Resource Allocation in a Multicast Tree

Murali Kodialam
Bell Labs , Lucent Technologies
Holmdel NJ, USA
muralik@lucent.com

Steven H. Low
Dept of EEE
University of Melbourne, Australia
slow@ee.mu.oz.au

Abstract— **We consider how to allocate bandwidth in a multicast tree so as to optimize some global measure of performance. In our model each receiver has a budget to be used for bandwidth reservation on links along its path from the source, and each link has a cost function depending on the amount of total bandwidth reserved at the link by all receivers using that link. We formulate and solve a problem of allocating bandwidth in the multicast tree such that the sum of link costs is minimized.**

Keywords—**Multicast, Resource Allocation, Convex Optimization, Flow Control**

I. Introduction

Multicasting is under intense research and development; see a survey in [2]. In order to provide quality of service (QoS) to receivers in a multicast session, mechanisms have been designed to perform resource reservation on a multicast tree [6], [5]. However algorithms to decide how much resource to reserve and where are relatively unexplored. In [3] such an algorithm is proposed where each receiver has an end-to-end QoS requirement that is divided into QoS requirement on each link in the path from the source to the receiver either evenly or in inverse proportion to the current load on the link. Bandwidth is then allocated on each link to provide the required link QoS, with adjustment to account for link sharing with other receivers. In this paper we propose a different approach that, instead of distributing QoS requirement along a receiver's path evenly or proportionally, directly distributes the resource budget of a receiver along its path.

Suppose we are given a multicast tree where associated with each node is a (possibly empty) set of receivers. Each receiver has a bandwidth budget and the network's goal is to distribute these budget in some globally optimal manner, subject to the constraint that a receiver's budget can only be used to reserve bandwidth on links in its own path from the source. We will give two equivalent problem formulations, the path formulation (Section 3) and the cut formulation (Section 4). We outline an iterative algorithm for the solution of the path formulation that converges to the optimal solution. We give a finite algorithm to solve the cut formulation. Using the dual to the cut formulation, we give a distributed algorithm to solve the resource allocation problem and outline some preliminary ideas on its implementation.

II. Model

Let $G = (V, E)$ represent the given directed multicast tree. We assume that there are n nodes and hence $n - 1$ arcs in the tree. Let $s \in V$ represent the source. Let $l \in E$ denote a directed edge (link) in the tree. Let $t(l)$ and $h(l)$ represent the tail and the head nodes of link l. Each node in the tree has a sub-tree that is rooted from it. For a given link l, let $T(l) = (V(l), L(l))$ represent the tree that is rooted at $h(l)$ where $V(l)$ represents the set of vertices (including $h(l)$) in $T(l)$ and $L(l)$ represent the set of links in $T(l)$. Note that $L(l)$ does not include the link l. We use $\bar{L}(l)$ to represent $L(l) \cup \{l\}$. Note that there is a unique path from s to any node in G. Let $P(l)$ represent the links in the path from s to $h(l)$. Each node $v \in V$ has associated with it a set of receivers R_v. Each receiver j has a budget b_j, The budget at a given node $h(l)$, represented by B_l, is the sum of the budgets of the receivers at $h(l)$, i.e., $B_l = \sum_{j \in R_{h(l)}} b_j$. We will refer to B_l as the budget on link l. We assume that B_l can only be used on the path $P(l)$.

III. Path formulation

Our aim is to allocate resources on the links so that some global objective function for the tree is optimized. We assume that the objective is to minimize the sum of some decreasing convex function on the links. Let σ_{kl} represent the amount of budget from link k that is allocated to link $l \in P(k)$. Given a link l note that this link lies on $P(k)$ for all $k \in T(l)$. The

0-7803-5420-6/99/$10.00 (c) 1999 IEEE

resource allocation problem formulated below will be termed the path formulation.

$$\min \sum_{l \in E} f_l \left(\sum_{k \in \bar{L}(l)} \sigma_{kl} \right) \qquad (1)$$

subject to

$$\begin{aligned} \sum_{l \in P(k)} \sigma_{kl} &\leq B_k, \quad \forall k \in E & (2) \\ \sigma_{kl} &\geq 0, \quad \forall k, l \in P(k) & (3) \end{aligned}$$

Here, each f_l represents a cost on link l, such as delay through link l, as a function of the total bandwidth reserved at link l. For instance, if each link can be modeled as a $M/M/1$ queue, then $f_l(\mu) = (\mu - \lambda)^{-1}$ represents the average delay through link l that is receiving at a rate of λ (We assume that the receiving rate λ is fixed by quality of service requirement of the receivers in the subtree fed by link l). The objective is to minimize the total cost subject to budget constraints of the receivers.

There is one constraint corresponding to each link that dictates that the allocation made from that link should not exceed the budget of the link. Note that the constraint set is separable by links. The only interaction between resources occurs in the objective function. Therefore one can use an iterative scheme to compute the optimal solution. In each iteration every link individually solves a single-link problem to distribute its own budget along its path, based on the allocations by all other links in the last iteration. The links exchange their results, and the cycle repeats until there is no change in allocations.

To describe the algorithm precisely assume that the budget for all links except link k has been allocated. Let $\sigma^* = (\sigma^*_{il}, i, l \in E)$ represent the allocation. The single-link optimization problem PROB(k, σ^*), for link k is the following:

$$\min \sum_{l \in P(k)} f_l \left(\sum_{k' \neq k,\ k' \in \bar{L}(l)} \sigma^*_{k'l} + \sigma_{kl} \right)$$

subject to

$$\begin{aligned} \sum_{l \in P(k)} \sigma_{kl} &\leq B_k \\ \sigma_{kl} &\geq 0, \quad \forall l \in P(k) \end{aligned}$$

Note that given σ^* this problem only involves link k's variable $(\sigma_{kl}, l \in P(k))$, and can be solved using techniques in [7]. The path formulation (1–3) is solved by cyclically solving the subproblems PROB(k, σ^*) for each link, according to algorithm $\mathcal{A}$ below.

Algorithm $\mathcal{A}$

```
1. INITIALIZATION
  • $t = 0$,   $\sigma^t_{kl} = 0$,   $\forall k, l \in E$
  • Number the links in some arbitrary order from 1 to $n$.

2. ITERATIVE STEP
  • $t = t + 1$; $\sigma^t_{kl} = \sigma^{t-1}_{kl}$, $\forall k, l \in E$.
  • For $i = 1, 2, \ldots n$,
    – Let $\sigma^*_{kl} = \sigma^t_{kl}$ for all $k < i$.
    – Let $\sigma^t_{il}$ solve PROB($i, \sigma^*$).

3. TERMINATION CHECK
  • If $\sigma^{t-1}_{il} = \sigma^t_{il}$,  for all $i, l$, then stop;
  • Else go to Step 2.
```

In practice, the check for equality in the termination test is replaced by

$$|\sigma^t_{il} - \sigma^{t-1}_{il}| \leq \epsilon, \quad \forall i, l$$

for some small ϵ.

Theorem 1: Algorithm $\mathcal{A}$ solves the path formulation.

Proof:

(Sketch) Associate dual multipliers $\delta_k \geq 0$, $\forall k \in E$, with each constraint (2) and $\theta_{kl} \leq 0$ with the nonnegativity constraints. The Kuhn-Tucker conditions for optimality for (1–3) are, for all $k \in \bar{L}(l)$, $l \in E$,

$$f'_l(\sum_{j \in \bar{L}(l)} \sigma^*_{jl}) + \delta^*_k 1(k \in \bar{L}(l)) + \theta^*_{kl} = 0$$

where δ^*_k and θ^*_{kl} represent the optimal dual multipliers and σ^*_{kl} satisfies (2) and (3) and

$$\theta^*_{kl}\sigma^*_{kl} = 0, \qquad \delta^*_k \left(B_k - \sum_{l \in P(k)} \sigma^*_{kl} \right) = 0$$

Note that optimality conditions are separable in k. These conditions are the same, if the budgets for all links except link k are allocated in an arbitrary manner and the budget for link l is allocated optimally on the path $P(l)$. Further, note that the objective function is non-increasing throughout the execution of the algorithm. This is because each subproblem PROB(k, σ^*) by link k decreases $\sum_{l \in P(k)} f_l$ but leaves other costs $f_{l'}, l' \notin P(k)$, unchanged, as the sets of variables $(\sigma_{k'l}, k' \in \bar{L}(l), l \in P(k))$ and $(\sigma_{k'l'}, k' \in \bar{L}(l'), l' \notin P(k))$ are disjoint. Therefore, there can be no cycling in the algorithm. If the solution does not change one full iteration, then the collection of Kuhn-Tucker conditions for all the individual subproblems is identical to the Kuhn-Tucker condition given above for the path formulation. Hence the solution is optimal. □

IV. Cut formulation

The path formulation is easier to motivate as noted in the last section. In this section we present an equivalent cut formulation that leads to a distributed solution, important for large networks.

In this formulation, instead of considering the resource allocation problem one path at a time, we look at different links in the network and determine how much resources can be allocated on that link. An upper bound on the amount of resources on certain sets of links can be determined as follows: for each link l, the sum of the allocations for all the links in $\bar{L}(l)$ has to be less than the sum of the budgets in that set $\bar{L}(l)$. One can write such a constraint for each cut in the tree. This formulation will be termed the cut formulation. It can be shown that the path formulation and the cut formulation are equivalent. Let μ_l represent the net amount of resource that is allocated to link l. Then the allocation problem is to find $\mu_l, l \in E$, in order to:

$$\min \sum_{l \in E} f_l(\mu_l) \tag{4}$$

subject to

$$\begin{aligned} \sum_{j \in \bar{L}(l)} \mu_j &\leq \sum_{j \in \bar{L}(l)} B_l, \quad \forall l \in E & (5) \\ \mu_l &\geq 0, \quad \forall l \in E & (6) \end{aligned}$$

We now present two algorithms to solve the optimization problem. The first one is a simple (centralized) greedy algorithm that terminates in a finite number of steps, and the second one is a distributed algorithm derived from the dual problem.

Before proceeding to the its solution, we first state formally without proof the equivalence of the two formulations.

Theorem 2: The path formulation and the cut formulation are equivalent in the following sense. If $(\sigma^*_{kl}, k, l \in E)$ is a solution to the path formulation, then $(\mu^*_l, l \in E)$ defined by $\mu^*_l := \sum_{k \in \bar{L}(l)} \sigma^*_{kl}$ is a solution to the cut formulation. Conversely, if $(\mu^*_l, l \in E)$ is a solution to the cut formulation, then one can construct a solution $(\sigma^*_{kl}, k, l \in E)$ to the path formulation that satisfies $\mu^*_l = \sum_{k \in \bar{L}(l)} \sigma^*_{kl}$.

A. Centralized algorithm

Note that there is one constraint corresponding to each link and the sum on both sides of the inequality in the constraints is over the set $\bar{L}(l)$. Since the problem is defined on a directed tree, if $L(l_1) \cap L(l_2) \neq \emptyset$ then $L(l_1) \subseteq L(l_2)$ or $L(l_2) \subseteq L(l_1)$. Therefore these sets form a laminar family [4]. As outlined in [4], a modified greedy algorithm can be used to optimize any separable convex objective function on a polyhedron described by a laminar family. In the description of the greedy algorithm $\mathcal{B}$ below, we assume that $f'_l(0) = c, \quad \forall\ l$. It is easy to make modifications to the algorithm when this condition does not hold. Let f'^{-1}_l denote the inverse of the marginal cost function f'_l.

Algorithm $\mathcal{B}$

```
1. INITIALIZATION
   (a) $FI = \emptyset, \quad FR = E, \quad \mu^*_l = 0, \ \forall l \in E.$

2. MINIMUM RATIO TEST
   (a) For each $l \in E$:
   • If $T(l) \cap FR \neq \emptyset$ then determine $\lambda(l)$ such that
         $\sum_{j \in T(l) \cap FR} f'^{-1}_j(\lambda(l)) = \sum_{j \in T(l) \cap FR} B_j - \sum_{j \in T(l) \cap FI} \mu^*_j$
   • If $T(l) \cap FR = \emptyset$ then set $\lambda(l) = \infty$.
   (b) Let $\lambda = \min\{\lambda(l)\}$ and $L = \arg\min\{\lambda(l)\}$.

3. UPDATE STEP
   (a) $\mu^*_j = f'^{-1}_j(\Lambda), \ \forall j \in T(L) \cap FR,$
   $FI = FI \cup T(L), \ FR = FR \setminus T(L).$
   (b) If $FR = \emptyset$ then stop else go to Step 2.
```

Theorem 3: Algorithm $\mathcal{B}$ solves the cut formulation.

Proof:
(Sketch) The sets for which there are constraints in the cut formulation form a laminar family. From [4] it is known that a polyhedron described by a laminar family can be extended to a submodular polyhedron and therefore a greedy algorithm can be used to solve separable convex minimization problems on this polyhedron. □

B. Decentralized algorithm

The dual of problem (4–6) is

$$\max_{p\geq 0} \quad D(p) = \sum_l D_l(p) - \sum_l p_l \sum_{k\in \bar{L}(l)} B_l \quad (7)$$

where

$$D_l(p) = \min_{\mu_l\geq 0} f_l(\mu_l) + \mu_l p^l \quad (8)$$

$$p^l = \sum_{k\in P(l)} p_k \quad (9)$$

Consider the iterative solution of the dual problem (7) using the gradient projection algorithm:

$$p_l(t+1) = [p_l(t) - \gamma \frac{\partial D}{\partial p_l}(p(t))]^+ \quad (10)$$

Here $\gamma > 0$ is a step size, and $(z)^+ = \max\{z, 0\}$. Let $\mu_l(p)$ be the unique maximizer in (8). Then (10) is

$$p_l(t+1) = [p_l(t) - \gamma \sum_{k\in \bar{L}(l)} (\mu_l(p(t)) - B_l)]^+ \quad (11)$$

By duality theory there exists a $p^* \geq 0$ such that the allocation vector $\mu(p^*)$ that minimizes $D_l(p^*), l \in E$, in (8) is indeed primal optimal. The following algorithm $\mathcal{C}$ uses the links as processors in a distributed computation system to solve the dual problem by iterating on (11) and (8–9).

Algorithm $\mathcal{C}$

1. INITIALIZATION

$$t = 0,\ p_l(t) = 0 \textit{ for all } l \in E.$$

2. BANDWIDTH UPDATE

$$p^l(t) = \sum_{k\in P(l)} p_k(t)$$

$$\mu_l(t) = f_l'^{-1}(p^l(t))$$

3. DUAL VARIABLE UPDATE

$$\Delta_l(t) = \sum_{k\in \bar{L}(l)} (\mu_l(t) - B_l)$$

$$p_l(t+1) = [p_l(t) - \gamma\Delta_l(t)]^+$$

4. Increment t and go to Step 2.

Algorithm $\mathcal{C}$ converges under the following assumptions on the cost functions f_l:
A1: The cost functions f_l are strictly convex decreasing and twice differentiable on $[0, B]$, where $B := \sum_l B_l$.
A2: $0 < 1/\bar{\varphi} \leq -f_l''(\mu_l)$ for all $\mu_l \in [0, B]$, for all l.

Let $W = \sum_l |\bar{L}(l)|$ where $|A|$ is the number of elements of set A.

Theorem 4: Suppose A1–A3 hold and the step size γ satisfies $0 < \gamma < 1/\bar{\varphi}W$. Then every limit point (μ^*, p^*) of the sequence $(\{\mu(t), p(t))\}$ generated by algorithm $\mathcal{C}$ is primal–dual optimal.

Proof:
(Sketch) Focus on solution of the dual problem. The dual objective function is clearly lower bounded by the primal objective value. In can be shown under assumptions A1 and A2 that it is also Lipschitz continuous. Then any limit point p^* of the sequence $\{p(t)\}$ generated by the gradient projection algorithm for the dual problem is dual optimal; see [1, pp.214]. Then $x^* = x(p^*)$ is primal optimal since $x(p)$ is continuous in p. □

Algorithm $\mathcal{C}$ can be implemented by periodically passing signalling messages up and down the multicast tree, as follows (the description is for synchronous algorithm but it can be extended to an asynchronous version in a straightforward manner). A signalling message contains two fields. The first field π collects p^l for each link l to adjust its allocation μ_l, going downward from the source. We will call the dual variable p_l computed at link l according to (11) a 'price'. The second field Δ collects the total slack Δ_l in tree $\bar{L}(l)$ for link l to adjust its price p_l, going upward towards the source.

Implementation

At times $t = 1, 3, 5, \ldots,$

1. A message is originated at source and broadcast towards all leaves with the field π set to zero.
2. Each link l, on receipt of a forward message from its parent:
   (a) Add its current price $p_l(t)$ to π: $\pi \leftarrow \pi + p_l(t)$.
   (b) Forward the message with new π to all its children.
   (c) Update its allocation: $\mu_l(t+1) = f_l'^{-1}(\pi)$.

At times $t = 2, 4, 6, \ldots$,
1. A message is originated at every leaf link l towards the source with the field Δ set to link l's slack: $\Delta \leftarrow \mu_l(t) - B_l$.
2. Each link l, on receipt of backward messages from *all* its children:
   (a) Generate a backward message to its parent with the field Δ set to:

$$\Delta \leftarrow (\mu_l(t) - B_l) + \sum_k \Delta_k$$

where Δ_k is the value of the field Δ in the backward message from link l's child link k.
   (b) Update its price: $p_l(t+1) = [p_l(t) + \gamma\Delta]^+$.

V. Conclusion

In this paper we have presented a model for optimal resource allocation in a multicast tree where the objective is to minimize total link costs subject to budget constraints at each receiver. We have presented two formulations and shown that they are equivalent. The first formulation is easier to motivate and admits a centralized solution. The advantage of the second formulation, however, is not only that it has a centralized finite solution, but also a distributed and decentralized solution that is more scalable.

References

[1] Dimitri P. Bertsekas and John N. Tsitsiklis. *Parallel and distributed computation.* Prentice-Hall, 1989.

[2] Christophe Diot, Walid Dabbous and Jon Crowcroft. Multipoint communication: A survey of protocols, functions, and Mechanisms. *IEEE Journal on Selected Areas in Communications*, 15(3):277–290, April 1997.

[3] Victor Firoiu and Don Towsley. Call admission and resource reservation for multicast sessions. *IEEE Proceedings of INFOCOM'96*, April 1996

[4] S.Fujishige. *Submodular functions and optimization.* North-Holland, 1991.

[5] C. Topolcic. *Experimental Internet Stream Protocol: Version 2 (ST-II).* Internet RFC 1190, October 1990.

[6] L. Zhang, S. E. Deering, D. Estrin, S. Shenker and D. Zappala. RSVP: A new Resource reSerVation Protocol. *IEEE Network*, 7(5):8–18, September 1993.

[7] P.H. Zipkin. Simple Ranking Method for the Allocation of one Resource *Management Science*, 26(5):34–43, 1980.

0-7803-5420-6/99/$10.00 (c) 1999 IEEE