2 = 0 to n_2
    ...
     for i_k = 0 to n_k
       compute τ(s, i_1, ..., i_k) using
       the relation in Lemma 4.
```

This dynamic program computes $O(k \times n_1 \times \ldots \times n_k)$ values of the form $\tau(s, i_1, \ldots, i_k)$. The computation of each such value requires $O(k \times i_1 \times \ldots \times i_k)$ steps. Since $i_1 \leq n_i \leq n$ for each $1 \leq i \leq k$, the total running time is $O(k^2 n^{2k}) = O(n^{2k})$ for any constant k. □

Note that the dynamic programming table in Theorem 2 contains the values of $\tau(s, i_1, \ldots, i_k)$ for all $1 \leq s \leq k$, $1 \leq i_j \leq n_j$, $1 \leq j \leq k$. Thus, for a network with small k it may be desirable to precompute the dynamic programming table and annotate each entry in the table with the optimal schedule. In this way, an optimal schedule can subsequently be found in constant time for any multicast in this network.

5 Conclusions

Although the optimal multicast problem for the heterogeneous receive-send communication model is NP-complete in the strong sense, we have demonstrated that a simple greedy algorithm is a polynomial time

approximation algorithm when receive-send ratios are constant bounded. In addition, we have shown that optimal solutions can be found in polynomial time when the number of distinct types of workstations is bounded by a constant.

A number of interesting and important directions remain for future research. Although the optimal multicast problem is NP-complete, the complexity for the case that receive-send ratios are constant bounded remains unsettled. It is conjectured that the problem remains NP-complete under this assumption. It is possible that a smaller ratio bound than the one presented here can be found for the greedy algorithm. In addition, other polynomial time approximation algorithms might exist for this problem. Finally, polynomial time algorithms and approximation algorithms should be examined for other collective communication operations.

References

[1] T. Anderson et al. A case for NOW (networks of workstations). *IEEE Micro*, pages 54–64, February 1995.

[2] M. Banikazemi, V. Moorthy, and D. Panda. Efficient collective communication on heterogeneous networks of workstations. In *Proc. of the 1998 Int. Conf. on Parallel Processing*, August 1998.

[3] M. Banikazemi, J. Sampathkumar, S. Prabhu, D. Panda, and P. Sadayappan. Communication modeling of heterogeneous networks of workstations for performance characterization of collective operations. In *Proc. of the Int. Workshop on Heterogeneous Computing*, pages 125–131, April 1999.

[4] A. Bar-Noy and S. Kipnis. Designing broadcast algorithms in the postal model for message-passing systems. *Mathematical Systems Theory*, 27:431–452, 1994.

[5] P. Bhat, C.S. Raghavendra, and V. Prasanna. Efficient collective communication in distributed heterogeneous systems. In *Proc. of the 19th IEEE Intl. Conf. on Distributed Computing and Systems*, 1999.

[6] N. Boden et al. A gigabit-per-second local-area network. *IEEE Micro*, 15(1):29–36, Feb. 1996.

[7] B. Chun, A. Mainwaring, and D. Culler. Virtual network transport protocols for myrinet. *IEEE Micro*, pages 53–63, January/February 1998.

[8] D. Culler et al. LogP: Towards a realistic model of parallel computation. In *Proc. of the Fourth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, San Diego, CA, May 1993.

[9] N. Hall, W.-P. Liu, and J. Sidney. Scheduling in broadcast networks. *Networks*, 32:233–253, 1998.

[10] G. Itkis, I. Newman, and A. Schuster. Broadcasting on a budget in the multi-service communication model. In *Proceedings of the Fifth International Conference on High Performance Computing*, December 1998.

[11] S. Johnsson and C.-T. Ho. Broadcasting and personalized communication in hypercubes. *IEEE Transactions on Computers*, 38(9):1249–1268, September 1989.

[12] R. Libeskind-Hadas and J. Hartline. Efficient multicast in the heterogeneous send-receive communication model. Technical Report HMC-CS-99-02, Department of Computer Science, Harvey Mudd College, 1999.

[13] R. Libeskind-Hadas, J. Hartline, P. Boothe, G. Rae, and J. Swisher. Multicast algorithms for heterogeneous networks of workstations. Technical Report HMC-CS-99-01, Department of Computer Science, Harvey Mudd College, 1999.

[14] J. Park, H. Choi, N. Nupairoj, and L. Ni. Construction of optimal multicast trees based on the parameterized communication model. In *Proc. of the Int. Conf. on Parallel Processing*, pages 180–187, Aug. 1996.

[15] M. Schroeder et al. Autonet: A high-speed, self-configuring local area network using point-to-point links. Technical Report 59, DEC SRC, April 1990.