

# Minimum Maximum-Degree Publish–Subscribe Overlay Network Design

Melih Onus and Andréa W. Richa

**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 that has least possible maximum degree 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 three heuristics for this problem that guarantee that each topic-connected overlay network will be of diameter 2 and that aim at keeping the overall average node degree low. Our experimental results validate our algorithms, showing that our algorithms are able to achieve very low diameter without increasing the average degree by much.

**Index Terms**—Communications technology, peer-to-peer computing.

## 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.

Pub/sub communication systems are scalable and simple to implement (see, e.g., [1], [3]–[5], [7]–[11], [16], [18], [23], [25], [27], [30], and [31]). Hence, there are many applications that are built on top of such systems, most notably a plethora of Internet-based applications, such as stock-market monitoring engines, RSS feeds [19], [30], online gaming, and several others. Publish/subscribe schemes have been implemented by many industrial strength solutions (e.g., Altherr *et al.* [2], Talarian Corporation [28], Skeen [26], Tibco [29]). For a survey on pub/sub systems, see [15].

In this paper, we design a (peer-to-peer) overlay network for each pub/sub topic such that, for each topic  $t$ , the subgraph induced by the nodes interested in  $t$  is connected. This translates into a *decentralized* topic-based pub/sub system in the sense that 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 tradeoff 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 as 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 [31]. 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,

Manuscript received February 23, 2010; revised October 05, 2010; accepted January 15, 2011; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor D. Rubenstein. Date of publication June 23, 2011; date of current version October 14, 2011. This work was supported in part by the NSF under Awards CCF-0830791 and CCF-0830704. A preliminary version containing some of the results in this paper appeared at the IEEE Conference on Computer Communications (INFOCOM) Rio de Janeiro, Brazil, April 19–25, 2009, and the IEEE International Conference on Distributed Computing Systems (ICDCS) Genoa, Italy, June 21–25, 2010.

M. Onus is with the Department of Computer Engineering, Bilkent University, Ankara 06560, Turkey (e-mail: onus@cs.bilkent.edu.tr).

A. W. Richa is with the Computer Science and Engineering Program, School of Computing Informatics and Decision Systems Engineering, Arizona State University, Tempe, AZ 85281 USA (e-mail: aricha@asu.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TNET.2011.2144999

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 (by aggregating traffic, we also reduce the number of packet headers, which can be responsible for a significant portion of the traffic for small messages). See [13] 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 the 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 [19], [30].

In this paper, 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 a node interest assignment  $I$ , connect the nodes in  $V$  into a topic-connected overlay network  $G$  that has the least possible maximum degree.

We present a logarithmic approximation algorithm for this problem. We also show that no polynomial-time algorithm can asymptotically approximate MinMax-TCO problem within less than a logarithmic factor (unless  $P = NP$ ), so our approximation guarantees are 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 three heuristics for this problem that guarantee that each topic's induced overlay subnetwork will be of diameter 2 and that aims at keeping the average node degree of the overall topic-connected overlay network low. We validate these algorithms through experimental results.

#### A. Related Work

The results in this paper are an improved and extended version of our results in [20]. The major extensions in this work are the following. We extended the hardness of approximability proof (Theorem 2) to show that one cannot approximate the MinMax-TCO problem within an asymptotically better than logarithmic ratio of optimal (unless  $P = NP$ ), rather than a constant ratio (as it appears in [20]). We introduced new and more thorough simulation results—in particular, all of the simulation results for zipf distributions (Section VII-D) and the more extensive simulation analysis for the uniform distribution (Sections VII-A–VII-C). We also significantly extended Section VIII by introducing two improved heuristics (Sections VIII-D–VIII-G) and corresponding simulations

(Section VIII-J), which provide a much more thorough approach to the problem than the very preliminary results presented in [20].

In [12], Chockler *et al.* introduced a closely related problem to the MinMax-TCO problem, which we call *MinAv-TCO*.<sup>1</sup> 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 toward 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 [12] have their roots in techniques used for the classical Set-Cover problem. We benefit from some of the ideas in [12] 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 the maximum degree or the diameter in topic-connected pub/sub overlay network design had not been directly addressed prior to this work [20], [21]. The overlay networks resulting from [3], [6], [11] are not required to be topic-connected. In [5], [10], [13], and [31], 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 [12], as we mentioned.

#### B. Our Contributions

Our main contribution in this paper is the formal design and analysis of the topic-connected overlay design algorithm (MinMax-ODA) that approximates the MinMax-TCO problem within a logarithmic factor. The MinMax-ODA algorithm is a greedy algorithm that relies on repeatedly using a greedy approach for finding matchings that connect a large (close to maximum) number of different connected components that emerge for the given topics. We also show that no polynomial-time algorithm can approximate the MinMax-TCO problem within asymptotically less than a logarithmic 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 the MinMax-ODA with experimental results.

In addition, we present three heuristics that build a topic-based pub/sub network such that each topic-connected component is guaranteed to be of constant diameter—more specifically

<sup>1</sup>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 [12] as MinAv-TCO in our paper.

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 the algorithms via 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.* [12], 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 proven in Section V. In Section VI, we present the hardness of approximation results for MinMax-TCO. Experimental results are presented in Section VII. Section VIII addresses the CD-TCO problem and presents our algorithms for the same. We conclude the paper, also presenting some future work, in Section IX.

II. PRELIMINARIES

We will start by presenting a set of basic definitions. 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  $V_t = \{v \in V | I(v, t) = 1\}$ .

We now look at the number of connected components for each topic. 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 topic is the length of the longest shortest path between the nodes subscribed to this topic. The diameter of a graph is equal to the maximum topic diameter. 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  $G$ .

We are now ready to formally state the MinMax-TCO 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  $G$  that has least possible maximum degree.

III. MINAV-TCO PROBLEM AND GREEDY MERGE (GM) ALGORITHM

The MinAv-TCO problem was introduced by Chockler *et al.* [12], in which they aim at minimizing the average node degree. In this section, we present a formal definition of the MinAv-TCO problem and reproduce the Greedy Merge (GM) algorithm, which will be useful for our

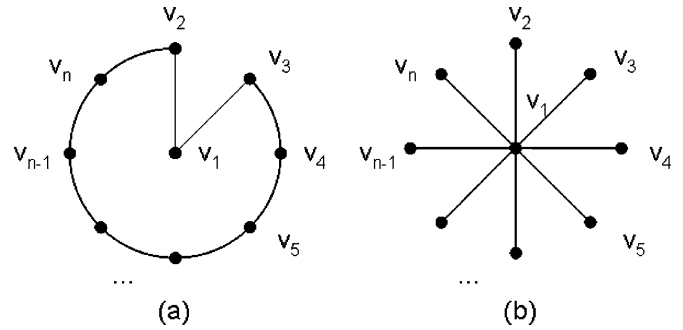


Fig. 1. (a) Overlay with optimal max degree. (b) Overlay constructed by GM.

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  $G$  that has the least possible total number of edges (and hence the least possible average node degree).

*The Greedy Merge (GM) Algorithm* [12]: Initially we have a the set of nodes  $V$  and no edges between the nodes. At each step, add the edge that maximally reduces the total number of topic-connected components.

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 high as  $\Theta(n)$ , as we show in the following lemma.

*Lemma 1:* The GM algorithm can only guarantee an approximation ratio of  $\Theta(n)$  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  $v_1, v_2, \dots, v_n$ , and  $n$  topics  $T = \{t_1, t_2, \dots, t_n\}$ . Node  $v_1$  is interested in all topics in  $T$ , and each  $v_i$  is interested in  $T \setminus \{t_i\}, 2 \leq i \leq n$ . The GM algorithm would produce an overlay network with max degree  $n - 1$ . The overlay network in Fig. 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  $E' = \{(v_i, v_{i+1}) | 3 \leq i < n\} \cup \{(v_n, v_2), (v_1, v_2), (v_1, v_3)\}$  [see Fig. 1(a)], which has maximum degree 2. Hence, the approximation ratio of the GM algorithm can be as large as  $(n - 1)/2 = \Theta(n)$ . ■

In the example above, there is a large discrepancy between the diameters of the overlay network produced by GM and the optimal solution for MinMax ODA. We now present another example where GM performs poorly with respect to the maximum degree, but where the diameters of the two respective solutions are the same.

Consider the example where we have  $n^2 + n + 1$  nodes  $u, u_i, v_{i,j}, 1 \leq i \leq n, 1 \leq j \leq n$ , and  $n^2$  topics  $T = \{x_{i,j} : 1 \leq i \leq n, 1 \leq j \leq n\}$ . Node  $u$  is interested in all topics in  $T$ , each node  $u_i$  is interested in  $n$  topics  $x_{i,j}$ , and each node  $v_{i,j}$  is interested in the topic  $x_{i,j}, 1 \leq i \leq n, 1 \leq j \leq n$ . A star

network with center node  $u$  can result from the GM algorithm. The maximum degree will be  $n^2 + n$  and the diameter will be 2. The optimal solution is a network with node  $u$  connected to all nodes  $u_i, 1 \leq i \leq n$ , and each node  $u_i$  connected to all nodes  $v_{i,j}, 1 \leq j \leq n$ . The maximum degree will be  $n + 1$ , and the diameter will be 2.

#### 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 that would result from the addition of  $(u, v)$  to the current overlay network—among the ones that minimally increase the maximum degree of the current graph is added to the edge set of the overlay network. Let  $NC(V, E)$  denote the total number of topic connected components in the overlay network given by  $(V, E)$ .

---

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

---

```

1: OverlayEdges  $\leftarrow \emptyset$ 
2:  $V \leftarrow$  Set of all nodes
3:  $G'(V, E') \leftarrow$  Complete graph on  $V$ 
4: for  $\{u, v\} \in E'$  do
5:    $w\{u, v\} \leftarrow$  Number of topics that both of nodes  $u$  and
      $v$  subscribe to
6: end for
7: while  $G(V, \text{OverlayEdges})$  is not topic-connected do
8:   Find maximum-weighted edge  $e$  on  $G'(V, E', w)$ 
     among the ones which increase the maximum degree
     of  $G(V, \text{OverlayEdges})$  minimally.
9:    $\text{OverlayEdges} = \text{OverlayEdges} \cup \{e\}$ 
10:   $E' \leftarrow E' - e$ 
11:  for  $\{u, v\} \in E'$  do
12:     $w\{u, v\} \leftarrow NC(V, \text{OverlayEdges})$ 
       $-NC(V, \text{OverlayEdges} \cup \{u, v\})$ 
13:  end for
14: end while
15: Discard all weight 0 edges from  $\text{OverlayEdges}$ .
```

---

Steps 1–6 of the 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  $\text{OverlayEdges}$ ). Initially, this amount will be equal to the number of topics that nodes  $u$  and  $v$  have in common.

The MinMax-ODA algorithm uses the GM algorithm with an additional constraint added to the edge selection process in the GM algorithm. We choose the maximum weighted edge among the ones that increase the maximum degree minimally. While at a first glance the MinMax-ODA algorithm may look very similar to the GM algorithm, it actually can be shown to work in phases. In each phase, a collection of edges that forms

a matching of the nodes in the pub/sub system and that also connects close to the maximum number of connected components for the different topics is selected, as we will explain. Basically, we show that the first  $n/2$  edges chosen by our algorithm form a matching, and the next  $n/2$  edges chosen by our algorithm forms a matching, etc.

At each iteration of the while loop, a maximum weight edge among the ones that 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  $\text{OverlayEdges}$  can either increase the maximum degree by 1 or not increase it at all. For ease of explanation, assume for now 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), the MinMax-ODA algorithm 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 MinMax-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 the 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 [12, 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, the 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 the MinMax-ODA has maximum node degree within a factor of  $O(\log(\sum_{v \in V} |\{t \in T | I(v, t) = 1\}|)) = O(\log n + \log |T|)$  from the minimum possible maximum node degree for any topic-connected overlay network on  $V$  with topic interest assignment  $I$ .

*Proof:* At a high level, the proof follows the general structure of 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 [12]). However, before we can apply the set cover framework, we first need to carefully show that

the 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 partition the edge set  $E$  into  $k = d + 1$  matchings  $M_i, 1 \leq i \leq k$ .

*Proof:* We can color the edges of any graph with  $d + 1$  colors such that any two adjacent edges will have different colors by Vizing’s Edge Coloring Theorem [14, Theorem 5.3.2]. 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  $M_i, 1 \leq i \leq k$ .

At the beginning of the algorithm, the total number of connected components is  $C_{start} = \sum_{v \in V} |\{t \in T | I(v, t) = 1\}|$  (i.e., there exists a singleton component for node  $v$  and topic  $t$  such that  $v$  is subscribed to  $t$ ) and at the end  $C_{end} = |\{t | t \in T \text{ and } \exists v \in V \text{ such that } I(v, t) = 1\}|$  (i.e., there exists exactly one component for each topic  $t$  in the system). 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., for each  $t$  such that there exists some  $v$  with  $I(v, t) = 1$ ), making the overlay network 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 that is not subscribed to any topic without affecting the final solution obtained by the MinMax-ODA; 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 that increase the maximum degree of the current graph minimally is added to the set of overlay edges. At the 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 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  $SA_i = S_1 \cup S_2 \cup \dots \cup S_{i-1}$  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 decreases the current total number of connected components by at least one third of the maximum possible amount. We say that a matching is *optimal* with respect to the current configuration of  $G$  if it reduces the number of connected components in  $G$  by the maximum possible amount.

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

*Proof:* Let  $P$  be the edge set of an optimal matching for  $G(V, SA_i)$ , which reduces the number of connected compo-

nents of  $G(V, SA_i)$  by  $c$ . Let  $S_i = \{e_1, e_2, \dots, e_j\}$ , where  $e_l = u_l v_l$ , for  $1 \leq l \leq j$ , and where  $a < b$  implies that  $e_a$  is found before  $e_b$  by our algorithm. Assume that  $S_i$  reduces the total number of connected components of the  $G(V, SA_i)$  by  $c'$ . Let  $G_0 = G(V, SA_i)$  and  $G_l = G_{l-1} \cup e_l$ , for  $1 \leq l \leq j$ . We use  $y_l$  to denote the reduction in the number of connected components incurred by the addition of  $e_l$  to  $G_{l-1}$ . Then

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

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

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  $u_{l'} \text{ or } v_{l'} 1 \leq l' \leq l - 1$ . 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  $S_i$  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 either of these two edges at that step, but chose  $e_l$  instead,  $0 \leq l \leq j$ . Since our algorithm greedily chooses 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  $y_l \geq x_l$ . Hence

$$\begin{aligned} y_l &\geq \frac{x_l}{2}, \quad 1 \leq l \leq j \\ \Rightarrow \sum_{1 \leq l \leq j} y_l &\geq \frac{1}{2} \sum_{1 \leq l \leq j} x_l. \end{aligned} \tag{3}$$

Since  $P_{l+1} = P_l - X_{l+1}$  and  $G_{l+1} = G_l \cup e_{l+1}, 0 \leq l \leq j - 1$ , the reduction by  $c_l$  on the number of connected components of  $G_l$  incurred by the addition of  $P_l$  is smaller than the sum of  $c_{l+1}$ , the reduction on the number of connected components in  $G_{l+1}$  incurred by  $P_{l+1}$ , and  $x_{l+1}$ , the reduction on the number of connected components of  $G_l$  due to the addition of  $X_{l+1}$ , and  $y_{l+1}$ , the reduction on the total number of connected components of  $G_l$  by the addition of  $e_{l+1}$ . Hence

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

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

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

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

From inequalities (3) and (6), we have that

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

From inequalities (1) and (7), we finally obtain that

$$c' \geq c/3. \tag{8}$$

■

Before the 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  $C_{\text{end}}$ . Therefore, there exists a matching  $M_j - SA_i$  that decreases the total number of connected components by at least  $(n_i - C_{\text{end}})/k$ . 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 one third of this amount. Therefore

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

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

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

implying that the degree of any node is at most a factor of  $\Theta(\log(\sum_{v \in V} |\{t \in T | I(v, t) = 1\}|))$  away from  $d_{\text{opt}}$ . ■

## VI. HARDNESS OF MINMAX-TCO PROBLEM

In this section, we will show that no polynomial-time algorithm can approximate the MinMax-TCO problem within less than a logarithmic factor (unless  $P = NP$ ). This is a tighter bound than the result in [20], where we showed that no constant-approximation polynomial-time algorithm was possible for this problem.

The proof below follows the general lines of a proof presented for [12, Theorem 5.3], which was adapted to handle the maximum node degree (instead of the average node degree as in [12]) and to show a logarithmic lower bound on the approximation factor for the MinMax-TCO problem unless  $P = NP$  (in [12], Theorem 5.3 proved a weaker, constant approximation lower bound on the average node degree).

*Theorem 2:* There exists no polynomial-time algorithm that approximates the MinMax-TCO problem within less than a logarithmic factor,  $\log |T|$ , unless  $P = NP$ .

*Proof:* The proof follows in general lines the Proof of [12, Theorem 5.3]. In fact, a more careful observation of the Proof of [12, Theorem 5.3] shows that the proof basically shows the hardness of approximation of the maximum degree, and not the average degree, as claimed in [12].

For the sake of completeness, we present here the sketch of the Proof of [12, Theorem 5.3] when directly applied to the 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  $V, T, I$ , and a node  $v \in V$ , connect the nodes in  $V$  into an overlay network  $G$  that has least possible degree for node  $v$  and that is topic-connected.

We can prove that SN-MinMax-TCO is NP-hard by reducing the minimum set cover problem to this problem. We then 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 [12, Theorem 5.3]. Assume that there is an algorithm A that approximates the MinMax-TCO problem within a factor of  $c$ . We will show that we can use this algorithm A to find a  $c$ -approximation to the minimum set cover problem.

Given an instance SN-MinMax-TCO( $V, T, I, v$ ), construct the corresponding instance MinMax-TCO( $V', T', I'$ ) following the construction in the Proof of [12, Lemma 5.1].

Let  $d_{\text{opt}}$  denote the maximum degree of an optimal solution (that is,  $d_{\text{opt}}$  denotes the degree of node  $v$  in an optimal solution) for this instance of the SN-MinMax-TCO problem. As shown in the Proof of [12, Lemma 5.1], the corresponding instance MinMax-TCO( $V', T', I'$ ) also has a solution with degree  $d_{\text{opt}}$ . Hence, algorithm A will find a solution to MinMax-TCO( $V', T', I'$ ) problem with degree at most  $c * d_{\text{opt}}$ . Using this solution, we can construct a solution to SN-MinMax-TCO( $V, T, I, v$ ) with degree at most  $c * d_{\text{opt}}$  as in [12, Lemma 5.1]. Thus, we have a  $c$ -approximation algorithm B for the SN-MinMax-TCO problem.

In the minimum set cover problem, we are given an instance  $(U, \mathcal{S})$ , where  $U$  is a set of elements and  $\mathcal{S}$  is a collection of subsets of  $U$ , and we would like to find  $\mathcal{C} \subseteq \mathcal{S}$  of smallest possible cardinality such that  $\cup_{x \in \mathcal{C}, C \in \mathcal{C}} x = U$ . Given an instance of the minimum set cover problem  $(U, \mathcal{S})$ , construct the corresponding SN-MinMax-TCO( $V, T, I, v$ ) instance following the construction as in the Proof of [12, Lemma 5.2]. Let  $s_{\text{opt}}$  denote an optimal solution for this instance of the minimum set cover problem. As shown in the Proof of [12, Lemma 5.2], the corresponding SN-MinMax-TCO( $V, T, I, v$ ) instance also has a solution with degree  $s_{\text{opt}}$ . Hence, algorithm B will find a solution to SN-MinMax-TCO( $V, T, I, v$ ) with degree at most  $c * s_{\text{opt}}$ .

Using this solution, we can construct a solution to the minimum set cover problem with cardinality at most  $c * s_{\text{opt}}$  as in [12, Lemma 5.2]. Thus, we have a  $c$ -approximation algorithm C for the minimum set cover problem. Given that we cannot approximate the minimum set cover within an  $o(\log |T|)$ -approximation factor [24], unless  $P = NP$ ,  $c = \Omega(\log |T|)$ , and the claim is proved. ■

Since it is trivial to show that MinMax-TCO is in NP, we have the following.

*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 the MinMax-ODA improves the maximum degree of the overlay network drastically at the cost of a small increase on the average degree and diameter.

### A. Maximum Node Degree

For these experiments, the number of nodes varies between 1000–10 000. In the first experiment (Fig. 2), the number of topics is 100, and in the second experiment (Fig. 3), the number of topics is 200. We fixed the number of subscriptions to  $s = 10$ .

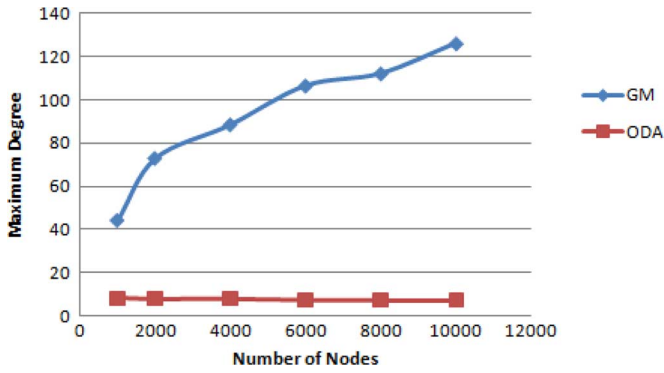


Fig. 2. Maximum node degree for GM and MinMax-ODA (number of topics is 100).

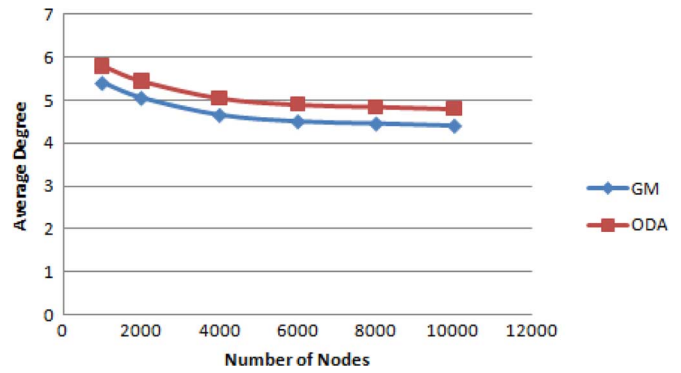


Fig. 4. Average node degree for GM and MinMax-ODA (number of topics is 100).

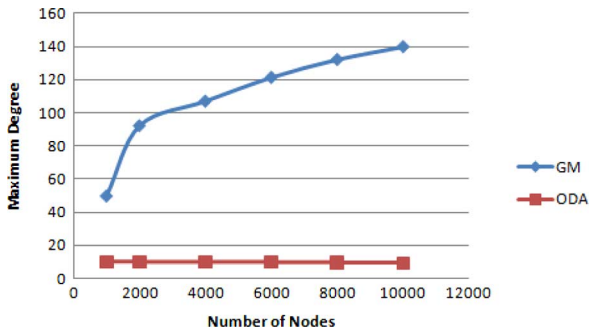


Fig. 3. Maximum node degree for GM and MinMax-ODA (number of topics is 200).

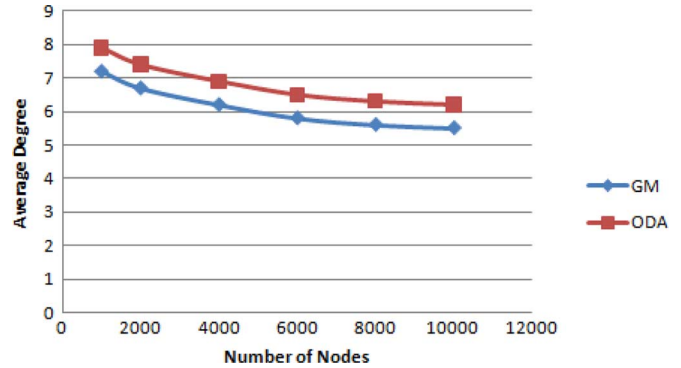


Fig. 5. Average node degree for GM and MinMax-ODA (number of topics is 200).

Each node is interested in each topic uniformly at random. This experimental setting is similar to previous studies [12].

Fig. 2 gives the comparison results for the GM and the MinMax-ODA algorithms for the maximum degree metric. The maximum degree of the graph decreases for the MinMax-ODA algorithm when the number of nodes increases since the MinMax-ODA algorithm can find more 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. The reason is a randomness of the subscription patterns. When the number of nodes increases, the chance for a node to find many neighbors with a big interest overlap increases. Basically as the number of nodes increases, the 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 the GM algorithm and the MinMax-ODA algorithm, the MinMax-ODA algorithm outperforms the GM algorithm by a factor of 10 on average (Fig. 2).

The same results are valid for Fig. 3. When we compare Figs. 2 and 3, the maximum node degree increases slightly for both the GM algorithm and the MinMax-ODA algorithm since edges will have less correlation when we increase the number of topics.

### B. Average Node Degree

We use the same experimental setting as in Section VII-A. Fig. 4 is a comparison of the GM algorithm and the MinMax-ODA algorithm for the average degree metric.

The average degree of the graph decreases for both the GM and the MinMax-ODA algorithms when the number of nodes increases since both algorithms can find edges with higher correlation. When we compare the results of the GM and the MinMax-ODA algorithms, the GM algorithm is slightly better than the MinMax-ODA algorithm, by 9% on average (Fig. 4). Similar results are valid for Fig. 5.

### C. Diameter

We use the same experimental setting as in Section VII-B. The number of topics is 100. Fig. 6 is a comparison of the GM algorithm and the MinMax-ODA algorithm for the diameter metric. The diameter of the graph increases for both the GM and the MinMax-ODA algorithms when the number of nodes increases since we have more nodes subscribed to each topic. When we compare the results of the GM and the MinMax-ODA algorithms, the GM algorithm is slightly better than the MinMax-ODA algorithm, by 30% on average (Fig. 6).

### D. Subscription Size

In these experiments, the number of nodes and the number of topics are fixed to 100. The subscription size varies between 10 and 50. Each node is interested in each topic uniformly at random. Fig. 7 is the comparison of the GM and the MinMax-ODA algorithms for the maximum degree metric. When the subscription size increases, the maximum degree of the overlay network decreases for the MinMax-ODA algorithm since the MinMax-ODA algorithm can find edges with higher

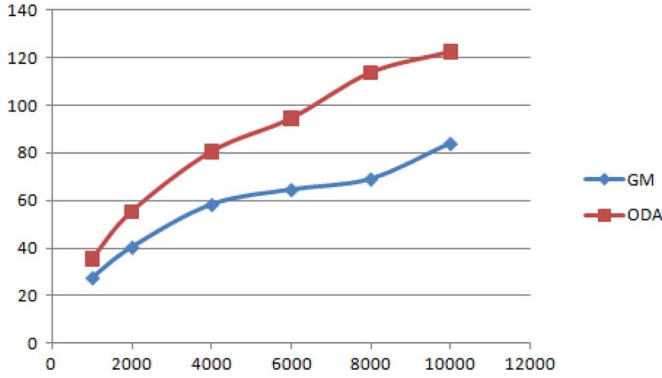


Fig. 6. Diameter for GM and MinMax-ODA.

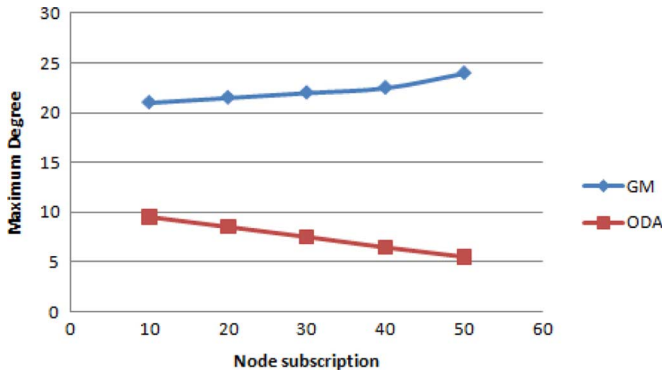


Fig. 7. Maximum node degree for different subscription sizes.

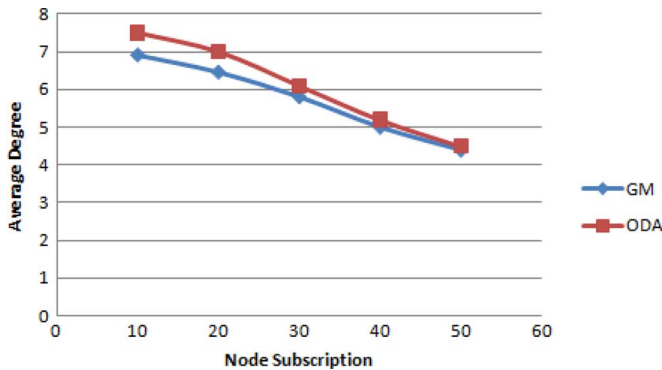


Fig. 8. Average node degree for different subscription sizes.

correlation. When the subscription size increases, the maximum degree of the overlay network increases for the GM algorithm. When we compare the current experimental results for the GM and the MinMax-ODA algorithms, the maximum degree obtained by the MinMax-ODA is a factor of 4 less than the maximum degree obtained by the GM algorithm, on average (Fig. 2).

Fig. 8 gives the comparison results for the GM and the MinMax-ODA algorithms for the average degree metric. The average degree of the overlay network decreases for both GM and the MinMax-ODA algorithms as the subscription size increases since both algorithms can find edges with higher correlations. When we compare the average degree obtained by the GM and the MinMax-ODA algorithms, the GM performs

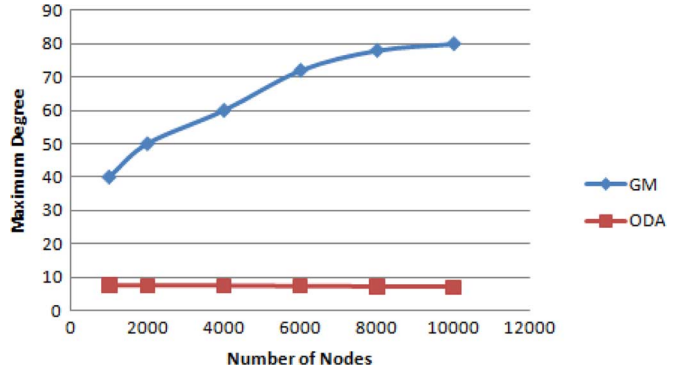


Fig. 9. Maximum node degrees for Zipf distribution.

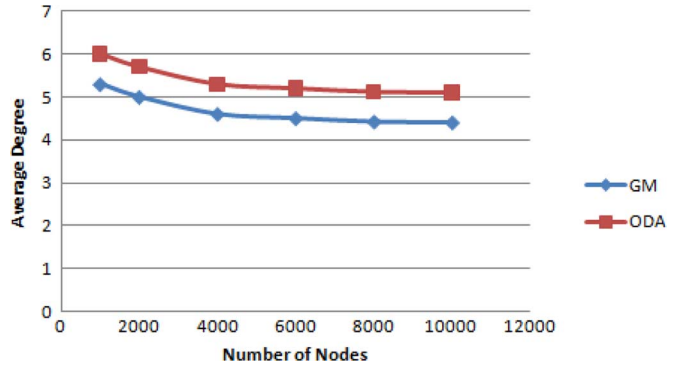


Fig. 10. Average node degrees for Zipf distribution.

slightly better than the MinMax-ODA—by close to 10% on average (Fig. 8).

### E. Zipf Distribution

For these experiments, the number of nodes varies between 1000 and 10000. The number of topics is fixed at 100, and the subscription size at  $s = 10$ . Each node subscribes to each topic  $t_i$  with probability  $p_i$ . Every time a node does not get exactly 10 subscriptions, we regenerate the node (i.e., we reassign topics to the node according to the probabilities  $p_i$ ). The value of  $p_i$  is distributed according to a Zipf distribution with  $\alpha = 0.5$  ( $p_i \propto 1/i^\alpha$ ). This experimental setting is similar to previous studies (e.g., [12]). Fig. 9 presents a comparison of the GM and MinMax-ODA algorithms for the maximum degree metric, while Fig. 10 is a comparison of the GM and MinMax-ODA algorithms for the average degree metric. The maximum degree obtained by the MinMax-ODA algorithm is on average less than 1/7 of the maximum degree obtained by the GM algorithms, at the expense of a small increase on the average degree.

## VIII. CONSTRUCTING CONSTANT DIAMETER OVERLAYS FOR PUBLISH-SUBSCRIBE

In this section, we study a variation of the low-degree topic-connected problems we considered so far, where we build a constant diameter overlay network for publish/subscribe communication with many topics [22]. We present three overlay network construction heuristics that guarantee constant diameter and topic-connectivity, which are important factors for efficient routing. We now formally define the problem.



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

This problem aims at minimizing the average degree, as does the MinAv-TCO problem introduced by Chockler *et al.* [12], with the additional requirement on the diameter. We focus on the case  $c = 2$  and present three heuristics for the 2-CD-TCO problem, which are validated via experimental results.

We first show that the GM algorithm and the MinMax-ODA algorithm do not work well for the 2-CD-TCO problem. Second, we present our 2-diameter algorithms and their performance evaluations.

In the 2-CD-TCO problem, there are few topologies (of diameter at most 2) that each topic-connected component can have. One such topology, which also minimizes the number of edges required for connectivity in each component, is the star topology. Hence, in all of our heuristics, we assume that the topology of each 2-diameter component that emerges is a star topology.

**A. GM Algorithm and the MinMax-ODA Algorithm for the CD-TCO Problem**

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

*Lemma 6:* The GM algorithm and the MinMax-ODA algorithm can produce an overlay network of diameter  $\Theta(n)$ , where  $n$  is number of nodes in the pub/sub system.

*Proof:* Consider the example where we have  $n$  nodes  $v_1, v_2, \dots, v_n$ , one topic  $t$ , and every node is interested in  $t$ . There exist 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  $n - 1$ . The MinMax-ODA algorithm will also result in this network. 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. Examples I and II**

In Fig. 11, each node is subscribed to topics 1 and 2. In an optimal solution, we have  $n - 1$  edges and we have only one star. This example illustrates the intuition that we should have a minimum number of stars that satisfy all connectivity requirements.

*Intuition 1:* Construct minimum number of stars.

In Fig. 12, node  $u$  is subscribed to topics 1–5.  $(n - 1)/4$  nodes are subscribed to topics 1 and 2.  $(n - 1)/4$  nodes are subscribed to topics 2 and 3.  $(n - 1)/4$  nodes are subscribed to topics 3 and 4.  $(n - 1)/4$  nodes are subscribed to topics 4 and 5. In the optimal solution, we have  $n - 1$  edges, and we have only one star

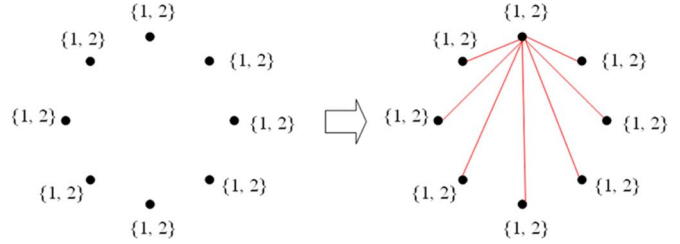


Fig. 11. Example I.

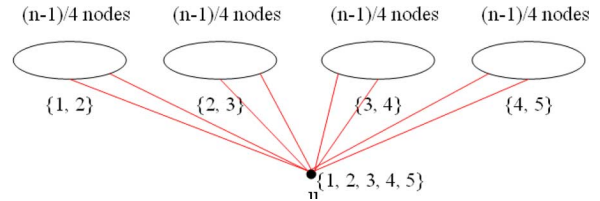


Fig. 12. Example II.

with center node  $u$ . From this example arises the intuition that nodes for which there are many nodes with overlapping interest are good candidates for being the center of stars.

*Intuition 2:* Nodes for which there are many nodes with overlapping interest assignment are good candidates for being the center of stars.

**C. Constant Diameter Overlay Design Algorithm (CD-ODA)**

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

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

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  $I(u, t) = I(v, t) = 1$ . 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  $t \in T$  with  $I(u, t) = I(v, t) = 1$ .
- 6:   Remove all topics  $t$  from  $T$  such that  $I(u, t) = 1$ .
- 7: **end while**

CD-ODA finds the optimal solution for Examples I and II in Fig. 11.

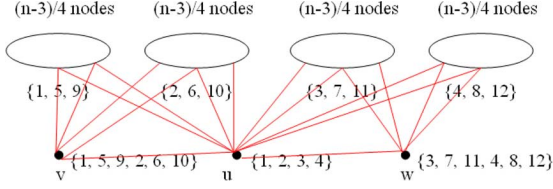


Fig. 13. CD-ODA on Example III.

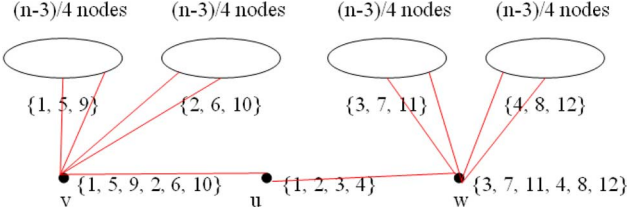


Fig. 14. Optimal solution for Example III.

#### D. Example III

In Fig. 13, node  $u$  is subscribed to topics 1–4. Node  $v$  is subscribed to topics 1, 5, 9, 2, 6, and 10. Node  $w$  is subscribed to topics 3, 7, 11, 4, 8, and 12.  $(n-3)/4$  nodes are subscribed to topics 1, 5, and 9.  $(n-3)/4$  nodes are subscribed to topics 2, 6, and 10.  $(n-3)/4$  nodes are subscribed to topics 3, 7, and 11.  $(n-3)/4$  nodes are subscribed to topics 4, 8, and 12. For this example, CD-ODA first puts edges between node  $u$  and all the other nodes, and then it puts edges between node  $v$  and the first two sets of  $(n-3)/4$  nodes, and it puts edges between node  $w$  and the last two sets of  $(n-3)/4$  nodes. Thus, CD-ODA uses  $(n-1) + (n-3)/2 + (n-3)/2 = (2n-4)$  edges.

In the optimal solution (Fig. 14), there are edges between node  $v$  and the first two sets of  $(n-3)/4$  nodes, there are edges between node  $w$  and the last two sets of  $(n-3)/4$  nodes, and there are edges  $(u, v)$  and  $(u, w)$ . In the optimal solution, only  $(n-3)/2 + (n-3)/2 + 2 = (n-1)$  edges are required. Let us define the number of weighted neighbors of a node  $z$  as

$$w_z = \sum_{t \in T} |\{v \in V \mid I(v, t) = I(z, t) = 1\}|.$$

This example illustrates the intuition that nodes with many weighted neighbors are good candidates for being the center of stars.

*Intuition 3:* Nodes with many weighted neighbors are good candidates for the center of stars.

Given examples I–III in Figs. 11, 12, and 14, respectively, and intuitions 1–3, we designed the CD-ODA-I algorithm. CD-ODA-I finds the optimal solution for Example III in Fig. 14.

#### E. Constant Diameter Overlay Design Algorithm I (CD-ODA-I)

CD-ODA-I starts with the overlay network  $G(V, \emptyset)$ . At each iteration of the CD-ODA-I, a node  $u$  that has the maximum number of weighted neighbors is chosen.

We add an edge between  $u$  and each of its neighbors and then remove the topics in this node's interest assignment from the set of topics.

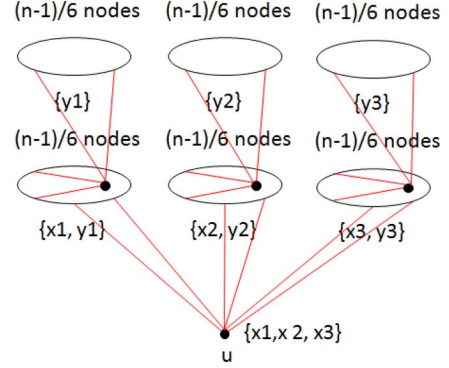


Fig. 15. CD-ODA and CD-ODA-I in Example IV.

#### Algorithm 3: Constant Diameter Overlay Design Algorithm I (CD-ODA-I)

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

We can also find an example that CD-ODA performs better than CD-ODA-I. Consider the following example. We have  $n$  nodes  $u, v_1, v_2, w, u_1, u_2, \dots, u_{n-4}$ . Nodes  $v_1$  and  $v_2$  are subscribed to topics  $x_1, x_2, \dots, x_n$ . Nodes  $u_1, u_2, \dots, u_{n-4}$  are subscribed to topics  $x_1, y$ . Node  $u$  is subscribed to topics  $x_1, y, z$ . Node  $w$  is subscribed to topic  $z$ . For this example, CD-ODA-I first puts edges between node  $v_1$  and nodes  $u, v_2, u_1, u_2, \dots, u_{n-4}$ , and then it puts edges between node  $u$  and nodes  $w, u_1, u_2, \dots, u_{n-4}$ . Thus, CD-ODA-I uses  $(n-2) + (n-3) = 2n-5$  edges. For this example, CD-ODA puts edges between node  $u$  and nodes  $v_1, v_2, w, u_1, u_2, \dots, u_{n-4}$ , and then it puts edges between nodes  $v_1$  and  $v_2$ . Thus, CD-ODA uses only  $(n-1) + 1 = n$  edges.

#### F. Example IV

In Fig. 15, node  $u$  is subscribed to topics  $x_1, x_2, x_3$ .  $(n-1)/6$  nodes (set  $B_1$ ) are subscribed to  $y_1$ ,  $(n-1)/6$  nodes (set  $B_2$ ) are subscribed to  $y_2$ , and  $(n-1)/6$  nodes (set  $B_3$ ) are subscribed to  $y_3$ .  $(n-1)/6$  nodes (set  $A_1$ ) are subscribed to  $x_1, y_1$ ,  $(n-1)/6$  nodes (set  $A_2$ ) are subscribed to  $x_2, y_2$ , and  $(n-1)/6$  nodes (set  $A_3$ ) are subscribed to  $x_3, y_3$ . For this example, CD-ODA and CD-ODA-I first put edges between node  $u$  and all the nodes in sets  $A_1, A_2, A_3$ , and then it puts edges between nodes  $u_i \in A_i$  and all nodes in  $A_i$  and  $B_i$ , for  $1 \leq i \leq 3$ . Thus, CD-ODA and CD-ODA-I use  $((n-1)/6) * 3 + ((n-1)/3 - 1) * 3 = 3 * (n-1)/2 - 3$  edges.

In the optimal solution (Fig. 16), there are edges between nodes  $u_i \in A_i$  and all nodes in  $A_i$  and  $B_i$ , for  $1 \leq i \leq 3$ .

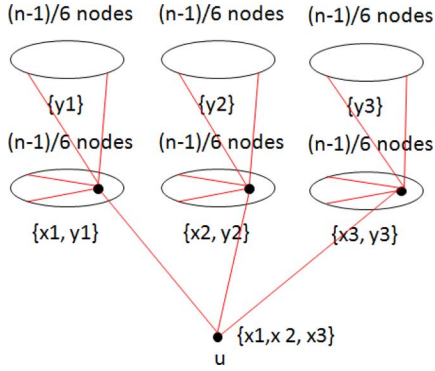


Fig. 16. Optimal solution for Example IV.

There are edges between node  $u$  and nodes  $u_i$ , for  $1 \leq i \leq 3$ . In the optimal solution, only  $(3 + ((n-1)/3 - 1) * 3 = (n-1))$  edges are required. Let us define the connection density of a node  $u$ ,  $d_u$ , as

$$d_u = \frac{w_u}{n_u}.$$

From this example, we can have the intuition that nodes with dense connections are good candidates for the center of stars.

*Intuition 4:* Nodes with dense connections are good candidates for the center of stars.

#### G. Constant Diameter Overlay Design Algorithm II (CD-ODA-II)

CD-ODA-II also starts with the overlay network  $G(V, \emptyset)$ . At each iteration of the CD-ODA-II, a node  $u$  that has maximum connection density  $d_u$  is chosen.

We add an edge between a node  $u$  with maximum density and each of its neighbors and then remove the topics in this node's interest assignment from the set of topics.

---

#### Algorithm 4: Constant Diameter Overlay Design Algorithm II (CD-ODA-II)

---

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

Given the example in Fig. 15 and Intuition 4, we designed the CD-ODA-II algorithm. CD-ODA-II finds an optimal solution for Example IV in Fig. 16.

We can also find an example that CD-ODA and CD-ODA-I perform better than CD-ODA-II. Consider the following example. We have  $2n + 1$  nodes  $w, v_1, v_2, \dots, v_n, u_1, u_2, \dots, u_n$ . For all  $1 \leq i \leq n$ , nodes  $v_i$  are subscribed to three topics  $x, y, z_i$ . For all  $1 \leq i \leq n$ , nodes  $u_i$  are subscribed to the topic  $t$ . Node  $w$  is subscribed to all topics. For this example,

CD-ODA-II first puts edges between nodes  $v_1$  and  $v_i$  for all  $2 \leq i \leq n$ . It puts an edge between nodes  $v_1$  and  $w$ . Then, it puts edges between node  $w$  and nodes  $v_i$  for all  $2 \leq i \leq n$ . It puts edges between node  $w$  and nodes  $u_i$  for all  $1 \leq i \leq n$ . Thus, CD-ODA-II uses  $n + 2n - 1 = 3n - 1$  edges. For this example, CD-ODA and CD-ODA-I puts edges between node  $w$  and all the other nodes. Thus, CD-ODA and CD-ODA-I use only  $2n$  edges.

#### H. Analysis of Algorithms

In this section, we show that all constant diameter algorithms presented (CD-ODA, CD-ODA I, CD-ODA II) terminate in  $O(|V|^2 * |T|^2)$  time-steps and generate a 2-diameter overlay for each topic.

*Lemma 7:* CD-ODA, CD-ODA-I, and CD-ODA-II terminate within  $O(|V|^2 * |T|^2)$  time.

*Proof:* In  $O(|V|^2 * |T|)$  time, we can find the total weight of the neighbors or the connection density for each node. We choose a node  $u$  with maximum weight/density. We remove all the topics to which node  $u$  is subscribed. We repeat this process until all topics have been considered. At each iteration, at least one topic is removed. Thus, the algorithm will take at most  $O(|V|^2 * |T|) * |T| = O(|V|^2 * |T|^2)$  time-steps. ■

*Lemma 8:* CD-ODA, CD-ODA-I, and CD-ODA-II generate a 2-diameter overlay for each topic.

*Proof:* Since any of the algorithms generates a star for each topic, each topic overlay network will have diameter 2. ■

#### I. Conjectures

*Conjecture 1:* All three algorithms approximate 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 a 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 the constant diameter overlay design problem within less than a logarithmic factor unless  $P = NP$ .

Our intuition for Conjecture 2 is that the set cover problem can probably be reduced to this problem in a similar fashion as it was done for the MinMax-TCO and MinAv-TCO problems.

#### J. Experimental Results

The GM algorithm [12] and our algorithms are implemented in Java. These algorithms are compared according to the average degree in the resulting graph. The diameter is always 2 for our algorithms, and it may be  $\Theta(n)$ , where  $n$  is the number of nodes, for the GM algorithm. When we compare the results of the GM algorithm and our algorithms according to the average degree, our algorithms require at most 2.3 times more edges than the GM algorithm, while improving the diameter drastically.

1) *Average Node Degree With Varying Number of Nodes:* For the first experiment, the number of nodes varies between

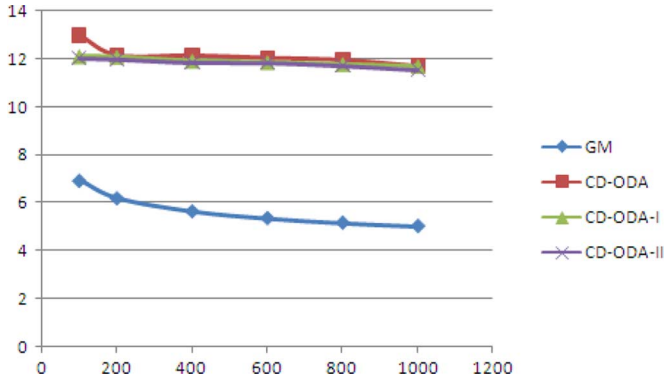


Fig. 17. Average node degree for GM, CD-ODA, CD-ODA-I, and CD-ODA-II.

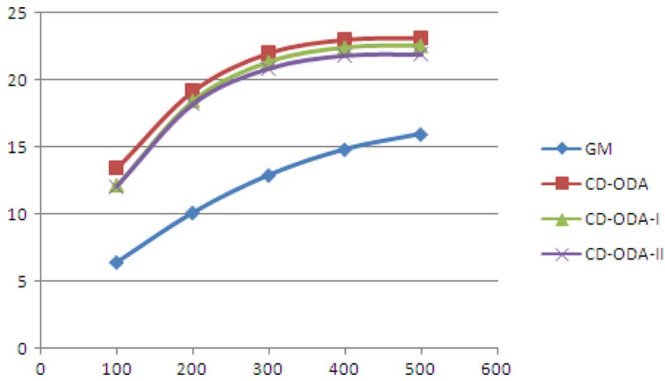


Fig. 18. Average node degree for GM, CD-ODA, CD-ODA-I, and CD-ODA-II.

100–1000. We fixed the number of topics at 100, and the number of subscriptions at  $s = 10$ . Each node is interested in each topic uniformly at random. This experimental setting is similar to previous studies [12], [20], [23].

Fig. 17 gives the comparison results for the GM and our algorithms according to the average degree. The average degree of the graph decreases for the GM algorithm when the number of nodes increases since there are more edges with higher correlation. The average degree of the graph slightly decreases for our algorithms. When we compare the results of GM and CD-ODA-II, our algorithm requires at most 2.3 times more edges than GM (Fig. 17). CD-ODA-II improves CD-ODA by a factor of 3% on average, and CD-ODA-I by a factor of 2% on average.

2) *Average Node Degree With Varying Number of Topics:* For the second experiment, the number of nodes is 100, the number of topics varies between 100 and 500, and the number of subscriptions is fixed at  $s = 20$ . Each node is interested in each topic uniformly at random. This experimental setting is also similar to previous studies [12], [23].

Fig. 18 gives the comparison results for the GM and our algorithms for the average degree metric. The average degree of the graph increases for our algorithms and the GM algorithm as the number of topics increases since the edges will have lower correlations. When we compare the results of GM and CD-ODA-II, our algorithm requires at most 1.9 times more edges than the GM (Fig. 18). CD-ODA-II improves CD-ODA by a factor of 6% on average, and CD-ODA-I by a factor of 2% on average.

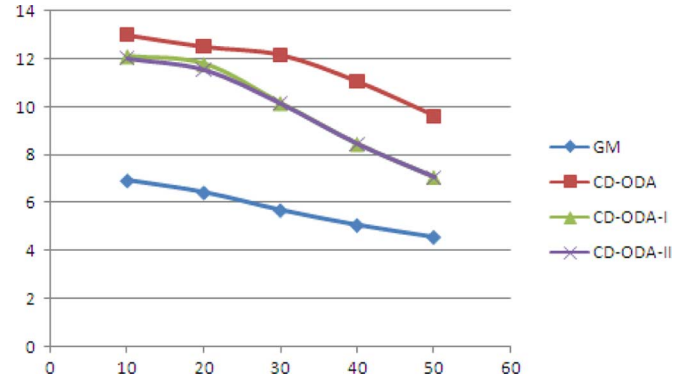


Fig. 19. Average node degree for GM, CD-ODA, CD-ODA-I, and CD-ODA-II.

3) *Average Node Degree With Varying Subscription Size:* For the third experiment, the number of nodes and the number of topics are fixed at 100. The subscription size varies between 10 and 50. Each node is interested in each topic uniformly at random. (The experimental setting is also similar to previous studies [12], [23].)

Fig. 19 gives the comparison results for the GM and our algorithms for the average degree metric. The average degree of the overlay network decreases for both GM and our algorithms when the subscription size increases since all algorithms can find edges with higher correlation. When we compare the results of GM and CD-ODA-II, our algorithm requires at most 1.8 times more edges than the GM (Fig. 19). CD-ODA-II improves CD-ODA by a factor of 20% on average, and CD-ODA-I by a factor of 1% on average.

## IX. CONCLUSION

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) that approximates the MinMax-TCO problem within a logarithmic factor. We also show that the approximation factor of the MinMax-ODA is almost tight since no less than logarithmic-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 the GM algorithm is used.

We present three heuristics 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 that achieves a “good” tradeoff 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 as 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*. Redwood Shores, CA: , Oracle.
- [2] M. Altherr, M. Erzberger, and S. Maffeis, "iBusa software bus middleware for the Java platform," in *Proc. Int. Workshop Rel. Middleware Syst.*, 1999, pp. 43–53.
- [3] E. Anceaume, M. Gradinariu, A. K. Datta, G. Simon, and A. Virgillito, "A semantic overlay for self-peer-to-peer publish/subscribe," in *Proc. 26th IEEE ICDCS*, 2006, p. 26.
- [4] S. Baehni, P. T. Eugster, and E. Guerraoui, "Data-aware multicast," in *Proc. DSN*, 2004, pp. 233–242.
- [5] R. Baldoni, R. Beraldi, V. Quema, L. Querzoni, and S. T. Piergiovanni, "TERA: Topic-based event routing for peer-to-peer architectures," in *Proc. 1st ACM DEBS*, 2007, pp. 2–13.
- [6] R. Baldoni, R. Beraldi, L. Querzoni, and A. Virgillito, "Efficient publish/subscribe through a self-organizing broker overlay and its application to SIENA," *Comput. J.*, vol. 50, no. 4, pp. 444–459, 2007.
- [7] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable application layer multicast," *SIGCOMM Comput. Commun. Rev.*, vol. 32, no. 4, pp. 205–217, 2002.
- [8] S. Bholra, R. Strom, S. Bagchi, Y. Zhao, and J. Auerbach, "Exactly-once delivery in a content-based publish-subscribe system," in *Proc. DSN*, 2002, pp. 7–16.
- [9] A. Carzaniga, M. J. Rutherford, and A. L. Wolf, "A routing scheme for content-based networking," in *Proc. 23rd Annu. IEEE INFOCOM*, Hong Kong, Mar. 2004, vol. 2, pp. 918–928.
- [10] M. Castro, P. Druschel, A. M. Kermarrec, and A. Rowstron, "SCRIBE: A large-scale and decentralized application-level multicast infrastructure," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 8, pp. 1489–1499, Oct. 2002.
- [11] 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*. New York: Springer-Verlag, 2005, vol. 3648, pp. 1194–1204.
- [12] G. Chockler, R. Melamed, Y. Tock, and R. Vitenberg, "Constructing scalable overlays for pub-sub with many topics," in *Proc. 26th ACM PODC*, 2007, pp. 109–118.
- [13] G. Chockler, R. Melamed, Y. Tock, and R. Vitenberg, "SpiderCast: A scalable interest-aware overlay for topic-based pub/sub communication," in *Proc. 1st ACM DEBS*, 2007, pp. 14–25.
- [14] R. Diestel, *Graph Theory*, 2nd ed. New York: Springer-Verlag, 2000.
- [15] P. T. Eugster, P. A. Felber, R. Guerraoui, and A. M. Kermarrec, "The many faces of publish/subscribe," *Comput. Surveys*, vol. 35, no. 2, pp. 114–131, 2003.
- [16] R. Guerraoui, S. Handurukande, and A. M. Kermarrec, "Gossip: A gossip-based structured overlay network for efficient content-based filtering," EPFL, Lausanne, Tech. Rep. IC/2004/95, 2004.
- [17] B. Korte and J. Vygen, *Combinatorial Optimization Theory and Algorithms*, 2nd ed. New York: Springer-Verlag, 2000.
- [18] R. Levis, *Advanced Massaging Applications With MSMQ and MQSeries*. Indianapolis, IN: QUE, 1999.
- [19] H. Liu, V. Ramasubramanian, and E. G. Sirer, "Client behavior and feed characteristics of RSS, a publish-subscribe system for web micronews," in *Proc. IMC*, Berkeley, CA, 2005, pp. 29–34.
- [20] M. Onus and A. W. Richa, "Minimum maximum degree publish-subscribe overlay network design," in *Proc. 28th Annu. IEEE INFOCOM*, Rio de Janeiro, Brazil, 2009, pp. 882–890.
- [21] M. Onus and A. W. Richa, "Brief announcement: Parameterized maximum and average degree approximation in topic-based publish-subscribe overlay network design," in *Proc. 21st ACM SPAA*, Calgary, AB, Canada, 2009, pp. 39–40.
- [22] M. Onus and A. W. Richa, "Parameterized maximum and average degree approximation in topic-based publish-subscribe overlay network design," in *Proc. 30th IEEE ICDCS*, Genoa, Italy, 2010, pp. 644–652.
- [23] V. Ramasubramanian, R. Peterson, and E. G. Sirer, "Corona: A high performance publish-subscribe system for the World Wide Web," in *Proc. 3rd NDSI*, 2006, pp. 15–28.
- [24] R. Raz and M. Safra, "A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP," in *Proc. 29th Annu. ACM STOC*, 1997, pp. 475–484.
- [25] D. Sandler, A. Mislove, A. Post, and P. Druschel, "FeedTree: Sharing Web micronews with peer-to-peer event notification," in *Proc. IPTPS*, 2005, pp. 141–151.
- [26] D. Skeen, "Vitrias publish-subscribe architecture: Publish-subscribe overview," Sunnyvale, CA, 1998 [Online]. Available: <http://www.vitria.com>
- [27] D. Tam, R. Azimi, and H.-A. Jacobsen, "Building content-based publish/subscribe systems with distributed hash tables," in *Proc. 1st ID-BISP2P*, Berlin, Germany, 2003, pp. 138–152.
- [28] "Everything you need to know about middleware: Mission-critical interprocess communication," Talarian Corporation, Los Altos, CA, White paper, 1999 [Online]. Available: <http://www.talarian.com/>
- [29] "TIB/Rendezvous," Tibco, Palo Alto, CA, White paper, 1999.
- [30] Y. Tock, N. Naaman, A. Harpaz, and G. Gershindky, "Hierarchical clustering of message flows in a multicast data dissemination system," in *Proc. 17th IASTED Int. Conf. Parallel Distrib. Comput. Syst.*, 2005, pp. 320–327.
- [31] 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 *Proc. IPTPS*, 2006, pp. 123–128.



**Melih Onus** received the B.S. degree in computer engineering from Bilkent University, Ankara, Turkey, in 2003, and the Ph.D. degree in computer science from Arizona State University (ASU), Tempe, in 2009.

He is currently an Instructor with Bilkent University. His research interests are in the areas of distributed computing, computer networks, and theoretical computer science.



**Andréa W. Richa** received the B.S. degree in computer science and M.S. degree in computer systems from the Federal University of Rio de Janeiro, Rio de Janeiro, Brazil, in 1990 and 1992, respectively, and the M.S. and Ph.D. degrees in computer science from Carnegie Mellon University, Pittsburgh, PA, in 1995 and 1998, respectively.

She has been an Associate Professor in computer science and engineering with Arizona State University, Tempe, since August 2004, having joined as an Assistant Professor in August 1998. For a selected list of her publications, CV, and current research projects, please visit <http://www.public.asu.edu/~aricha>. Her main area of research is in network algorithms. Some of the topics she has worked on include packet scheduling, distributed load balancing, packet routing, ad hoc wireless network clustering and routing, wireless network modeling, and distributed hash tables.

Dr. Richa was the recipient of a National Science Foundation (NSF) CAREER Award in 1999.