Channelization Problem In Large Scale Data Dissemination

Zihui Ge\(^1\)  Micah Adler\(^1\)  James F. Kurose\(^1\)  Don Towsley\(^1\)  Stephen Zabele\(^2\)

\(^1\)Department of Computer Science  University of Massachusetts  gszabele@tasc.com
\(^2\)Litton-TASC Inc.
{micah,gezihui,kurose,towsley}@cs.umass.edu

June 30, 2001

Abstract

In many large scale data dissemination systems, a large number of information flows must be delivered to a large number of information receivers. However, because of differences in interests among receivers, not all receivers are interested in all of the information flows. Multicasting provides the opportunity to deliver a subset of the information flows to a subset of the receivers. With a limited number of multicast groups available, the channelization problem is to find an optimal mapping of information flows to a fixed number of multicast groups, and a subscription mapping of receivers to multicast groups so as to minimize a function of the total bandwidth consumed and the amount of unwanted information received by receivers. In this paper, we formally define two versions of the channelization problem and the subscription problem (a subcomponent of the channelization problem). We analyze the complexity of each version of the channelization problem and show they are both NP-complete. We also find that the subscription problem is NP-complete when one flow could be assigned to multiple multicast groups. We also study and compare different approximation algorithms to solve the channelization problem, finding that one particular heuristic, flow-based-merge, finds good solutions over a range of problem configurations.

1 Introduction:

Large scale data dissemination applications, such as distributed interactive simulation (DIS) [2, 7], multi-player games [6], publish-subscribe systems [1] and distributed event notification systems [3], are characterized by a large number of information sources and a large number of information consumers. However, the nature of these applications is such that individual users are not interested in all of the published content [8]. Thus, flooding is not attractive in that a significant portion of network bandwidth and end host resources will be wasted by delivering and processing messages in which the receiver is not interested.

An alternative to flooding is multicasting, in which one or more information flows are sent into a so-called multicast group. Multiple multicast groups may be used. A receiver subscribes to one or more multicast groups, receiving only the information that was transmitted over the multicast groups to which it has subscribed. Multicast groups, however, require resources (e.g., router state) and management overhead (e.g., to set up and maintain the
multicast routes); it is thus often not feasible or desirable to allocate a separate multicast group to each information flow. With a limited number of multicast groups available, the *channelization problem* is to find an optimal mapping of information flows to a fixed number of multicast groups and a subscription mapping of receivers to multicast groups so as to minimize a cost function involving the total bandwidth consumed and the amount of unwanted information received by receivers. Previous studies [8] have conjectured that the channelization problem is a computationally hard problem. However, no formal analysis has been presented.

In this paper, we formally define two versions of the channelization problem. In the first version of this problem, a given flow can be assigned to multiple multicast groups. In the second version of this problem, a given flow can be assigned to only one multicast group. We also consider a subcomponent of the channelization problem, called the “subscription problem,” in which the information-flow-to-multicast-group mapping is predetermined, and only the receiver-to-multicast-group subscription question is considered. We analyze the complexity of each version of the channelization problem and show that although they are different in total number of assignment combinations, they are both NP-complete. We also find that the subscription problem is NP-complete when one flow could be assigned to multiple multicast groups, while its complexity is greatly reduced and the subscription problem becomes solvable in linear time when one flow is restricted to belong to only one multicast group. Finally, we study and compare different approximation algorithms to solve the channelization problem and evaluate them over a randomly generated set of problem configurations. We find that one particular heuristic, flow-based-merge, finds good solutions over a range of problem configurations.

The remainder of this paper is organized as follows. Section 2 introduces the channelization and subscription problems with and without the constraint that one flow be assigned to only one multicast group and presents a model that characterizes multicast data dissemination systems. Section 3 provides formal definitions of each of the problems and analyzes their complexity. Section 4 presents several heuristic approaches for obtaining an approximate solution for the channelization problem. Section 5 presents simulation results comparing different heuristic algorithms in different problem settings. Finally, Section 6 concludes our study.

## 2 Problem Description and Model

### 2.1 Problem Description

Since multicast groups are a limited resource, a challenging and important assignment problem arises when mapping information flows to multicast groups and mapping users to the multicast groups containing the flows of interest to a user. The *Channelization Problem*, as shown in Figure 1, is this two-phase mapping problem, which carries with it several requirements that stem from robustness considerations [10]. Specifically:

- **No false exclusion** – The mapping must be such that all data needed by a user is mapped to one or more multicast groups to which the user is subscribed.

- **Minimum false inclusion** – The mapping should be such that the amount of unneeded data received by users by subscribing to various multicast groups carrying the needed data is minimized.
A subcomponent of the channelization problem is the subscription problem, where the mapping of fbws to multicast groups is fixed, and the decision needs to be made as to which groups each user should subscribe to such that the requirements of no false exclusion and minimum false inclusion are satisfied.

2.2 Model and Notations

In this section, we present a model which characterizes the channelization problem and subscription problem. A data dissemination system consists of:

- A set of fbws of interest(source): \( S, |S| = N \). Flow \( i \) has rate \( \lambda_i, i \in S \).
- A set of multicast groups: \( G, |G| = K \).
- A set of independent users: \( U, |U| = M \).

Each user is only interested in some of the fbws, and different users can share common interests. We define the interests matrix as

\[
W = (w_{ji}) \quad \text{where } w_{ji} = \begin{cases} 1 & \text{user } j \text{ is interested in information fbw } i \\
0 & \text{otherwise} \end{cases} \quad i \in S, j \in U.
\]

Information (fbws) is distributed through multicast groups, with each fbw being assigned to one or more multicast groups. We define the fbw-to-group mapping matrix as

\[
X = (x_{im}) \quad \text{where } x_{im} = \begin{cases} 1 & \text{fbw } i \text{ is assigned to multicast group } m \\
0 & \text{otherwise} \end{cases} \quad i \in S, m \in G.
\]

Users receive information by subscribing to multicast groups and each user can subscribe to multiple multicast groups. We define the subscription matrix as

\[
Y = (y_{jm}) \quad \text{where } y_{jm} = \begin{cases} 1 & \text{user } j \text{ subscribes to multicast group } m \\
0 & \text{otherwise} \end{cases} \quad j \in U, m \in G.
\]
Upon joining a multicast group, the user will receive all fbws assigned to that group. Some of these fbws will be of interest to the user, while others may not.

Having defined the notation for mapping fbws, let us next consider the cost of a given mapping. Each fbw \(i\), on being assigned to one multicast group, increases the system cost by \(w_2 \lambda_i\). Each user \(j\), on receiving one copy of fbw \(i\), increases the system cost by \(w_1 c_{ji} \lambda_i\), where \(c_{ji}\) are topology dependent coefficients. We define the overall cost function associate with a mapping to be

\[
C(X, Y) = w_1 \sum_{i \in S} \sum_{j \in U} \sum_{m \in G} x_{im} y_{jm} c_{ji} \lambda_i + w_2 \sum_{m \in G} \sum_{i \in S} x_{im} \lambda_i
\]

where \(w_1 + w_2 = 1\). Here \(w_1\) and \(w_2\) are introduced to provide relative weights to the two costs.

The no false exclusion property corresponds to \(\sum_{m \in G} x_{im} \geq w_{ji}, \forall i \in S, j \in U\), and the minimum inclusion requirement as minimizing \(C(X, Y)\).

In Section 3, we will present a formal definition and complexity analysis for both the channelization problem and the subscription problem based on this model.

### 2.3 Unconstrained and Constrained Channelization and Subscription Problem

The mapping of fbws to multicast groups falls into two categories. The unconstrained version allows each fbw to be assigned to multiple multicast groups and the constrained version requires that the multicast groups form a partition of all the information fbws, i.e., requires that each fbw be assigned to only one multicast group. The choice of using constrained or unconstrained mappings is determined by the nature of problem or by considering a design tradeoff involving issues of system complexity, efficiency, flexibility, requirements of data consistency, etc.

For example, using the constrained version of fbw-to-group mapping instead of the unconstrained version may sacrifice system efficiency. This would happen if, in the unconstrained version of the channelization problem, the optimal configuration required that one fbw be assigned to multiple multicast groups.

<table>
<thead>
<tr>
<th></th>
<th>(S_1)</th>
<th>(S_2)</th>
<th>(S_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\lambda_1 = 1)</td>
<td>(\lambda_2 = 100)</td>
<td>(\lambda_3 = 100)</td>
<td></td>
</tr>
<tr>
<td>(U_1)</td>
<td>(\checkmark)</td>
<td>(\checkmark)</td>
<td>-</td>
</tr>
<tr>
<td>(U_2)</td>
<td>(\checkmark)</td>
<td>-</td>
<td>(\checkmark)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>(G_1)</th>
<th>(G_2)</th>
<th>(G_1)</th>
<th>(G_2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(S_1, S_2)</td>
<td>(S_1, S_3)</td>
<td>(S_1, S_2)</td>
<td>(S_1, S_3)</td>
<td></td>
</tr>
<tr>
<td>(U_1)</td>
<td>101</td>
<td>-</td>
<td>(U_1)</td>
<td>101</td>
</tr>
<tr>
<td>(U_2)</td>
<td>-</td>
<td>101</td>
<td>(U_2)</td>
<td>101</td>
</tr>
</tbody>
</table>

Table 1: example that constraint leads to suboptimal configuration

To illustrate this issue, consider the example presented in Table 1. Three fbws \(S_1, S_2, S_3\) are disseminated to two users \(U_1, U_2\) and only two multicast groups \(G_1\) and \(G_2\) are available. Flow \(S_1\) transmits at rate \(\lambda_1 = 1\),
while \( S_2 \) and \( S_3 \) transmit at rates \( \lambda_2 = \lambda_3 = 100 \). User \( U_1 \) is interested in fbws \( S_1 \) and \( S_2 \). User \( U_2 \) is interested in fbws \( S_1 \) and \( S_3 \). If one fbw is allowed to belong to multiple multicast groups, the optimal configuration is \( G_1 = \{ S_1, S_2 \} \), \( G_2 = \{ S_1, S_3 \} \). Thus, \( U_1 \) will only need to subscribe to \( G_1 \) and \( U_2 \) will only need to subscribe to \( G_2 \). Each user receives exactly the data it wants and no more. When each fbw can be assigned to only one group, the best configuration is \( G_1 = \{ S_1, S_2 \} \), \( G_2 = \{ S_3 \} \). Thus \( U_1 \) will subscribe to \( G_1 \) and get only the fbws in which it is interested. However \( U_2 \) must subscribe to both \( G_2 \) and \( G_1 \), and receives unwanted traffic at rate 100.

<table>
<thead>
<tr>
<th>Unconstrained group mapping</th>
<th>Channelization Problem</th>
<th>Subscription Problem</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NP-complete</td>
<td>NP-complete</td>
</tr>
<tr>
<td>Constrained group mapping</td>
<td>NP-complete</td>
<td>Linear Time</td>
</tr>
</tbody>
</table>

Table 2: Complexity structure of unconstrained and constrained Channelization problem and Subscription problem

Given the distinctions between these two kinds of fbw-to-multicast-group mappings — unconstrained and constrained, we studied the complexity of two different versions of the channelization problem and the subscription problem individually. Table 2 presents the results of complexity analyses of the problems. In the following section, we will present in detail the formal definitions and proofs of complexity for the unconstrained channelization problem [Section 3.1], the constrained channelization problem [Section 3.2], the unconstrained subscription problem [Section 3.3], and the unconstrained subscription problem [Section 3.4].

3 Complexity Study

3.1 Unconstrained Channelization Problem

In the framework described Section 2.2, given a set of fbws \( S \), a set of multicast groups \( G \), a set of users \( U \), and an interests matrix \( W \), the Unconstrained Channelization Problem is to obtain values for \( X \) and \( Y \) that minimize overall system cost while ensuring that each user receives all the information in which it is interested when \( K < N \).

More formally, the unconstrained channelization problem is to minimize \( C(X, Y) \) subject to \( \sum_{m \in G} x_{im} y_{jm} \geq w_{ji} \), for all \( i \in S, j \in U \), where

\[
C(X, Y) = w_1 \sum_{i \in S} \sum_{j \in U} \sum_{m \in G} x_{im} y_{jm} c_{ji} \lambda_i + w_2 \sum_{m \in G} \sum_{i \in S} x_{im} \lambda_i
\]

The unconstrained channelization problem has a very large solution space. The number of ways to assign each of \( N \) fbws to one or more of \( K \) multicast groups exponentially increases with the size of the problem:

\[
\text{No. of different mappings} = \frac{1}{K!} (2^K - 1)^N
\]

Thus, determining the proper mappings for the unconstrained channelization problem is potentially computationally quite expensive. In fact, we can show that the unconstrained channelization problem is NP-Complete. Before proceeding with the proof, we first introduce a well-studied NP-Complete problem, SET BASIS[5]:

5
INSTANCE: Collection $C$ of subsets of a finite set $S$ and a positive integer $K \leq |C|$.

QUESTION: Is there a collection $B$ of subsets of $S$ with $|B| = K$ such that, for each $c \in C$, there is a subcollection of $B$ whose union is exactly $c$?

By transforming from the VERTEX COVER problem, Stockmeyer showed the SET BASIS problem to be NP-complete [9].

**Theorem 1** The Unconstrained Channelization Problem is NP-Complete.

**Proof:** It is easy to show that Unconstrained Channelization Problem $\in$ NP. Given values for $X, Y$, validating that $X, Y$ satisfies the constraints and computing $C(X, Y)$ can be done in polynomial time.

To show that it is NP-hard, we prove that SET BASIS is polynomially reducible to the Unconstrained Channelization problem. i.e. SET BASIS $\leq P$ Unconstrained Channelization.

Given an instance of SET BASIS with set $S$ and set $C = \{C_1, C_2, ..., C_M\}$, where $C_j \subseteq S$ for $1 \leq j \leq M$, and positive integer $K$, we can formulate an instance of the unconstrained channelization problem:

- Let the set of fbws be $S$.
- Let the flow rate for $i \in S$ be $\lambda_i = 1$.
- Let the number of multicast groups be $K$.
- Let the number of users be $M$.
- Let the interests matrix be $W = (w_{ji})$ where $w_{ji} = \begin{cases} 1 & \text{if } i \in C_j, \\ 0 & \text{if } i \notin C_j, \end{cases}$, $i \in S$.
- Let the cost factor be $c_{ji} = \begin{cases} 1 & \text{if } i \notin C_j, \\ 0 & \text{if } i \in C_j, \end{cases}$, $i \in S$. $w_1 = 1$, $w_2 = 0$.

Thus, the system cost only depends on the sum of the number of excessive fbws received by users. By solving the unconstrained channelization problem, we can get an optimal configuration $X$ and $Y$, that minimizes the system cost. If the cost is 0, then we answer "YES" for the SET BASIS problem and the collection of each row of $X$ is the desired subcollection $B$. Otherwise we answer "NO" for the SET BASIS problem.

Since the reduction is in polynomial time and since the SET BASIS problem has been shown to be NP-Complete, the channelization problem is NP-hard. This completes the proof.

**3.2 Constrained Channelization Problem**

Since, in the constrained fbw-to-group mapping scenario, one fbw is allowed to be assigned to only one multicast group, each user has to subscribe to the multicast groups that contain the fbws that the user is interested in. Thus the subscription matrix $Y$ is a function of the fbw-to-group mapping matrix $X$ and the interests matrix $W$,

$$ Y(X, W) = (y_{jm}) \quad \text{where } y_{jm} = \begin{cases} 1 & \text{if } w_{ji} = 1, x_{im} = 1, \\ 0 & \text{otherwise} \end{cases} \quad i \in S, m \in G, j \in U $$

Also, since each fbw is assigned to exactly one multicast group, we are no longer interested in the cost contribution of the fbw-to-multicast mapping, which is a constant given $S, G$ and $W$. We thus define the cost function for the
constrained channelization problem as

\[ C(X) = \sum_{i \in S} \sum_{j \in U} \sum_{m \in G} x_{im} y_{jm} c_{ji} \lambda_i. \]

Given a set of fbws \( S \), a set of multicast groups \( G \), a set of users \( U \), and an interests matrix \( W \), the **Constrained Channelization Problem** is to find \( X \) that minimizes system cost while ensuring that each fbw is assigned to only one multicast group and each user receives all interested information when \( K < N \), i.e., the constrained channelization problem is to minimize \( C(X) \) subject to \( \sum_{m \in G} x_{im} = 1 \) and \( \sum_{m \in G} x_{im} y_{jm} \geq w_{ji} \), for all \( i \in S, j \in U \).

A preliminary analysis in [7][10] shows that the brute force approach to solving the constrained channelization problem is intractable for most problems of interest: for example, the number of ways to assign each of \( N \) fbws to one and only one of \( K \) multicast groups is given by the Stirling number of the second kind:

\[ \text{No. of different settings} = S_N^{(K)} = \frac{1}{K!} \sum_{j=0}^{K} (-1)^{K-j} (\frac{K}{j}) j^N \]

Where \( N = 25, K = 10, S_N^{(K)} = 1, 203, 163, 392, 175, 387, 500 \). This suggests that the Constrained version of Channelization Problem may not be easy to solve. In fact, by reducing from the MINIMUM SUM OF SQUARES problem, we are able to show that the Constrained Channelization Problem is still \( \text{NP} \)-complete. The MINIMUM SUM OF SQUARES is described below[5]:

- **INSTANCE:** Finite set \( A \), a function \( s : A \rightarrow \mathbb{Z}^+ \) and positive integers \( K \leq |A| \) and \( J \).
- **QUESTION:** Can \( A \) be partitioned into \( K \) disjoint sets \( A_1, A_2, ..., A_K \) such that

\[ \sum_{i=1}^{K} \left( \sum_{a \in A_i} s(a) \right)^2 \leq J? \]

Based on the \( \text{NP} \)-completeness of MINIMUM SUM OF SQUARES [4], we have the following theorem:

**Theorem 2** The Constrained Channelization Problem is \( \text{NP} \)-Complete.

**Proof:** The Constrained Channelization Problem \( \in \text{NP} \), since, given \( X \), validating \( X \) and computing \( C(X) \) can be done in polynomial time.

To show that it is \( \text{NP} \)-hard, we prove that MINIMUM SUM OF SQUARES is polynomially reducible to Constrained Channelization problem. i.e. MINIMUM SUM OF SQUARES \( \leq \text{P} \) Constrained Channelization problem.

Given an instance of MINIMUM SUM OF SQUARES with \( A = \{a_1, a_2, ..., a_N \}, s(a_i) \in \mathbb{Z}^+ \) for \( 1 \leq i \leq N \), positive integers \( K \leq |A| \) and \( J \), we can formulate an instance of the Constrained Channelization problem:

- Let \( S = A \) and let the fbw rate be \( \lambda_i = s(a_i) \), where \( 1 \leq i \leq N \).
- Let \( U = \{(a, j) | a \in A, j = 1, ..., s(a)\} \) be the set of users.
• Let each user \((a, j) \in U\) be interested only in fbw \(a\), where \(a \in A\). Thus

\[
W = (w_{(a,j)i}) \quad \text{where } w_{(a,j)i} = \begin{cases} 
1 & a = a_i, 1 \leq j \leq s(a) \\
0 & \text{otherwise}
\end{cases}
\]

• Let all \(c_{(a_i,j)i} = 1\) for \(1 \leq j \leq s(a_i)\) and \(1 \leq i \leq N\).

• Let the multicast groups available be \(G = \{1, 2, \ldots, K\}\), where \(|G| = K\).

Because of the constraint that one fbw can only be assigned to one multicast group, i.e., \(\sum_{m \in G} x_{im} = 1\), we have

\[x_{im} = 1 \text{ and } w_{(a_i,j)i} = 1 \Leftrightarrow y_{(a_i,j)m} = 1\]

And the system cost is

\[
C(X) = \sum_{i=1}^{N} \sum_{t=1}^{N} \sum_{j=1}^{K} \sum_{m=1}^{K} x_{im} y_{(a_t,j)m} c_{(a_t,j)i} \lambda_i = \sum_{m=1}^{K} \left[ \sum_{i=1}^{N} x_{im} \lambda_i \right] \left[ \sum_{t=1}^{N} \sum_{j=1}^{K} y_{(a_t,j)m} \right]
\]

\[
= \sum_{m=1}^{K} \left[ \sum_{i=1}^{N} x_{im} s(a_i) \right] \left[ \sum_{t=1}^{N} s(a_t) x_{im} \right] = \sum_{m=1}^{K} \left[ \sum_{i=1}^{N} x_{im} s(a_i) \right]^2
\]

Thus, by solving the Constrained Channelization Problem, we can get a fbw assignment, \(X\), that minimizes \(C(X)\), which is essentially the sum of squares for the current set partition \(X\). If the cost is less than or equal to \(J\), the answer is ”YES” to the MINIMUM SUM OF SQUARES problem. Otherwise, the answer is ”NO”.

Since the reduction is in polynomial time and since the MINIMUM SUM OF SQUARES problem is known to be NP-Complete, the Constrained Channelization Problem is NP-hard. This completes the proof.

From Theorem 2, we know that introducing the constraint that one fbw can be assigned to only one multicast group will not change the fact that the channelization problem is NP-hard. However, it helps to reduce the complexity of the subscription problem, as we show below.

### 3.3 Unconstrained Subscription Problem

In the previous section we considered the channelization problem. In this section, we consider a subproblem of the channelization problem known as the subscription problem. In the subscription problem, the mapping of fbws to multicast group is fixed. The subscription problem is to determine, for each user, the set of multicast groups which it should subscribe to in order to receive all needed information, and to do so at minimal cost.

Instead of considering a set of users \(U\), we can focus on only one user who is interested in receiving some of the fbws. We define the interests vector as

\[
W = (w_i) \quad \text{where } w_i = \begin{cases} 
1 & \text{user is interested in information fbw } i \\
0 & \text{otherwise}
\end{cases} \quad i \in S.
\]
Define the subscription vector as

\[ Y = (y_m) \quad \text{where} \quad y_m = \begin{cases} 1 & \text{user subscribes to multicast group } m \\ 0 & \text{otherwise} \end{cases} \quad m \in G. \]

Also, in the subscription problem, we are only interested in the cost associated with mapping multicast group to users. We define the cost function as

\[ C(Y) = \sum_{i \in S} \sum_{m \in G} \lambda_p x_{im} y_m. \]

The Unconstrained Subscription Problem is, given a set of fbws \( S \), a set of multicast groups \( G \), a fbw-to-group mapping matrix \( X \) and an interests vector \( W \), to find a subscription vector \( Y \) so that the user receives all the information fbws in which it is interested, while minimizing cost. More formally, the unconstrained subscription problem is to minimize \( C(Y) \) subject to \( \sum_{m \in G} x_{im} y_m \geq w_i \), for all \( i \in S \).

When each fbw is assigned to one or more multicast groups, this unconstrained subscription problem, like the channelization problem before it, is very hard to solve, as indicated by the following theorem.

**Theorem 3** The Unconstrained Subscription Problem is NP-Complete.

By reducing the SET COVER problem to the Unconstrained Subscription problem, we can show that the Unconstrained Subscription problem is NP-hard. The SET COVER problem is described below [5]:

- **INSTANCE:** Collection \( C \) of subsets of a finite set \( R \) and a positive integer \( J \).
- **QUESTION:** Is there a subset \( B \) of \( C \) such that \( |B| \leq J \) and \( \bigcup_{b \in B} b = R \).

**Proof:** The Unconstrained Subscription Problem \( \in \) NP, since given \( Y \), validating \( Y \) and computing \( C(Y) \) can be done in polynomial time.

To show that it is NP-hard, we prove that SET COVER is polynomially reducible to the Unconstrained Subscription problem, i.e. SET COVER \( \leq_P \) Unconstrained Subscription problem.

Given an instance of the SET COVER problem with \( C, R \) and \( J \) where \( \bigcup_{i \in C} C_i = R \), we can formulate an instance of the unconstrained subscription problem:

- For each \( C_i \in C \), create a fbw \( S_i \), where \( 1 \leq i \leq K \); let the fbw rate be \( \lambda_i = 1 \).
- For each \( R_j \in R \), create a fbw \( S_{j+K} \), where \( 1 \leq j \leq |R| \); let the fbw rate be \( \lambda_{j+K} = 0 \).

Then the set of fbws is \( S = \{S_1, S_2, ..., S_K, S_{K+1}, ..., S_{K+|R|}\} \)

- For each \( i \in C \), create a multicast group and set the fbw-to-group mapping matrix as follows

\[ X = (x_{im}) \quad \text{where} \quad x_{im} = \begin{cases} 1 & i = m \text{ or } R_i \in C_m \\ 0 & \text{otherwise} \end{cases} \]

- Let the user be interested in fbws \( S_{K+1}, S_{K+2}, ..., S_{K+|R|} \):

\[ W = (w_i) \quad \text{where} \quad w_i = \begin{cases} 1 & i > K \\ 0 & i \leq K \end{cases} \]
The cost for subscription $Y$ is

$$C(Y) = \sum_{i=1}^{K+|R|} \sum_{m=1}^{K} \lambda_i x_{im} y_m = \sum_{i=1}^{K} \sum_{m=1}^{K} x_{im} y_m = \sum_{m=1}^{K} y_i$$

Since $\sum_{m=1}^{K} x_{im} y_m \geq w_i$ for all $1 \leq i \leq K + |R|$, $\sum_{m=1}^{K} x_{im} y_m \leq 1$ for all $|K| \leq i \leq K + |R|$ i.e., the multicast groups that the user joins covers all the fbws from $S_{K+1}, \ldots, S_{K+|R|}$, the corresponding subsets will cover the original set $R$. Thus, by solving the unconstrained subscription problem, we obtain a subscription—$Y$, that minimizes the system cost. If the cost is less than or equal to $J$, we answer “YES” for the SET COVER problem, and the subset $B = \{C_i | y_i = 1\}$ Otherwise, we answer “NO”.

Since the reduction is in polynomial time, and the SET COVER problem is known NP-Complete, the Unconstrained Subscription problem is NP-hard. This completes the proof.

### 3.4 Constrained Subscription Problem

The constrained subscription problem is different from the unconstrained subscription problem defined in Section 3.3 in that one fbw is allowed be assigned to only one multicast group, i.e., $\sum_{m \in G} x_{im} = 1$ for all $i \in S$. More formally, the Constrained Subscription Problem is, given a set of fbws $S$, a set of multicast groups $G$, a fbw-to-group mapping matrix $X$ and an interests vector $W$, find subscription vector $Y$ that minimizes $C(Y)$ subject to $\sum_{m \in G} x_{im} y_m \geq w_i$, $\sum_{m \in G} x_{im} = 1$, for all $i \in S$, where

$$C(Y) = \sum_{i \in S} \sum_{m \in G} \lambda_i x_{im} y_m.$$  

Because of the “no false exclusion” requirement, an efficient algorithm that assign $y_m = 1$ if and only if there exists a fbw $i$, $w_i = 1$ and $x_{im} = 1$ will achieve the minimum cost. Since for all $i \in S$, $\sum_{m \in G} x_{im} = 1$, assigning $y_m = \sum_{i \in S} x_{im} w_i$ will give the optimal subscription. This computation can be done in linear time ($O(|X|)$).

We have observed that adding the constraint that one fbw can be assigned to only one multicast group may not achieve the optimal configuration in some cases, and the channelization problem remains NP-hard under the constraint. However, we found that imposing this constraint can greatly reduce the complexity of finding a solution to the user subscription problem. In the next section, we will examine heuristics to find approximate solutions to the channelization problem with or without this constraint.

### 4 Channelization Heuristics

Since the brute force approach of exhaustive search is infeasible, and the NP-completeness of the channelization problem implies that any attempt to find the optimal solution will have exponential computational complexity, we
focus our attention on finding approximations for the channelization problem. Specifically, we investigate random assignment, two simple heuristics (to balance the multicast group size and to balance multicast group rate sum) and two greedy approaches (Flow Based Merge and User Based Merge). A short description of these heuristics is given below.

- **random assignment (RAN):** Randomly pick a flow. Uniformly assign the flow to one and only one of the $K$ multicast group until all flows have been assigned. For each user, use the algorithm in Section 3.4 to solve the constrained subscription problem. The run time for this algorithm is $O(KN)$.

- **random assignment with heuristic of balancing group size (RSE):** Randomly pick a flow. Assign the flow to the multicast group that currently contains the least number of flows. For each user, use the algorithm in Section 3.4 to solve the constrained subscription problem. The run time for this algorithm is $O(KN)$.

- **random assignment with heuristic of balancing group rate sum (RRE):** Randomly pick a flow. Assign the flow to the multicast group with the least total flow rate. For each user, use the algorithm in Section 3.4 to solve the constrained subscription problem. The run time for this algorithm is $O(KN)$.

- **flow based merge (FBM):** Start with $N$ multicast groups. Assign each flow to a different multicast group. This gives the perfect multicast group assignment since each user receives and only receives the flow(s) that it wants. By merging two multicast groups into one multicast group, we reduce the total number of multicast group needed by one. However, some receivers will have to receive more than they want — this happens when a receiver is interested in flow(s) that is contained in one of the multicast group but not the other. Thus we can define a pairwise merging cost as the sum of the excessive flow received by each user over all users when merging a pair of multicast groups. Having defined this, for each step, we merge the pair of multicast groups that minimizes the pairwise merging cost and repeat this $N - K$ times until the total number of groups is within the limit. To implement the flow based merge algorithm, it is necessary to maintain a table of pairwise merging costs. $O(M)$ operations are needed to compute the merge cost of each pair of multicast groups. Initially, there are $N \times N$ table entries, and, after each merge, $N - K$ entries need to be recomputed. Thus the run time for executing the flow based merge algorithm is $O(N^2M + (N - K)^2M)$, with $O(N^2)$ space requirement.

Table 4 shows an example where 4 flows are disseminated to 4 receivers through 2 multicast groups. Initially, 4 multicast groups are created for each flow. According to the pairwise merging costs, $G_2$ and $G_4$ are first merged into one group, and furthermore, $G_1$ and $G_3$ are merged together, resulting a total of 2 multicast groups used.

- **user based merge (UBM):** Start with $M$ multicast groups — create one multicast group for each user, which is equivalent to a unicast assignment. At each step, merge the pair of multicast groups that minimizes the pairwise merging cost. Repeat $M - K$ times. To implement the user based merge algorithm, it is necessary to maintain a pairwise merging cost table. $O(N)$ operations are required to compute the merge cost of each pair of multicast groups. Initially, there are $M \times M$ table entries, and, after each merge, $M - K$ entries need to be recomputed. Thus the run time for executing the user based merge algorithm is $O(M^2N + (M - K)^2N)$, with $O(M^2)$ space requirement.
At first glance, flow based merge and user based merge look very similar. However, there is a significant difference between them – flow based merge (FBM), as well as random assignment (RAN) and simple heuristics (RSE, RRE), implicitly place the constraint that no flow will be assigned to more than one multicast group, while user based merge (UBM) allows for the possibility that one flow will be assigned to multiple multicast groups.

In Section 5, we compare the cost associated with the solutions produced by these approximation algorithms.

5 Evaluation of the Approximation Algorithms

5.1 Simulation setting

To test the goodness of the approximation algorithms in Section 4, we implemented and tested them on randomly generated problem instances. The sets of fbws and users were generated with regard to the following considerations:

- high rate v.s. low rate — Each information fbw, once created, is assigned a transmission rate \( \lambda \), which takes one of two values \( \lambda_H \) or \( \lambda_L \), where \( \lambda_L < \lambda_H \). A fbw is assigned to rate \( \lambda_H \) with probability \( \alpha \) and rate \( \lambda_L \) with probability \( 1 - \alpha \).
- popular v.s. unpopular — Each information fbw is either popular or unpopular. A fbw is popular with probability \( \beta \) and unpopular with probability \( 1 - \beta \). The popularity of a fbw is uncorrelated of its transmission rate. A user is interested in a popular fbw with probability \( P_{pop} \) and an unpopular fbw with probability \( P_{unp} \).
- inactive fbws and users — After all fbws and all users are created and table of interests are initiated, inactive users (users that are not interested in any of the fbws) and inactive fbws (fbws that are not interested by any users) are eliminated from the test set.

For the sake of simplicity, we evaluate the cost function \( C(X, Y) \) with parameter setting \( w_1 = w_2 \) and \( C_{ji} = 1 \) for all \( j \in U, i \in S \). We can rewrite the cost function with fbw set \( S \), user set \( U \), and table of interests
\( W = (w_{ji}) \ (j \in U, i \in S) \) as follows:

\[
C(X,Y) = \sum_{i \in S} \sum_{j \in U} \sum_{m \in G} x_{im} y_{jm} \lambda_i + \sum_{m \in G} \sum_{i \in S} x_{im} \lambda_i
\]

where \( X = (x_{im}) \) is the information flow to multicast group matrix, \( Y = (y_{jm}) \) is the user subscription matrix and \( \sum_{m \in G} x_{im} y_{jm} \geq w_{ji} \). Also, we define \( C_\infty \) as the cost when there are an infinite number of multicast groups available. \( C_\infty \) can be evaluated from \( S, U, W \) directly by equation

\[
C_\infty = \sum_{j \in U} \sum_{i \in S} w_{ji} \lambda_i + \sum_{i \in S} \lambda_i
\]

and provides a lower bound for the cost function of any group assignment, \( C(X,Y) \geq C_\infty, \forall X,Y \).

### 5.2 Simulation results

#### 5.2.1 Comparison with exhaustive search

We begin by comparing the approximation solutions with the optimal solutions for small problem sizes. In Figure 2, we plot 50 experiments with problem parameters \( N = 9, M = 9, K = 3, \alpha = 0.1, \lambda_H = 10, \lambda_L = 1, \beta = 0.2, P_{pop} = 0.6, P_{unp} = 0.1 \). For each experiment, we compared the costs of the following algorithms:

- FBM, UBM, RAN, RSE, RRE as in Section 4.
- Exhaustive search (Optimal Configuration) with the constraint that one flow be assigned to only one multicast group (OCC)
- Exhaustive search (Optimal Configuration) of all possible settings (OCA)

![Figure 2: Cost Comparison on Different Algorithms](image-url)
The x-axis is the cost when there are infinite number of multicast group available and the y-axis values represent the cost of solutions found by performing the different algorithms described above. The line $y = x$ indicates the lower bound of cost. Figure 2 shows that neither the random approach (RAN) nor the simple heuristics (RSE, RRE) can find a close to optimal solution, while both fbw-based merge (FBM) and user-based merge (UBM) provide fairly good approximations. Indeed, in 29 out of the 50 (58%) experiments, FBM finds the same solution as OCC and in 46 (92%) cases, FBM finds a configuration with a cost that is within 5% of the optimal cost (with constraint) found by OCC. In 5 out of above 50 (10%) experiments, UMB finds the same solution as OCA and in 31 (62%) cases, UMB finds a configuration with a cost that is within 5% of the optimal cost found by OCA. As we predicted, adding the constraint that one flow can be assigned to only one multicast group can result in excluding the optimal solution of the unconstrained case. However, the performance degradation is acceptable for these problem instances. In 18 out of the 50 (36%) experiments, OCC finds the same solution as OCA and 42 out of above 50 (84%) cases, OCC finds a configuration with a cost that is within 5% of that found by OCA.

5.2.2 Effect of flow rate heterogeneity, traffic density and number of multicast group

In Figures 3 to 6, we consider the effects of the number of multicast groups in four different problem settings. Each point corresponds to the average of 100 randomly generated problem instances with a given set of parameters:

- Figure 3: $N = 100$, $M = 100$, $\alpha = 0.1$, $\lambda_H = 10$, $\lambda_L = 1$, $\beta = 0.2$, $P_{\text{pop}} = 0.4$, $P_{\text{unp}} = 0.05$.
- Figure 4: $N = 100$, $M = 100$, $\alpha = 0.1$, $\lambda_H = 10$, $\lambda_L = 1$, $\beta = 0.2$, $P_{\text{pop}} = 0.7$, $P_{\text{unp}} = 0.1$.
- Figure 5: $N = 100$, $M = 100$, $\alpha = 0.05$, $\lambda_H = 30$, $\lambda_L = 1$, $\beta = 0.2$, $P_{\text{pop}} = 0.4$, $P_{\text{unp}} = 0.05$.
- Figure 6: $N = 100$, $M = 100$, $\alpha = 0.05$, $\lambda_H = 30$, $\lambda_L = 1$, $\beta = 0.2$, $P_{\text{pop}} = 0.7$, $P_{\text{unp}} = 0.1$.

Since performing exhaustive search is infeasible at this problem scale, we are unable to compare the result with the optimal solution. Instead, we plot the cost overhead, $C(X, Y) - C_{\infty}$ as a function of the number of available multicast groups, $K$. In all of these cases, FBM outperforms UBM and RAN. The cost overhead of configurations generated by FBM is monotonically decreasing as the number of multicast groups increases. When fbw rates are less balanced, indicated by $\alpha$, $\lambda_H$ and $\lambda_L$, FBM shows more significant advantages in that it finds configurations that are close to optimal.
with low cost overheads, even when there are only a small number of available multicast groups.

UBM only performs well when the number of multicast groups is small and the traffic density, indicated by \( \beta \), \( P_{\text{pop}} \) and \( P_{\text{unp}} \), are low. When traffic density is high, many users are interested in most of the fbws. UBM will generate configurations in which most multicast groups contain large number of fbws and are only slightly distinct to each other. In such cases, the second component in the cost function — the cost of assigning a fbw to a multicast group will dominate. This explains the fact that the cost overhead for UBM may increase as more multicast groups are used. In an extreme example, if all users are interested in all the fbws, assigning one multicast group for each user will obviously lead to the worst configuration.

Random approaches with simple heuristics, RSE and RRE, do not perform much better than RAN, especially when the number of multicast group available is only a small percentage of all the fbws. Among these two simple heuristics, RRE is consistently better than RSE.
5.2.3 Scale of Problem

In Figure 7, we investigate the performance of our approximation approaches for different size problems. We fix the problem setting as \( \alpha = 0.05, \lambda_H = 20, \lambda_L = 1, \beta = 0.1, P_{pop} = 0.5, P_{unp} = 0.1 \) and fix the number of users be the same as the number of flows and the relative size of the number of multicast group allowed be 10% of the number of flows while we increase the number of flows in the system. We plot both the absolute cost values, \( C(X,Y) \), and relative cost values, \( C(X,Y)/C_{\infty} \) with different number of flows in the system. The result shows that as the problem becomes larger, the effectiveness of FBM remains — the ratio of \( C(X,Y)/C_{\infty} \) remains constant.

We also investigated asymmetric cases where the number of flows and number of users are different. Figure 8 shows the set of the experiments in which the number of flows is set to 1000 while the user population varies from 200 to 800. Figure 9 shows the set of the experiments in which the number of users is set to 1000 while the total flows varies from 200 to 800. We observe that user based approach (UBM) can outperform flow based approaches (FBM, etc.) when the number of total users is much smaller than the number of total flows, while in
all the other cases, the FBM is the algorithm of best performance.

6 Conclusions

This paper has investigated the constrained/unconstrained channelization and subscription problem in large scale data dissemination with IP multicast. We have formalized these problems and analyzed their complexities. By showing that both the constrained and unconstrained channelization problems are NP-Complete, we proved the intractability of finding the optimal solution for the channelization problem. We also proved that the unconstrained subscription problem is NP-Complete, while the constrained subscription problem can be solved in linear time. This provides the possibility of approximation. Given the difficulty of the channelization problem, we compared several polynomial time approximation schemes including simple heuristics such as to balance group size, to balance group rate sum, and greedy approaches such as flow based merge and user based merge. With randomly generated problems, we compared the performance of these approaches. Our results show that fbw-based merge is a good approximation, and that it can find solutions of relative low cost for a wide range of problem scales. The
user-based merge algorithm is only having advantage when the number of flows is much larger than the number
of users. We found that simple heuristics generally do not provide much improvement over a random assignment
scheme.

References

efficient multicast protocol for content-based publish-subscribe systems. International Conference on Distributed


internet-scale event notification service. In 19th ACM Symposium on Principles of Distributed Computing (PODC


[7] Katherine L. Morse, Lubomir Bic, Michael Dillencourt, and Kevin Tsai. Multicast grouping for dynamic data distribu-


Simulation Interoperability Workshop(SIW), March 2000.