On Multicast Flow Control for Heterogeneous Receivers

Rung-Hung Gau, *Member, IEEE*, Zygmunt J. Haas, *Senior Member, IEEE*, and Bhaskar Krishnamachari, *Student Member, IEEE*

***Abstract*—In this paper, we study the impact of heterogeneous receivers on the throughput of multicast flow control and propose a new multicast flow control algorithm to optimally partition group members into multiple subgroups. Our main contributions are as follows. First, we cast the multicast flow control problem in the Internet as the list partition problem and then prove that the list partition problem is equivalent to the optimal paging problem in cellular networks. The result is not only interesting in itself but also essential to derive the first known analytical bounds for the throughput of multicast flow control. Furthermore, we propose an algorithm to solve not only the list partition problem but also the optimal paging problem and the problem of bulk data transfer using multiple multicast groups. The complexity of our algorithm is one order less than the best known algorithm designed only for the problem of bulk data transfer using multiple multicast groups in the literature. While earlier work uses simulations to justify the usage of multiple subgroups to deliver information to a large amount of receivers in heterogeneous networks, we provide the first analytical support.**

***Index Terms*—Multicast, flow control, partition, optimization.**

I. Introduction

In THE PAST few years, numerous research projects have been carried out to explore how to support multicast in various networking environments. Especially, these include systems that use multicast to deliver data and multimedia traffic [1]–[3]. Other systems support reliable and unreliable multicast over LAN's [4]–[6], Internet [7]–[15], ATM [16], [17], and networks including mobile hosts [18]–[22].

Multicast flow control is essential for high-performance multicast applications. Mishra and Wu [23] studied several techniques of flow control for atomic multicast protocols by simulations. Wang and Schwartz [24] proposed a multicast flow control framework for combined wired/wireless networks. They focused on the fundamental theory for controlling source rate, when source sends packets at a single rate.

McCanne, Jacobson, and Vetterli [25] proposed layered multicast to divide receivers into multiple subgroups. In layered multicast, several layers of information are provided and each receiver subscribes to one specific layer. Li, Paul and Ammar [26] proposed the use of retransmissions in a layered environment and the use of hierarchical control to manage the addition and removal of video layers by receivers. Vicisano, Crowcroft, and Rizzo [27] proposed a TCP-like congestion control algorithm for layered multicast data transfer. Ammar and Wu [28] proposed to improve throughput of point-to-multipoint ARQ protocols through destination set splitting. Shacham [29] proposed to use hierarchically encoded data to maximize utility. The study was mainly algorithmic in nature. Bhattacharyya, Kurose, Towsley, and Nagarajan [30] considered the problem of finding the optimal rate at each layer to minimize the completion time of a fixed-size file. They proposed a cubic-time algorithm to obtain the optimal rates. However, their work does not provide analytical solution to the tradeoff between the throughput and the total number of subgroups or the impact of the distribution of receiver capacities on the throughput.

In this paper, we study the impact of the distribution of the receiver capacities on the throughput of multicast flow control. As in layered multicast [25], we assume that the sender can deliver data to n heterogeneous receivers at up to w distinct rates. In order to maximize the throughput, we propose an efficient algorithm to optimally choose the w distinct rates and to partition the n receivers into w subgroups. More importantly, we derive analytical results of the impact of the distribution of receiver capacities on the throughput. Unlike earlier work that used simulations to justify the usage of delivering information to heterogeneous receivers at distinct rates, our work provides the first analytical support. To adapt to the time-varying capacities of the receivers, we propose to periodically update the information about the capacities of the receivers and periodically perform the optimal partition of the receivers. We emphasize that our proposal aims to reconfigure the optimal partition in the order of every few minutes. Unlike congestion control schemes that provide packet-level adjustment, our flow control scheme only supports session-level adaptation.

The problem of reliable distribution of bulk data to many receivers was studied extensively. Proposed solutions included those that use techniques such as local repair, polling or hierarchy [14], [10], [25], [13], [11]. Additionally, the data carousel approach [51] was proposed to eliminate retransmission and to ensure full reliability at the expense of high overhead. Furthermore, forward error correction based on erasure codes was proposed to achieve reliable multicast [44], [45], [13], [46], [47], [49], [50], [27]. Recently, Nonnenmacher, Biersack and Towsley [48] proposed a parity-based scheme for loss recovery to achieve reliable multicast. A novel digital fountain approach [43] was proposed to allow heterogeneous receivers to reliably and efficiently obtain bulk data. The topic is beyond the scope of this paper.

Manuscript received October 3, 2000; revised March 23, 2001 and June 14, 2001; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor E. Biersack.

The authors are with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14853 USA (e-mail: runghung@ece.cornell.edu; haas@ece.cornell.edu; bhaskar@ece.cornell.edu).

Publisher Item Identifier S 1063-6692(02)01080-4.

1063–6692/02$17.00 © 2002 IEEE

Many multicast routing algorithms (e.g., [33]–[41]) have been proposed in the past few years. We feel that integrating multicast flow control and multicast routing is an interesting research field. In this paper, however, we consider these two problems as orthogonal and assume that a reasonable and well-behaved multicast routing algorithm is implemented at the network layer.

The rest of the paper is organized as follows: Section II introduces our assumptions. Section III presents the formulation of the problem of maximizing throughput by optimal partition of receivers. Some general results of the problem are also described. In Section IV, we present the impact of distribution of receiver capacities on the throughput and show that a small number of subgroups are sufficient to significantly improve the throughput. Section V presents the efficient algorithms for the list partition problem. Section VI shows the simulation results of the satisfaction of receivers' requirement. Section VII describes some design issues and Section VIII presents concluding comments. The proofs of the various results referenced to in the paper can be found in the Appendix.

II. Assumptions

We assume that a basic multicast group is composed of one sender and many receivers. A multicast group, which includes many senders can be seen as a superposition of many basic multicast groups. Therefore, we only consider a basic multicast group in this paper.

We now elaborate on the concept of the capacity of a path. We assume that during each short period of time, the maximum achievable data rate of the path from the sender to each group member can be measured. If the path is a dedicated circuit or a virtual circuit, the data rate is constant. However, in a general packet network, the data rate changes with time. For example, a computer with a 10 Mbps Ethernet connection will most often get a small fraction of the 10 Mbps capacity. The allocated bandwidth depends on the average load of the Ethernet, which changes with time. A computer with 56 kbps modem cannot always send data at 56 kbps; it depends on the quality of the channel, for example, the signal to noise ratio of the twisted pair. Moreover, the queueing delay at each intermediate router also changes with time. Thus, even though, in principle, there are only a limited number of nominal data rates, the actual data rates in a practical network can vary in time over wide ranges around these nominal values.

III. List Partition Problem

In this section, we define the list partition problem and the multicast flow control problem. We first define several terms.

n	total number of receivers in a multicast group;
w	total number of subgroups;
q_i	instant capacity of the path to the ith receiver;
Q	$= \{q_1, q_2, .., q_n\}$ is the list of path capacities of the receivers;
Q_i	list of path capacities of the receivers in the ith subgroup;
$\|Q_i\|$	cardinality of Q_i;
$\min(Q_i)$	value of the minimum element in Q_i;
$\max(Q_i)$	value of the maximum element in Q_i;
$\text{sum}(Q_i)$	sum of all elements in Q_i.

In this paper, we denote a list Q by $\{q_1, q_2, \ldots, q_n\}$, where $q_1, q_2, \ldots, q_n$ are elements belong to the list Q. Although the concept of a list is very similar to the concept of a set, we use the terminology list instead of set to emphasize that there can be repeated elements in a list and the elements in a list are ordered so that q_i, where $1 \le i \le n$, is the ith element in the above list Q. Furthermore, in this paper, the list Q is always composed of nonnegative real numbers.

The definition of a partition of a list is similar to the definition of a partition of a set. Now, we define the wth-order list partition problem as follows.

Given a list Q and a natural number w, find the partition $Q_1, Q_2, \ldots, Q_w$ to maximize $\sum_{i=1}^{w} \min(Q_i) \cdot |Q_i|$.

We can also use notations similar to the linear programming problem to formulate the above list partition problem as follows:

$$\text{maximize} \sum_{i=1}^{w} \min(Q_i) \cdot |Q_i|$$

subject to

Q is a given list of finite cardinality

w is a given natural number

$Q_1, Q_2, \ldots, Q_w$ is a partition of the list Q.

For example, assume that $Q = \{1, 2, 3, 4\}$ and $w = 2$. $Q_1 = \{1, 2\}, Q_2 = \{3, 4\}$ is a partition of Q. The associated value of the object function is equal to $1 \cdot 2 + 3 \cdot 2 = 8$. Similarly, $Q_1 = \{1, 3\}, Q_2 = \{2, 4\}$ is also a partition of Q. The associated value of the object function is equal to $1 \cdot 2 + 2 \cdot 2 = 6$.

For any w and Q, a partition $Q_1, Q_2, \ldots, Q_w$ is said to be an optimal wth-order partition of Q if it maximizes the value of the object function $\sum_{i=1}^{w} \min(Q_i) \cdot |Q_i|$. For example, when $Q = \{1, 2, 3, 4\}$ and $w = 2, Q_1 = \{1, 2\}, Q_2 = \{3, 4\}$ is the optimal second order partition of Q.

To solve the wth-order list partition problem, we have to find the maximum value of the object function and the associated optimal wth-order partition. For any Q and w, we denote the maximum value of the object function by $T_Q(w)$. For example, $T_{\{1,2,3,4\}}(2) = 8$.

We now introduce the multicast flow control problem. Suppose there are n receivers in a multicast group. Let q_i be the capacity of the path from the sender to the ith receiver. Let $Q = \{q_1, q_2, \ldots, q_n\}$. As in layered multicast [25], we assume that the sender can support sending data with up to w distinct transmission rates, $s_1, s_2, \ldots, s_w$. The sender delivers data to a receiver at a single rate, which is equal to one of the above w rates. Receivers that obtain data at the same rate form a subgroup. Therefore, all receivers are partitioned into w subgroups as shown in Fig. 1. Our goal is to maximize the weighted sum of rates at which the sender delivers data to receivers. We define the aggregate throughput as the weighted sum of rates at which the sender delivers data to receivers. Assume that the sender delivers data to n_i receivers at rate s_i, where $1 \le i \le w$ and $n = \sum_{i=1}^{w} n_i$. Then, the aggregate throughput is $\sum_{i=1}^{w} n_i \cdot s_i$.

In general, to maximize the aggregate throughput, each of the w rates must be equal to one element in Q. On the other hand,

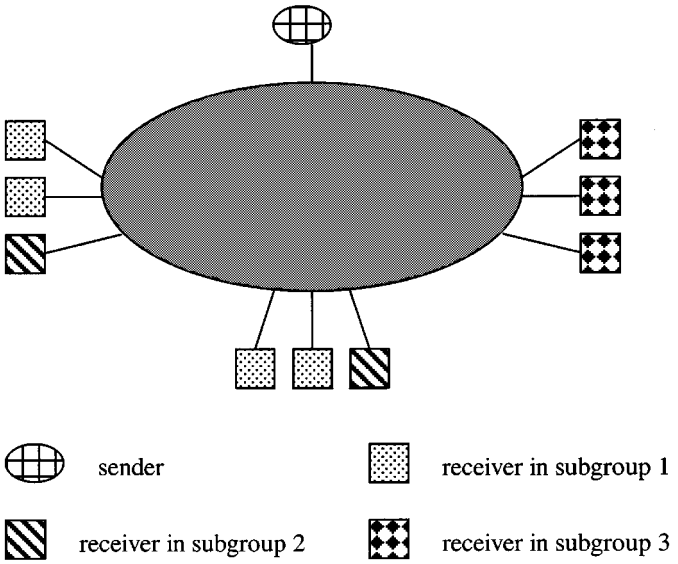

Fig. 1. The multicast flow control problem.

to avoid overflow at each receiver, the sender can not deliver data to the receivers in the ith subgroup at a rate greater than $\min(Q_i)$. Therefore, $s_j = \min(Q_j)$, where $1 \leq j \leq w$, is a necessary condition to achieve maximum aggregate throughput. It can be shown that the above multicast flow control problem is equivalent to the list partition problem.

For example, assume again that $Q = \{1,2,3,4\}$ and $w = 2$. Suppose that the sender delivers data to the first receiver at a rate equal to 1, while the sender delivers data to the other three receivers at a rate equal to 2. The associated aggregate throughput is equal to $1+2+2+2 = 7$. We can increase the aggregate throughput by delivering data to the first two receivers at a rate equal to 1 and delivering data to the last two receivers at a rate equal to 3. The throughput will be then equal to $1+1+3+3 = 8$.

In this paper, we consider the list partition problem only in the nontrivial case in which $w < \text{rank}(Q)$, where $\text{rank}(Q)$ is the total number of distinct elements in the list Q. Furthermore, without loss of generality, we assume that the list $Q = \{q_1, q_2, \ldots, q_n\}$ has been sorted, so that $q_i \leq q_{i+1}, \forall 1 \leq i \leq n-1$.

We define ordered optimal partition as follows.

Definition: For any list Q and any natural number w, a partition $Q_1, Q_2, \ldots, Q_w$ of the list Q is said to be an **ordered partition**, if $\alpha \leq \beta$, for every α, β, i, j, where $\alpha \in Q_i, \beta \in Q_j$ and $i \leq j$.

For example, $Q_1 = \{1,2\}, Q_2 = \{3,4\}$ is an ordered partition of Q. On the other hand, $Q_1 = \{1,3\}, Q_2 = \{2,4\}$ is not an ordered partition of Q.

Definition: A partition is an **ordered optimal partition**, if it is **both** optimal and ordered.

In the above example, $Q_1 = \{1,2\}, Q_2 = \{3,4\}$ is an ordered optimal partition of $Q = \{1,2,3,4\}$. On the other hand, $Q_1 = \{1\}, Q_2 = \{2,3,4\}$ is an ordered partition but not an optimal partition.

We now present two theorems that create the theoretical basis for our fast algorithm. In the first theorem, we prove that there exists an ordered optimal partition for any wth-order list partition problem. In the second theorem, we show the optimal sub-structure of an optimal solution of the list partition problem.

Theorem 3.1: There is at least one ordered optimal partition for the wth-order list partition problem.

Proof: See Appendix. □

Although it can be proved that an optimal partition must be an ordered partition, the proof is lengthy. Therefore, we only use the above theorem that is sufficient for designing the algorithms in the paper.

Theorem 3.2 (Optimal Substructure): Let $Q_1, Q_2, \ldots, Q_w$ be an ordered optimal partition for Q. $\forall 2 \leq i \leq w-1$, let $L = Q_1 \cup Q_2 \cup \cdots \cup Q_i$ and $R = Q_{i+1} \cup Q_{i+2} \cup \cdots \cup Q_w$. Furthermore, let $L_j = Q_j$, where $1 \leq j \leq i$, and $R_k = Q_{i+k}$, where $1 \leq k \leq w-i$. Then, $L_1, L_2, \ldots, L_i$ is an ordered optimal partition of L and $R_1, R_2, \ldots, R_{w-i}$ is an ordered optimal partition of R.

Proof: See Appendix. □

IV. Analytical Results

In this section, we use the notion of the normalized throughput to study the impact of the distribution of receiver capacities on the throughput of multicast flow control and provide analytical results. For any list Q of cardinality n and any natural number w, we define the normalized throughput $\Gamma_Q(w)$ to be $T_Q(w)/T_Q(n)$, where $T_Q(w)$ is the maximum value of the object function of the wth-order list partition problem for the list Q. We show that when a single subgroup is used, the normalized throughput tends to zero as the total number of receivers increases. On the other hand, when two or more subgroups are used, the normalized throughput is always positive, regardless of the total number of receivers and the distribution of receiver capacities. In addition, we provide the first known lower bounds to the normalized throughput. Furthermore, we prove that the list partition problem and the optimal paging problem are equivalent. The result is not only interesting but also important to derive the above lower bounds of the normalized throughput.

A. The Uniform Distribution: Why Two Is Much Better Than One

We first consider a list $U_n = \{1,2,\ldots,n\}$, which corresponds to a group of receivers with uniformly distributed capacities. We can derive $\Gamma_{U_n}(w)$ as follows. For $w = 1$ and $w = n$, it is easy to see that $T_{U_n}(1) = n$ and $T_{U_n}(n) = ((n(n+1))/2)$, respectively. Therefore, $\Gamma_{U_n}(1) = (2/(n+1))$.

More importantly, as n goes to infinity, $\Gamma_{U_n}(1)$ goes to zero. This means that the slowest receiver in a group dominates the overall performance and the relative performance becomes worse when the diversity of receivers, which is defined as the capacity of the fastest receiver divided by the capacity of the slowest receiver, increases.

For $w = 2$, we obtain the $\Gamma_{U_n}(2)$ as follows. For simplicity, we assume that n is an even number; the analysis can be easily extended to the general case. First, by simple calculations, it is easy to derive that

the optimal partition should be $\{1,2,\ldots,(n/2)\}$ and $\{(n/2)+1,(n/2)+2,\ldots,n\}$, while the optimal object function is $T_{U_n}(2)=(1)\cdot(n/2)+((n/2)+1)\cdot(n/2)=(n^2/4)+n$. Therefore, $\Gamma_{U_n}(2)=((n^2/4+n)/(n^2/2+n/2))$. Moreover, as n goes to infinity, $\Gamma_{U_n}(2)=(1/2)$. We can extend the above result to the cases when $w\geq 3$, as in the following theorem.

Lemma 4.1: If n is a multiple of w, then $\{1,2,\ldots,(n/w)\}$, $\{(n/w)+1,(n/w)+2,\ldots,(2n/w)\},\ldots,\{((n(w-1))/w)+1,((n(w-1))/w)+2,\ldots,n\}$ is an optimal solution of the wth-order list partition problem for U_n.

Proof: See Appendix. □

Theorem 4.1: When n is a multiple of w, $\lim_{n\to\infty}\Gamma_{U_n}(w)=1-(1/w)$.

Proof:

1) By the above lemma

$$\begin{aligned}T_{U_n}(w)&=\sum_{i=0}^{w-1}\left(\frac{ni}{w}+1\right)\cdot\left(\frac{n}{w}\right)\\&=\frac{n^2}{w^2}\sum_{i=0}^{w-1}i+n\\&=\left(\frac{n^2}{2}\right)\cdot\left(1-\frac{1}{w}\right)+n\end{aligned}$$

2)

$$\begin{aligned}\lim_{n\to\infty}\Gamma_{U_n}(w)&=\left(1-\frac{1}{w}\right)\lim_{n\to\infty}\frac{\frac{n^2}{2}}{\frac{n^2}{2}+\frac{n}{2}}+\lim_{n\to\infty}\frac{n}{\frac{n^2}{2}+\frac{n}{2}}\\&=1-\frac{1}{w}\end{aligned}$$

□

Theorem 4.2: $\lim_{n\to\infty}\Gamma_{U_n}(w)=1-1/w$

Proof: See Appendix. □

We now explain the importance of the above theorem. The theorem tells us how good the performance of a system using w subgroups is, compared to the performance of a system using n subgroups. Since the limit of $\Gamma_{U_n}(1)$ is zero, we know that a single subgroup scheme is not appropriate in the limit for receivers with diverse capacities. On the other hand, the limit of $\Gamma_{U_n}(2)$ is 0.5. This means that we can always use two subgroups to achieve 50% of the throughput of the limiting case of n subgroups, even when the receivers have diverse capacities. We also observe that $\Gamma_{U_n}(w)$ is close to 1.0 even for small w. For example, since $\Gamma_{U_n}(5)=0.8$, only five subgroups will achieve already 80% of the throughput of the n subgroups case. Similarly, we can use 8 subgroups to achieve 87.5% of the throughput of using n subgroups.

B. The Optimal Paging Problem and the List Partition Problem Are Equivalent

In this section, we prove that the optimal paging problem is a special case of the list partition problem and we will use the result to derive a lower bound of the normalized throughput.

We now briefly explain the optimal paging problem. In a cellular network, a service region is divided into smaller areas. Each area is called a cell. In the center of a cell, there is a base station with an antenna that provides a wireless link to cellular phones in the cell. Since a person who carries a cellular phone moves and can be in any place in the service region, one of the essential task of the cellular network is to find the cell in which the cellular phone currently resides. The task is performed by sequential paging.

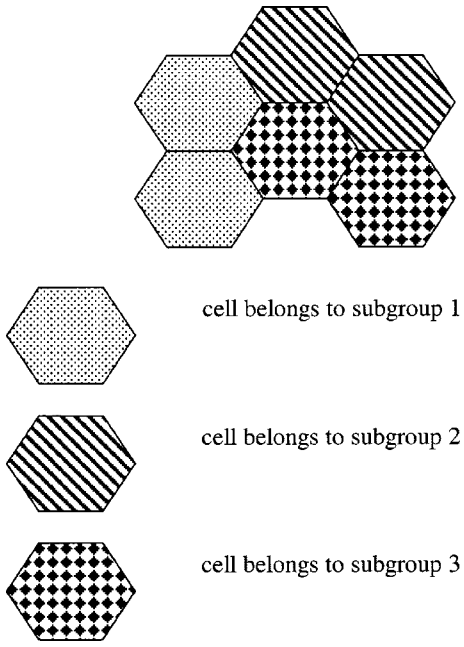

Fig. 2. Optimal paging problem.

For simplicity, we assume that the cellular phone is always inside the service region and turned on. First, each base station in one subgroup of cells broadcasts a paging message. If the cellular phone is in a cell that belongs to the subgroup, the cellular phone responds. Therefore, we know the cell in which the cellular phone resides. If the cellular phone is not in any cell that belongs to the subgroup, there will be no response before timeout. Then, the base stations in another subgroup of cells broadcast the paging message. The procedure is repeated until the cellular phone responds and is, thus, located.

Rose and Yates [31] studied the problem of optimal paging, when the probability that the cellular phone appears in each cell is given. Suppose there are n cells inside the service region. Let d_i be the probability that the cellular phone appears in the ith cell inside the service region and $D=\{d_1,d_2,\ldots,d_n\}$. We illustrate the optimal paging problem in Fig. 2. Assume that the cells are partitioned into w subgroups, so that the cells in the ith subgroup will be paged in the ith round if necessary. Let D_i be the list of associated probabilities of cells in the ith subgroup. Then, we can calculate the average number of paging messages required to find the cellular phone as follows.

With probability equal to $\mathrm{sum}(D_1)$, the cellular phone is in a cell that belongs to the first subgroup and the required number of paging messages is $|D_1|$. With probability equal to $\mathrm{sum}(D_2)$, the cellular phone is in a cell that belongs to the second subgroup and the required number of paging messages is $(|D_1|+|D_2|)$. In general, the average number of required paging messages can be calculated as $\sum_{i=1}^{w}(\sum_{j=1}^{i}|D_j|)\cdot\mathrm{sum}(D_i)$. Rose and Yates [31] assumed that the average paging cost is equal to the average number of required paging messages to find the location of the cellular phone. The optimal paging problem is to find the optimal partition of cells to groups to minimize the average paging

cost. We formulate the wth-order optimal paging problem as follows.

Given a list D, where $\mathrm{sum}(D) = 1$, and a natural number w, find the partition $D_1, D_2, \ldots, D_w$ to minimize $\sum_{i=1}^{w}(\sum_{j=1}^{i}|D_j|)\cdot \mathrm{sum}(D_i)$.

For any list D and natural number w, we denote the minimum value of the object function in the wth-order optimal paging problem by $P_D(w)$.

Rose and Yates [31] proved that in order to minimize the average paging cost, cells with higher probabilities must be paged before cells with lower probabilities are paged. Akyildiz [32] showed that in the optimal paging problem, w may be driven by delay constraints. We focus on the impact of the probability distribution function, D, on the average paging cost.

Lemma 4.2: Let $D_{\mathrm{worst}} = \{1/n, 1/n, \ldots, 1/n\}$, where $|D_{\mathrm{worst}}| = n$. For any natural number w and any list D, where $|D| = n$ and $\mathrm{sum}(D) = 1, P_{D_{\mathrm{worst}}}(w) \geq P_D(w)$.

Proof: See Rose and Yates [31]. □

For example, when $n = 4, D_{\mathrm{worst}} = \{1/4, 1/4, 1/4, 1/4\}$. Let $D = \{0.4, 0.3, 0.2, 0.1\}$. According to the above lemma, $P_{D_{\mathrm{worst}}}(2) \geq P_D(2)$. In fact, $P_{D_{\mathrm{worst}}}(2) = 0.5\cdot 2 + 0.5\cdot 4 = 3$, while $P_D(2) = 0.7\cdot 2 + 0.3\cdot 4 = 2.6$.

The above result matches our intuition. In the optimal paging problem, each element in the list represents the probability that a cellular phone appears in the corresponding cell. The case in which the cellular phone appears equal-likely in each cell is the worst case, since the entropy is maximum.

We now define some terms that will be used to prove the equivalence of the optimal paging problem and the list partition problem.

Definitions:

1. Concave list) For any finite list $A = \{a_1, a_2, \ldots, a_n\}$, define a sequence $d_k = a_{k+1} - a_k$, where $1 \leq k \leq n-1$. The list A is said to be a concave list if the sequence d_k is nonincreasing.
2. Basic concave list) For any finite list $A = \{a_1, a_2, \ldots, a_n\}$, define $A^+ = \{a_1, a_2, \ldots, a_n, a_{n+1}\}$, where $a_{n+1} = 1$. The list A is said to be a basic concave list if $a_1 = 0$, $\forall 2 \leq i \leq n, 0 \leq a_i \leq 1$, $\forall 1 \leq i \leq n-1, a_{i+1} \geq a_i$ and A^+ is a concave list.
3. Quasi-basic list) A list $A = \{a_1, a_2, \ldots, a_n\}$ is said to be a quasi-basic list if $a_1 = 0, \forall 2 \leq i \leq n, 0 \leq a_i \leq 1$ and $\forall 1 \leq i \leq n-1, a_{i+1} \geq a_i$.
4. Difference list) For any quasi-basic list $A = \{a_1, a_2, \ldots, a_n\}$, construct the associated difference list $D = \{d_1, d_2, \ldots, d_n\}$ as follows: $d_k = a_{k+1} - a_k$, where $1 \leq k \leq n-1$ and $d_n = 1 - a_n$.
5. Accumulation list) For any list $D = \{d_1, d_2, \ldots, d_n\}$, where $\sum_{i=1}^{n} d_i = 1$ and $\forall 1 \leq i \leq n-1, 0 \leq d_{i+1} \leq d_i$, construct the associated accumulation list $A = \{a_1, a_2, \ldots, a_n\}$ as follows: $a_1 = 0$ and $a_{k+1} = \sum_{i=1}^{k} d_i$, where $1 \leq k \leq n-1$.

For example, $\{1, 5, 8, 10\}$ is a concave list, since $5-1 \geq 8-5 \geq 10-8$. In addition, $\{0, 0.4, 0.7, 0.9\}$ is a basic concave list. We choose the word "basic" primarily to reflect that each element in a basic concave list is between 0 and 1. It is clear that a basic concave list is also a quasi-basic list. Furthermore, when $A = \{0/n, 1/n, 2/n, \ldots, ((n-1)/n)\}$, the associated difference list is $\{(1/n)-(0/n), (2/n)-(1/n), \ldots, 1-((n-1)/n)\} = \{1/n, 1/n, \ldots, 1/n\}$. When $D = \{1/n, 1/n, \ldots, 1/n\}$ and $n = |D|$, the associated accumulation list is $\{0, 1/n, (1/n)+(1/n), \ldots, (1/n)+(1/n)+\cdots+(1/n)\} = \{0/n, 1/n, 2/n, \ldots, ((n-1)/n)\}$.

We now prove a theorem that reveals the relation between the list partition problem and the optimal paging problem.

Theorem 4.3:

a) Let $D = \{d_1, d_2, \ldots, d_n\}$, where $\sum_{i=1}^{n} d_i = 1$ and $\forall 1 \leq i \leq n-1, 0 \leq d_{i+1} \leq d_i$. Let A be the accumulation list of D. Then, $T_A(w) = n - P_D(w)$.
b) Let $A = \{a_1, a_2, \ldots, a_n\}$ be a basic concave list. Let D be the difference list of A. Then, $P_D(w) = n - T_A(w)$.

Proof: See Appendix. □

The above theorem states that the list partition problem and the optimal paging problem are equivalent under some conditions. This is a very interesting result, since the two problems are from different fields: the list partition problem is from multicast flow control in the Internet, while the optimal paging problem is from wireless networks.

To ensure a one-to-one correspondence between a nondecreasingly ordered partition in the list partition problem and a nonincreasingly ordered partition in the optimal paging problem, that the list A is basic concave is necessary. The first three conditions of a basic concave list ensure that every element in the corresponding difference list is nonnegative, while that the list A^+ is concave guarantees that the elements in the corresponding difference list are nonincreasingly ordered. For example, assume that $A = \{0, 0.1, 0.3, 0.4\}$, which is not basic concave. The partition $\{0, 0.1\}, \{0.3, 0.4\}$ is a nondecreasingly ordered partition in the list partition problem. However, the partition $\{0.1, 0.2\}, \{0.1, 0.6\}$ is not a nonincreasingly ordered partition for the associated difference list $D = \{0.1, 0.2, 0.1, 0.6\}$ in the optimal paging problem, since $0.2 < 0.6$.

For any nonincreasingly ordered partition in the optimal paging problem, we can derive an upper bound of the associated value of the object function. Thus, we can derive a lower bound of $T_A(w)$ based on the above theorem. However, for a disordered partition in the optimal paging problem, we do not have a nontrivial upper bound of the associated value of the object function. Therefore, we can not use Theorem 4.3 to derive a lower bound of $T_A(w)$. We will explore this case later in the paper.

C. Lower Bounds for Concave Lists

In this section, we derive lower bounds for the normalized throughput, when the associated list is concave. We emphasize

that these are the first known analytical bounds for the normalized throughput.

Theorem 4.7:

a) For any basic concave list $A = \{a_1, a_2, \ldots, a_n\}$, $\Gamma_A(2) \geq (1/3) - \epsilon(n)$, where $\lim_{n\to\infty} \epsilon(n) = 0$.
b) For any nonnegative, nondecreasing, concave list $Q = \{q_1, q_2, \ldots, q_n\}, \Gamma_Q(2) \geq (1/3) - \epsilon(n)$, where $\lim_{n\to\infty} \epsilon(n) = 0$.

Proof:

1) Let $D = \{d_1, d_2, \ldots, d_n\}$ be the associated difference list of A and $p = ((\sum_{k=1}^{n} k \cdot d_k)/n)$. We only consider the case when $p \in [0, 0.5]$, since $\sum_{k=1}^{n} k \cdot d_k \leq ((n+1)/2)$. Let $x = \sum_{i=\lceil 2\cdot p\cdot n\rceil}^{n} d_i$. We claim that $x \leq 0.5$, since otherwise
$$\sum_{k=1}^{n} k \cdot d_k \geq x \cdot \lceil 2 \cdot p \cdot n \rceil > 0.5 \cdot (2 \cdot p \cdot n) = p \cdot n$$
which is contradictory to $p = ((\sum_{k=1}^{n} k \cdot d_k)/n)$.
2) Since $\{d_1, d_2, \ldots, d_{\lceil 2\cdot p\cdot n\rceil - 1}\}, \{d_{\lceil 2\cdot p\cdot n\rceil}, d_{\lceil 2\cdot p\cdot n\rceil + 1}, \ldots, d_n\}$ is a second order partition of D,
$$\begin{aligned} P_D(2) &\leq (1-x) \cdot (\lceil 2 \cdot p \cdot n \rceil - 1) + x \cdot n \\ &\leq (1-x) \cdot (2 \cdot p \cdot n) + x \cdot n \end{aligned}$$
3)
$$\begin{aligned} n - P_D(2) &\geq (1-x) \cdot n - (1-x) \cdot (2 \cdot p \cdot n) \\ &= n \cdot (1 - 2 \cdot p) \cdot (1 - x) \\ &\geq \left(\frac{n}{2}\right) \cdot (1 - 2 \cdot p) \end{aligned}$$
4)
$$\begin{aligned} \Gamma_A(2) &= \frac{n - P_D(2)}{n - \sum_{k=1}^{n} k \cdot d_k} \\ &\geq \frac{\left(\frac{n}{2}\right) \cdot (1 - 2 \cdot p)}{n \cdot (1-p)} \\ &= \frac{1 - 2 \cdot p}{2 - 2 \cdot p} \end{aligned}$$
5) Based on the proof of Theorem 4.5, we have
$$\begin{aligned} \Gamma_A(2) = \frac{T_A(2)}{T_A(n)} &\geq \frac{\frac{n}{2} \cdot \left(1 - \frac{1}{2}\right) - \epsilon(n)}{n \cdot (1-p)} \\ &\geq \frac{1}{4 \cdot (1-p)} - \frac{\epsilon(n)}{n \cdot (1-p)} \end{aligned}$$
where $\lim_{n\to\infty} \epsilon(n) = 0$.
6) Then, $\forall p \in [0, 0.5], \Gamma_A(2) \geq \max((1 - 2 \cdot p)/(2 - 2 \cdot p)), (1/(4 \cdot (1-p))) - (\epsilon(n)/(n \cdot (1-p)))$, where $\lim_{n\to\infty} \epsilon(n) = 0$. Therefore, $\Gamma_A(2) \geq (1/3) - \epsilon'(n)$, where $\lim_{n\to\infty} \epsilon'(n) = 0$. Thus, we have completed the proof of property a. Following the proof of Theorem 4.6, it is easy to prove property b, which we will omit here. □

D. Lower Bounds for Arbitrary Lists

In this section, we derive lower bounds of the normalized throughput for arbitrary lists. These bounds are the first known bounds for the normalized throughput and are applicable for arbitrary lists.

We first introduce a variant of the optimal paging problem, which will be used to derive the lower bounds for the normalized throughput.

Given a list $D = \{d_1, d_2, \ldots, d_n\}$, where $\mathrm{sum}(D) = 1$, and a natural number w, find the partition $D_1, D_2, \ldots, D_w$, where $\forall p, q, s, t$, if $p < q, d_p \in D_s$ and $d_q \in D_t$, then $s \leq t$, to minimize $\sum_{i=1}^{w} (\sum_{j=1}^{i} |D_j|) \cdot \mathrm{sum}(D_i)$.

Let $VP_D(w)$ be the minimum value of the above object function. Compared to the original optimal paging problem, there is one more constraint in the above problem. In the optimal paging problem, an optimal partition must be a nonincreasingly ordered partition. In the above problem, however, an optimal partition does not need to be a nonincreasingly ordered partition due to the last constraint. In fact, there are cases in which any nontrivial feasible partition is not a nonincreasingly ordered partition. For example, when $D = \{0.3, 0.1, 0.2, 0.4\}$, the partition $\{0.3\}, \{0.1, 0.2, 0.4\}$ is not nonincreasingly ordered, since $0.3 < 0.4$. Similarly, the partition $\{0.3, 0.1\}, \{0.2, 0.4\}$ is not nonincreasingly ordered, since $0.1 < 0.4$.

Now, we derive a similar relation between the list partition problem and the variant of the optimal paging problem as in the following theorem.

Theorem 4.8: For any quasi-basic list $A = \{a_1, a_2, \ldots, a_n\}$ and the associated difference list $D = \{d_1, d_2, \ldots, d_n\}, ((T_A(w))/(T_A(n))) = ((n - VP_D(w))/(n - \sum_{k=1}^{n} k \cdot d_k))$.

Proof: Similar to the proof of Theorem 4.3 and the proof of Theorem 4.4 and will be omitted. □

Now, we derive the lower bound of the normalized throughput, when two subgroups are used.

Theorem 4.10: For any nonnegative, nondecreasing list $Q = \{q_1, q_2, \ldots, q_n\}$, if $q_n \neq q_1$, then

a) There exists a quasi-basic list $A = \{a_1, a_2, \ldots, a_n\}$ and two real numbers $\alpha > 0$ and $\beta \geq 0$, such that $Q = \alpha \cdot A + \beta$.
b) Let $D = \{d_1, d_2, \ldots, d_n\}$ be the associated difference list of A and $p = ((\sum_{k=1}^{n} k \cdot d_k)/n)$. Then, $((T_Q(2))/(T_Q(n))) \geq (1/2) \cdot (-1 + (2/(1+p))) > 0$.

Proof:

1) We now prove property a. Choose $\alpha = q_n - q_1, \beta = q_1$ and $A = ((Q - \beta)/\alpha)$. The rest of the proof is similar to the proof of property a in Theorem 4.6 and will be omitted.
2) We now prove the second property. It is clear that $p \in [0, 1]$.
$$\begin{aligned} \frac{T_Q(2)}{T_Q(n)} &= \frac{\alpha \cdot T_A(2) + \beta \cdot n}{\alpha \cdot T_A(n) + \beta \cdot n} \\ &\geq \frac{T_A(2)}{T_A(n)} \left(\frac{\alpha \cdot T_A(w)}{\alpha \cdot T_A(n)} \leq 1, \beta \cdot n \geq 0\right) \\ &\geq \frac{1}{2} \cdot \left(-1 + \frac{2}{1+p}\right) \end{aligned}$$
(Theorem 4.8 and Theorem 4.9)
3) $D = \{((q_2 - q_1)/\alpha), ((q_3 - q_2)/\alpha), \ldots, ((q_n - q_{n-1})/\alpha), 1 - ((q_n - q_1)/\alpha)\}$. Since $q_n \neq q_1, d_n \neq 1$. Therefore, $\sum_{k=1}^{n} k \cdot d_k \neq n$ and $p \neq 1$. There-

fore, $p \in [0,1)$. Furthermore, $((T_Q(2))/(T_Q(n))) \geq (1/2)\cdot(-1+(2/(1+p))) > 0$. □

We can use the above theorem to estimate the normalized throughput. For example, consider $Q = \{1^2, 2^2, \ldots, n^2\}$. In this case, $\lim_{n\to\infty} p = 2/3$. Therefore, $\lim_{n\to\infty}((T_Q(2))/(T_Q(n))) \geq (1/10) = 0.1$. In fact, $\lim_{n\to\infty}((T_Q(2))/(T_Q(n))) = 4/9 \approx 0.444$. For any list with $p = 1/4$, we can show that $((T_Q(2))/(T_Q(n))) \geq 0.3$.

Based on a method similar to the method in the above theorem, we can derive a better lower bound of the normalized throughput. We show the result in the following theorem.

Theorem 4.11: For any nonnegative, nondecreasing list $Q = \{q_1, q_2, \ldots, q_n\}$, if $q_n \neq q_1, ((T_Q(2))/(T_Q(n))) \geq ((1-2\cdot\sqrt{p}+p)/(1-p))$, where $p \in (0,1)$ is defined as in the above theorem.

Proof: See Appendix. □

Consider the same case in which $Q = \{1^2, 2^2, \ldots, n^2\}$. Based on the above theorem, we can show that $\lim_{n\to\infty}((T_Q(2))/(T_Q(n))) \geq 3\cdot((5/3)-2\cdot\sqrt{(2/3)}) \approx 0.101\,02$. For any list with $p = 1/4$, we can show that $((T_Q(2))/(T_Q(n))) \geq (1/3) \approx 33\%$. We also observe that we can use Theorem 4.6 and Theorem 4.11 to derive Theorem 4.7.

E. The Problem of Minimizing the Completion Time

In this section, we will derive an upper bound of the completion time for a very general class of lists. We first briefly introduce the problem of minimizing the completion time. Suppose we want to transfer a file of fixed size L to n receivers. Furthermore, assume that we are allowed to use w distinct transmitting rates. Let q_i be the capacity of the path from the sender to the ith receiver and $Q = \{q_1, q_2, \ldots, q_n\}$. As in the list partition problem, we can partition the list Q into w sublists, $Q_1, Q_2, \ldots, Q_w$. Namely, we partition n receivers into w subgroups. To avoid overflow and to minimize the completion time, the sender must deliver packets to the jth subgroup at rate equal to $\min(Q_j)$. The completion time of each receiver in the jth subgroup is defined as $(L/(\min(Q_j)))$. Therefore, the average completion time is defined as $\sum_{j=1}^{w}(L/(\min(Q_j)))\cdot(|Q_j|/n) = L/n\cdot\sum_{j=1}^{w}(|Q_j|/(\min(Q_j)))$. The problem of minimizing the completion time is to find the optimal partition of Q to minimize $L/n\cdot\sum_{j=1}^{w}(|Q_j|/(\min(Q_j)))$.

Since L and n are constants, without loss of generality, we assume that $L/n = 1$. Furthermore, we define $B = Q^{-1} = \{q_1^{-1}, q_2^{-1}, \ldots, q_n^{-1}\}$ and let $B_1, B_2, \ldots, B_w$ be a partition of B. Therefore, we obtain the following concise version of the problem of minimizing the completion time.

Given a list B and a natural number w, find the partition $B_1, B_2, \ldots, B_w$ to minimize $\sum_{i=1}^{w}|B_i|\cdot\max(B_i)$.

We denote the minimum value of the above object function by $\mathrm{CT}_B(w)$.

Bhattacharyya, Kurose, Towsley and Nagarajan [30] first studied the above problem. They proposed an $O(wn^3)$ time algorithm to find the optimal rates to minimize the completion time. We derive here an upper bound of the completion time. We now define a quasi-basic convex list.

Definition: A list $B = \{b_1, b_2, \ldots, b_n\}$ is said to be a quasi-basic convex list if:

1) $b_1 = 1$;
2) $\forall 2 \leq i \leq n, 0 \leq b_i < 1$;
3) $\forall 1 \leq i \leq n-1, b_{i+1} \leq b_i$;
4) the sequence d_k is nondecreasing, where $\forall 1 \leq k \leq n-1, d_k = b_{k+1} - b_k$.

In the above definition, the first three conditions are similar to the conditions in the definition of a quasi-basic list. The last condition ensures that the list is convex.

We now prove that under some conditions, the optimal paging problem is equivalent to the problem of minimizing the completion time.

Theorem 4.12:

a) Let $D = \{d_1, d_2, \ldots, d_n\}$, where $\sum_{i=1}^{n} d_i = 1$ and $\forall 1 \leq i \leq n-1, 0 \leq d_{i+1} \leq d_i$. Construct $B = \{b_1, b_2, \ldots, b_n\}$ such that $b_k = \sum_{i=k}^{n} d_i$, where $1 \leq k \leq n$. Then, $\mathrm{CT}_B(w) = P_D(w)$.
b) Let $B = \{b_1, b_2, \ldots, b_n\}$ be a quasi-basic convex list, where $1 \geq b_i \geq b_{i+1} \geq 0$. Construct $D = \{d_1, d_2, \ldots, d_n\}$ such that $d_k = b_k - b_{k+1}$, where $1 \leq k \leq n$ and $b_{n+1} = 0$. Then, $P_D(w) = \mathrm{CT}_B(w)$.

Proof: Similar to the proof of Theorem 4.3 and will be omitted here. □

We now derive an upper bound for the quasi-basic convex list and show that the upper bound is achieved by the list $B = \{n/n, ((n-1)/n), \ldots, 2/n, 1/n\}$, where $n = |B|$.

Theorem 4.13: Let $B = \{n/n, ((n-1)/n), \ldots, 2/n, 1/n\}$, where $n = |B|$.

a) If n is a multiple of $w, ((\mathrm{CT}_B(w))/(\mathrm{CT}_B(1))) = 1/2\cdot(1+(1/w))$
b) $\lim_{n\to\infty}((\mathrm{CT}_B(w))/(\mathrm{CT}_B(1))) = 1/2\cdot(1+(1/w))$

Proof:

1) Let $D = \{1/n, 1/n, \ldots, 1/n\}$. Then,

$$\mathrm{CT}_B(w) = P_D(w) \text{ (Theorem 4.12)} = \frac{n}{2}\cdot\left(1+\frac{1}{w}\right) (n \text{ is a multiple of } w)$$

2) Since $\mathrm{CT}_B(1) = n$, we have $((\mathrm{CT}_B(w))/(\mathrm{CT}_B(1))) = (1/2)\cdot(1+(1/w))$.
3) The proof of property b is similar to that of Theorem 4.2 and will be omitted here. □

Theorem 4.14:

a) For any quasi-basic convex list $B, ((\mathrm{CT}_B(w))/(\mathrm{CT}_B(1))) \leq (1/2)\cdot(1+(1/w))+\epsilon(n)$, where $\lim_{n\to\infty}\epsilon(n) = 0$.
b) For any nonnegative, nonincreasing, convex list $B, ((\mathrm{CT}_B(w))/(\mathrm{CT}_B(1))) \leq (1/2)\cdot(1+(1/w))+\epsilon(n)$, where $\lim_{n\to\infty}\epsilon(n) = 0$.

Proof:

1) For any quasi-basic convex list $B = \{b_1, b_2, \ldots, b_n\}$, construct $D = \{d_1, d_2, \ldots, d_n\}$, so that $\forall 1 \leq k \leq n, d_k = b_k - b_{k+1}$, where $b_{n+1} = 0$. Let $D_{\text{worst}} = \{1/n, 1/n, \ldots, 1/n\}$.

2)

$$\begin{aligned}\mathrm{CT}_B(w) &= P_D(w) \text{ (Theorem 4.12)}\\ &\le P_{D_{\mathrm{worst}}}(w) \text{ (Lemma 4.2)}\\ &\le \frac{n}{2}\cdot\left(1+\frac{1}{w}\right)+\left(w+1+\frac{w}{n}\right)\\ &\quad \text{(proof of Theorem 4.5)}\end{aligned}$$

3) Since $\mathrm{CT}_B(1)=n$, we have $((\mathrm{CT}_B(w))/(\mathrm{CT}_B(1))) \le (1/2)\cdot(1+(1/w))+\epsilon(n)$, where $\epsilon(n)=((w+1)/n)+(w/n^2)$ and $\lim_{n\to\infty}\epsilon(n)=0$.

4) We now prove property b. Let $Y=(B/(\max(B)))$. It is clear that Y is a quasi-basic convex list. Then,

$$\begin{aligned}((\mathrm{CT}_B(w))/(\mathrm{CT}_B(1))) &= \frac{\max(B)\cdot \mathrm{CT}_Y(w)}{\max(B)\cdot \mathrm{CT}_Y(1)}\\ &= \frac{\mathrm{CT}_Y(w)}{\mathrm{CT}_Y(1)}\\ &\le \frac{1}{2}\cdot\left(1+\frac{1}{w}\right)+\epsilon(n) \text{ (property a)}\end{aligned}$$

□

Based on the above theorem, we know that as long as B is nonnegative, nonincreasing, and convex, the completion time when 5 subgroups are used is at most 60% of the completion time when 1 subgroup is used. To illustrate that this condition is true for a wide range of lists, let's consider the following three lists. In the first case, $Q=\{1,2,\ldots,n\}$. In the second case, $Q=\{1^2,2^2,\ldots,n^2\}$. In the third case, $Q=\{\sqrt{1},\sqrt{2},\ldots,\sqrt{n}\}$. The list Q is linear in the first case, convex in the second case, and concave in the third case. However, the corresponding $B=Q^{-1}$ is always nonnegative, nonincreasing and convex.

V. A Fast Algorithm

We propose here a fast algorithm to solve the list partition problem. The algorithm requires $O(wn^2)$ in time and $O(wn)$ in space and is the most efficient algorithm known up to date. Furthermore, the fast algorithm can be used to solve not only the list partition problem but also the optimal sequential paging problem and the problem of minimizing the completion time.

We now present an auxiliary problem. Without loss of generality, we assume that $Q=\{q_1,q_2,\ldots,q_n\}$ has been sorted such that $q_i \le q_{i+1}$, where $1\le i\le n-1$.

Given two natural numbers w and e and a list $R=\{q_1,q_2,\ldots,q_e\}$, find the partition $R_1,R_2,\ldots,R_w$ to maximize $\sum_{i=1}^{w}\min(R_i)\cdot|R_i|$.

Let $h[w,e]$ be the maximum value of the object function in the above auxiliary problem. Then, we can derive the following recursive relation:

$$\begin{aligned}h[2,e] &= \max_{j=1}^{e-1}\{q_1\cdot j+q_{j+1}\cdot(e-j)\}\\ h[w,e] &= \max_{j=1}^{e-1}\{h[w-1,j]+q_{j+1}\cdot(e-j)\},\quad w\ge 3\end{aligned}$$

Since $T_Q(w)=h[w,n]$, we can solve the list partition problem by solving multiple instances of the auxiliary problem.

We now present the complete algorithm.

```
for e = 1 to n
  h[2,e] = max_{j=1}^{e-1}{q_1 · j + q_{j+1}(e − j)}
end for
for k = 3 to w
  for e = 1 to n
    h[k,e] = max_{j=1}^{e-1}{h[k − 1, j] + q_{j+1} · (e − j)}
  end for
end for
```

Since there are two nested loops and each statement takes $O(n)$ time, the time complexity of the algorithm is $O(w\cdot n\cdot n)=O(un^2)$ and the space complexity is clearly $O(w\cdot n)=O(wn)$.

VI. Simulation Results

A. Normalized Throughput

In this section, we use simulations to study $\Gamma_Q(w)$, where Q is a list of size n and can be seen as a list of clusters of elements. By adjusting the total number of clusters and the distributions within the clusters, we can capture the essential nature of capacities of various receivers. Receivers with the same type of network connection devices usually can be seen as a cluster, since their instant capacities vary around the nominal capacity. Besides, same-type receivers in the same region usually can be seen as a cluster, since all these receivers share large portions of the same path to the sender.

In our simulations, to create c clusters of k receivers, we first create c cluster heads. Each cluster head is a receiver whose capacity is the mean capacity of all receivers in a cluster. To determine the capacity of cluster heads, we first generate c random variables that are uniformly distributed in [0,1] and divide these random variables by the maximum one to obtain c normalized random variables. We observe that the largest normalized random variable is 1. The normalized random variables are the normalized speeds of cluster heads. For each cluster, we generate k receivers with normalized speeds according to a Gaussian distribution with mean equals to the normalized speed of the cluster head and variance equals to a chosen value. The variance is small enough so that the overlapping probability of two clusters is negligible.

In our simulations, n, the size of Q, is equivalent to 200. The capacities of the receivers are clustered; namely, receivers can be seen as c clusters of receivers, where $c\in\{200,20,10,5,4,2\}$. In each cluster, there are $k=n/c$ receivers. For each fixed c, we run the simulation 10 000 times to get the probability density distribution of the normalized throughput $\Gamma_Q(w)$. Due to the limit of the space, we only show some representative results in the paper.

We first compare the performance of using 2 subgroups/windows and the performance of using 1 subgroup/window, which corresponds to "listen to the slowest". We show the simulation results in Fig. 3. The normalized throughput is equivalent to $\Gamma_Q(w)$. We first note that in all cases the performance of using two subgroups/windows is much better than using only one subgroup/window. Second, as in Fig. 3, we note that in the worse case, when the capacities of receivers are uniformly distributed, the normalized throughput of using two subgroups/windows is

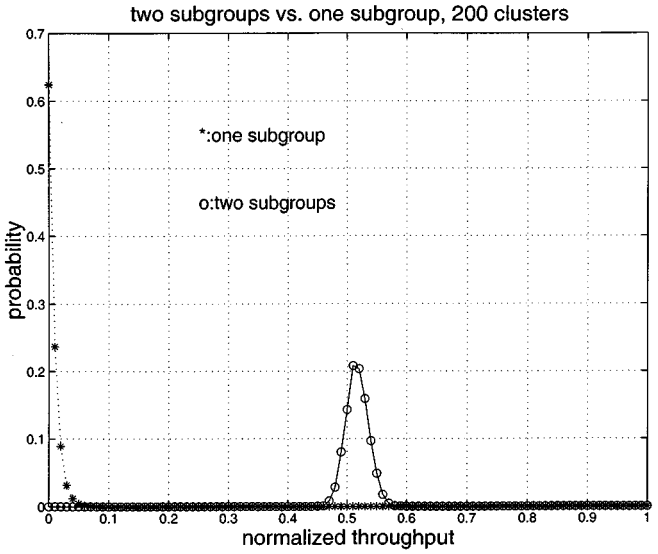

Fig. 3. Normalized throughput of using one subgroup/window or two subgroups/windows.

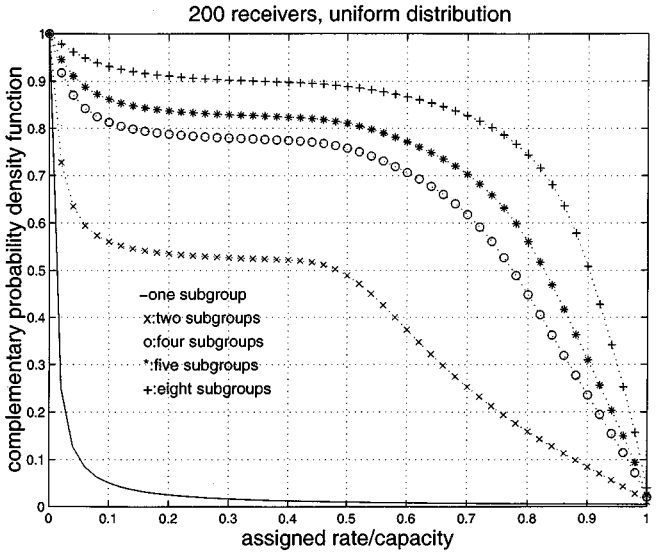

Fig. 4. Receiver satisfaction, uniformly distributed.

about 0.5, while the normalized throughput of using one subgroup/window is less than 0.1. When there are 800 receivers, the normalized throughput of using one subgroup/window decreases to 0.01. On the other hand, the normalized throughput of using two subgroups/windows continues to be about 0.5.

B. Will the Receivers' Requirements be Satisfied?

In this section, we show the level of satisfactions of receivers' requirements, when they are partitioned into subgroups. For each list of receivers, we perform the optimal partition and measure the percentage of receivers that is assigned a rate higher than a specific fraction of its capacity. For each type of receiver group, we run the simulation 10 000 times and obtain a probability distribution function by averaging the simulation results. In Fig. 4, the x-axis represents the ratio of the assigned rate of a receiver to the capacity of the receiver. The y-axis shows the probability that a receiver is assigned a rate with an associated ratio higher than the corresponding value of x-axis. For example, a point $(x, y) = (0.4, 0.6)$ in the figure means that 60% of the receivers are assigned rates higher than 40% of their capacities.

We first show the results when the capacities of the receivers are uniformly distributed. From Fig. 4, we know that when only one subgroup is used, about 3% of the receivers are assigned a rate higher than 20% of the corresponding capacities. On the other hand, we know that when 2 subgroups are used, more than 50% of the receivers are assigned a rate higher than 20% of the corresponding capacities. Furthermore, when 5 subgroups are used about 80% of the receivers are assigned a rate higher than 50% of the corresponding capacities.

VII. Design Considerations

A. How to Measure the Capacities of the Paths?

We discuss two approaches for measuring the capacities of the paths from the sender to group members. In the first approach, the sender periodically measures the round trip times of packets from the sender to group members. As in the sliding window scheme, the capacity of each group member is approximately inversely proportional to the associated round trip time. We assume that the path used to deliver data packets from the sender to a receiver is identical to the path used for measuring the round trip times from the sender to the receiver. This assumption holds in most of the proposed routing algorithms in which routes do not change very fast. The main disadvantage of this approach is that the sender becomes the bottleneck.

Paul [10] and Holbrook [15] have used hierarchy to improve the scalability of their reliable multicast protocols. However, they focus on error control instead of flow control. To improve scalability, we propose a receiver-driven scheme to divide the network into a hierarchy of zones and select a group manager in each zone. We also assume that all group managers are connected and form a hierarchical virtual backbone. Each group member decides when to report to the group manager. A group member reports to its group manager by providing its current capacity of the link in the path to the sender to the group manager. Each group manager optimally partitions the reported group members into k subgroups, which is configurable. Let $r_1, r_2, \ldots, r_k$ be the optimal rates in the k subgroups. Then, the group manager reports to its group manager in the next level as if there are k group members with capacities equal to $r_1, r_2, \ldots, r_k$. Therefore, we can dramatically reduce the traffic to the sender. Furthermore, there is no unique bottleneck node in the scheme.

B. Should a Group Member Report?

We now consider the question of when it is beneficial for a group member to report to its group manager. Suppose the group manager provides k layers of service based on the optimal partition of receivers in its zone. Let $r_1, r_2, \ldots, r_k$ be the data rates at these layers. Without loss of generality, we assume that $r_1 \leq r_2 \leq \cdots \leq r_k$. Let the current capacity of a specific group member in the zone be equal to x. We assume that the group member decide not to report to the group manager and consider three cases based on the value of x.

If $x < r_1$, the group member will experience constant buffer overflow. For the benefit of the group member, it should report to the group manager, so that the group manager can take care of the slow group member. If $x > r_k$, the group member will never get the most desired quality of service. Similarly, the group member should report to the group manager so that the group manager can optimally repartition the group members to increase the aggregate throughput and in return the fast group member has an opportunity to get a better quality of service. Third, if $r_1 \le x \le r_k$, it may be acceptable for a group member not to report to the group manager to reduce the amount of control messages. However, this is a sub-optimal solution in terms of the aggregate throughput. How to find the optimal moments for each receiver to report to the group manager remains an open problem for future consideration.

VIII. Conclusion

The problem of multicast flow control for heterogeneous receivers has been considered in this paper. In particular, we have concentrated on maximizing aggregate throughput when information about receiver capacity is used to optimally divide receivers into subgroups. We have focused on the impact of the distribution of receiver capacities on the maximal throughput and have obtained the first analytical results in the literature.

We have cast the multicast flow control problem as the list partition problem and have solved the list partition problem to determine the optimal rate at each layer to maximize the throughput. Interestingly, we have proven that the list partition problem from multicast flow control in the Internet is equivalent to the optimal paging problem in wireless networks. The result is essential to analyze the impact of distribution of receiver capacities on the throughput of multicast flow control. In addition, we have used the notion of normalized throughput to study the impact of receiver capacities on the throughput, when optimal partition is used. We have defined this normalized throughput as the ratio of the maximum value of throughput using w subgroups to the value of throughput when each receiver forms a subgroup.

We have provided analytical results for two types of list of receiver capacities: concave and not concave. When one subgroup is used, we have shown that the normalized throughput in the limit tends to zero. When receivers are optimally partitioned into $w \ge 2$ subgroups, we have proven that when the list of the capacities of the receivers is concave, the normalized throughput in the limit is always greater than or equal to $(1/2)\cdot(1-(1/w))$. We have also proven that when the list is concave and two subgroups are used, the normalized throughput in the limit is at least 33%. When the list is not concave, we have proven that when two subgroups are used, the normalized throughput in the limit is always positive. Furthermore, we have derived lower bounds to estimate the normalized throughput.

We have proposed a quadratic-time algorithm to perform the optimal partition, showing that the throughput improvement can be achieved in practical system design. Since the optimal paging problem in cellular networks is equivalent to the list partition problem, the quadratic-time algorithm can be applied there as well. Moreover, we have observed that the problem of minimizing the completion time is similar to the list partition problem. Therefore, with a minor modification, our algorithm can be used to solve the former problem as well. For the problem of minimizing the completion time, our algorithm provides the exact solution with one-order lower complexity compared to the best-known algorithm in the literature [30].

In another paper [42], we had proved that a small number of paging zones is sufficient to obtain the majority of paging cost reduction. Similarly, in this paper, we have shown that for a wide range of receiver capacities, a small number of subgroups are sufficient to achieve a significant reduction of completion time. While earlier work relies on simulations to justify the the usage of multiple subgroups to deliver information to a large amount of receivers in heterogeneous networks, our results provide analytical support.

Appendix

A. The List Partition Problem

Before we prove Theorem 3.1, we introduce some notations. Given a list Q of cardinality n and a natural number w, we denote a wth-order partition of Q by $Q_1, Q_2, \ldots, Q_w$. For the purpose of proof, we also represent a wth-order partition of the above list Q by a sequence $a_1(t), a_2(t), \ldots, a_n(t)$ and the other sequence $b_1, b_2, \ldots, b_{w-1}$. During the following procedure of proof, reordering sublists $Q_1, Q_2, \ldots, Q_w$ and/or swapping elements in sublists $Q_1, Q_2, \ldots, Q_w$ may occur in each round. Therefore, a discrete-time index t, where $t \ge 1$, appears in the sequence $\{a_i(t)\}$. For each $t \ge 1$, the sequence $\{a_i(t)\}$ is constructed by concatenating elements in the sublists $Q_1, Q_2, \ldots, Q_w$. Suppose at time $t, Q_i = \{q_{i,1}, q_{i,2}, \ldots, q_{i,e(i)}\}$, where $1 \le i \le w$ and each $e(i)$ is the total number of elements in Q_i. Then, $a_1(t) = q_{1,1}, a_2(t) = q_{1,2}, \ldots, a_{e(1)}(t) = q_{1,e(1)}, a_{e(1)+1}(t) = q_{2,1}$, etc. In addition, $\forall 1 \le i \le w-1, b_i = \sum_{k=1}^{i} e(k)$.

The proof is based on a sorting algorithm that puts the ith smallest element in the list Q in the right position in the ith round and either increases or remains the value of the object function in each round. The algorithm is very similar to the well-known bubble sort algorithm. When the algorithm updates the sublists $Q_1, Q_2, \ldots, Q_w$, it also updates the sequences $\{a_i(t)\}$ and $\{b_i\}$, and vice versa.

Theorem 3.1: There is at least one ordered optimal partition for the wth-order list partition problem.

Proof:

1) We prove the theorem by showing that for every disordered partition, there exists an ordered partition with higher or equivalent value of the object function. Without loss of essential generality, we assume that $a_1(1) = \min_{1 \le i \le n} a_i(1)$. (We can always reorder the sublists or reorder the elements in a sublist without changing the value of the object function.) Let $t = 2$.
2) If $t = n$, stop. Otherwise, let $m(t) = \min_{i \ge t} a_i(t-1)$.
3) If $a_t(t-1) = m(t)$, set $a_i(t) = a_i(t-1)$, where $1 \le i \le n$, increase the value of t by one and then go to step 2. Otherwise, let $\alpha = t$ and $\beta = \arg\min_{i \ge t} a_i(t-1)$. (If there are several elements equivalent to $m(t)$, choose one

TABLE I
AN EXAMPLE OF THE ALGORITHM IN THE PROOF OF THEOREM 3.1

t	$a_1(t)$	$a_2(t)$	$a_3(t)$	$a_4(t)$	$a_5(t)$	$a_6(t)$	Q_1	Q_2	Q_3	value of the object function
1	1	6	2	3	4	5	1,6	2,3	4,5	14
2	1	2	6	3	4	5	1,2	6,3	4,5	16
3	1	2	3	6	4	5	1,2	3,6	4,5	16
4	1	2	3	4	6	5	1,2	3,4	6,5	18
5	1	2	3	4	5	6	1,2	3,4	5,6	18

arbitrarily.) Namely, $a_\beta(t-1) = \min_{i \geq t} a_i(t-1)$. Let $f(\alpha)$ be the index such that $a_\alpha(t-1) \in Q_{f(\alpha)}$ and $f(\beta)$ be the index such that $a_\beta(t-1) \in Q_{f(\beta)}$.

4) If $a_\alpha(t-1) = \min(Q_{f(\alpha)})$, increase the value of t by one, exchange the indexes of $Q_{f(\alpha)}$ and $Q_{f(\beta)}$, update $Q_1, Q_2, \ldots, Q_w, a_1(t), a_2(t), \ldots, a_n(t)$ and $b_1, b_2, \ldots, b_{w-1}$ accordingly and then go to step 2. It is clear that the value of the object function does not change after this step, since the algorithm only rearranges the order of sublists and does not add/remove elements into/from any sublists.

5) If $a_\alpha(t-1) > \min(Q_{f(\alpha)})$, swap $a_\alpha(t-1)$ and $a_\beta(t-1)$ so that $a_\alpha(t) = a_\beta(t-1), a_\beta(t) = a_\alpha(t-1)$ and $a_i(t) = a_i(t-1)$, $\forall 1 \leq i \leq n, i \neq \alpha, i \neq \beta$, increase the value of t by one and then go to step 2. Let T be the value of the object function before the swapping and T' be the value of the object function after the swapping. Let $Q_1, Q_2, \ldots, Q_w$ be the partition before the swapping and $Q'_1, Q'_2, \ldots, Q'_w$ be the partition after the swapping. We now prove that $T' \geq T$ as follows. If $f(\alpha) = f(\beta)$, it is clear that $T' = T$. We now prove the case that $f(\alpha) \neq f(\beta)$.

$$
\begin{aligned}
T' - T &= \sum_{i=1}^{w} \min(Q'_i) \cdot |Q'_i| - \sum_{i=1}^{w} \min(Q_i) \cdot |Q_i| \\
&= \sum_{i \neq f(\alpha), f(\beta)} \min(Q'_i) \cdot |Q'_i| - \min(Q_i) \cdot |Q_i| \\
&\quad + \min\left(Q'_{f(\alpha)}\right) \cdot |Q'_{f(\alpha)}| - \min\left(Q_{f(\alpha)}\right) \cdot |Q_{f(\alpha)}| \\
&\quad + \min\left(Q'_{f(\beta)}\right) \cdot |Q'_{f(\beta)}| - \min\left(Q_{f(\beta)}\right) \cdot |Q_{f(\beta)}| \\
&= 0 + 0 + \left(\min\left(Q'_{f(\beta)}\right) - \min\left(Q_{f(\beta)}\right)\right) \cdot |Q_{f(\beta)}| \\
&\geq 0
\end{aligned}
$$

The second term is zero, since before the swapping, $\min(Q_{f(\alpha)}) \leq \min\{a_\alpha(t-1), a_\beta(t-1)\}$ and therefore after the swapping, $\min(Q'_{f(\alpha)}) = \min(Q_{f(\alpha)})$. The third term is nonnegative, since $a_\alpha(t-1) > a_\beta(t-1)$ and therefore after the swapping, $\min(Q'_{f(\beta)}) \geq \min(Q_{f(\beta)})$.

6) Since the value of t increases by one in each round and n is finite, the above algorithm terminates within finite rounds. In addition, in each round of the algorithm, the value of the object function never decreases. Furthermore, when the algorithm terminates, the sequence $a_1(t), a_2(t), \ldots, a_n(t)$ is sorted in nondecreasing order, since the algorithm makes $a_i(i)$ the ith smallest element in the ith round and never moves it after that. Therefore, for every disordered partition, there exists an ordered partition with higher or equivalent value of the object function. Since the total number of partitions is finite, we have completed the proof. □

In Table I, we use an example to illustrate the algorithm in the above proof. Let $Q = \{1,2,3,4,5,6\}$ and $w = 3$. Assume that at $t = 1, Q_1 = \{1,6\}, Q_2 = \{2,3\}$ and $Q_3 = \{4,5\}$. Meanwhile, $a_1(1) = 1, a_2(1) = 6, a_3(1) = 2, a_4(1) = 3, a_5(1) = 4, a_6(1) = 5$. At $t = 2$, since 2 is the second smallest element in the list Q, the algorithm swaps 2 and 6 and, therefore, $a_1(2) = 1, a_2(2) = 2, a_3(2) = 6, a_4(2) = 3, a_5(2) = 4, a_6(2) = 5$. Meanwhile, $Q_1 = \{1,2\}, Q_2 = \{6,3\}$ and $Q_3 = \{4,5\}$. At $t = 3$, since 3 is the third smallest element in the list Q, the algorithm swaps 3 and 6 and, therefore, $a_1(3) = 1, a_2(3) = 2, a_3(3) = 3, a_4(3) = 6, a_5(3) = 4, a_6(3) = 5$. Similarly, $a_1(4) = 1, a_2(4) = 2, a_3(4) = 3, a_4(4) = 4, a_5(4) = 6, a_6(4) = 5$ and $a_1(5) = 1, a_2(5) = 2, a_3(5) = 3, a_4(5) = 4, a_5(5) = 5, a_6(5) = 6$.

Theorem 3.2 (Optimal Substructure): Let $Q_1, Q_2, \ldots, Q_w$ be an ordered optimal partition for Q. $\forall 2 \leq i \leq w-1$, let $L = Q_1 \cup Q_2 \cup \cdots \cup Q_i$ and $R = Q_{i+1} \cup Q_{i+2} \cup \cdots \cup Q_w$. Furthermore, let $L_j = Q_j$, where $1 \leq j \leq i$, and $R_k = Q_{i+k}$, where $1 \leq k \leq w-i$. Then, $L_1, L_2, \ldots, L_i$ is an ordered optimal partition of L and $R_1, R_2, \ldots, R_{w-i}$ is an ordered optimal partition of R.

Proof:

1) Due to the construction of $L_1, L_2, \ldots, L_i$ and L, it is clear that $L_1, L_2, \ldots, L_i$ is an ordered partition of L.

2) Suppose $L_1, L_2, \ldots, L_i$ is not an optimal partition of L. Let $L^*_1, L^*_2, \ldots, L^*_i$ be an ordered optimal partition of L. (The existence of such a partition has been proved in Theorem 3.1.) It is clear that $L_1, L_2, \ldots, L_i, R_1, R_2, \ldots, R_{w-i}$ is a partition of Q. Similarly, $L^*_1, L^*_2, \ldots, L^*_i, R_1, R_2, \ldots, R_{w-i}$ is also a partition of Q. Then,

$$
\begin{aligned}
&\sum_{s=1}^{w} \min(Q_s) \cdot |Q_s| \\
&\quad = \sum_{s=1}^{i} \min(L_s) \cdot |L_s| + \sum_{k=1}^{w-i} \min(R_k) \cdot |R_k| \\
&\quad < \sum_{s=1}^{i} \min(L^*_s) \cdot |L^*_s| + \sum_{k=1}^{w-i} \min(R_k) \cdot |R_k|.
\end{aligned}
$$

The last inequality is due to that $L_1, L_2, \ldots, L_i$ is not an optimal partition of L. However, the last inequality is contradictory to that $Q_1, Q_2, \ldots, Q_w$ is an optimal partition

of Q. Therefore, $L_1, L_2, \ldots, L_i$ must be an optimal partition of L.

3) Similarly, it is easy to prove that $R_1, R_2, \ldots, R_{w-i}$ is an ordered optimal partition of R. □

B. Analytical Results

1) The Uniform Distribution: Why Two Is Much Better Than One:

Lemma 4.1: If n is a multiple of w, then $\{1, 2, \ldots, n/w\}$, $\{(n/w)+1, (n/w)+2, \ldots, 2n/w\}, \ldots, \{((n(w-1))/w)+1, ((n(w-1))/w)+2, \ldots, n\}$ is an optimal solution of the wth-order list partition problem for U_n.

Proof:

1) Define a quadratic function $f(x_1, x_2, \ldots, x_{w-1})$ as follows:

$$\begin{aligned} &f(x_1, x_2, \ldots, x_{w-1}) \\ &= x_1 + \sum_{i=1}^{w-2} (x_i + 1) \cdot (x_{i+1} - x_i) \\ &\quad + (x_{w-1} + 1) \cdot (n - x_{w-1}) \\ &= \sum_{i=1}^{w-1} -x_i^2 + \sum_{i=1}^{w-2} x_i \cdot x_{i+1} + n \cdot x_{w-1} + n. \end{aligned}$$

Note that $f(x_1, x_2, \ldots, x_{w-1})$ is the object function of the wth-order list partition problem for $U_n = \{1, 2, \ldots, n\}$, when $x_1, x_2, \ldots, x_{w-1}$ are partition points. (i.e., the corresponding partition is $\{1, 2, \ldots, x_1\}, \{x_1 + 1, x_1 + 2, \ldots, x_2\}, \ldots, \{x_{w-1} + 1, x_{w-1} + 2, \ldots, n\}$.

2) We use partial derivative to obtain its local optimal point. First, since $(\partial f/\partial x_1) = 0, x_2 = 2x_1$. Second, since $(\partial f/\partial x_j) = 0, x_{j+1} - 2x_j + x_{j-1} = 0, \forall 2 \le j \le w-2$. Third, $(\partial f/\partial x_{w-1}) = 0, 2x_{w-1} - x_{w-2} = n$.

3) From 2 and the fact that $x_{w-1} < n$, we derive that $x_j = (nj/w)$, $\forall 1 \le j \le w-1$.

4) To prove that the local optimal point $x_j = (nj/w)$, where $1 \le j \le w-1$, is the global maximum point, we prove that the function $f(x_1, x_2, \ldots, x_{w-1})$ is concave as follows. First, $\sum_{i=1}^{w-2} x_i \cdot x_{i+1} = (1/2)\sum_{i=1}^{w-2} x_i^2 + x_{i+1}^2 - (x_i - x_{i+1})^2$. Then

$$\begin{aligned} &f(x_1, x_2, \ldots, x_{w-1}) \\ &= \sum_{i=1}^{w-1} -x_i^2 + \sum_{i=1}^{w-2} x_i \cdot x_{i+1} + n \cdot x_{w-1} + n \\ &= -\frac{1}{2}x_1^2 - \frac{1}{2}x_{w-1}^2 - \frac{1}{2}\sum_{i=1}^{w-2}(x_i - x_{i+1})^2 + n \cdot x_{w-1} + n. \end{aligned}$$

5) Since $-(1/2)x_1^2, -(1/2)x_{w-1}^2, -(1/2)\sum_{i=1}^{w-2}(x_i - x_{i+1})^2$ and $n \cdot x_{w-1} + n$ are all concave functions, $f(x_1, x_2, \ldots, x_{w-1})$ is also a concave function and therefore the local optimal point is the global optimal point.

Theorem 4.2: $\lim_{n\to\infty} \Gamma_{U_n}(w) = 1 - (1/w)$.

Proof:

1) Let $f(x) = f(x_1, x_2, \ldots, x_{w-1}) = \sum_{i=1}^{w-1} -x_i^2 + \sum_{i=1}^{w-2} x_i \cdot x_{i+1} + n \cdot x_{w-1} + n$ and $\bar{x} = (\overline{x_1}, \overline{x_2}, \ldots, \overline{x_{w-1}})$ be the real maximum of the function. Let $\hat{x} = (\widehat{x_1}, \widehat{x_2}, \ldots, \widehat{x_{w-1}})$ be the *integer* maximum of the function f. Let $x^* = (x_1^*, x_2^*, \ldots, x_{w-1}^*)$ be any rounding of $\bar{x}$ to nearest integers. Then obviously, $f(x^*) \le f(\hat{x}) \le f(\bar{x})$.

2) The evaluation of f at $\bar{x}$ gives $f(\bar{x}) = n^2/2(1-(1/w)) + n$.

3) Since $1 \le w \le n$ and $1 \le j \le w-1, 0 < ((nj)/w) \le \lceil((nj)/w)\rceil \le ((nj)/w) + 1$ and $0 \le ((nj)/w) - 1 \le \lfloor((nj)/w)\rfloor \le ((nj)/w)$. Therefore,

$$\begin{aligned} &f(x_1^*, x_2^*, \ldots, x_{w-1}^*) \\ &\ge \sum_{i=1}^{w-1} -\left(\frac{ni}{w} + 1\right)^2 + \sum_{i=1}^{w-2}\left(\frac{ni}{w} - 1\right) \\ &\quad \cdot \left(\frac{n \cdot (i+1)}{w} - 1\right) + n \cdot \left(\frac{n \cdot (w-1)}{w} - 1\right) + n \\ &= \frac{-n^2}{w^2} \cdot \sum_{i=1}^{w-1} i^2 + \frac{n^2}{w^2} \cdot \sum_{i=1}^{w-2} i \cdot (i+1) \\ &\quad + \left(1 - \frac{1}{w}\right) \cdot n^2 + \epsilon_1(n) \\ &= \frac{1}{2} \cdot \left(1 - \frac{1}{w}\right) \cdot n^2 + \epsilon_1(n) \end{aligned}$$

where $\epsilon_1(n)$ is a polynomial of n such that the degree of the variable n is at most one (w is seen as a constant).

4) Therefore, $\lim_{n\to\infty} \Gamma_{U_n}(w) = \lim_{n\to\infty}(f(\hat{x})/(n^2/2 + (n/2))) = 1 - (1/w)$ □

2) The Optimal Paging Problem and the List Partition Problem Are Equivalent:

Theorem 4.3:

a) Let $D = \{d_1, d_2, \ldots, d_n\}$, where $\sum_{i=1}^{n} d_i = 1$ and $\forall 1 \le i \le n-1, 0 \le d_{i+1} \le d_i$. Let A be the accumulation list of D. Then, $P_D(w) = n - T_A(w)$.

b) Let $A = \{a_1, a_2, \ldots, a_n\}$ be a concave list, where $1 \ge a_{i+1} \ge a_i \ge 0$ and $1 - a_n \le a_n - a_{n-1}$. Let d be the difference list of A. Then, $P_D(w) = n - T_A(w)$.

Proof:

1) Let $D_1, D_2, \ldots, D_w$ be an arbitrary wth-order partition of D and $A_1, A_2, \ldots, A_w$ be the associated partition of A. Then

$$\begin{aligned} &\sum_{i=1}^{w}\sum_{j=1}^{i} |D_j| \cdot \mathrm{sum}(D_i) \\ &= \sum_{j=1}^{w}\sum_{i=j}^{w} \mathrm{sum}(D_i) \cdot |D_j| \\ &= \sum_{j=1}^{w}\left(1 - \sum_{i=1}^{j-1} \mathrm{sum}(D_i)\right) \cdot |D_j| \\ &= \sum_{j=1}^{w} |D_j| - \sum_{j=1}^{w}\sum_{i=1}^{j-1} \mathrm{sum}(D_i) \cdot |D_j| \\ &= n - \sum_{j=1}^{w} \min(A_j) \cdot |A_j|, \left(\sum_{i=1}^{j-1} \mathrm{sum}(D_i) = \min(A_j)\right). \end{aligned}$$

2) Since D is nonnegative, A is nondecreasingly ordered. Therefore

$$\begin{aligned} P_D(w) &= \text{minimize} \sum_{i=1}^{w} \sum_{j=1}^{i} |D_j| \cdot \text{sum}(D_i) \\ &= n - \text{maximize} \sum_{i=1}^{w} \min(A_i) \cdot |A_i| \\ &= n - T_A(w). \end{aligned}$$

3) We now prove property b. It is clear that the result in 1 still holds. Since A is concave, nondecreasingly ordered and $1 - a_n \le a_n - a_{n-1}$, the sequence d_i is nonincreasing and nonnegative. Therefore

$$\begin{aligned} T_A(w) &= \text{maximize} \sum_{i=1}^{w} \min(A_i) \cdot |A_i| \\ &= n - \text{minimize} \sum_{i=1}^{w} \sum_{j=1}^{i} |D_j| \cdot \text{sum}(D_i) \\ &= n - P_D(w). \end{aligned}$$

□

3) Lower Bounds for Concave Lists: We first define a linear operation on a list and then prove that the optimality of a partition is preserved under a linear function in terms of throughput.

Definition: For any finite list $Q = \{q_1, q_2, \ldots, q_n\}$, we define $\alpha \cdot Q + \beta = \{\alpha \cdot q_1 + \beta, \alpha \cdot q_2 + \beta, \ldots, \alpha \cdot q_n + \beta\}$

For example, assume that $Q = \{0.1, 0.2, 0.3, 0.4\}, \alpha = 4$ and $\beta = 0.01$. Then, $\alpha \cdot Q + \beta = \{4 \cdot 0.1 + 0.01, 4 \cdot 0.2 + 0.01, 4 \cdot 0.3 + 0.01, 4 \cdot 0.4 + 0.01\} = \{0.41, 0.81, 1.21, 1.61\}$.

Lemma 4.3: If $Q_1^*, Q_2^*, \ldots, Q_w^*$ is an optimal partition of the wth-order list partition problem for the list Q of size n, then:

a) for any $\alpha > 0, T_{\alpha \cdot Q + \beta}(w) = \alpha \cdot T_Q(w) + \beta \cdot n$;
b) for any $\alpha > 0, \alpha \cdot Q_1^* + \beta, \alpha \cdot Q_2^* + \beta, \ldots, \alpha \cdot Q_w^* + \beta$ is an optimal partition of the wth-order list partition problem for the list $\alpha \cdot Q + \beta$.

Proof: The proof is straightforward and will be omitted here. □

Lemma 4.4: For any list $D = \{d_1, d_2, \ldots, d_n\}$, where $\forall 1 \le i \le n-1, 0 \le d_{i+1} \le d_i \le 1$ and $\sum_{i=1}^{n} d_i = 1, \sum_{i=1}^{n} i \cdot d_i \le ((n+1)/2)$.

Proof: 1. According to Corollary 2 of Theorem 2 in [31], $\sum_{i=1}^{n} i \cdot d_i$ is maximized when $d_1 = d_2 = \cdots = d_n = 1/n$. Therefore, for an arbitrary list D, we have $\sum_{i=1}^{n} i \cdot d_i \le \sum_{i=1}^{n} i \cdot (1/n) = ((n+1)/2)$. □

Theorem 4.4: For any basic concave list A of cardinality n, Let $D = \{d_1, d_2, \ldots, d_n\}$ be the associated difference list. Then:

a) for any natural number $w, ((T_A(w))/(T_A(n))) = ((n - P_D(w))/(n - \sum_{k=1}^{n} k \cdot d_k))$.
b) For any fixed natural number w, define $L(d_1, d_2, \ldots, d_n) = ((n - P_D(w))/(n - \sum_{k=1}^{n} k \cdot d_k))$. Let $S = \{(d_1, d_2, \ldots, d_n) : \sum_{i=1}^{n} d_i = 1, 0 \le d_n \le d_{n-1} \le \cdots \le d_1 \le 1\}$. Then, $\min_{(d_1, d_2, \ldots, d_n) \in S} L(d_1, d_2, \ldots, d_n)$ exists.

Proof:

1) According to Theorem 4.3, $T_A(w) = n - P_D(w)$. For any basic concave list $A = \{a_1, a_2, \ldots, a_n\}$, let $D = \{d_1, d_2, \ldots, d_n\}$ be the associated difference list. First, due to the construction of the difference list, $\sum_{i=1}^{n} d_i = 1$. Then

$$\begin{aligned} T_A(n) &= \sum_{i=1}^{n} a_i = \sum_{i=1}^{n-1} \sum_{j=1}^{i} d_j \left(a_1 = 0, a_{i+1} = \sum_{j=1}^{i} d_j \right) \\ &= \sum_{j=1}^{n} (n-j) \cdot d_j = n - \sum_{j=1}^{n} j \cdot d_j \left(\sum_{j=1}^{n} d_j = 1 \right). \end{aligned}$$

Then, $((T_A(w))/(T_A(n))) = ((n - P_D(w))/(n - \sum_{k=1}^{n} k \cdot d_k))$.

2) We now prove property b. It is easy to see that for any $w, L(\cdot)$ is a continuous function, since both $n - P_D(w)$ and $n - \sum_{k=1}^{n} k \cdot d_k$ are continuous. Since, S is closed and bounded, S is a compact set. Therefore, the image of S under $L(\cdot)$ is also a compact set, in which the minimum element exists. □

The first property of theorem 4.4 allows us to derive the normalized throughput by solving the optimal paging problem. The second property shows the existence of the minimum normalized throughput among basic concave lists. Since the object function $L(d_1, d_2, \ldots, d_n)$ is nonlinear and it includes a discrete optimization problem, it is difficult to analytically derive the minimum normalized throughput. Therefore, we derive the following lower bound of the normalized throughput.

Theorem 4.5: For any basic concave list A of cardinality n and any natural number w:

a) if n is a multiple of $w, \Gamma_A(w) \ge (1/2) \cdot (1 - (1/w))$;
b) $\Gamma_A(w) \ge (1/2) \cdot (1 - (1/w)) - \epsilon(n)$, where $\lim_{n \to \infty} \epsilon(n) = 0$.

Proof:

1) Let $D_{\text{worst}} = \{1/n, 1/n, \ldots, 1/n\}$, where $n = |D_{\text{worst}}|$. When n is a multiple of $w, P_{D_{\text{worst}}}(w) = \sum_{i=1}^{w} (1/w) \cdot ((n \cdot i)/w) = (n/2) \cdot (1 + (1/w))$.

$$\begin{aligned} T_A(w) &= n - P_D(w) \text{ (Theorem 4.3)} \\ &\ge n - P_{D_{\text{worst}}}(w) \text{ (Lemma 4.2)} \\ &= n - \left(\frac{n}{2}\right) \cdot \left(1 + \frac{1}{w}\right) \\ &= \left(\frac{n}{2}\right) \cdot \left(1 - \frac{1}{w}\right) \end{aligned}$$

2) For any basic concave list $A = \{a_1, a_2, \ldots, a_n\}$, let $D = \{d_1, d_2, \ldots, d_n\}$ be the associated difference list. We know that $\sum_{i=1}^{n} d_i = 1$ (See proof in Theorem 4.4). Then

$$\begin{aligned} T_A(n) &= n - \sum_{j=1}^{n} j \cdot d_j \text{ (Proof in Theorem 4.4)} \\ &\le n - 1 \end{aligned}$$

3) Based on 1 and 2

$$\frac{T_A(w)}{T_A(n)} \geq \frac{\frac{n}{2}}{n-1} \cdot \left(1-\frac{1}{w}\right) \geq \frac{1}{2} \cdot \left(1-\frac{1}{w}\right)$$

4) We now prove property b. First

$$P_{D_{\text{worst}}} \leq \sum_{i=1}^{w} \left(\frac{\frac{ni}{w}+1}{n}\right) \cdot \left(\frac{ni}{w}+1\right) = \sum_{i=1}^{w} \left(\frac{1}{n}\right) \cdot \left(\frac{ni}{w}+1\right)^2 = \frac{n}{2} \cdot \left(1+\frac{1}{w}\right) + w + 1 + \frac{w}{n}.$$

5) Then

$$T_A(w) = n - P_D(w) \geq n - P_{D_{\text{worst}}}(w) = \frac{n}{2} \cdot \left(1-\frac{1}{w}\right) - \left(w+1+\frac{w}{n}\right).$$

6) Similar to the proof of property a, we have $((T_A(w))/(T_A(n))) \geq (1/2) \cdot (1-(1/w)) - \epsilon(n)$, where $\epsilon(n) = ((w+1)/n) + (w/n^2)$ and clearly $\lim_{n\to\infty} \epsilon(n) = 0$. □

We have derived a lower bound of the normalized throughput for any basic concave list in Theorem 4.5. We now extend the result. We first define a nonnegative list and a nondecreasing list.

Definitions:

1) A list $Q = \{q_1, q_2, \ldots, q_n\}$ is said to be nonnegative if each element q_i is nonnegative, where $1 \leq i \leq n$.
2) A list $Q = \{q_1, q_2, \ldots, q_n\}$ is said to be nondecreasing if $q_1 \leq q_2 \leq \cdots \leq q_n$

We now present an extended version of Theorem 4.5.

Theorem 4.6:

a) For any nonnegative, nondecreasing, concave list $Q = \{q_1, q_2, \ldots, q_n\}$, if $q_n \neq q_1$, there exists a basic concave list $A = \{a_1, a_2, \ldots, a_n\}$, two real numbers $\alpha > 0$ and $\beta \geq 0$, such that $Q = \alpha \cdot A + \beta$.

b) For any nonnegative, nondecreasing, concave list $Q = \{q_1, q_2, \ldots, q_n\}$, if $q_n \neq q_1, \Gamma_Q(w) \geq (1/2) \cdot (1-(1/w)) - \epsilon(n)$, where $\lim_{n\to\infty} \epsilon(n) = 0$.

Proof:

1) Choose $\alpha = (q_n - q_{n-1}) + (q_n - q_1)$ and $\beta = q_1$. First, since Q is nondecreasing and $q_n \neq q_1, \alpha > 0$. Since Q is nonnegative, $\beta \geq 0$.
2) $Q = \alpha \cdot ((Q-\beta)/\alpha) + \beta$.
3) We now prove that $A = ((Q-\beta)/\alpha) = \{0, ((q_2 - q_1)/\alpha), ((q_3 - q_1)/\alpha), \ldots, ((q_n - q_1)/\alpha)\}$ is a basic concave list. Since $\alpha > 0$ and Q is concave, A is also concave. Clearly, $a_1 = 0$. Since $\alpha \geq q_n - \beta$ and Q is nondecreasing, $\forall 1 \leq i \leq n, 0 \leq a_i \leq 1$. Since $\alpha > 0$ and Q is nondecreasing, A is also nondecreasing. Moreover,

$$1 - a_n = 1 - \frac{q_n - q_1}{\alpha} = \frac{q_n - q_{n-1}}{\alpha} = \frac{q_n - q_1}{\alpha} - \frac{q_{n-1} - q_1}{\alpha} = a_n - a_{n-1}$$

Thus, A is a basic concave list.

4) We now prove the property b.

$$\frac{T_Q(w)}{T_Q(n)} = \frac{\alpha \cdot T_A(w) + \beta \cdot n}{\alpha \cdot T_A(n) + \beta \cdot n} \text{ (Lemma 4.3)} \geq \frac{\alpha \cdot T_A(w)}{\alpha \cdot T_A(n)} \left(\frac{\alpha \cdot T_A(w)}{\alpha \cdot T_A(n)} \leq 1, \beta \cdot n \geq 0\right) = \frac{T_A(w)}{T_A(n)} \geq \frac{1}{2} \cdot \left(1-\frac{1}{w}\right) - \epsilon(n), \text{ (Theorem 4.5)}$$

where $\lim_{n\to\infty} \epsilon(n) = 0$. □

4) Lower Bounds for Arbitrary Lists:

Theorem 4.9: Let $D = \{d_1, d_2, \ldots, d_n\}$, where $\forall 1 \leq i \leq n, d_i \in [0,1]$ and $\sum_{i=1}^{n} d_i = 1$. Then, $((n - VP_D(2))/(n - \sum_{k=1}^{n} k \cdot d_k)) > (1/2) \cdot (-1 + (2/(1+p)))$, where $p = ((\sum_{k=1}^{n} k \cdot d_k)/n)$.

Proof:

1) Let $p = ((\sum_{k=1}^{n} k \cdot d_k)/n)$. Clearly, $p \in [0,1]$. Let $x = \sum_{i=\lceil((1+p)/2)\cdot n\rceil}^{n} d_i$. We claim that $x \leq ((2 \cdot p)/(1+p))$, since otherwise

$$\sum_{k=1}^{n} k \cdot d_k \geq \left(\left\lceil \left(\frac{1+p}{2}\right) \cdot n \right\rceil\right) \cdot x > \left(\frac{1+p}{2}\right) \cdot n \cdot \left(\frac{2 \cdot p}{1+p}\right) = p \cdot n.$$

which is contradictory to $p = ((\sum_{k=1}^{n} k \cdot d_k)/n)$.

2)

$$VP_D(2) \leq (1-x) \cdot \left(\left\lceil \left(\frac{1+p}{2}\right) \cdot n \right\rceil - 1\right) + x \cdot n < (1-x) \cdot \left(\frac{1+p}{2}\right) \cdot n + x \cdot n.$$

Then,

$$n - VP_D(2) > n \cdot (1-x) - n \cdot (1-x) \cdot \left(\frac{1+p}{2}\right) = n \cdot (1-x) \cdot \left(\frac{1-p}{2}\right) \geq n \cdot \left(1 - \frac{2 \cdot p}{1+p}\right) \cdot \left(\frac{1-p}{2}\right) = n \cdot \left(\frac{(1-p)^2}{2+2 \cdot p}\right).$$

3)

$$\frac{n - VP_D(2)}{n - \sum_{k=1}^{n} k \cdot d_k} > \frac{1-p}{2+2\cdot p} = \frac{1}{2}\cdot\left(-1+\frac{2}{1+p}\right).$$

□

Theorem 4.11: For any nonnegative, nondecreasing list $Q = \{q_1, q_2, \ldots, q_n\}$, if $q_n \neq q_1, ((T_Q(2))/(T_Q(n))) \geq ((1-2\cdot\sqrt{p}+p)/(1-p))$, where $p \in (0,1)$ is defined as in the above theorem.

Proof:

1) According to the previous theorem, There exists a quasi-basic list $A = \{a_1, a_2, \ldots, a_n\}$ and two real numbers $\alpha > 0$ and $\beta \geq 0$, such that $Q = \alpha \cdot A + \beta$. Let $D = \{d_1, d_2, \ldots, d_n\}$ be the associated difference list of A and $p = ((\sum_{k=1}^{n} k \cdot d_k)/n)$. Clearly, $p \in (0,1)$. (That $p = 1$ results in $q_1 = q_2 = \cdots = q_n$ and violates the condition that $q_1 \neq q_n$.) Let $x = \sum_{i=\lceil (r)\cdot(p\cdot n)+(1-r)\cdot(n)\rceil}^{n} d_i$, where $r \in [0,1]$. We claim that $x \leq (p/(r\cdot p+(1-r)))$, since otherwise

$$\sum_{k=1}^{n} k \cdot d_k \geq x \cdot \lceil (r)\cdot(p\cdot n)+(1-r)\cdot(n)\rceil > \frac{p}{r\cdot p+(1-r)}\cdot(r\cdot p+(1-r))\cdot n = p\cdot n.$$

2)

$$\begin{aligned} P_D(2) &\leq (1-x)\cdot(\lceil (r)\cdot(p\cdot n)+(1-r)\cdot(n)\rceil - 1) + x\cdot n \\ &\leq (1-x)\cdot((r)\cdot(p\cdot n)+(1-r)\cdot(n)) + x\cdot n \\ &\leq n\cdot\left\{\left(1-\frac{p}{r\cdot p+(1-r)}\right)\cdot[r\cdot p+(1-r)] + \frac{p}{r\cdot p+(1-r)}\right\}. \end{aligned}$$

3) $((n-P_D(2))/(n-\sum_{k=1}^{n} k\cdot d_k)) \geq (((1-r)\cdot p + r - (p/(r\cdot p+1-r)))/(1-p))$.

4) Let $f(r,p) = (((1-r)\cdot p + r - (p/(r\cdot p+1-r)))/(1-p)) = ((r(1-r)(1-p))/(1-r(1-p)))$. For each $p \in (0,1)$, we find the optimal r to maximize $f(r,p)$ as follows. First, $(\partial f/\partial r) = (1/(1-p))\cdot(-p+1+((p\cdot(p-1))/((r\cdot p+1-r)^2)))$. Since $r = ((1-\sqrt{p})/(1-p))$ is the unique solution for $(\partial f/\partial r) = 0$ in $[0,1]$, it is therefore the unique local optimal in $[0,1]$.

5) We now prove that for each fixed $p \in (0,1)$, the function f is a concave function of r. First, $((\partial^2 f)/(\partial r^2)) = ((2\cdot p\cdot(p-1))/((1-r\cdot(1-p))^3))$. Since $p \in (0,1)$, it is clear $2\cdot p\cdot(p-1) < 0$. Furthermore,

$$1 - r\cdot(1-p) > 1 - r, (p \in (0,1)) \geq 0, (r \in [0,1]).$$

Then, for each fixed $p \in (0,1), ((\partial^2 f)/(\partial r^2)) < 0$ and, therefore, f is a concave function. Thus, $r = ((1-\sqrt{p})/(1-p))$ is the global maximum point.

6) Then

$$\frac{n - P_D(2)}{n - \sum_{k=1}^{n} k\cdot d_k} \geq \max_{r\in[0,1]} f(r,p) = f\left(\frac{1-\sqrt{p}}{1-p}, p\right) = \frac{1-2\cdot\sqrt{p}+p}{1-p}.$$

□

ACKNOWLEDGMENT

The authors thank the editor and the reviewers for their help to improve our paper. The authors particularly thank one of the anonymous reviewers for providing in-depth and valuable suggestions. The authors also thank M. R. Perlman for reading the first draft of the paper and providing suggestions.

REFERENCES

[1] Y. Amir, D. Dolev, S. Kramer, and D. Malki, "Transis: A communication sub-system for high availability," in *IEEE 22nd Int. Symp. on Fault-Tolerant Computing*, July 1992, pp. 76–84.

[2] K. P. Birman and R. V. Renesse, *Reliable Distributed Computing With the ISIS Toolkit*. Los Alamitos, CA: IEEE Computer Society Press, 1994.

[3] R. V. Renesse, T. M. Hickey, and K. P. Birman, "Design and Performance of Horus: A Lightweight Group Communications System," Dept. Computer Science, Cornell Univ., Ithaca, NY, Tech. Rep. TR94-1442, Aug. 1994.

[4] G. D. H. Hunt, "Multicast flow control on local area networks," Ph.D. dissertation, Dept. Computer Science, Cornell Univ., Ithaca, NY, May 1995.

[5] X.-M. Chen, "Gigabit Network Multicast Protocols. Chap 4, Flow Control Techniques," Ph.D. dissertation, Dept. ECE, Univ. of California, Santa Barbara, 1996.

[6] A. P. Naryan, "Reliable transfer of data in a local area network with multicast distribution," in *IEEE Proc. 15th Conf. Local Computer Networks*, Sept. 1990, pp. 310–319.

[7] B. J. Dempsey, J. C. Fenton, and A. C. Weaver, "The multidriver: A reliable multicast service using Xpress transfer protocol," in *IEEE Proc. 15th Conf. Local Computer Networks*, Sept. 1990, pp. 351–358.

[8] S. Armstrong, A. Freier, and K. Marzullo, "Multicast transport protocol," Xerox, Apple, Cornell Univ., RFC 1301, 1992.

[9] R. Braudes and S. Zabele, "Requirements for multicast protocols," TSAC, RFC 1458, 1993.

[10] J. C. Lin and S. Paul, "RMTP: A reliable multicast transport protocol," in *Proc. INFOCOM*, San Francisco, CA, Mar. 1996, pp. 1414–1424.

[11] R. Yavatkar, J. Griffoen, and M. Sudan, "A reliable dissemination protocol for interactive collaborative applications," in *Proc. ACM Multimedia*, San Francisco, CA, 1995, pp. 333–344.

[12] Reliable multicast transport (RMT) [Online]. Available: http://research.ivv.nasa.gov/RMP/Docs/index.html

[13] C. K. Miller, "Reliable multicast protocols: A practical view," *IEEE Local Computer Network*, 1997.

[14] S. Floyd, V. Jacobson, C.-g. Liu, S. McCanne, and L. Zhang, "A reliable multicast framework for light-weight sessions and application level framing," *IEEE/ACM Trans. Networking*, vol. 5, pp. 784–803, Dec. 1997.

[15] H. W. Holbrook, S. K. Singhal, and D. R. Cheriton, "Log-based receiver-reliable multicast for distributed interactive simulation," in *ACM SIGCOMM*, Oct. 1995, pp. 328–341.

[16] *Traffic Management Specification Version 4.0*, Mar. 1996.

[17] D. Cavendish, S. Mascolo, and M. Gerla, "Rate-based congestion control for multicast ABR traffic," in *Proc. GLOBECOM*, London, U.K., Nov. 1996, pp. 1114–1118.

[18] A. Acharya and B. R. Badrinath, "A framework for delivering multicast messages in networks with mobile hosts," *J. Special Topics Mobile Networks and Applications (MONET)*, vol. 1, no. 2, pp. 199–219.

[19] C. L. Williamson, T. G. Harrison, W. L. Mackrell, and R. B. Bunt, "Performance evaluation of the MoM mobile multicast protocol," *Mobile Networks and Applications*, pp. 189–201, 1998.
[20] A. Bartoli, "Group-based multicast and dynamic membership in wireless networks with incomplete spatial coverage," *Mobile Networks and Applications*, pp. 175–188, 1998.
[21] M. Gerla, C.-C. Chiang, and L. Zhang, "Tree multicast strategies in mobile, multihop wireless networks," *Mobile Networks and Applications*, pp. 193–207, 1999.
[22] S. Basagni, I. Chlamtac, and V. R. Syrotiuk, "Location aware one-to-many communication in mobile multi-hop wireless networks," in *IEEE Vehicular Technology Conf.*, 2000, pp. 288–292.
[23] S. Mishra and L. Wu, "An evaluation of flow control in group communication," *IEEE/ACM Trans. Networking*, vol. 6, pp. 571–587, Oct. 1998.
[24] H. Wang and M. Schwartz, "Performance analysis of multicast flow control algorithms over combined wired/wireless networks," *IEEE J. Select. Areas Commun.*, vol. 15, no. 7, pp. 1349–1363, Sept. 1997.
[25] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," in *Proc. ACM SIGCOMM*, 1996.
[26] X. Li, S. Paul, and M. Ammar, "Layered video multicast with retransmissions (LVMR): Evaluation of hierarchical rate control," in *Proc. IEEE INFOCOM*, 1998.
[27] L. Vicisano, J. Crowcroft, and L. Rizzo, "TCP-like congestion control for layered multicast data transfer," in *Proc. IEEE INFOCOM*, 1998.
[28] M. H. Ammar and L.-R. Wu, "Improving the throughput of point-to-multipoint ARQ protocols through destination set splitting," in *Proc. IEEE INFOCOM*, 1992.
[29] N. Shacham, "Multipoint communication by hierarchically encoded data," in *Proc. IEEE INFOCOM*, 1992.
[30] S. Bhattacharyya, J. F. Kurose, D. Towsley, and R. Nagarajan, "Efficient rate-controlled bulk data transfer using multiple multicast groups," in *Proc. IEEE INFOCOM*, 1998.
[31] C. Rose and R. Yates, "Minimizing the average cost of paging under delay constraints," *Wireless Networks*, 1995.
[32] J. S. M. Ho and I. F. Akyildiz, "Mobile user location update and paging under delay constraint," *Wireless Networks*, vol. 1, no. 4, pp. 413–425, 1995.
[33] A. Ballardie, "Core based trees (CBT Version 2) Multicast routing—Protocol specification,", RFC 2189, Sept. 1997.
[34] T. Ballardie, P. Francis, and J. Crowcroft, "Core based trees (CBT): An architecture for scalable inter-domain multicast routing," in *Proc. ACM SIGCOMM*, Sept. 1993.
[35] S. E. Deering, "Multicast routing in internetworks and extended LANs," *ACM Comput. Commun. Rev.*, vol. 19, no. 4, pp. 55–64, 1998.
[36] ——, "Host extension for IP multicasting,", RFC 1112, Aug. 1989.
[37] S. E. Deering, D. Estrin, D. Farinacci, V. Jacobson, C.-G. Liu, and L. Wei, "An architecture for wide-area multicast routing," in *Proc. ACM SIGCOMM'91*, Oct. 1994, pp. 126–135.
[38] ——, "The PIM architecture for wide-area multicast routing," *IEEE/ACM Trans. Networking*, vol. 4, pp. 153–162, Apr. 1996.
[39] J. Moy, "Multicast extensions to OSPF,", RFC-1584, Mar. 1994.
[40] D. Thaler, D. Estrin, and D. Meyer, Border Multicast Routing Protocol (BGMP): Protocol Specification, Apr. 1998. Internet Draft, draft-ietf-idmrgum-01.txt.
[41] S. Keshav and S. Paul, "Centralized multicast," in *Proc. 7th Int. Conf. Network Protocols (ICNP'99)*, 1999.
[42] B. Krishnamachari, R.-H. Gau, S. B. Wicker, and Z. J. Haas, "Optimal sequential paging in cellular networks," , submitted for publication.
[43] J. W. Byers, M. Luby, M. Mitzenmacher, and A. Rege, "A digital fountain approach to reliable distribution of bulk data," in *Proc. ACM SIGCOMM*, 1998, pp. 56–67.
[44] J. Gemmell, "ECSRM—Erasure correcting scalable reliable multicast," Microsoft Research, Tech. Rep. MS-TR-97-20, June 1997.
[45] C. Huitema, "The case for packet level FEC," in *Proc. IFIP 5th Int. Workshop Protocols for High Speed Networks*, Sophia Antipolis, France, Oct. 1996, pp. 134–148.
[46] J. Nonnenmacher and E. W. Biersack, "Reliable multicast: Where to use forward error correction," in *Proc. IFIP 5th Int. Workshop Protocols for High Speed Networks*, Sophia Antipolis, France, Oct. 1996, pp. 134–148.
[47] ——, "Asynchronous multicast push: AMP," in *Proc. Int. Conf. Computer Communications*, Cannes, France, Nov. 1997.
[48] J. Nonnenmacher, E. W. Biersack, and D. Towsley, "Parity-based loss recovery for reliable multicast transmission," in *Proc. ACM SIGCOMM*, 1997.
[49] L. Rizzo and L. Vicisano, "A reliable multicast data distribution protocol based on software FEC techniques," in *Proc. HPCS*, Greece, June 1997.
[50] E. Schooler and J. Gemmell, "Using multicast FEC to solve the midnight madness problem," Microsoft Research, Tech. Rep. MS-TR-97-25, Sept. 1997.
[51] S. Acharya, M. Franklin, and S. Zdonik, "Dissemination-based data delivery using broadcast disks," *IEEE Person. Commun.*, pp. 50–60, Dec. 1995.
[52] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*. New York: McGraw-Hill, 1990, ch. 16.

Rung-Hung Gau (M'01) received the B.S.E.E. degree from National Taiwan University, Taipei, Taiwan, R.O.C., in 1994 and the M.S.E.E. degree from the University of California, Los Angeles, CA, in 1997. In 2001, he received the Ph.D. degree from Cornell University, Ithaca, NY.

Since February 2002, he has been an Assistant Professor with National Sun Yet-Sen University, Kaoshiung, Taiwan. His research interests include mobility management, flow and congestion control, quality of service issues, and application of optimization techniques in network problems.

Zygmunt J. Haas (S'84–M'88–SM'90) received the B.Sc. and M.Sc. degrees from Technion and Tel-Aviv University in 1979 and 1985, respectively, both in electrical engineering, and the Ph.D. degree from Stanford University, Stanford, CA, in 1988.

He subsequently joined AT&T Bell Laboratories in the Network Research Department, where he pursued research on wireless communications, mobility management, fast protocols, optical networks, and optical switching. From September 1994 until July 1995, he worked for the AT&T Wireless Center of Excellence, where he investigated various aspects of wireless and mobile networking, concentrating on TCP/IP networks. In August 1995, he joined the faculty of the School of Electrical and Computer Engineering at Cornell University, Ithaca, NY. He is an author of numerous technical papers and holds 14 patents in the fields of high-speed networking, wireless networks, and optical switching. His interests include mobile and wireless communication and networks, personal communication service, and high-speed communication and protocols.

Dr. Haas has organized several workshops, delivered tutorials at major IEEE and ACM conferences, and serves as editor of several journals and magazines, including the IEEE TRANSACTIONS ON NETWORKING. He has been a guest editor of three IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS issues ("Gigabit Networks," "Mobile Computing Networks," and "Ad-Hoc Networks"). He is the Chair of the IEEE Technical Committee on Personal Communications.

Bhaskar Krishnamachari (S'98) received the B.E. degree in electrical engineering (summa cum laude) from The Cooper Union for Advancement of Science and Art, New York, in 1998, and the M.S.E.E. from Cornell University, Ithaca, NY, in 1999, where he expects to receive the Ph.D. degree in 2002.

His research is focused on algorithmic and complexity issues in wireless networks.

Mr. Krishnamachari is a member of the Eta Kappa Nu and Tau Beta Pi engineering honor societies. He is the recipient of a three-year graduate fellowship and the Olin presidential fellowship.