Minimum Maximum Degree Publish-Subscribe Overlay Network Design

Melih Onus
Department of Computer Science and Engineering
Arizona State University
Tempe, AZ 85281
Email: melih@asu.edu

Andréa W. Richa
Department of Computer Science and Engineering
Arizona State University
Tempe, AZ 85281
Email: aricha@asu.edu

***Abstract*—Designing an overlay network for publish/subscribe communication in a system where nodes may subscribe to many different topics of interest is of fundamental importance. For scalability and efficiency, it is important to keep the degree of the nodes in the publish/subscribe system low. It is only natural then to formalize the following problem: Given a collection of nodes and their topic subscriptions connect the nodes into a graph which has least possible maximum degree and in such a way that for each topic t, the graph induced by the nodes interested in t is connected. We present the first polynomial time logarithmic approximation algorithm for this problem and prove an almost tight lower bound on the approximation ratio. Our experimental results show that our algorithm drastically improves the maximum degree of publish/subscribe overlay systems.**

We also propose a variation of the problem by enforcing that each topic-connected overlay network be of constant diameter, while keeping the average degree low. We present a heuristic for this problem which guarantees that each topic-connected overlay network will be of diameter 2 and which aims at keeping the overall average node degree low. Our experimental results validate our algorithm showing that our algorithm is able to achieve very low diameter without increasing the average degree by much.

I. Introduction

In the publish/subscribe (pub/sub) communication paradigm, publishers and subscribers interact in a decoupled fashion. Publishers publish their messages through logical channels and subscribers receive the messages they are interested in by subscribing to the appropriate services, which deliver messages through these channels.

A pub/sub system may be *topic-based*, if messages are published to "topics", where each topic is uniquely associated with a logical channel. Subscribers in a topic-based system will receive all messages published to the topics to which they subscribe. The publisher is responsible for defining the classes of messages to which subscribers can subscribe. In a *content-based* system, messages are only delivered to a subscriber if the attributes of those messages match constraints defined by the subscriber; each logical channel is characterized by a subset of these attributes. The subscriber is responsible for classifying the messages.

This work was supported in part by NSF awards CCF-0830791 and CCF-0830704.

Pub/sub communication systems are scalable and simple to implement (see e.g., [1]–[4], [6]–[10], [15], [17], [19]).

Hence there are many applications which are built on top of such systems, most notably a plethora of Internet-based applications, such as stock-market monitoring engines, RSS feeds [18], on-line gaming and several others. For a survey on pub/sub systems, see [14].

In this paper, we will design a (peer-to-peer) overlay network for each pub/sub topic, in the sense that for each topic t, the subgraph induced by the nodes interested in t will be connected. This translates into a *fully decentralized* topic-based pub/sub system since any given topic-based overlay network will be connected and thus nodes subscribed to a given topic do not need to rely on other nodes (agents) for forwarding their messages. Such an overlay network is called *topic-connected*.

We can evaluate the complexity of a pub/sub overlay network in terms of the cost of topic-based broadcasts on the network. As in many other systems, a space-time trade-off exists: On one hand, one would like the total time taken by the broadcast (which directly depends on the diameter of each topic-based subnetwork) to be as small as possible; on the other hand, for memory and node bandwidth considerations, one would like to keep the total degree of a node small. Those two measures are often conflicting. For example, take the simple scenario where all nodes are subscribed to the same topic: A star overlay would result in the best possible diameter but worst possible degree for the nodes. Even if we were to maintain a balanced structure (e.g., a balanced binary tree) for each topic, it is not clear how to achieve that without letting the node degrees grow as large as the sizes of the node subscription sets.

Some of the current solutions adopted in practice actually fail at maintaining *both* the diameter and the node degrees low. A naive, albeit popular, solution to topic connected-overlay network design is to construct a cycle (or a tree or any other separate overlay structure) connecting all nodes interested in a topic independently for each given topic [19]: This construction may result in a network with node degrees proportional to the nodes' subscription sizes, whereas a more careful construction, taking into account the correlations among the node subscription sets might result in much smaller

node degrees (and total number of edges).

Low node degrees are desirable in practice for scalability and also due to bandwidth constraints. Nodes with a high number of adjacent links will have to manage all these links (e.g., monitor the availability of its neighbors, incurring in heartbeats and keep-alive state costs, and connection state costs in TCP) and the traffic going through each of the links, without being able to take great advantage of aggregating the traffic (which would also reduce the number of packet headers, which can be responsible for a significant portion of the traffic for small messages). See [12] for further motivation.

The node degrees and number of edges required by a topic-connected overlay network will be low if the node subscriptions are well-correlated. In this case, by connecting two nodes with many coincident topics, one can satisfy connectivity of many topics for those two nodes with just one edge. Several recent empirical studies suggest that correlated workloads are indeed common in practice [18].

In this work, we first consider the problem of devising topic-based pub/sub overlay networks with low node degrees. More specifically, we consider the following problem:

Minimum Maximum Degree Topic-Connected Overlay (MinMax-TCO) Problem: Given a collection of nodes V, a set of topics T, and the node interest assignment I, connect the nodes in V into a topic-connected overlay network G which has the least possible maximum degree.

We present a logarithmic approximation algorithm for this problem. We also show that no polynomial time algorithm can approximate MinMax-TCO problem within a constant factor (unless P=NP), so our approximation guarantees are almost tight. We further validate our algorithm with experimental results.

We also propose a variation of the MinMax-TCO problem by enforcing that each topic-connected overlay network be of constant diameter, while keeping the average degree low (see MinAv-TCO problem defined in the next section). We present a heuristic for this problem which guarantees that each topic's induced overlay subnetwork will be of diameter 2 and which aims at keeping the average node degree of the overall topic-connected overlay network low. We validate this algorithm through experimental results.

A. Related Work

In [11], Chockler et al. introduced a closely related problem to the MinMax-TCO problem, which we call *MinAv-TCO* [In the original paper, this problem was called Min-TCO; since it aims at *minimizing the average* degree of the overlay network, and in order to avoid confusion with the MinMax-TCO problem considered in this paper, we will refer to the problem considered by Chockler et al. in [11] as MinAv-TCO in our work.]. The MinAv-TCO problem aims at minimizing the average degree of the nodes rather than the maximum degree. They present an algorithm, called GM, which achieves a logarithmic approximation on the minimum average degree of the overlay network. While minimizing the average degree is a step forward towards improving the scalability and practicality of the pub/sub system, their algorithm may still produce overlay networks of very uneven node degrees where the maximum degree may be unnecessarily high: As we will show in Section III, their algorithm may produce a network with maximum degree $|V|$ while a topic-connected overlay network of constant degree exists for the same configuration of I. Some of the high level ideas and proof techniques of [11] have their roots in techniques used for the classical Set-Cover problem. We benefit from some of the ideas in [11] and also build upon the constructions for Set-Cover, extending and modifying them to be able to handle the maximum degree case.

To the best of our knowledge, minimizing max-degree or diameter in topic-connected pub/sub overlay network design had not been directly addressed prior to this work. The overlay networks resulting from [2], [5], [10] are not required to be topic-connected. In [4], [9], [12], [19], topic-connected overlay networks are constructed, but they make no attempt to minimize the average or maximum node degree. The first to directly consider node degrees when building topic-connected pub/sub systems were Chockler et al. in [11], as we mentioned above.

B. Our Contributions

Our main contribution in this paper is the formal design and analysis of the topic-connected overlay design algorithm (MinMax-ODA) which approximates the MinMax-TCO problem within a logarithmic factor. The MinMax-ODA algorithm is a greedy algorithm which relies on repeatedly using a greedy approach for finding matchings that connect a large (close to maximum) number of different connected components which emerge for the given topics. We also show that no polynomial time algorithm can approximate the MinMax-TCO problem within a constant factor (unless P=NP), and so our MinMax-ODA algorithm is almost tight. No previous algorithm with sublinear approximation guarantees on the maximum degree of a topic-connected pub/sub overlay network was known prior to this work. Furthermore, we validate the performance of MinMax-ODA with experimental results.

In addition, we present an algorithm, CD-ODA, which builds a topic-based pub/sub network, where each topic-connected component is guaranteed to be of constant diameter — more specifically of diameter 2 —and where we aim at keeping the average degree low. While we do not have a formal proof on any approximation guarantees on the average node degree, we present steps of a possible formal proof, intuitions and conjectures. Furthermore, we validate the performance of CD-ODA with experimental results.

C. Structure of the paper

In Section II, we present some definitions and restate the formal problem definition. In Section III, we present an outline of the related problem of minimizing the average node

degree, namely the MinAv-TCO problem, and the corresponding logarithmic approximation algorithm GM proposed by Chockler et al. [11], since some of the ideas presented will be useful for the minMax-TCO problem. Section IV presents our topic-connected overlay design algorithm MinMax-ODA, whose approximation ratio is proved in Section V; in Section VI, we present the hardness of approximation results for MinMax-TCO. Section VIII addresses the CD-TCO problem and presents our algorithm, CD-ODA, for the same. We conclude the paper, also presenting some future work, in Section IX.

II. Preliminaries

Let V be the set of nodes, and T be the set of topics. Let $n = |V|$. The interest function I is defined as $I : V \times T \rightarrow \{0, 1\}$. For a node $v \in V$ and topic $t \in T$, $I(v, t) = 1$ if and only if node v is subscribed to topic t, and $I(v, t) = 0$ otherwise.

For a set of nodes V, an overlay network $G(V, E)$ is an undirected graph on the node set V with edge set $E \subseteq V \times V$. For a topic $t \in T$, let $V_t = \{v \in V | I(v, t) = 1\}$. Given a topic $t \in T$ and an overlay network $G(V, E)$, the number of topic-connected components of G for topic t is equal to the number of connected components of the subgraph of G induced by V_t. An overlay network G is *topic-connected* if and only if it has one topic-connected component for each topic $t \in T$. The diameter of a graph is the length of the longest shortest path in the graph. The degree of a node v in an overlay network $G(V, E)$ is equal to the total number of edges adjacent to v in G.

Minimum Maximum Degree Topic-Connected Overlay (MinMax-TCO) Problem: Given a collection of nodes V, a set of topics T, and the node interest assignment I, connect the nodes in V into a topic-connected overlay network G which has least possible maximum degree.

III. MinAv-TCO problem and Greedy Merge(GM) Algorithm

The MinAv-TCO problem was introduced by Chockler et. al. [11] in which they aim at minimizing the average node degree. In this section we present a formal definition of the MinAv-TCO problem and outline the main techniques in the corresponding Greedy Merge (GM) algorithm, which will be useful for our approach to MinMax-TCO. We start with a formal definition of the MinAv-TCO problem.

Minimum Topic Connected Overlay Problem (MinAv-TCO): Given a collection of nodes V, a set of topics T, and a node interest assignment I, connect the nodes in V into a topic-connected overlay network G which has the least possible total number of edges (and hence the least possible average node degree).

The Greedy Merge (GM) Algorithm [11]: Initially we a have the set of nodes V and no edges between the nodes. At each step, add the edge which maximally reduces the total number of topic-connected components.

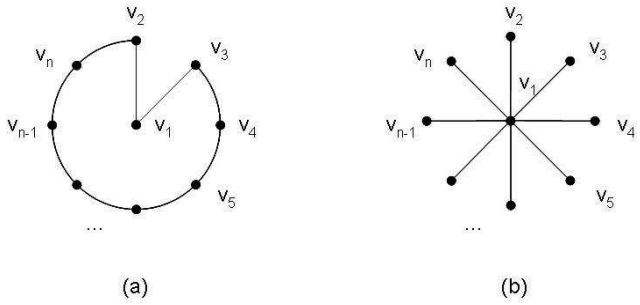


Fig. 1. (a) Overlay with optimal max degree (b) Overlay constructed by GM

The GM algorithm does not work well for the MinMax-TCO problem: The approximation ratio on the maximum degree obtained by the GM algorithm may be as bad as $\Theta(n)$, as we show in the lemma below.

Lemma 1. *The GM algorithm can only guarantee an approximation ratio of at least $\Theta(n)$ for the MinMax-TCO problem, where n is number of nodes in the pub/sub system.*

Proof: Consider the example where we have n nodes $v_1, v_2, ..., v_n$, and n topics $T = \{t_1, t_2, ..., t_n\}$. Node v_1 is interested in all topics in T and each v_i is interested in $T - t_i$, $2 \leq i \leq n$. The GM algorithm would produce an overlay network with max degree $n-1$. The overlay network in Figure 1 (b), where $E = \{(v_1, v_i) | 1 < i \leq n\}$, would result from the GM algorithm – the maximum degree of this overlay network is $n - 1$. The optimal solution for the MinMax-TCO on the same configuration for the nodes V is the overlay network $G(V, E')$, where $E' = \{(v_i, v_{i+1}) | 3 \leq i < n\} \bigcup \{(v_n, v_2), (v_1, v_2), (v_1, v_3)\}$ (see Figure 1 (a)), which has has maximum degree 2. Hence the approximation ratio of the GM algorithm can be as large as $(n-1)/2 = \Theta(n)$. ■

IV. Overlay Design Algorithm

In this section we present our overlay design algorithm (MinMax-ODA) for the MinMax-TCO problem. MinMax-ODA starts with the overlay network $G(V, \emptyset)$. At each iteration of MinMax-ODA, a maximum weight edge — where the weight of an edge (u, v) is given by the reduction on the number of topic-connected components which would result from the addition of (u, v) to the current overlay network — among the ones which minimally increases maximum degree of the current graph is added to edge set of the overlay network. Let $NC(V, E)$ denote total number of topic connected components in the overlay network given by (V, E).

Steps 1-6 of MinMax-ODA build an initial weighted graph $G'(V, E', w)$ on V, where $E' = V \times V$ and $w(\{u, v\})$ is equal to the amount of decrease in the number of topic-connected components resulting from the addition of the edge (u, v) to the current overlay network (represented by the edges in OverlayEdges). Initially, this amount will be equal to the number of topics that nodes u and v have in common.

Algorithm 1 Minimum Maximum Degree Overlay Design Algorithm (MinMax-ODA)

```
OverlayEdges ← ∅
V ← Set of all nodes
G'(V, E') ← Complete graph on V
for {u, v} ∈ E' do
    w{u, v} ← Number of topics that both of nodes u and v have
end for
while G(V,OverlayEdges) is not topic-connected do
    Find maximum-weighted edge e on G'(V, E', w) among the ones which increase the maximum degree of G(V,OverlayEdges) minimally.
    OverlayEdges = OverlayEdges ⋃ e
    E' ← E' − e
    for {u, v} ∈ E' do
        w{u, v} ← NC(V, OverlayEdges) - NC(V,OverlayEdges ⋃{u, v} )
    end for
end while
```

While at a first glance MinMax-ODA may look very similar to GM, it actually can be shown to work in phases, where in each phase a collection of edges that form matching of the nodes in the pub/sub system that connects close to the maximum number of connected components for the different topics is selected, as we explain below. Such a matching decomposition of the selected edges, crucial for our approximation ratio analysis, was not possible for GM.

At each iteration of the while loop, a maximum weight edge among the ones which increase the maximum degree of the current graph minimally is added to the set of overlay edges. Note that the addition of an edge to OverlayEdges can either increase the maximum degree by 1 or not increase it at all. For ease of explanation, assume that we have an even number of nodes in the pub/sub system. Since we start with all nodes having equal degree (equal to 0), MinMax-ODA will first select a set of edges that increases the degree of every node to 1 (i.e., a matching), and then a set of edges (another matching) that increases the degree of each node to 2, etc. Note that some of these edges may have weight 0 (i.e., they do not really contribute to the construction of the topic-connected components), but they will not affect the final solution obtained (the 0-weight edges will be discarded at the end of the algorithm). The crux in the analysis of this algorithm is to show that each of these matchings will reduce the number of connected components by a "large" amount.

Before we proceed in proving the approximation ratio on the maximum degree guaranteed by MaxMin-ODA, we prove that the algorithm terminates in $O(|V|^4|T|)$ time.

Lemma 2. *The MinMax-ODA algorithm terminates within $O(|V|^2)$ iterations on the while loop.*

Proof: At each iteration of the while loop, at least one edge is added to the current overlay network. Hence the algorithm will terminate in at most $O(|V|^2)$ iterations. ■

Lemma 3. *The running time of MinMax-ODA is $O(|V|^4|T|)$.*

Proof: The weight initialization takes $O(|V|^2|T|)$ time. Updating the weight of each of the remaining edges takes $O(1)$ time([11], Lemma 6.4). Finding the edge with max weight will take at most $O(|V|^2)$ time. Since total weight of the edges is $O(|V|^2|T|)$ at the beginning and greater than 0 at the end, MinMax-ODA takes $O(|V|^2|T|) * O(|V|^2) = O(|V|^4|T|)$ time. ■

V. Approximation Ratio

In this section, we will prove that our overlay design algorithm (MinMax-ODA) approximates the MinMax-TCO problem within a logarithmic factor.

Theorem 1. *The overlay network output by MinMax-ODA has maximum node degree within a factor of $O(\log(\sum_{v\in V}|\{t \in T|I(v,t) = 1\}|))$ from the minimum possible maximum node degree for any topic-connected overlay network on V.*

Proof: At a high level, the proof follows the general lines as the proof of the logarithmic approximation ratio for the classic set cover problem (which was also the basis for the approximation ratio proof of the GM algorithm for the MinAv-TCO problem [11]). However, before we can apply the set cover framework, we first need to carefully show that MinMax-ODA works as if we had many applications of a greedy matching algorithm that aims at reducing the number of connected components maximally and then relate our network overlay construction to a matching decomposition of an optimal (i.e., a minimum maximum degree) overlay network.

Assume we have an instance of the MinMax-TCO problem and that $G(V, E_{opt})$ is an optimum solution for this instance with maximum degree d_{opt}. We will use the following well-known result in graph theory for the proof.

Lemma 4. *Given a graph $G(V, E)$ with maximum degree d, we can divide the edge set E into $k = d + 1$ matchings M_i, $1 \le i \le k$.*

Proof: We can color the edges of any graph with $d + 1$ colors such that any adjacent edge will have different color. This is Vizing's Edge Coloring Theorem (Theorem 5.3.2 in [13]). Since each coloring class is a matching, we can divide the edge set into $d + 1$ matchings. ■

Using the lemma above, we can divide the edge set E_{opt} of the optimum solution into $k = d_{opt} + 1$ matchings M_i, $1 \le i \le k$.

At the beginning of the algorithm, the total number of connected components is $C_{start} = \sum_{v\in V}|\{t \in T|I(v,t) = 1\}|$ and at the end $C_{end} = |\{t|t \in T$ and $\exists v \in V$ such that $I(v,t) = 1\}|$. Note that since we count the connected components for each topic separately, once we get down to C_{end} components, there must exist *exactly one* component for each active topic t (i.e., each t such that there exists some

v with $I(v,t) = 1$) — i.e., the overlay network is topic-connected.

For ease of explanation, assume that we have an even number of nodes in the pub/sub system (if there are an odd number of nodes, one can always add a "dummy" node which is not subscribed to any topic, without affecting the final solution; or we can handle small deviations from a perfect matching decomposition in the analysis). At each iteration of the while loop, a maximum weight edge among the ones which increases the maximum degree of the current graph minimally is added to the set of overlay edges. At start all nodes have degree 0. After a number of iterations the edges added will form a perfect matching and then the next edge added will increase the max degree of the graph by 1.

Let S_i be the edge set of the i^{th} matching added to the set by the algorithm MinMax-ODA, $1 \leq i \leq k$. Let n_i be total number of connected components before we add i^{th} matching, so $n_1 = C_{start}$. Let $SA_i = S_1 \bigcup S_2 \bigcup ... \bigcup S_{i-1}$ be the union of all matchings found before the algorithm starts adding the i-th matching.

The following lemma proves that each matching S_i chosen by our algorithm decrease the current total number of connected components at least $(1/3)$- optimally.

Lemma 5. *The matching S_i reduces the total number of connected components of $G(V, SA_i)$ by at least $1/3$ of any optimal matching which reduces by maximum amount.*

Proof: Let P be the edge set of the matching which reduces the total number of connected components of the $G(V, SA_i)$ by the maximum amount, which we denote by c. Let $Q = \{e_1, e_2, \ldots, e_j\}$ be the edge set of the matching S_i that our algorithm finds. Let $e_l = u_l v_l$ for $1 \leq l \leq j$. For e_a and e_b, if $a < b$, then e_a is found before e_b by our algorithm. Let Q reduce the total number of connected components of the $G(V, SA_i)$ by c'. Let $G_0 = G(V, SA_i)$ and $G_l = G_{l-1} \bigcup e_l$, for $1 \leq l \leq j$. Let e_l reduce the total number of connected components of G_{l-1} by y_l. Then,

$$c' = \textstyle\sum_{1 \leq l \leq j} y_l \tag{1}$$

$$y_a \geq y_b, \text{ for } 1 \leq a \leq b \leq j \tag{2}$$

Let X_l be the set of edges in P which are incident to u_l or v_l, $1 \leq l \leq j$, but not incident to $u_{l'}$ or $v_{l'}$ $1 \leq l' \leq l-1$. Thus, X_l will have zero or one or two edges for $1 \leq l \leq j$. Let $P_0 = P$ and $P_l = P_{l-1} - X_l$ for $1 \leq l \leq j$. Since Q is a maximal matching, $P_j = \emptyset$. Let X_l reduce the total number of connected components of G_{l-1} by x_l for $1 \leq l \leq j$. Let P_l reduce the total number of connected components of G_l by c_l for $0 \leq l \leq j$.

If X_l has two edges, then our algorithm did not choose one of these two edges at that step and choose e_l instead, $0 \leq l \leq j$. Since our algorithm greedily choose the edges, e_l reduces the total number of connected components of G_{l-1} by at least as much as each of the edges in X_l. Hence, $y_l \geq x_l/2$. Similarly, if X_l has one or zero edges, then $y_l \geq x_l$. So,

$$y_l \geq \tfrac{x_l}{2},\ 1 \leq l \leq j$$

$$\Rightarrow \textstyle\sum_{1 \leq l \leq j} y_l \geq \frac{1}{2} \sum_{1 \leq l \leq j} x_l \tag{3}$$

Since $P_{l+1} = P_l - X_{l+1}$ and $G_{l+1} = G_l \bigcup e_{l+1}$, $0 \leq l \leq j-1$, the amount that P_l reduces the total number of connected components of G_l is smaller than sum of the amount that P_{l+1} reduces the total number of connected components of G_{l+1} and the amount that e_{l+1} reduces the total number of connected components of G_l and the amount X_{l+1} reduces the total number of connected components of G_l. Hence,

$$c_0 = c,\ c_j = 0 \tag{4}$$

$$c_{l+1} \geq c_l - (x_{l+1} + y_{l+1}) \text{ for } 0 \leq l \leq j-1 \tag{5}$$

If we add all the inequalities (4) and (5), we will have

$$\textstyle\sum_{1 \leq l \leq j} x_l + \sum_{1 \leq l \leq j} y_l \geq c \tag{6}$$

From the inequalities (3) and (6), we will have

$$3 \textstyle\sum_{1 \leq l \leq j} y_l \geq c \tag{7}$$

From the inequalities (1) and (7), we will have

$$c' \geq c/3$$

■

Before MinMax-ODA starts adding the i^{th} matching, we have n_i components and we know that if we add all the $k = d_{opt} + 1$ matchings $M_j - SA_i$, $1 \leq j \leq k$, to the current solution, the total number of connected components will be reduced to C_{end}. Therefore, there exists a matching $M_j - SA_i$ which decreases the total number of connected components by at least $(n_i - C_{end})/k$. Since our algorithm always finds at least a $(1/3)$-optimal matching (Lemma 5), the matching S_i that our algorithm uses must decrease the total number of connected components at that time by at least $(1/3)$ of this amount. Therefore,

$$n_i - n_{i+1} \geq (n_i - C_{end})/(3k)$$
$$\Rightarrow n_{i+1} - C_{end} \leq (1 - 1/(3k))(n_i - C_{end}).$$

Hence, the number of iterations for our algorithm MinMax-ODA is less than or equal to the smallest m which satisfies

$$1 > (n_1 - C_{end})(1 - 1/(3k))^m$$
$$\Rightarrow m \leq 3kln(C_{start} - C_{end})$$
$$\Rightarrow m \leq 3kln(C_{start})$$
$$\Rightarrow m \leq 3(d_{opt} + 1) * ln(C_{start})$$

■

VI. Hardness of MinMax-TCO Problem

In this section, we will show that no polynomial time algorithm can approximate the MinMax-TCO problem within a constant factor.

Theorem 2. *There exists no polynomial time algorithm that approximates the MinMax-TCO problem within a constant factor unless P=NP.*

Proof: The proof follows the proof of Theorem 5.3 in [11]. In fact, a more careful observation of the proof of Theorem 5.3 in [11] shows that the proof basically shows the hardness of approximation of the maximum degree (in [11], the authors consider the problem of approximating the average degree). For the sake of completeness, we present here the sketch of the proof of Theorem 5.3 [11] when directly applied to maximum degree. We first define the single node version of the MinMax-TCO problem.

Single node version of the MinMax-TCO problem(SN-MinMax-TCO): Given V, T, I, a node $v \in V$, connect the nodes in V into an overlay network G which has least possible degree for node v and G is topic-connected.

We can prove that SN-MinMax-TCO is NP-hard by reducing the minimum set cover problem to this problem. Then we show how to reduce the SN-MinMax-TCO problem to the MinMax-TCO problem, thus showing that the MinMax-TCO problem is NP-hard.

Now, we are ready to prove our inapproximability result. We will use the same reduction as in the proof of Theorem 5.3 in [11]. Assume that there is an algorithm A which approximates MinMax-TCO problem within a constant factor. We will show we can use this algorithm A to find a constant factor approximation to the minimum set cover problem, which is known to be impossible unless P = NP.

Given an instance of $SN - MinMax - TCO(V, T, I, v)$ problem, construct the corresponding instance of $MinMax - TCO(V', T', I')$ problem as in the construction of proof of Lemma 5.1 [11]. Let d_{opt} denote the maximum degree of an optimal solution for this instance of $SN - MinMax - TCO(V, T, I, v)$ problem. As shown in proof of Lemma 5.1 [11], the corresponding $MinMax - TCO(V', T', I')$ problem also has a solution with degree d_{opt}. So, the algorithm A will find a solution to the corresponding $MinMax - TCO(V', T', I')$ problem with degree at most $c * d_{opt}$. With using this solution, we can construct a solution to $SN - MinMax - TCO(V, T, I, v)$ problem with degree at most $c*d_{opt}$ as in Lemma 5.1 [11]. Thus, we have a constant approximation algorithm B for $SN - MinMax - TCO(V, T, I, v)$ problem.

Given an instance of the minimum set cover problem (U, S), construct the corresponding $SN - MinMax - TCO(V, T, I, v)$ problem same as in the construction of proof of Lemma 5.2 [11]. Denote s_{opt} an optimal solution for this instance of minimum set cover problem. As shown in proof of Lemma 5.2 [11], the corresponding $SN - MinMax - TCO(V, T, I, v)$ problem has a solution with degree s_{opt}. So, the algorithm B will find a solution to the corresponding $SN - MinMax - TCO(V, T, I, v)$ problem with degree at most $c' * s_{opt}$. With using this solution, we can construct a solution to minimum set cover problem with number of sets at most $c' * s_{opt}$ as in Lemma 5.2 [11]. Thus, we have a constant approximation algorithm C for the minimum set cover problem.

■

Since it is trivial to show that MinMax-TCO is in NP, it follows

Corollary 1. *The MinMax-TCO problem is NP-complete.*

VII. Experimental Results

The GM algorithm [12] and our MinMax-ODA algorithm are implemented in Java. These two algorithms are compared according to maximum degree and average degree in the resulting overlay graphs. Experimental results show that MinMax-ODA improves the maximum degree of the overlay network drastically at the cost of a small increase on the average degree.

A. Maximum Node Degree

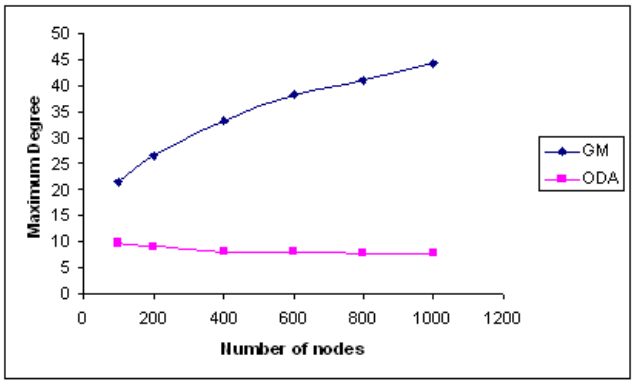
For these experiments, the number of nodes varies between 100 to 1000. In the first experiment (Figure 2(a)), the number of topics is 100 and in the second experiment (Figure 2(b)) the number of topics is 200. We fixed number of subscriptions to $s = 10$. Each node is interested in each topic uniformly at random. This experimental setting is similar to previous studies [12].

Figure 2(a) is a comparison of GM and MinMax-ODA algorithm according to the maximum degree. The maximum degree of the graph decreases for MinMax-ODA algorithm when the number of nodes increases since MinMax-ODA algorithm can find edges with higher correlation as the number of nodes increases. Interestingly, the maximum degree of the graph increases for the GM algorithm as the number of nodes increases. Basically as the number of nodes increase, the GM algorithm will assign more edges to same the nodes since now we have more nodes with higher correlation for each node. When we compare the results of GM and MinMax-ODA algorithm, MinMax-ODA improves GM by factor 3 on average (Figure 2(a)).

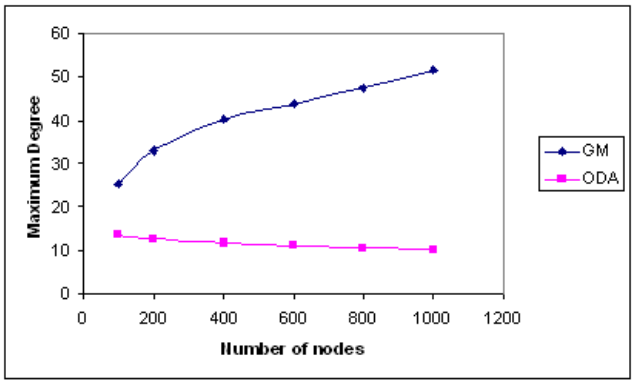
The same results are valid for Figure 2(b). When we compare Figure 2(a) and Figure 2(b), max node degree increases slightly for both GM and MinMax-ODA since edges will have less correlation when we increase the number of topics.

B. Average Node Degree

Experimental setting is same as previous subsection. Figure 3(a) is comparison of GM and MinMax-ODA algorithm according to average degree. The average degree of the graph decreases for both GM and MinMax-ODA algorithms when the number of nodes increases since algorithms can find edges with higher correlation when the number of nodes increases.

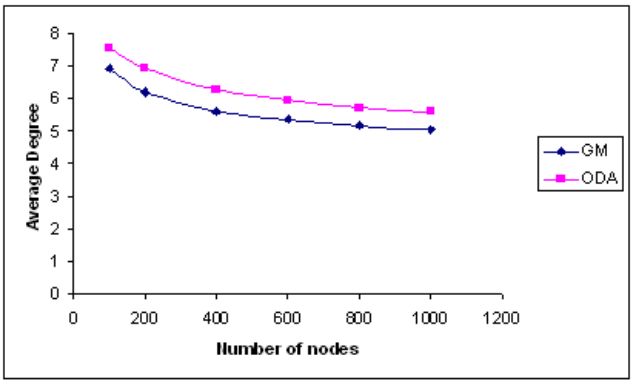


(a)

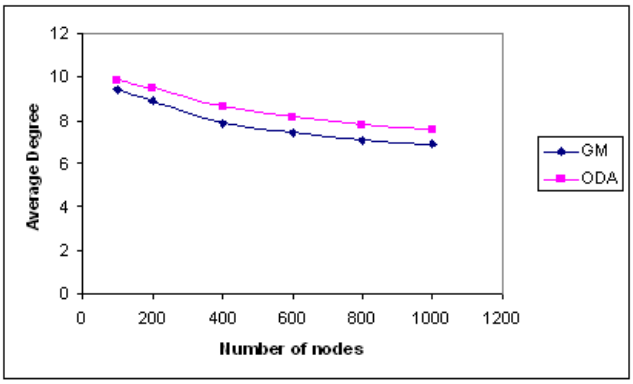


(b)

Fig. 2. Maximum node degree for GM and MinMax-ODA (a) Number of topics is 100 (b) Number of topics is 200



(a)



(b)

Fig. 3. Average node degree for GM and MinMax-ODA (a) Number of topics is 100 (b) Number of topics is 200

When we compare the results of GM and MinMax-ODA algorithm, GM is slightly better than MinMax-ODA, 7% on average, (Figure 3(a)). Similar results are valid for Figure 3(b).

C. Subscription Size

In these experiment, the number of nodes and the number of topics are fixed to 100. The subscription size varies between 10 to 50. Each node is interested in each topic uniformly randomly. Figure 4(a) is the comparison of GM and MinMax-ODA algorithm according to the maximum degree. The maximum degree of the overlay network decreases for the MinMax-ODA algorithm as the subscription size increases since the MinMax-ODA algorithm can find edges with higher correlation when the subscription size increases. Interestingly, the maximum degree of the overlay network increases for GM algorithm as the subscription size increases. When the subscription size increases, GM algorithm will assign more edges to the same nodes since now we have more nodes with higher correlation for each node. When we compare the results of GM and MinMax-ODA algorithms, MinMax-ODA improves GM by factor a 4 on average (Figure 2(a)).

Figure 4(b) is comparison of GM and MinMax-ODA algorithm according to the average degree. The average degree of the overlay network decreases for both GM and MinMax-ODA algorithms when subscription size increases since algorithms can find edges with more correlation. When we compare the results of GM and MinMax-ODA algorithm, GM is slightly better than MinMax-ODA, %10 on average, (Figure 4(b)).

VIII. Constructing Constant Diameter Overlays for Publish-Subscribe

In this section, we study a new optimization problem that constructs a constant diameter overlay network for publish/subscribe communication with many topics. We present an overlay network construction heuristic that guarantees constant diameter and topic-connectivity which are most important factors for efficient routing. The formal problem is as follows:

Constant Diameter Topic-Connected Overlay (CD-TCO) Problem: Given a collection of nodes V, a set of topics T, and the node interest assignment I, connect the nodes in V into a topic-connected overlay network G which has least possible average degree and constant diameter.

This problem aims at minimizing the average degree, as does the MinAv-TCO problem introduced by Chockler et. al. [11], with the additional requirement on diameter. We present a heuristic for this problem and validate our heuristic via experimental results.

We first show that the GM algorithm does not work well for the CD-TCO problem. Second, we present our 2-diameter

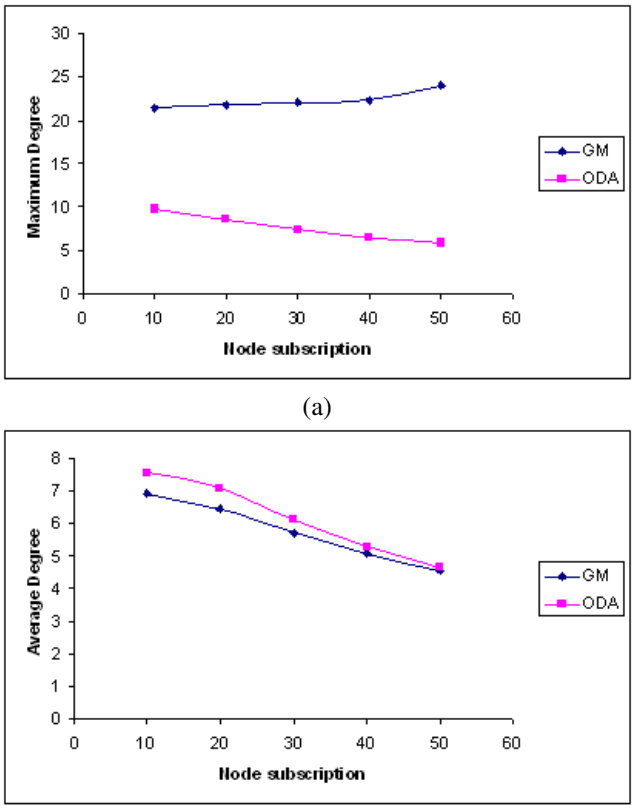


Fig. 4. Average and maximum node degree for different subscription size (Number of nodes and number of topics is 100)

algorithm and its performance evaluation.

A. The GM Algorithm and the CD-TCO Problem

The GM algorithm does not work well for the CD-TCO problem: The diameter obtained by the GM algorithm may be as bad as $\Theta(n)$, as we show in the lemma below.

Lemma 6. *The GM algorithm can have diameter of $\Theta(n)$, where n is number of nodes in the pub/sub system.*

Proof: Consider the example where we have n nodes $v_1, v_2, ..., v_n$, one topic t and every node is interested in t. There exists many orderings of the edges of G for which the GM algorithm would produce an overlay network with diameter $n-1$. For example, the overlay network , $G(V,E)$, where $E = \{(v_i, v_{i+1}) | 1 \leq i < n\}$, can result from the GM algorithm – the diameter of this overlay network is $n-1$. Another solution for the CD-TCO on the same configuration for the nodes V is the overlay network $G(V,E')$, where $E' = \{(v_1, v_i) | 1 < i \leq n\}$, which has diameter 2. ■

B. Constant Diameter Overlay Design Algorithm (CD-ODA)

We will present a greedy algorithm (CD-ODA) for the problem. Our algorithm generates a star for each topic and hence each topic-connected component will have diameter 2.

CD-ODA starts with the overlay network $G(V,\emptyset)$. At each iteration of the CD-ODA, a node which has maximum number neighbors with non-empty interest intersection is chosen. The number of neighbors of a node u is equal to

$$n_u = |\{v \in V | \exists t \in T, Int(v,t) = Int(u,t) = 1\}|$$

We then put an edge between this node and each of its neighbors, and remove all the topics in this node's interest assignment from the set of topics.

Algorithm 2 Constant Diameter Overlay Design Algorithm (CD-ODA)

1: $T \leftarrow$ Set of all topics
2: **while** T is not empty **do**
3: For each node u, calculate number of nodes v such that there exists a topic t in T and $Int(u,t) = Int(v,t) = 1$. Denote this number by n_u.
4: Find node u with maximum n_u.
5: Put an edge between u and all nodes v such that there exists a topic $t \in T$ and $Int(u,t) = Int(v,t) = 1$.
6: Remove all topics t from T such that $Int(u,t) = 1$.
7: **end while**

C. Analysis of Algorithms

Lemma 7. *CD-ODA terminates within $O(|V|^2 * |T|)$ time.*

Lemma 8. *CD-ODA generates a 2-diameter overlay for each topic.*

Proof: Since the algorithm generates a star for each topic, each topic overlay network will have diameter 2. ■

D. Conjectures

Conjecture 1. *CD-ODA approximates the constant diameter overlay design problem within a logarithmic factor.*

Our first intuition is that if there exists a k-edge overlay network with constant diameter for a graph G, then there exists a constant c such that there exists a $c * k$-edge overlay with diameter 2 for a graph G. This step will make the reduction from constant diameter overlay to 2-diameter overlay.

For 2-diameter overlay, the most efficient graph for each topic alone will have a star structure, since a star has diameter 2 and optimal number of edges (n-1 edges). Combining these two steps, we have Conjecture 1.

Conjecture 2. *There exists no polynomial time algorithm that approximates constant diameter overlay design problem with a constant factor unless P=NP.*

Our intuition for Conjecture 2 is that the set cover problem can be reduced to this problem as for the MinMax-TCO and Min-TCO problems.

E. Experimental Results

The GM algorithm [12] and CD-ODA are implemented in Java. These two algorithms are compared according to the average degree in the resulting graph. Experimental results show that CD-ODA improve the diameter of the overlay

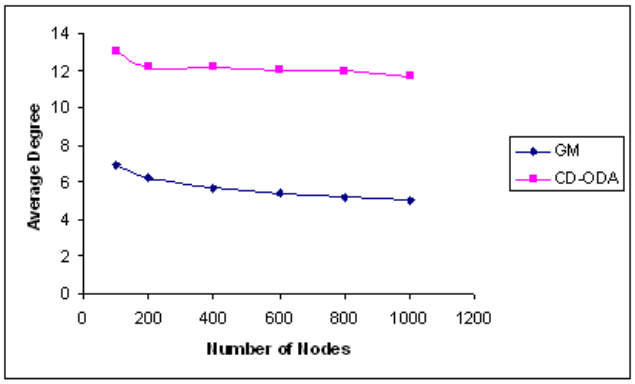


Fig. 5. Average node degree for GM and CD-ODA

network drastically at the cost of a small increase on the average degree. The diameter is always 2 for CD-ODA and it may be $\theta(n)$ for GM algorithm. When we compare the results of GM and CD-ODA according to average degree, CD-ODA requires at most 2.3 times more edges.

In the experiment, the number of nodes varies between 100 to 1000. The number of topics is 100. We fixed number of subscriptions to $s = 10$. Each node is interested in each topic uniformly at random. This experimental setting is similar to previous studies [11].

Figure 5 is a comparison of GM and CD-ODA according to the average degree. The average degree of the graph decreases for GM algorithm when the number of nodes increases since GM algorithm can find edges with higher correlation as the number of nodes increases. The average degree of the graph slightly decreases for our algorithms. When we compare the results of GM and CD-ODA, our algorithms requires at most 2.3 times more edges than GM (Figure 5).

IX. Conclusions

In this paper, we study a new optimization problem (MinMax-TCO) that constructs a practical and scalable overlay network for publish/subscribe communication with many topics. We present a topic-connected overlay network design algorithm (MinMax-ODA) which approximates the MinMax-TCO problem within a logarithmic factor. We also show that the approximation factor of MinMax-ODA is almost tight, since no constant-approximation polynomial-time algorithm can exist for the MinMax-TCO problem (unless P=NP).

Our experimental results validate our formal analysis of the MinMax-ODA algorithm, showing that the maximum degree obtained by our algorithm clearly outperforms the maximum degree obtained when GM is used.

We present a heuristic for constructing constant diameter overlay networks. Our experimental results show that the diameter obtained by our heuristics outperforms the diameter obtained when GM is used while only increasing the average degree by a factor of 2.3.

As future work, we would like to build upon our CD-ODA algorithm, by formally and experimentally evaluating the hardness of obtaining a topic-connected overlay design algorithm which achieves a "good" trade-off between low diameter and low node degree. This basically amounts to a bicriteria optimization problem and we have to be able to "quantify" the relative importance of optimizing over these two parameters (e.g., in the CD-ODA algorithm we restrict our attention to networks of diameter 2, while aiming at maintaining the average degree low).

Two other important lines for future work would be to design efficient *distributed* algorithms for the MinMax-TCO problem, and to look at this problem under the line of a dynamic configuration of the node set V and the interest assignment I.

References

[1] *Oracle9i Application Developers Guide Advanced Queuing*, Oracle, Redwood Shores, CA.

[2] E. Anceaume, M. Gradinariu, A. K. Datta, G. Simon, and A. Virgillito, *A semantic overlay for self- peer-to-peer publish/subscribe*, In ICDCS, 2006.

[3] S. Baehni, P. T. Eugster, and E. Guerraoui, *Data-aware multicast.* In DSN, 2004.

[4] R. Baldoni, R. Beraldi, V. Quema, L. Querzoni, and S. T. Piergiovanni, *TERA: Topic-based Event Routing for Peer-to-Peer Architectures*, 1^{st} International Conference on Distributed Event-Based Systems (DEBS). ACM, 6 2007.

[5] R. Baldoni, R. Beraldi, L. Querzoni, and A. Virgillito, *Efficient publish/subscribe through a self-organizing broker overlay and its application to SIENA*, The Computer Journal, 2007.

[6] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, *Scalable application layer multicast*, SIGCOMM Comput. Commun. Rev, 32(4):205-217, 2002.

[7] S. Bhola, R. Strom, S. Bagchi, Y. Zhao, and J. Auerbach, *Exactly-once delivery in a content-based publish-subscribe system.* In DSN, 2002.

[8] A. Carzaniga, M. J. Rutherford, and A. L. Wolf, *A routing scheme for content-based networking.*, IEEE INFOCOM 2004, Hon Kong, China, Mar. 2004.

[9] M. Castro, P. Druschel, A. M. Kermarrec, and A. Rowstron, *SCRIBE: a large-scale and decentralized application-level multicast infrastructure*, IEEE J. Selected Areas in Comm. (JSAC), 20(8):14891499, 2002.

[10] R. Chand and P. Felber, *Semantic peer-to-peer overlays for publish/subscribe networks*, In Euro-Par 2005 Parallel Processing, Lecture Notes in Computer Science, volume 3648, pages 11941204. Springer Verlag, 2005.

[11] G. Chockler, R. Melamed, Y. Tock and R. Vitenberg, *Constructing scalable overlays for pub-sub with many topics*, Proc. of the 26^{th} ACM Symp. on Principles of Distributed Computing (PODC), 2007, pp. 109–118.

[12] G. Chockler, R. Melamed, Y. Tock, and R. Vitenberg, *SpiderCast: A Scalable Interest-Aware Overlay for Topic-Based Pub/Sub Communication*, 1^{st} International Conference on Distributed Event-Based Systems (DEBS). ACM, 6 2007.

[13] R. Diestel, *Graph Theory*, Springer-Verlag, 2^{nd} edition, New York, 2000.

[14] P. T. Eugster, P. A. Felber, R. Guerraoui, and A. M. Kermarrec. *The many faces of publish/subscribe*. ACM Computing Surveys, 35(2):114131, 2003.

[15] R. Guerraoui, S. Handurukande, and A. M. Kermarrec, *Gossip: a gossip-based structured overlay network for efficient content-based filtering*, Technical Report IC/2004/95, EPFL, Lausanne, 2004.

[16] B. Korte, J. Vygen, *Combinatorial Optimization Theory and Algorithms*, Springer-Verlag, 2^{nd} edition, 2000.

[17] R. Levis, *Advanced Massaging Applications with MSMQ and MQSeries.* QUE, 1999.

[18] H. Liu, V. Ramasubramanian, and E. G. Sirer. *Client behavior and feed characteristics of rss, a publish-subscribe system for web micronews*. In Internet Measurement Conference (IMC), Berkeley, California, October 2005.

[19] S. Voulgaris, E. Riviere, A. M. Kermarrec, and M. van Steen, *Sub-2-sub: Self-organizing content-based publish subscribe for dynamic large scale collaborative networks*, In IPTPS, 2006.