Multicast Routing with Heterogeneous Quality

Michalis Faloutsos*	Rajesh Pankaj	Kenneth C. Sevcik
University	of	Toronto
Dept. of Computer Science	Dept. of Electrical Eng.	Dept. of Computer Science
mfalou@cs.toronto.edu	pankaj@comm.toronto.edu	kcs@cs.toronto.edu

Abstract

In the common multicast problem, it is fundamental that destinations receive exactly the same information. In this paper, we examine the heterogeneous problem where destinations receive subsets of common information. It is worth noting that this scenario can be met in practice in audio and video sessions, as long as the data is encoded with the hierarchical encoding scheme. In this paper, we focus on the case where each destination requires a specific subset of the total information. For this problem, we design polynomial algorithms that meet some efficiency criteria. Theoretical and experimental work demonstrates the advantage of the proposed algorithms over previous solutions.

Keywords: Heterogeneous multicast routing

1 Introduction

The main idea behind multicasting is to save network resources by not "repeating" information. Namely, instead of multiple point-to-point connections of the same information, it is more efficient to distribute the information on a tree. Therefore, at first, the idea of multicasting "different" information seems strange. However, with the hierarchical encoding that we describe below, we see that heterogeneity is not only possible, but it can further increase the efficiency of network use.

Hierarchical encoding is a layered way of encoding information that can appear in different quality levels, such as audio and video data. This encoding method has received a lot of attention from the networking community [Gha89] [KV89]. In hierarchical encoding, the information of a session is divided into an enumerated set of **levels**. Any continuous sequence of levels, including the first level corresponds to a **stream**. A destination can receive a stream, which corresponds to a different quality. It is important for streams to include the lower levels since higher levels can only be used on top of the lower levels. The lower streams correspond to lower quality, while higher streams to higher quality.

Considering the above, we can see how the notion of multicast can be extended. A node that we call source, distributes information to a number of destinations. Destinations may receive different streams. We call this problem the **heterogeneous multicast problem**. In contrast, we refer to the one-stream problem as the **homogeneous multicast problem**. The purpose of this paper is to propose efficient algorithms for the heterogeneous problem. We believe that an efficient algorithm has to satisfy the following three requirements.

Cost sensitivity. Given that higher streams use more resources, we have to consider the different resource requirements of the streams. In addition, we want our algorithms to be as efficient homogeneous algorithms for instances of the heterogeneous problem that degenerate to homogeneous

*This research and the author himself would have not been possible without the existence of Nikolaos Faloutsos. Part of this paper was written during the last month of Nikos, who left early leaving a lot of painting undone.

problems.

Low complexity. The heterogeneous problem is NP-complete, as a general case of the homogeneous problem, which is NP-complete [DW72]. For this we are interested in polynomial approximation algorithms only. For the rest of tis paper, the term *algorithm* will imply an approximation algorithm. Ideally, we want to maintain the complexity of the homogeneous algorithms.

Flexibility. The heterogeneous algorithms should be flexible to incorporate additional functional constraints, and thus be suitable for use in different variations of the multicast problem, that we list below. User synchronization and round-loop requirements introduce end-to-end delay constraints that have been addressed by Kompella et al. [KPP93] [KPP92] and by Zhu et al. [ZPGLA95]. Physical limitations of the switches bring forward degree constraints, an issue that has been studied by Bauer and Verma [BV95]. The dynamic user-participation requirement introduces the dynamic problem that has been studied theoretically by Bharath-Kumar and Jaffe [BKJ83], Imase and Waxman [IW91], Aspens et al. [AAF+93] Awerbuch and Azar [AA95] and more practically by Doar and Leslie [DL93] and Waxman [Wax88] [Wax93].

Finally, we want to stress the practical aspect of heterogeneous multicasting. First, heterogeneous multicasting can be used in most audio or video applications such as, (near) video-on-demand, teleconferencing, computer supported collaborative work (CSCW). Second, heterogeneity may even be compulsory in some cases. The great variety of machines and technologies that interoperate require this user-oriented adaptation. For example, in a teleconference with mobile and workstation participants, the mobile users may not be able to handle the higher streams, while the workstation users may demand it. Third, user-defined quality of service (QoS) is a fundamental notion for modern broadband networks. Furthermore, one can expect in the future more services for which the users will have to pay and expect quality guarantees, as opposed to the current "free but no guarantees" mentality of the Internet. Finally, current technology is able to support the necessary filtering. The RSVP protocol [ZDE+93] is a step in this direction, which also shows the trend for user-specified quality of service.

The rest of this paper is structured as follows. Section 2 presents our model and lists previous work. Section 3 describes our algorithms for the fixed heterogeneous problem. Section 4 presents our simulation results.

2 Model Definition and Related Work

First, we give an overview of the commonly used model of the problem. Given an undirected weighted graph $G(V, E)$ and a subset of **participant** nodes $S \subseteq V$, we want to find the minimum weight tree that spans the participants. The number of nodes $|V|$ is N and the number of participants $|S|$ is M. A number of algorithms exist with a WCI bounded by 2 [TM80] [KMB81] [RSC86] [Meh88]. A comprehensive survey of the algorithms is provided by Winter [Win87]. A metric of the quality of an approximation is the **worst-case inefficiency (WCI)** or competitive ratio of a multicast algorithm, which is defined as the maximum ratio of the cost of the tree of the algorithm over the optimal one.

Note that for the homogeneous case, the routing cost of an edge is its weight, in other words the cost is the distance. However, as we will see in the heterogeneous problem, the routing cost becomes a function of the required resources. The cost of a path is the sum of the routing cost of its edges. In this paper, when we refer to the path between two nodes, we imply the minimum cost or shortest path. Along these lines, the term "nearest" refers to the minimum cost path.

The union of the shortest paths between source and each destination forms a solution with a WCI of $O(M)$. We call this algorithm **Shortest Paths**. Note that Shortest Paths routes every destination separately independently of the other destinations.

An intuitive algorithm for the multicast problem was suggested by Takahashi and Matsuyama [TM80]. We start from the root and construct the multicast tree by adding one participant at a time. At each step we connect the participant that is nearest to some node of the partial multicast tree. We call this algorithm **Nearest Participant First (NPF)**. For undirected graphs, the worst case inefficiency of NPF is bounded by 2. More accurately, the bound is $2\ (1 - \frac{1}{l})$ where l is the leaves of the optimal tree [FPS97], which is lower than $2\ (1 - \frac{1}{M})$ [TM80]. Finally, in the case where the destinations arrive on-line (but do not leave), the worst case inefficiency is tightly bounded by $O(\log M)$ [BKJ83] [IW91].

Previous research on the routing aspect of the heterogeneous problem was conducted by Shacham [Sha92b] [Sha92a]. However, Shacham examines only a special case of the heterogeneous problem, which we call **best-effort problem**. The problem we address in this paper includes best-effort problem as a special case. In a best-effort problem, all destinations receive the highest stream of the session that the network congestion can allow, i.e., a stream is offered as long as there is enough bandwidth to route it. Clearly, this formulation fails to consider that some destinations may prefer not to receive the highest stream as we already mentioned. The algorithm proposed by Shacham is an extension of the Shortest Paths algorithm, and thus its WCI is at least that of the homogeneous case, which is $O(M)$. The complexity of the algorithm is polynomial and bounded by the complexity of the Dijkstra algorithm $O(N^2)$. In a more recent work, Shacham and Meditch [SM94] examine the problem of distributing unrelated streams over a given tree. Although they address a significantly different problem, their ideas inspired our work. More recently, MacCanne et al. [MJV96] proposed a receiver-driven adaptive scheme for heterogeneous multicasting, but they do not discuss routing issues.

Our work relies on one assumption: all the levels of a session have to use the same route. If this is not necessary, we can multicast each level independently only to the appropriate destinations. However, this independence requires synchronization of the levels coming from different routes, which is not trivial. On the other hand, conforming to this assumption can even be beneficial. Routing the levels together can facilitate the network management through the enforcement of priority policies. For example, giving higher priority to the lower and fundamental levels, we can shift the losses to the higher levels, and thus achieve graceful degradation of the quality.

Note that our algorithms do not need to assume undirected graphs: they can be used on directed graphs in a straightforward way. Note, however, that the performance of greedy algorithms degrades with the asymmetry of a graph as shown by Ramanathan [Ram96] and Faloutsos et al. [FPS96b]. However, our simulations are conducted on undirected graphs. We are currently studying the effect of the asymmetry of a graph in the heterogeneous problem.

3 The Fixed Heterogeneous Problem

In a heterogeneous session, destinations can define a range of streams, with which they are willing to participate in the session. Each stream in this range is a **preferred** stream for that destination. We call **fixed heterogeneous problem** the case where each destination defines only one preferred stream. The more general problem (multiple preferred streams) has been studied by Faloutsos et al. [FPS96a]. The fixed problem can be stated as follows: given a graph, a source and a set of destinations with one preferred stream each, we want to identify the minimum cost tree from the source that offers the preferred stream to each destination.

At this point, we have to redefine the cost on an edge to accommodate the cost sensitivity requirement that we mentioned in the introduction. The routing cost of an edge is defined as a function of the routing requirements of the connection. For the rest of this paper, we will use the concept of flow (in terms of bandwidth) to represent all network resources. We define the **cost function** $cost(e, f)$, which represents the cost of passing flow f on edge e. We define the cost of an edge e on which we want to route a connection of bandwidth b as: $cost(e, f+b) - cost(e, f)$, where f is the existing flow. We define the cost of a path/session as the sum of all the cost increases of all edges before and after routing the required bandwidth. Given a partial tree, the cost of connecting a destination with a stream is called **joining cost**, and depends on the destination, the stream and the partial tree of that moment. Note that the actual mathematical function can be tailored to reflect conditions and assumptions of the problem. A commonly-used cost function is the *usage*

$$cost(e, f) = \begin{cases} \alpha + \beta\ f, & \text{if } f < Cap \text{ and } \alpha, \beta \in \mathcal{R} \\ \infty, & \text{otherwise} \end{cases}$$

Alternatively, theoretical work suggests the *exponential* function as a requirement for competitive solutions [Plo95] [AA95].

$$cost(e, f) = e^f$$

Note that the algorithms we present below are compatible with any cost function.

For the heterogeneous problem, one way to extend the NPF algorithm, is to keep the selection criterion as "minimum cost" destination and use the cost as defined above. Observe that low stream nodes have an advantage over the high stream ones, since higher streams are more costly. We call this algorithm **Minimum Cost Participant First** or **MinCostPF** [FPS96a].

In another approach, we observe that lower stream destinations can tap into the higher streams and acquire the levels they want. Therefore, we route the higher stream destinations first. Among destinations of the higher stream, we choose the one nearest to the partial multicast tree. This strategy gives rise to the **Maximum Stream Participant First** or **MaxStrPF** algorithm.

1. START from the source.
2. REPEAT until all destinations are reached
   (a) SELECT destination: Find the set of unreached destinations that have maximum preferred stream. From this set, select the nearest (minimum cost) destination to the partial multicast tree, d.
   (b) JOIN: Include d and the related path in the tree.

Note that, when a destination intends to join the tree with stream s at a node that carries a lower stream, the joining cost includes the cost of bringing the required levels to the connection point.

Efficient implementations of both algorithms [FPS96a] have complexity equal to that of the homogeneous NPF algorithm, that is $O(M\ N^2)$. This satisfies the *low complexity* requirement. The *cost sensitivity* requirement is met by the new definition of the cost. Finally, the *flexibility* requirement is also satisfied since it follows the framework of NPF.

Note that the inefficiency of both algorithms is bounded by 2 for the cases where the problem approaches the homogeneous case. If we assume a linear relationship between cost and bandwidth, the following theorem holds for MaxStrPF.

Theorem 1 *Assume that the cost of an edge is linear to its flow. In the fixed heterogeneous problem with M participants, the WCI of MaxStrPF is bounded by*

$$WCI_{MaxStrPF} \leq \log(M)$$

The proof can be found in [FPS96a]. Intuitively, we can consider that MaxStrPF "shuffles" the joining order. This theorem says that this "shuffling" in the heterogeneous problem is never worse than an arbitrary shuffling of the destinations in a homogeneous case ($\Theta(\log M)$). In comparison, recall that the Shortest Paths [Sha92b] algorithm has a worst case inefficiency of ($\Theta(M)$).

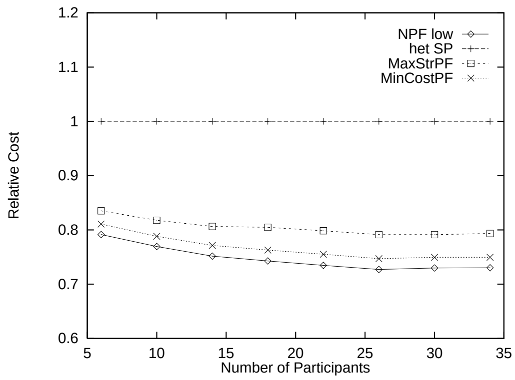

Figure 1: Relative cost versus the number of participants for stream cost 0.01.

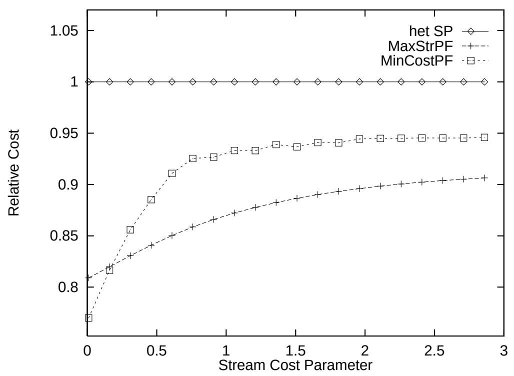

Figure 2: Relative cost versus the stream cost parameter for 14 participants.

4 Experimental Results

We implemented a simulator in C++ to examine the performance of the above algorithms by simulation. In the implementation, we had to make a lot of choices in terms of parameters and models. As a first step, we wanted to examine the existence of cases that differentiate the algorithms. As a next step, we would like to identify and model realistic scenarios. However, this is a non-trivial task given the diversity and volatility of real life. Furthermore, such experiments may interest only specialized applications and parties.

We chose to experiment on the family of graphs that we call **random euclidean** graphs. These graphs are widely used in network simulations, because they seem to have some characteristics of real networks [DL93]. We can create these graphs by uniformly distributing nodes on a euclidean plane, and set edges between nodes on a probabilistic basis that favors nodes of small euclidean distance, for example, for $v, w \in V$:

$$Prob\_Connect(v, w) = B\,\frac{e^{-Dist(v,w)}}{A\,L}, \quad A, B \in \mathcal{R}$$

In the above formula, $Dist()$ is the euclidean distance, and L is the maximum possible distance between nodes. A more detailed description of these graphs can be found by Doar and Leslie [DL93] or Waxman [Wax88]. By setting $A = 0.13$ and $B = 0.3$, we created a 40-node random euclidean with 108 edges. The participants are chosen randomly, and so is the root of the session among them. We used the average of thirty-two different executions in an attempt to smooth the irregularities of special cases. We assume that the information is divided into five distinct levels. The choice of preferred streams follows a uniform probability. We consider that each level requires a unit of flow, and assume the cost of an edge to be exponential with its flow: $cost(f, edg) = e^{K\,f}$, for any edge edg. The parameter K is a measure of the cost of routing, and we call it the **stream cost**. We can think of stream cost as measure of the cost of bandwidth and its value is crucial in the comparison between the algorithms.

The algorithms we compare with are the following:

1. *NPF low:* We perform NPF and offer the minimum stream of the session to all destinations. The cost NPF low is shown for comparison, but offering only the lowest stream is not acceptable in our formulation of the problem.

2. *hetSP:* We perform the Shortest Paths algorithm and offer the preferred stream to each destinations along the shortest path. Recall that hetSP corresponds to the algorithm suggested by Shacham [Sha92b].

3. *MinCostPF:* We perform Minimum Cost Participant First as described in the previous section.

4. *MaxStrPF:* We perform Maximum Stream Participant First as described in the previous section.

Figure 1 shows the relative cost of the solution versus the number of participants for a stream cost of $K = 0.01$. The cost of all algorithms is normalized by the cost of hetSP. It is easy to see that the number of participants does not have a great impact on the relative performance of the algorithms. However, we can observe a slight 2-3 per cent relative improvement of our algorithms (compared with hetSP). For this set of experiments, MinCostPF is as expensive as the NPF low, while MaxStrPF is costlier than these two. Finally, hetSP is less efficient comparatively giving worse solutions by approximately 25 per cent worse than MinCostPFand 20 per cent worse than MaxStrPF.

We wanted to examine the effect of the stream cost parameter on the behavior of the algorithms. Figure 2 shows the relative cost of the algorithms for varying values of stream cost and for 14 participants. In all the cases, MinCostPF and MaxStrPF outperform hetSP by as much as 25 and 17 per cent respectively for low stream cost values. Note that for low stream cost values, MinCostPF performs better, but as the stream cost parameter increases the MaxStrPF prevails. This can be explained if we recall that the main idea of MaxStrPF considers first the high stream connections. Finally, the consistently higher cost of the hetSP can be attributed to the "multicast-oblivious" nature of the algorithm.

5 Conclusions

In this paper, we examine the fixed heterogeneous problem, which is a generalization of the multicast problem. In the fixed problem, the variability of the offered data challenges the uniformity of the common multicast problem. In this paper, we suggest routing algorithms following the requirements with which we determine efficiency.

We propose the MaxStrPF and MinCostPF algorithms that are extensions of the NPF greedy algorithm. These algorithms meet our requirements of efficiency and outperform existing heterogeneous solutions. First, we prove that MaxStrPF has WCI of $\log(M)$, while a previous heterogeneous algorithm, hetSP [Sha92b], has a WCI of $\Theta(M)$. Second, experimental results indicate the advantage of both the proposed algorithms over hetSP: the difference reached a maximum of 25 per cent in our set of experiments.

Open Problems and Future Work. From a theoretical point of view, it would be interesting to examine the worst case inefficiency performance of MaxStrPF when the cost is an exponential function of the flow and the performance of MinCostPF for both cost models. More practically, we would like to examine our algorithms on specific applications. Fixing our simulation parameters to realistic values can offer valuable feedback on the design of the algorithms, which can be tailored to address successfully the given problem.

ACKNOWLEDGMENTS. The authors would like to thank Manolis Skordalakis for his support and Christos Faloutsos for several inspiring discussions.

References

[AA95] Baruch Awerbuch and Yossi

Azar. Competitive multicast routing. *Wireless Networks*, 1:107–114, 1995.

[AAF+93] J. Aspens, Y. Azar, A. Fiat, S. Plotkin, and O. Waarts. Online load balancing with applications to machine scheduling and virtual circuit routing. *Proc. of 25th ACM Symposium of the Theory of Computing (STOC)*, pages 623–631, 1993.

[BKJ83] K. Bharath-Kumar and J.M. Jaffe. Routing to multiple destinations in computer networks. *IEEE Trans. on Communications*, 31:343–351, 1983.

[BV95] Fred Bauer and Anujan Verma. Degree-constrained multicasting in point-to-point networks. *Proc. IEEE INFOCOM*, pages 369– 376, 1995.

[DL93] Matthew Doar and Ian Leslie. How bad is naive multicast routing? *Proc. IEEE INFOCOM*, pages 82–89, 1993.

[DW72] S.E. Dreyfus and R.A. Wagner. The Steiner problem in graphs. *Networks*, 1:195–207, 1972.

[FPS96a] Michalis Faloutsos, Rajesh Pankaj, and Kenneth C. Sevcik. Audio/video multicasting with hierarchical encoding: The fixed and the general problem. *University of Toronto, Technical Report*, 1996.

[FPS96b] Michalis Faloutsos, Rajesh Pankaj, and Kenneth C. Sevcik. Multicasting in directed graphs. *18th Biennial Symposium on Communications, June 2-5*, pages 29–32, 1996.

[FPS97] Michalis Faloutsos, Rajesh Pankaj, and Kenneth C. Sevcik. Tight(er) bounds for multicasting on directed graphs. *Technical Report U. of Toronto*, 1997.

[Gha89] M. Ghanbari. Two-layer coding of video signals for VBR networks. *IEEE Journal on Selected Areas on Communications*, 7(5):771–781, 1989.

[IW91] M. Imase and B.M. Waxman. Dynamic Steiner tree problem. *SIAM Journal on Discrete Mathematics*, 4:369–384, 1991.

[KMB81] L. Kou, G. Markowsky, and L. Berman. A fast algorithm for Steiner trees. *Acta Informatica, Springer-Verlag*, 15:141–145, 1981.

[KPP92] V. P. Kompella, J. C. Pasquale, and G. C. Polyzos. Multicasting for multimedia applications. *Proc. IEEE INFOCOM*, pages 2078–2085, 1992.

[KPP93] V. P. Kompella, J. C. Pasquale, and G. C. Polyzos. Multicast routing for multimedia communication. *IEEE/ACM Transaction on Networking*, pages 286–292, June 1993.

[KV89] M. Karlsson and M. Vetterli. Packet video and its integration into the network architecture. *IEEE Journal on Selected Areas on Communications*, 7(5):739–751, 1989.

[Meh88] Kurt Mehlhorn. A faster approximation algorithm for the Steiner problem in graphs. *Information Processing Letters*, pages 125–128, 1988.

[MJV96] S. McCanne, V. Jacobson, and M. Vetterli. Receiver-driven layered multicast. *ACM SIGCOMM*, pages 117–130, August, Stanford 1996.

[Plo95] S. Plotkin. Competitive routing in ATM networks. *IEEE Journal on Selected Areas in Communications*, pages 1128–1136, August 1995. Special issue on Advances in the Fundamentals of Networking. (Invited paper).

[Ram96] S. Ramanathan. An algorithms for multicast tree generation in networks with asummetric links. *Proc. IEEE INFOCOM '96*, pages 337–344, 1996.

[RSC86] V. J. Rayward-Smith and A. Clare. On finding Steiner vertices. *Networks*, 16:283–294, 1986.

[Sha92a] N. Shacham. Multicast routing of hierarchical data. *Proc. IEEE International Conference on Communication (ICC)*, pages 1217–1221, 1992.

[Sha92b] N. Shacham. Multipoint communication by hierarchically encoded data. *Proc. IEEE INFOCOM*, pages 2107–2114, 1992.

[SM94] N. Shacham and James S. Meditch. An algorithm for optimal multicast of multimedia streams. *Proc. IEEE INFOCOM*, pages 856–864, 1994.

[TM80] H. Takahashi and A. Matsuyama. An approximate solution for the Steiner problem in graphs. *Math. Japonica*, 24(6):573–577, 1980.

[Wax88] B.M. Waxman. Routing of multipoint connections. *IEEE Journal of Selected Areas in Communications*, pages 1617–1622, 1988.

[Wax93] B.M. Waxman. Performance evaluation of multipoint routing algorithms. *Proc. IEEE INFOCOM*, pages 980–986, 1993.

[Win87] P. Winter. Steiner problem in networks: a survey. *Networks*, 17:129–167, 1987.

[ZDE+93] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala. Rsvp: A new resource reservation protocol. *IEEE Network*, September 1993.

[ZPGLA95] Q. Zhu, M. Parsa, and J.J. Garcia-Luna-Aceves. A source-based algorithms for delay constrained minimum cost multicasting. *Proc. IEEE INFOCOM*, pages 377–385, 1995.