International Journal of Management, IT & Engineering
(ISSN: 2249-0558)

CONTENTS

Editorial Board

Title

ACHIEVING OPTIMAL DOS RESISTANT P2P TOPOLOGIES FOR LIVE MULTIMEDIA STREAMING USING COST FUNCTION ALGORITHM

Author(s)

A. L.Srinivasulu
M.Tech (CSE)
Asst Professor
Intel Engineering College,
Anantapur, A. P.

S. Jaya Bhaskar
M.Tech (CSE)
Intel Engineering College,
Anantapur, A. P.

Ms. K. Deepthi
Asst. Professor
Dept. of Comp. Sc. & Engineering
GITAM University,
Hyderabad, A.P

Dr. Sudarson Jena
Associate Professor & Head
Dept. of Computer Science and Application
GITAM University,
Hyderabad, A.P

ABSTRACT:

The peer-to-peer approach for live multimedia streaming applications offers the promise to obtain a highly scalable, decentralized, and robust distribution service. When constructing streaming topologies, however, specific care has to be taken in order to ensure that quality of service requirements in terms of delay, jitter, packet loss, and stability against deliberate denial of service attacks are met. In this paper, we concentrate on the latter requirement of stability against denial-of-service attacks. We present an analytical model to assess the stability of overlay streaming topologies and describe attack strategies. Building on this, we describe topologies, which are optimally stable toward perfect attacks based on global knowledge, and give a mathematical proof of their optimality. The formal construction and analysis of these topologies using global knowledge lead us to strategies for distributed procedures, which are able to construct resilient topologies in scenarios, where global knowledge can not be gathered. Experimental results show that the topologies created in such a real-world scenario are close to optimally stable toward perfect denial of service attacks.

Keywords: Reliability, Fault Resilience, Attack Resilience, Media Streaming, Peer-to-Peer, Overlay.

1. INTRODUCTION:

Due to its scalability, cooperative streaming, sometimes called application layer multicast (ALM) [1], has become increasingly interesting system architecture for live content distribution over the last years. These systems use the resources of end hosts and integrate participants as service proxies for other subscribers. While client-server solutions introduce a bottleneck and single point of failure at the server, cooperative streaming systems gain additional bandwidth and backup resources as the number of participants increases. As content forwarded to the receivers by other participants of the system, each subscriber is dependent on all preceding participants in the overlay path to the original source. In consequence, participants with many successors, which are topologically close to the source, have a higher relevance to the overall system than participants near the leaves of the multicast trees that have fewer successors. Systems designed for an application layer multicast are commonly classified into push- or pull-based approaches [2].Push-based approaches create and maintain an explicit topology for the content dissemination. In pull-based approaches, each node explicitly requests the transfer of Systems of the latter category have to preload the requested parts well in advance of the play-out and their applicability to live streaming in consequence is limited, as this characteristic causes rather high delays [3], [4]. Push-based approaches again are commonly further classified into the categories mesh-first or tree-first [5], [6]. While mesh-first approaches create a management overlay first and set up the content dissemination topology using this mesh, tree first approaches create the content dissemination topologies directly and use them for the distribution of management traffic, as well. However, using an overlay for streaming multimedia from a source to multiple receivers, each packet of the content is distributed along a set of links which connect all participating nodes. These links always form spanning trees, which are rooted at the source of the stream and concise of all participating nodes as either inner- or leaf-nodes. This characteristic applies for both push- and pull-based approaches, even though in pull-based approaches, these trees are neither created explicitly nor managed for multiple transmissions, and hence, possibly very short-lived. In general, cooperative streaming systems, consisting of self-organizing hosts, show inherent stability against node failures.

This property stems from the domain of peer to-peer systems, which is characterized by a high churn of node arrivals and node departures. In consequence, discovery and selection of alternative serving nodes as a fallback strategy are an integral part of these systems. However, each reconnection of nodes due to dynamics in the system comes with the cost of additional messaging and topology management, leading to delays, jitter, and possible packet loss in the data transmission. Furthermore, while node failures and intentional departures usually happen at random locations in the system, a malicious attacker will try to gather information about the overlay and deliberately attack nodes which are important for the overall service. Therefore, in order to create systems which are also resilient to attacks, appropriate overlay topologies have to be constructed. In this paper, we focus on improving the stability of peer-to-peer-based distribution of multimedia streams against deliberate Denial of Service (DoS) attacks. Intuitively, a couple of simple strategies to construct attack resilient overlay streaming topologies come to mind: The first idea is to keep the dependency of each node to other nodes low. This dependency is twofold: on the one hand, it is important for a node to minimize the amount of other nodes that it is dependent on, in order to minimize the chance of a predecessor leaving and, hence, to avoid quality degradations due to node departures. On the other hand, it needs to minimize its dependency to any single other node, in order to keep the impact of a leaving predecessor as low as possible. A second guideline is to keep the relevance of nodes balanced in order to avoid single nodes to become very important and a good target in consequence. third important issue is to keep information about the topology as secret as possible, in order to make it hard for a malicious node to find good targets for attacking. However, this last requirement is difficult to meet in practice, as with knowledge about participants and about the distributed algorithm, it is always possible to make good estimations about the evolving topology. In this paper, we make the following contributions: we provide an analytical model, which can be used to describe ALM systems. Using this model, we derive properties of topologies, which are optimally stable against attacks and give analytical proof of their optimality. These can serve as an upper bound to the resilience of topologies, both toward random node departures, caused by failure or churn, and DoS attacks on the system.The topologies additionally are characterized by the fact that they lead to a minimum deterioration of the quality of the delivered service for any number of failing nodes.

With knowledge about these properties, we are able to design a distributed procedure which creates close to optimally stable topologies in real scenarios, based on local knowledge only. The remainder of the paper is organized as follows: In Section 2 we present problem and approaches to the construction of stable streaming overlays. Subsequently, we define an analytical model for overlay streaming systems that we use to describe different damage functions as well as different attacker models in Section 2.1, in section 3 We define cost function algorithm for the attack and failure stability of ALM topologies, which we use to derive different types of attack strategies. In Section 4 Analytical Results how optimal streaming topologies can be constructed and proves their optimal stability against perfect attacks based on global knowledge. Following this formal approach, in Section 5, we design a distributed procedure to construct stable topologies in real environments, based on local knowledge of the participating nodes only, followed by a simulative evaluation of the stability of different ALM topologies. This performance comparison consists of three different types of topologies: optimally stable topologies, topologies that are created with the proposed distributed procedure, as well as topologies from previous ALM systems, and thus numerical results are evaluated. In Section 6, we conclude our paper and give some directions for further work.

2. PROBLEM FORMULATION:

To model the streaming overlays, it suffices to focus on end to-end links. The characteristics of the underlying network infrastructure do not need to be considered, as backbone and network routing decisions are not influenced by the systems and all overlay nodes additionally are able to establish a connection to any of the other overlay nodes. The abstraction from the underlying network topology leads to the inability to examine the behavior in circumstances of failing or attacked routers; however, the model suffices our needs, as we currently focus on end hosts only. Generally, one source node S is the originator of the streaming content. All other joining nodes locate participants as potential sources, which have joined the service at an earlier time, select some of them as parents and offer the service of forwarding the content, in turn. This system of potential and selected neighbors can be modeled as an undirected graph.

2.1 ATTACKS AND FAILURES

It is always possible to interrupt a multimedia stream by destroying the camera or the filmed scene. Strategies to secure the source of the stream like backup servers are conceivable. However, as we are concerned about constructing stable overlay topologies, we consider the source node s being hidden and assume that it cannot be the target of attacks. Hence, we only consider the failure of peers. Usually, the vertex connectivity is used as a stability metric for topologies, as it gives the minimum set of nodes that have to be removed in order to split the system into two separate fragments, one being completely disconnected from the data source. However, as the multimedia streaming service already has to be rendered useless as soon as the amount lost packets exceeds a certain threshold, a different metric is needed.

3. ALGORITHMS:

It becomes apparent that optimally stable topologies have a distinct set of three properties: 1) every node forwards data in only one spanning tree; 2) the number of distinctive direct child nodes of the source and of all heads are maximized; 3) the difference of the number of successors of all heads is at most 1.To achieve these properties, we design a distributed procedure. It creates overlay live streaming topologies in a tree-first approach based on local knowledge only. The main aims of the procedure are to keep the overall topology balanced and as low as possible, with each node forwarding in only one of the spanning trees.

```
   Input: v, i,
   Childs_{T_i}(v) {child nodes of node v in T_i}
1  d ← ∅ {link to drop};
2  a ← ∅ {alternative parent};
3  b ← deg(v);
4  gain ← true;
5  i ← preferred stripe;
6  while gain do
7  |   gain ← false;
8  |   d ← argmax {K(v, w, i) | w ∈ Childs_{T_i}(v)} ;
9  |   a ← argmax {G(v, w, i) | w ∈ Childs_{T_i}(v)\{d}};
10 |   if G(e',i) ≥ Θ_pass(v) then
11 |   |   drop(d, a);
12 |   |   gain ← true;
13 |   end
14 end
15 while b < c(v) do
16 |   a ← w = rand{Childs_{T_i}};
17 |   childRequest(a, Θ_pass(v), (succ_{T_i}(v) / fanout_{T_i}(v) − 1));
18 |   b ← b + 1;
19 end
```

After analyzing the local situation and repeating the three steps of phase one until no more child nodes are dropped to alternative parents, the nodes check their bandwidth consumption. If they have bandwidth capacities available, they request successors from their child nodes in the second phase, in order to decrease the amount of levels of the topology again (lines 14 . . . 18). In order to avoid the topologies to oscillate, the parent node sends information about its situation to the selected child, which then is able to predetermine that it does not pass a successor, which will be dropped back immediately.

Successors	12	14	16	18	20	22	28
Number of Nodes	143	72	16	1	1	1	1

Figure 1: The distribution of the number of packets depending on nodes forwarding in more than one stripe.

Height	3	4	5	6	7
Number of Heads	233	44	32	9	2

Figure 2: The number heads of a specific height

4. ANALYTICAL RESULTS:

In order to evaluate the stability of the optimally stable topologies and compare them with topologies of both the proposed procedure and existing ALM approaches, we compared their stability under attacks. For this purpose, we analyzed instances of the different topologies with algorithms implementing an optimal attack, based on global knowledge. The experiments were conducted in two steps. At first, we created topologies resembling the optimal topologies using a generator. The generator constructs topologies following the approach of creating balanced spanning trees for all stripes with minimum height and disjoint sets as forwarding nodes. For comparison, we then constructed topologies with simulation models of the cost-based approach, described in Section 5, and with strategies from our own previous work, constructing DAG Topologies with BCBS.

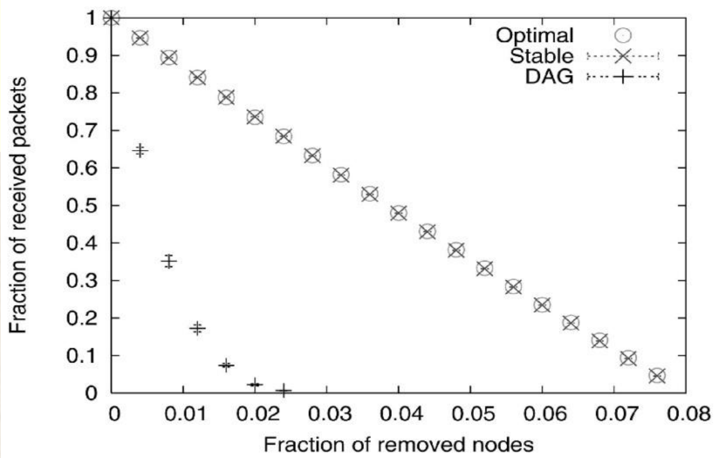

Figure 3: Stability of different ALM topologies

5. NUMERICAL RESULTS:

We expected the DAG topologies that were created using BCBS to experience a higher loss of received stripes with each successfully attacked node. Thus, they are expected to being less stable than the topologies that were created using the cost-based approach, as they are neither optimized to being well balanced nor to being low. In consequence, the stable topologies, which were constructed using the cost- based approach, are expected to be more stable than the DAG topologies. However, as the cost-based system is implemented in a distributed procedure that optimizes the neighborhood of the nodes based on local knowledge only, it is assumed that successful attacks on nodes will still lead to a higher number of stripes not being received than in the optimally stable topologies, created using the generator with global knowledge. The stability toward an optimal attack of the topologies generated using global knowledge, finally, should meet the theoretical values. For the last case, the optimally stable topologies, we realized that our assumptions hold true and the calculated quality drop equals the theoretical values. As Fig. 3 shows, the optimally stable topologies additionally are much more robust to attacks based on global knowledge than the DAG topologies described in [7]. However, the topologies generated using the cost-based approach were much more stable than the DAG topologies as well and even almost matched the stability of the optimally stable topologies In order to compare the topologies to other systems, sample topologies were obtained through the authors, where possible, and analyzed as well. It turned out that the topologies constructed using Dag Stream showed an almost identical stability as the topologies which were generated using BCBS, with the latter ones being slightly more stable toward attacks. This fact is not surprising, as the approaches are very similar, BCBS only allows for a more fine-grained splitting of the stream, as it creates topologies of spanning trees at packet level rather than at a stripe level. However, a more fine-grained partition of the stream with DagStream should lead to topologies which are as stable as the topologies created using BCBS. Both for Magellan and Split Stream, not enough topologies could be acquired which would be needed to conduct experiments leading to significant results. However, both systems create unbalanced and rather high topologies. These properties lead to the fact that while they are stable to random node failures, they cannot be stable toward optimal attacks.

In order to further evaluate the topologies of the cost-based approach, we analyzed to what extent they complied to the three properties required in Section 3: 1) each node forwarding in only one stripe, 2) the number of distinctive successors of the heads being maximized, and 3) the number of successors of all heads differing by 1 at maximum. Regarding the number of stripes in which each node is forwarding, it became apparent that an average of 6 percent of the nodes in all simulations where actually internal nodes in two, and thus in more than one stripes, with an overall minimum of 4 percent and an overall maximum of 10 percent of the nodes in any simulation. In total, we found 235 nodes in the 16 topologies, which forwarded in more than one stripe, with a minimum of nine and a maximum of 24. Even though these nodes forward more than one stripe, the experiments indicate that the low number of packets depending on them does not cause the topologies to be less stable. In fact, we conjecture that the condition for optimally stable topologies that a node forwards only in one stripe can be relaxed. popularity rank scores. How does the size of the competing set for the true popularity ranks in our data set across differ with suggestion set sizes? Competing set size and the suggestion set size for suggestion set sizes ranging from 1 to 40 tags. We observe that the competing set size is about twice or less than the suggestion set size, for suggestion set sizes tags. This suggests that the competing set tends to increase proportionally with the suggestion set size. Finally, we examined the height of the resulting trees to obtain a simple measure for latency. To obtain a detailed picture, we examined the height of the heads instead of the height of the root.

6. CONCLUDING REMARKS:

In this paper, we introduced a graph theoretic model for application layer multicast overlays. We subsequently described different types of strategies to attack topologies, including the definition of an optimal attack strategy. Based on the knowledge of optimal strategies, we were able to describe overlay streaming topologies, which are optimally stable toward perfect attacks and gave proof of their optimality. Furthermore, using this class of optimally stable topologies, we were then able to present a distributed procedure, which creates stable Topologies through cost based optimizations of the neighborhood of each node based on local knowledge only. The different topologies were evaluated in a simulation study, which revealed that the topologies created through the distributed procedure were much more stable than topologies from previous approaches, and that they were actually close to optimally stable against worstcase node failures and perfect attacks. As real-world scenarios are commonly assumed to be characterized by a high churn of arriving and departing nodes and a high rate of random node failures, we plan to extend our research toward analyzing, how stable our new topologies are in such an environment. We expect them to also be more stable in this case. However, they might not be more stable to a very high extent than topologies created using Split Stream or DagStream /BCBS, when subject to a random removal of nodes. If the topologies created with competing approaches developed scale free characteristics with respect to a power-law-distributed amount of successors, they could even be more stable to random node failure. This would be consequence of the fact that in such topologies, the chances of an important node with many successors to fail are very low, compared to the chances of one of the many nodes with only very few successors to fail. However, first results with a random attack strategy show that the gap between the stability of our topologies and the topologies of previous approaches shrinks, with our new topologies still being more stable under all circumstances. This result most probably stems from the fact that the topologies of existing approaches do not display very strong scale free characteristics. Further studies are planned to analyze the convergence behavior of the cost-based approach in adverse environments with high churn. The preliminary results are very promising, as they show that the measures taken in our approach are effective. Additionally, since the simulation study we conducted takes a user model into account that introduces a very high but decreasing churn at the start of the simulation, and which has been developed following real-world measurements of multimedia services on the Internet, we are confident that it will perform well in these studies. By applying some basic rules in our procedure it should be possible to create the topologies such that the number of successors is distributed—at least approximately—by a Power Law. In this way, the constructed topologies will be close to optimally stable and may show a high resilience against random failures.

REFERENCES:

- Y.H. Chu, S.G. Rao, S. Seshan, and H. Zhang, "A Case for End System Multicast," IEEE J. Selected Areas in Comm., vol. 20, no. 8,pp. 1456-1471, Oct. 2002.
- V. Pai, K. Kumar, K. Tamilmani, V. Sambamurthy, and A. Mohr, "Chainsaw: Eliminating Trees from Overlay Multicast," Proc. Fourth Int'l Workshop Peer-to-Peer Systems,

2005.

- X. Hei, Y. Liu, and K. Ross, "IPTV over P2P Streaming Networks: The Mesh-Pull Approach," IEEE Comm. Magazine, pp. 86-92, 2008.
- M. Zhang, Q. Zhang, L. Sun, and S. Yang, "Understanding the Power of Pull-Based Streaming Protocol: Can We Do Better?" IEEE J. Selected Areas in Comm., vol. 25, pp. 1678- 1694, 2007.
- S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable Application Layer Multicast," ACM Computer Comm. Rev., pp. 205-217, 2002.
- D. Carra, R.L. Cigno, and E.W. Biersack, "Graph-Based Analysis of Mesh Overlay Streaming Systems," IEEE J. Selected Areas in Comm., vol. 25, pp. 1667-1677, 2007.
- T. Strufe, G. Scha¨fer, and A. Chang, "BCBS: An Efficient Load Balancing Strategy for Cooperative Overlay Live-Streaming," Proc. IEEE Int'l Congress on Communication 2006.