Optimization of Tree-like Core Overlay in Hybrid-structured Application-layer Multicast

Jianguang Weng [1], Xuelan Zou [1] and Minhong Wang [2]
[1] Zhejiang Institute of Media and Communications
Hangzhou 310018, - China
[2] Division of Information & Technology Studies, Faculty of Education
The University of Hongkong, - Hongkong
[e-mail: wengjg@163.com; zxl_wn@163.com, magwang@hku.hk]
*Corresponding author: Xuelan Zou

Received August 15, 2012; revised October 25, 2012; accepted November 13, 2012; published December 27, 2012

Abstract

The tree topology in multicast systems has high transmission efficiency, low latency, but poor resilience to node failures. In our work, some nodes are selected as backbone nodes to construct a tree-like core overlay. Backbone nodes are reliable enough and have strong upload capacity as well, which is helpful to overcome the shortcomings of tree topology. The core overlay is organized into a spanning tree while the whole overlay is of mesh-like topology. This paper focuses on improving the performance of the application-layer multicast overlay by optimizing the core overlay which is periodically adjusted with the proposed optimization algorithm. Our approach is to construct the overlay tree based on the out-degree weighted reliability where the reliability of a node is weighted by its upload bandwidth (out-degree). There is no illegal solution during the evolution which ensures the evolution efficiency. Simulation results show that the proposed approach greatly enhances the reliability of the tree-like core overlay systems and achieves shorter delay simultaneously. Its reliability performance is better than the reliability-first algorithm and its delay is very close to that of the degree-first algorithm. The complexity of the proposed algorithm is acceptable for application. Therefore the proposed approach is efficient for the topology optimization of a real multicast overlay.

Keywords: Tree-like core overlay, application-layer multicast, out-degree weighted reliability, evolutionary programming

---

This research was supported by National Natural Science Foundation of China (No. 61003088) and Zhejiang Provincial Natural Science Foundation of China (No.Y1100189).

http://dx.doi.org/10.3837/tiis.2012.12.005

1. Introduction

Application-layer multicast has become a very popular research topic due to the lack of widespread support for IP multicast and the high cost of infrastructure-based approaches using dedicated servers. According to their overlay structures, the multicast networks are classified as the tree-like overlay systems [1][2], the mesh-like overlay systems [3][4] and the hybrid systems [5]. In a tree-like overlay multicast network, nodes are interconnected in a spanning tree and the media information is pushed from parent node to child node and the receiving child node replicates and sends packets to other members connected with it. NICE is a typical tree-like overlay system, which has a small control overhead and produces low latency distribution trees [6]. However the tree-like overlay systems have poor resilience to node failures. When a node on a multicast tree leaves or fails, all the nodes receiving media data directly and indirectly from it will lose their streaming service. The approach to the mesh-like overlay multicast is to organize nodes into a mesh-structured overlay by selecting multiple nodes as neighbors to exchange data with. Compared with the tree-like overlay, the mesh-like overlay is more resilient against node failures; but it has additional control overhead due to the data driven technique and high transmission delay. The hybrid systems combine both advantages of the tree-like and mesh-like overlay multicast. In a hybrid-based overlay multicast a tree-like overlay is used as the backbone network for high efficient media streaming transmission while the mesh-like overlay is used as the non-backbone network for enhancing the system's failure resilience [5][7]. In this paper we focus on the construction of the tree-like backbone overlay in a hybrid-based multicast system. Some nodes are selected as the backbone nodes which are reliable enough and have good upload performance as well. The backbone nodes are organized into a spanning tree. We aim at optimizingthe topology of the backbone tree to enhance its reliability and to achieve short transmission delay as well.

There are some constraints that might be imposed on the overlay construction such as traffic cost, link availability, upload capacity, and time delay etc. This paper is mainly concerned with the topology optimization of the core overlay network with upload capacity, i.e. out-degree constraint. It has been proved that the degree constrained spanning tree problem is NP-complete [8]. We propose an optimization algorithm which is the evolutionary programming algorithm based on the out-degree weighted reliability. We name it ODW-reliability algorithm. There is no illegal tree problem during the evolution. Our contributions are summarized below:

(1) the proposed ODW-reliability algorithm increases the reliability of the core overlay tree and achieves short delay at the same time.
(2) we use the out-degree weighted array as the encoding approach, which is efficient because it incurs no illegal solution during the evolution process.
(3) we evaluate the proposed algorithm and compare it with other optimization algorithms. Simulation results show that our algorithm has good performance with an acceptable complexity.
(4) the proposed algorithm could also be used to optimize the topology of a tree-like multicast overlay system.

The rest of the paper is organized as follows. In Section 2 we present briefly previous research on node failure resilience of tree-like overlay. In section 3 we describe the topology optimization problem for the core overlay multicast tree and give the mathematical model. The proposed optimization algorithm to optimize the topology of the core overlay tree is described in detail in section 4. In section 5 we give the computational results which demonstrate the effectiveness of the proposed algorithm. Finally the paper concludes in section 6.

2. Related Work

The research on node failure resilience of tree-like topology could be summerized into three kinds of methods, i.e. Multiple-Pathes, Redundant-Parents , and Reliability-Optimization.

Multiple-Pathes methods try to send stream data by more than two trees that work simutaniously or alternatively. Authors of [9] suggest to use multiple diverse distribution trees to supply redundancy in network paths and multiple description coding (MDC). A simple tree management algorithm that provides the necessary path diversity and minimizes peak a signal-to-noise ratio (PSNR) parameter is proposed. [10] focuses on the problem of reliable video broadcasts in optical networks. They propose constructing two disjoint trees to increase the network survivability. Since the backbone optical network is considered, the authors use Steiner trees to formulate the MIP model of the multicasting. The optimization of [11] covers multicast flow arrangement on multiple paths in order to minimize the overall streaming cost. It applies a basic hop-constrained spanning tree and capacitated spanning tree problems and defines level-constrained multiple trees problem with bandwidth capacity constraints for multicast flow assignment in overlay system. In [12], the main idea to provide protection of the system against network failures is to establish several (at least two) disjoint multicasting trees. The discussion in this paper centers on the problem of how additional survivability constraints to provide failure-disjoint trees can make impacts on the operation of P2P multicasting systems.

Redundant-Parents methods provide some backup nodes as parents when a node requests its parent. Zigzag[13] is a method for clustering peers into a hierarchy called the administrative organization for easy management, and a method for building the multicast tree where each cluster has serveral nodes which could be head alternatively. A P2P multicast protocol called Nemo is described in [14]. The Nemo offers high delivery ratio without sacrificing end-to-end latency or incurring additional costs. The protocol applies two techniques: co-leaders to minimize dependencies and triggered negative acknowledgments (NACKs) to detect lost packets. [15] proposes a new P2P streaming framework Dagstream which produces a structure of peers in the form of a directed acyclic graph (DAG). To assure the system resilience each peer keeps at least k parents. Authors of [16] let each non-leaf node precalculates a parent-to-be for each of its children rather than let downstream nodes try to find a new parent after a node departure. When this non-leaf node is gone, all its children can find their respective new parents immediately. The salient feature of the approach is that rescue plans for multiple non-leaf nodes can work together for their respective children when they fail or leave at the same time. [17] proposes to use multiple cooperative error recovery sources, called a minimum-losscorrelation group (MLC group), to help a node that suffers from a stream disruption find the lost data while it is looking for a new parent. Considering significant spatial correlation in loss among the receiver nodes, authors of [18] propose a scalable recovery tree construction scheme scheme distributes some tasks normally handled by the sender node to specific nodes acting as repair node distributors.

Reliability-Optimiazation methods improve the tree reliability by optimizing node position in the tree. An important approach to this is to build a short tree. Intuitively, the shortness helps to reduce the number of descendant nodes that will be affected by a failed node. An additional merit of this approach is that it generally leads to a small average service delay from the source. The minimum depth tree is an example of this approach[14][19]. The high-bandwidth-first algorithm achieves minimum tree depth by placing the nodes from high to low layers in a non-increasing order of bandwidths[19]. This algorithm achieves a global optimization. In contrast to the depth-optimizing approach using the members' bandwidth properties, another approach leverages the member's time property: if the members' lifetimes follow a distribution with a long tail, then the older members are less likely to leave before the younger ones [17][20]. Authors of [21] suggest a reliable overlay multicast tree based on members' sojourn probabilities. Path reliabilities from a source to member nodes are considered to maximize the reliability of an overlay multicast tree.

Each aformentioned methods have its advantages and some common or special shortcomings. Protocol overload is a common problem of Multiple-Pathes and Redundant-Parents methods[9][10][11][12][13][14][15][16][17]. Reliability-Optimiazation methods have less overload but they have other shortcomings such as local optimization[17], tall tree[20], large compuation cost[21], and still heavy overload[19]. Our approach is one of Reliability-Optimiazation methods, which can optimize depth and realibility at the same time with pratical computation cost and not much overload.

3. Problem Description

In an application level multicast system, a node may subscribe and withdraw the streaming service at any time, which is different from the IP multicast system. For the application level multicasting, when the user behind a node on the overlay quits, the node fails. When a node fails, all its child nodes will lose their streaming services. For a tree-like overlay network, if a node leaves or joins the system frequently, the maintenance of the network is greatly increased. In a hybrid-based multicast system some special nodes are chosen as the backbone nodes according to their degree and reliability. These backbone nodes are reliable enough and have strong upload capacity as well and it is helpful to overcome shortcomings of tree topology. In the paper the degree means a node's out-degree or upload bandwidth. The selected backbone nodes could be those whose out-degree is larger than the media stream rate or whose reliability is higher than the given threshold. For a real network design the reliability threshold could be a given value or be determined by the ratio of backbone nodes to all the nodes in the network. Then the selected nodes are used to construct a tree-like overlay multicast with the proposed algorithm to provide high quality media stream service.

An overlay multicast tree can be model as a graph $G=(V,E)$ with $n=|V|$ nodes and $m=|E|$ edges. Let the degree constraint $d_{\max}(k)$ represent the upload capacity limit of node *k* in an overlay tree T. Suppose that $d_T(k)$ is the degree of node *k* in the tree T, then $d_T(k)\le d_{\max}(k)$. In our study *n* is the number of the selected backbone nodes. Generally the more the layer of a tree, the higher the delay. We limit the depth of the core overlay tree for the goal of satisfying delay constaints. The maximum depth of the tree depends on the node number and the average node degree. Suppose that d_{ave} ($d_{ave}>1$) is the average node degree and *l* is the depth of the tree (the root node is in the zero layer). Suppose these nodes are used to construct a full tree. Then we have

$$n=1+d_{ave}+d_{ave}^2++d_{ave}^3\cdots d_{ave}^{l-1}+d_{ave}^l \tag{1}$$

According to the geometric progression properties, we have

$$1 + d_{ave} + d_{ave}^2 + d_{ave}^3 + \cdots d_{ave}^{l-1} + d_{ave}^l = \frac{d_{ave}^{l+1} - 1}{d_{ave} - 1} \tag{2}$$

Then l is calculated by (3)

$$l = \log_{d_{ave}} [n(d_{ave} - 1) + 1] - 1 \tag{3}$$

To simplify the computation from (3), we have

$$l < \log_{d_{ave}} (n) \tag{4}$$

In a full tree each node is either a leaf node or has exactly d_{ave} children. For the given node number, the full tree has the minimum depth. In our study we aim at constructing a stable tree to provide high quality media service. The depth of the core overlay tree in our study is given by

$$l = \lfloor \log_{d_{ave}} (n) + 1 \rfloor \tag{5}$$

which is also the length of the out-degree weighted array in the proposed algorithm.

Note that a node fails in an application level multicast system much more frequently than a link does [22][23]. The reliability of a multicast tree depends more on the path reliability than on the link reliability. In our work we consider the node failure as the only cause of loss of multimedia information transmission. The reliability of an overlay multicast tree is defined as

$$R(T) = \sum_{k \in S} r_a(k) \tag{6}$$

where S is the node set of an overlay multicast tree. $r_a(k)$ is the accumulated reliability of node k which is calculated by

$$r_a(k) = r(k) \times \prod_{i \in S_k} r(i) \tag{7}$$

S_k is the node set which contains all the other nodes except node k in the path from source node to node k. $r(k)$ is the sojourn probability of node k.

Then the problem for topology optimization of a core overlay tree is described as

$$\max \; R(T) = \sum_{k \in S} r_a(k) \tag{8}$$

$$\text{subject to} \qquad d_T(k) \le d_{\max}(k), \forall k \in V \tag{9}$$

3. The Proposed Optimization Algorithm for Core Overlay Tree

As aforementioned, evolutionary programming is one of the four major evolutionary algorithms including genetic algorithms along with genetic programming and evolutionary strategies. The primary difference between evolutionary programming and the other evolutionary approaches is that there is no recombination. Mutation is the main genetic operator for evolutionary programming, which produces spontaneous random changes to generate offspring. In this section we provide our ODW-reliability algorithm to optimize the topology of the core overlay tree which is shown in **Fig. 1**. The main idea of the proposed approach includes four components: 1) design parameters calculation, 2) initial solution, 3) mutation, 4) selection. **Table 1** lists the main variables used in this paper. As shown in **Fig. 1**, the first step is to determine the parameters n and l. n is the backbone node number and l is calculated by (5) as described in section 2. The second step is to generate the initial solutions, then do mutation on each individual to generate offspring. Selection is determinstic according to the reliability defined by (6). Those individuals whose reliability is in the top pop_size will be selected as the new population. If the termination conditions are not satisfied, continue the above procedure. Otherwise, select the optimal one from the current generation which is the solution to the overlay optimization problem. Note that the fitness function in the evolution is the reliability of the overlay tree defined by (6) and mutation is based on out-degree weighted array. The steps in the ODW-reliability algorithm in this research are discussed below.

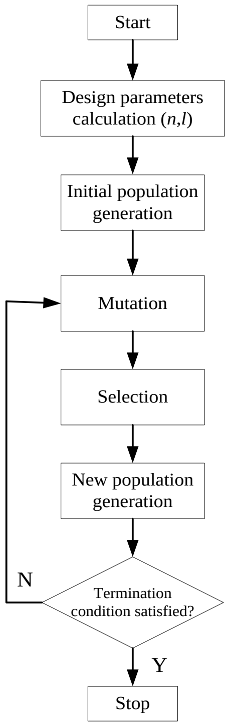


Fig. 1. Proposed optimization algorithm

Table 1. Denotation

n	node number of the overlay tree
l	depth of the overlay tree
pop_size	population size of evolutionary programming
d_{av}	average node degree
$d(k)$	degree of node k
$r(k)$	reliability of node k
$r_w(k)$	out-degree weighted reliability of node k
w_i	out-degree weighted value for the nodes in the ith layer of the overlay tree
g	generation number of evolutionary programming
g_max	maximum generation number of evolutionary programming
r	reliability ranking of the individual in population
δ	standard deviation of Gaussian distribution
δ_r	weighted standard deviation
f_g	factor of evolution generation
f_r	factor of individual ranking

3.1 Construction of Initial Tree

Randomly generate an out-degree weighted array $w=[w_1,w_2,\cdots,w_l]$ with the constraints of $l=\lfloor \log_{d_{ave}}(n)+1 \rfloor$ and $w_i \in (0,1]$ $i=1,2,\cdots,l$. Suppose that S is the node set of a core overlay tree with *n* nodes and $\Re$ is the source node in the tree. The initial tree is constructed as following.

Step 1: Select $\Re$ from node set S and set it as the root node of the tree. Delete $\Re$ from node set S.

Step 2: Randomly choose a node *k* from node set S and delete node *k* from S. Let node *k* as the child node of node $\Re$. Note that in our tree construction the root node $\Re$ is in the zero layer of the tree. Then *k* is in the first layer of the tree.

Step 3: Randomly choose another node *j* from S, then delete node *j* from S. if $d_T(\Re) < \lfloor d_{\max}(\Re) \rfloor$, add node *j* as the child node of node $\Re$ and *j* is also in the first layer of the tree.

Step 4: If $d_T(\Re) = \lfloor d_{\max}(\Re) \rfloor$, then *j* should be added to the second layer of the tree. Select a node as node *j*' parent node from the left to the right which is in the first layer and which satisfies the degree constraint.

Step 5: For all the other nodes in S, randomly choose the node one by one and add it to the tree as done in Step 3 and Step 4.

3.2 Mutation Operator

Mutation is the main genetic operation of evolutionary programming. It produces random changes in the selected parent and forms new offspring. In general the mutation in evolutionary programming is Gaussian mutation. It is clear that good parameter values facilitate good performance. In the paper the mutation operator is based on the out-degree weighted array which is used to choose nodes to construct new population during the evolution. To control the algorithm convergence, the effect of the ranking of reliability in the population and evolution generation is considered to control the mutation step size. The mutation operator is shown as following:

Step 1: Generate mutation step size parameters:

$$f_g = 1 - \frac{g}{g\_\max} \tag{10}$$

$$f_r = \frac{r}{pop\_size} \tag{11}$$

$$\delta_r = \delta \times f_r \tag{12}$$

Step 2: For each $w_i \in w$ ($i=1,2,\cdots,l$), generate a random number θ_i between 0 and 1, then

$$\theta_i = \theta_i \times f_g \tag{13}$$

$$\Delta_1 = \max(w_i - \theta_i, 0) \tag{14}$$

$$\Delta_2 = \min(w_i + \theta_i, 1) \tag{15}$$

Generate a Gaussian distributed number

$$\alpha_i = N(w_i, \delta_r) \tag{16}$$

$$\text{subject to: } \Delta_1 < \alpha_i < \Delta_2 \tag{17}$$

Step 3: Let $w_i = \alpha_i$ and w_i is the new out-degree weighted array which is used to produce new offspring.

3.3 New Population Generation

Note that the proposed mutation changes the out-degree weighted array not the topology of the tree. In order to generate new offsprings, we need to re-construct trees according to the new out-degree weighted array. The out-degree weighted reliability is defined by (18)

$$r_w(k) = r(k) + (d(k) - 1) \times w_i \tag{18}$$

The tree re-construction is based on the above out-degree weighted reliability which is shown as following:

Step 1: Calculate the out-degree weighted reliability for all the nodes in node set S except for the root node $\Re$.

Step 2: Sort the nodes in decreasing order according to their out-degree weighted reliability where node $\Re$ is in the first position.

Step 3: $\Re$ is the root node and delete it from the node set S. Select nodes in decreasing order of out-degree weighted reliability from S to re-construct the overlay tree which is similar to the procedure described in section 3.1. The difference is that in the initial tree construction nodes are randomly selected while during the tree re-construction nodes are selected according to their out-degree weighted reliability. Therefore nodes with the higher out-degree weighted reliability move to the higher positions in the tree.

The re-constructed trees are new offsprings and now there are $2 \times pop\_size$ individuals in the current population. The new population is generated as following:

Step 1: Calculate the fitness value for each individual in the current population which is defined by (6).

Step 2: Sort the individuals in decreasing order according to their fitness values.

Step 3: The individuals whose fitness values are in top *pop_size* are survivals.

Step 4: The new population is made up of these survival individuals.

4. Computation Results and Discussion

We generate four node set groups to compare the perfomance of ODW-reliability algorithm with random algorithm, the degree-first algorithm , and the reliability-first algorithm. Each group is compared from twoperspectives, i.e. tree reliabitliy and transmission delay. The difference between the random algorithm, the degree-first algorithm, and the reliability-first algorithm is the node selection strategy. In the degree-first algorithm the node is selected in the order of its degree, while in the reliability-first algorithm the node which has maximum reliability is chosen first to the source node. All the simulations were performed with implementations in C executed on Intel Core i5 (2.27GHz clock).

The node set groups are the tree-like core overlay networks with different design parameters including network size, population size and node capacity. Node propertities are generated randomly according to long-tailed lifetime. The details of the test problems are available from authors.The property of each node set group is as the followings:

(1) The first group compares the effect of network size where the network size is 100, 200, 500, 1000 and 2000 nodes respectively.

(2) The second group is different in average node degree where the average node degree

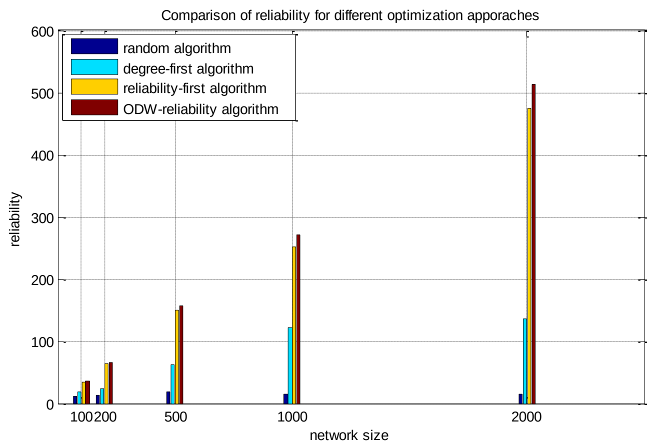
is 1.69, 2.09 and 3.28.

(3) The third group is used to studied the influence of average node reliability where the average node reliability is set as 0.46, 0.52 and 0.63 respectively.

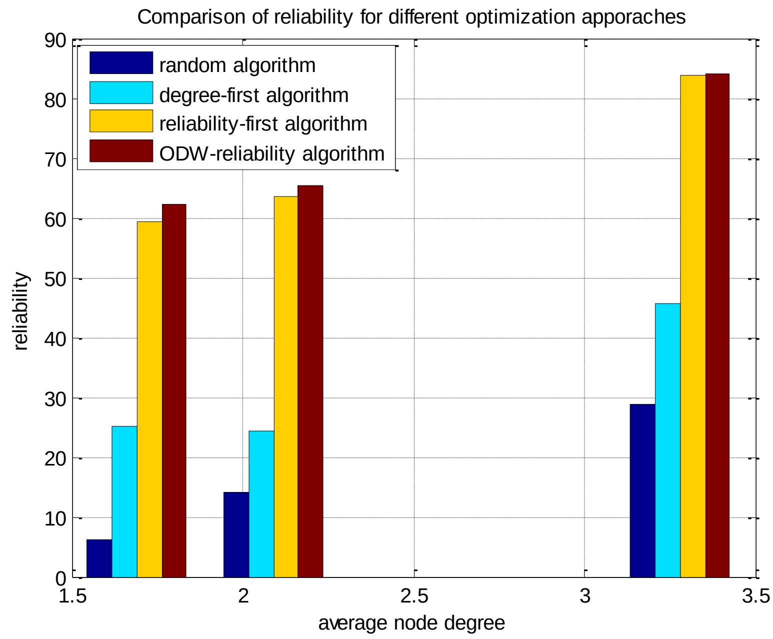
(4) The fourth group is to compare the effect of population size on the performance of the proposed algorithm.The population size for all the networks is 20 except for networks in the fourth group.

The halting criterion, population size, node capacity are important parameters for evolutionary programming performance. There are several halting criteria to stop evolutionary programming including generation number, computing time, mininum reliability, maxinum depth, and fitness convergence. Mininum reliability and maxinum depth get from computation results of other contrast algorithms. Fitness convergence is the key criteria which is the only contributing criteria in our experiments. The fitness convergence criterion is to compare the improvement of average reliability and optimal reliability with the given threshold. If any of the average reliability and optimal reliability is satisfied, the evolutionary programming will be stopped.

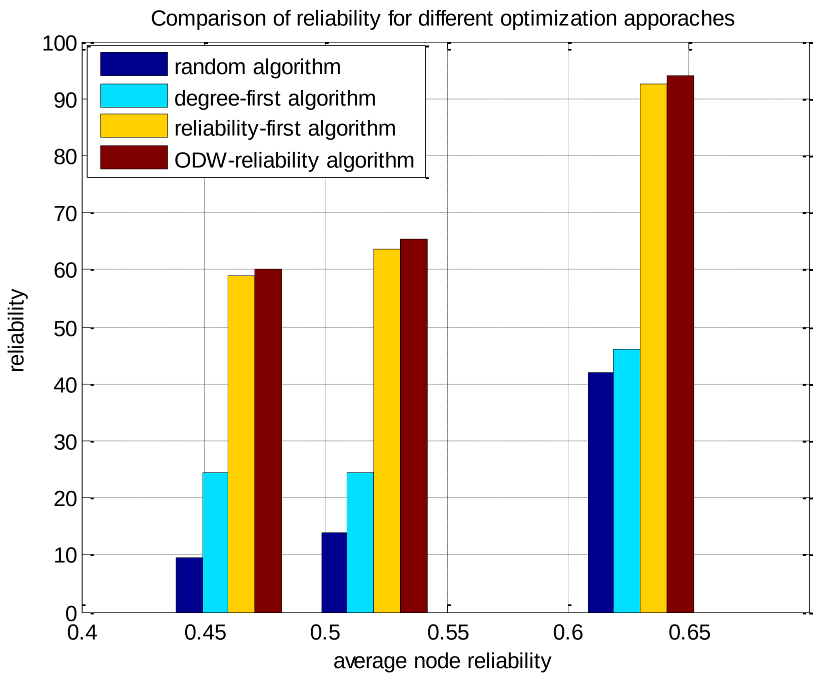
Fig. 2 shows the reliability comparison of the proposed ODW-reliability algorithm with the other three optimization approaches. It is shown that for all the tested problems the proposed algorithm has the best reliability performance which is better than that of the reliability-first algorithm. The performance of the random algorithm is the worst. **Fig. 2 (a)** shows that the performance gap increases with the increase of network size. For example, when the network size is 200 nodes, compared with random algorithm, the degree-first algorithm's reliability increases by 73.61%. The reliability-first algorithm's reliability increases by 352.54% and the proposed ODW-reliability algorithm's reliability increases by 365.66%. When the network size is 2000 nodes, the performance improvement is 774.65%, 2955.28% and 3208.65% repectively for the above approaches. It is shown from **Fig. 2 (b)** and **Fig. 2 (c)** that the reliability of the overlay tree increases with the increase of average node degree and average node reliability for all the optimization methods. Therefore the core overlay tree constructed by the proposed algorithm is more reliable for media streaming and it helps to improve the reliability of the hybrid overlay system. As we know population size is a key design parameter for evolutionary programming. Results from **Fig. 2 (d)** show that the selection of population size of the evolutionary programming has little effect on the reliability performance. Therefore the proposed ODW-reliability algorithm is efficient for the core overlay tree's reliability optimization.



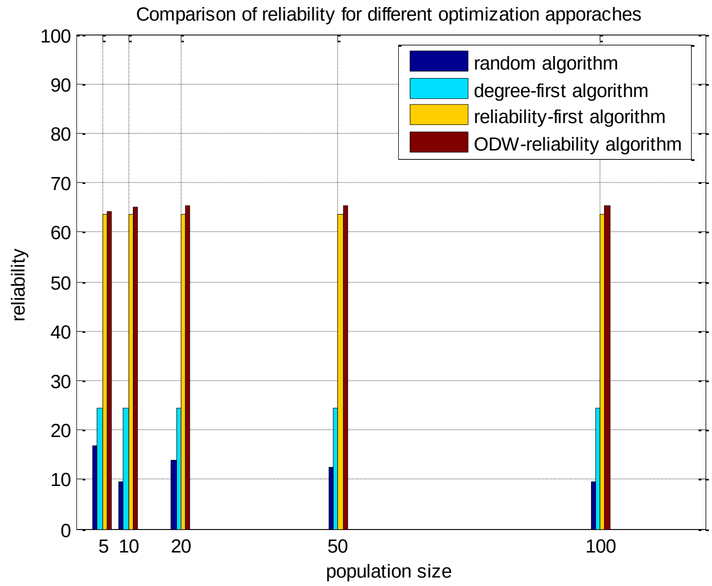
(a) network size=100,200,500,1000,2000 nodes



(b) average node degree=1.69, 2.09, 3.28



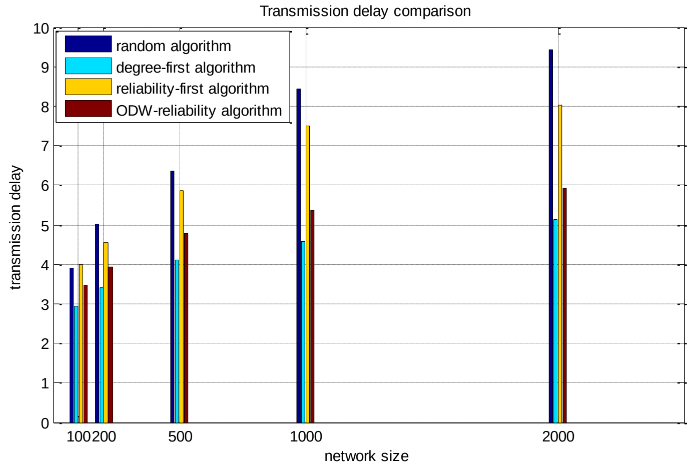
(c) average node reliability= 0.46, 0.52 , 0.63



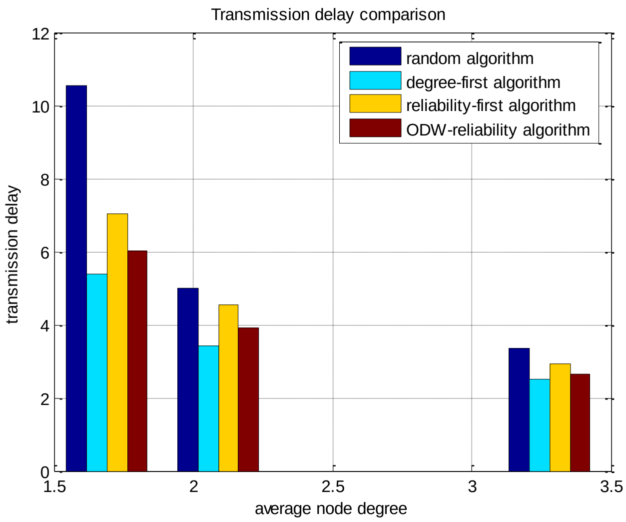
(d) population size=5,10,20,50,100

Fig. 2. Reliability comparison of the proposed algorithm with other optimization approaches

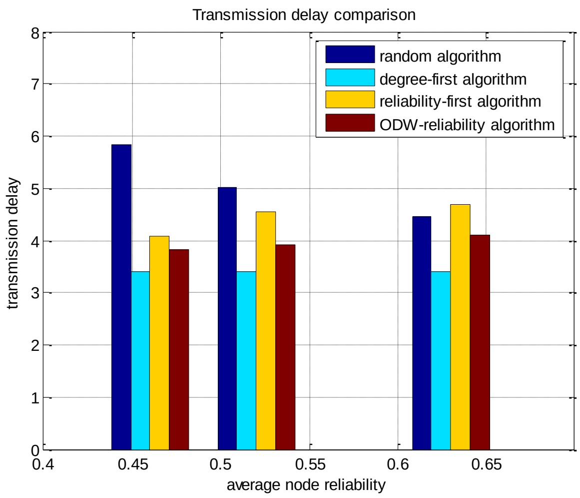
Fig. 2 shows that the proposed algorithm improves the reliability of the core network of a hybrid-structured application-layer multicast overlay. It is also important to examine the quality of service experienced by end users in terms of end-to-end delay. Now we study the node's service delay for the multicast overlays under different optimization approaches. In this paper the node's service delay is measured by transmission delay which is defined as the depth of the overlay tree. **Fig. 3** is the results of transmission delay comparison of the four optimization approaches. It is shown that for all the test problems the degree-first algorithm has the shortest transmission delay. As mentioned before the tree constructed by the degree-first algorithm selects the nodes with larger degree to the higher position, it is clear that it constructs the shortest tree and has the shortest delay. **Fig. 3 (a)** shows that the larger the network size,the longer the transmission delay is produced. For a 200-node tree, compared with the degree-first algorithm, the proposed ODW-reliability algorithm's delay increase by 14.79%, the reliability-first algorithm's delay increase by 33.53% and the random algorithm's delay increase by 46.71%. When constructing a 2000-node tree, compared with the degree-first algorithm, the proposed ODW-reliability algorithm's delay increases by 15.37%, the reliability-first algorithm's delay increases by 55.92% and the random algorithm's delay increases by 83.65%. Therefore the proposed ODW-reliability algorithm's delay is very close to the degree-first algorithm, and it is better than the other two optimization approaches. **Fig. 3 (b)** shows that the larger the average node degree, the shorter the transmission delay is produced for all the optimization algorithms. The performance gap between the degree-first algorithm and proposed ODW-reliability algorithm decreases with the increase of the average node degree. Therefore the network size and average node degree are the key parameters for the transmission delay of the overlay tree. Note that the average node reliability has little effect on the delay of the overlay tree as shown in **Fig. 3 (c)**, which is different from the case of reliability optimization. It is also shown from **Fig. 3 (d)** that population size has little effect on the core overlay tree's delay for the proposed ODW-reliability algorithm. Therefore in terms of reliability and delay the population size is not important for our ODW-reliability algorithm.



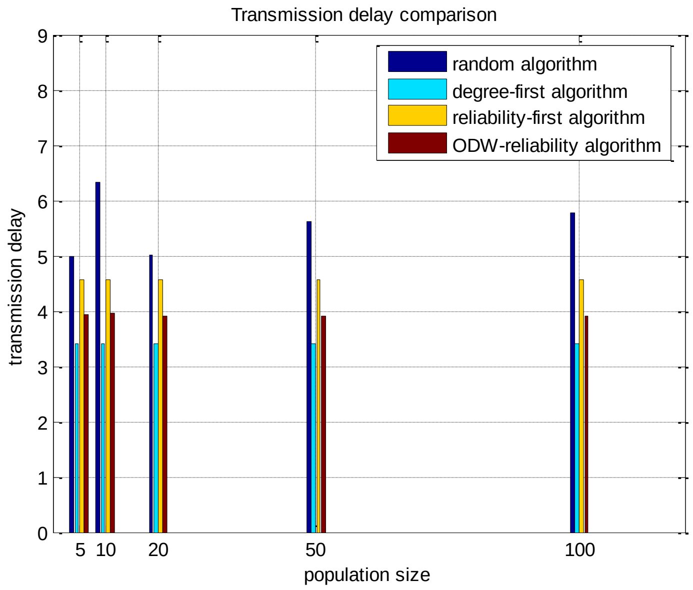
(a) network size=100,200,500,1000,2000 nodes



(b) average node degree=1.69, 2.09, 3.28



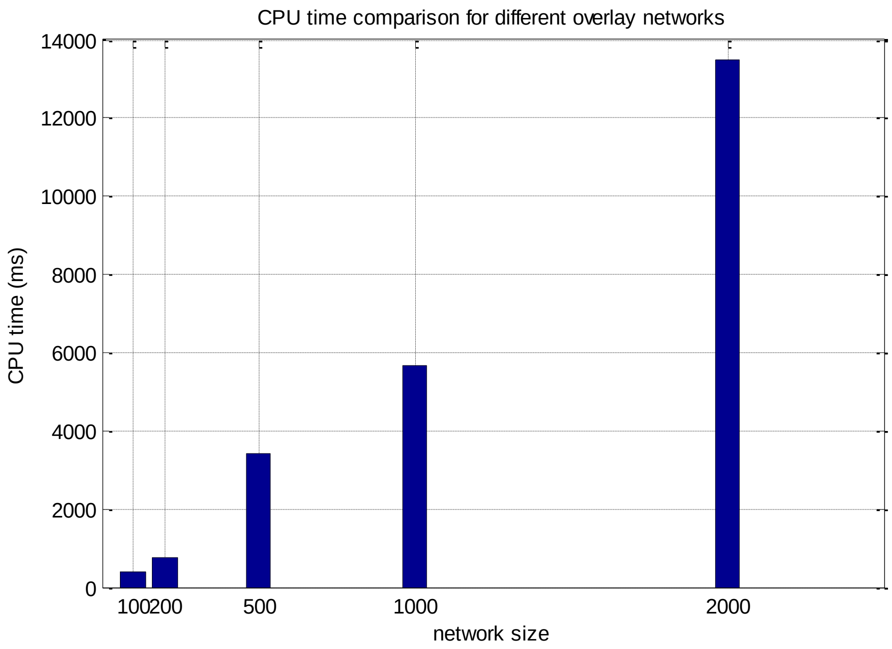
(c) average node reliability= 0.46, 0.52 , 0.63



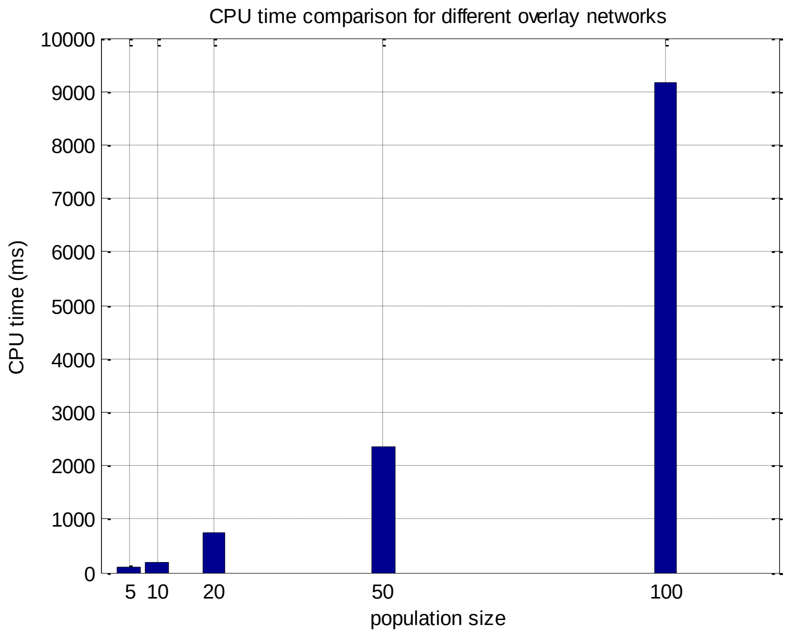
(d) population size=5,10,20,50,100

Fig. 3. Transmission delay comparison of the proposed algorithm with other optimization approaches

Although we focus on improving the hybrid-structured application-layer multicast overlay's performance, it is also important to investigate the time cost of the proposed approach. As mentioned, network size and population size are two important parameters for evolutionary programming, the time cost of the proposed algorithm in different tree-like overlay systems are shown in **Fig. 4**.



(a) population size =20



(b) network size=200 nodes

Fig. 4. CPU time comparison for different overlays with the proposed approach

Fig. 4 (a) shows that the CPU time increases when the network size increases. But even for la arge scale network, such as a 2000-node network, the required time is only 13475 ms. This is acceptable for the real network design. **Fig. 4 (b)** provides the comparison of CPU time for a 200-node overlay system with different population size. It is shown in **Fig. 4 (b)** that the larger the population size, the more the required CPU time. The CPU time is 90 ms when the population size is 5, while it is 9159 ms when population size is 100, which is increased by a factor of 100. Note that the population size has little effect on the performance of the core overlay tree in terms of reliability and delay, which is demonstrated by **Fig. 2** and **Fig. 3**. A small value of population size is advisable when using the proposed method to optimize the overlay's topology. From the experiment results we can find that the proposed ODW-reliability algorithm has the best reliability among the four investigated optimization approaches and its delay is very close to that of the degree-first algorithm with a low time cost.

5. Conclusion

In this paper we address the topology optimization issue for a hybrid-structured application-layer multicast overlay system. We first choose some nodes which are reliable enough and have strong upload capacity, and use these nodes as backbone nodes. These backbone nodes are organized into a tree-like core overlay. The core overlay is periodically adjusted by the proposed ODW-reliability algorithm. The proposed algorithm is based on the out-degree weighted reliability which incurs no illegal tree problem during the evolution. Simulation results show that compared with the random algorithm, the degree-first algorithm, and the reliability-first algorithm, our proposed approach greatly improves the reliability of the tree-like core overlay. The degree-first algorithm has the shortest delay and the proposed ODW-reliability algorithm's delay is very close to it. The population size is a key parameter of evolutionary programming but it has little effect on the performance of the proposed algorithm in terms of reliability and delay. The complexity of the proposed algorithm is measured by CPU time. Simulation results show that the required CPU time depends on the network size and population size and it is very satisfactory for a real multicast overlay's optimization. The proposed ODW-reliability algorithm can also be used to optimize a tree-like application-layer multicast overlay where all the nodes in the tree are selected.

References

[1] D. A. Tran, K. Hua and T. Do, "A peer-to-peer architecture for media streaming," *IEEE Journal on Sel. Areas in Commun.*, vol. 22, no. 1, pp. 121-133, 2004. Article (CrossRef Link)

[2] M. Castro, P. Druschel, A. M. Kermarrec and A. I. T. Rowstron, "Scribe: a large-scale and decentralized Application-Layer multicast infrastructure," *IEEE Journal on Sel. Areas in Commun,*, vol. 20, no. 8, pp. 1489-1499, 2002. Article (CrossRef Link)

[3] N. Magharei and R. Rejaie, "PRIME: Peer-to-peer receiver-driven mesh-based streaming," In *Proc. of IEEE INFOCOM'07*, Anchorage, AK, USA, May, 2007. Article (CrossRef Link)

[4] X. Zhang, J. Liu and T. P. Yum, "Coolstreaming/DONet: A data-driven overlay network for efficient live media streaming," in *Proc. of IEEE INFOCOM'05*, Miami, FL, USA, Mar. 2005. Article (CrossRef Link)

[5] F. Wang, Y. Xiong and J. Liu, "mTreebone: A hybrid tree/mesh overlay for application-layer live video multicast," in *Proc. of ICDCS'07*, Toronto, Canada, Jun. 2007. Article (CrossRef Link)

[6] S. Banerjee, B. Bhattacharjee and C. Kommareddy, "Scalable application layer multicast," in *Proc. of ACM SIGCOMM*, Pittsburgh, PA, USA, Aug. 2002. Article (CrossRef Link)

[7] D. Kostic, A. Rodriguez, J. Albrecht and A. Vahdat, *"Bullet: High bandwidth data dissemination using an overlay mesh,"* In *Proc. of SOSP*, Lake Bolton, New York, USA, Oct. 2003. Article (CrossRef Link)

[8] M. Garey and D. Johnson, "Computers and Intractability: A Guide to the Theory of NP-completeness," SanFrancisco: W. H. Freeman and Company, 1979.

[9] V. Padmanabhan, H. Wang, P. Chou, "Resilient peer-to-peer streaming," In *Proc. Of IEEE International Conference on Network Protocols*, Atlanta, GA, USA, pp. 16-27, 2003. Article (CrossRef Link)

[10] J. Allen, P. Kubat, Reliable video broadcast via protected steiner trees, *IEEE Commun. Mag.,* vol. 48, no. 2, pp. 70-76, 2010. Article (CrossRef Link)

[11] M. Kucharzak, K. Walkowiak, "Optimization of Flows in Level-Constrained Multiple Trees for P2P Multicast System," In *Proc. of International Conference on Advances in P2P Systems*, Florence, Italy, pp. 106-111, 2010.

[12] K. Walkowiak, M. Przewoniczek. "Modeling and optimization of survivable P2P multicasting," *Journal of Computer Communications*, vol. 34, no. 12, pp. 1410-1424, Aug. 2011. Article (CrossRef Link)

[13] D. A. Tran, K. A. Hua, and T. T. Do. "A peer-to-peer architecture for media streaming," *IEEE J. on Sel. Areas in Commun*. vol. 22, no. 1, pp. 121-133, Jan. 2004. Article (CrossRef Link)

[14] S. Birrer, D. Lu, F. Bustamante, Y. Qiao, P. Dinda, "FatNemo: building a resilient multi-source multicast fat-tree," *Lecture Notes Comput. Sci*. vol.3293, 182-196, 2004. Article (CrossRef Link)

[15] J. Liang, K. Nahrstedt, "DagStream: locality aware and failure resilient peer-to-peer streaming," in *Proc. of SPIE Multimedia Computing and Networking Conference*, San Jose, CA, 2006.

[16] Z. Fei, M. Yang, "A proactive tree recovery mechanism for resilient overlay multicast," *IEEE/ACM Trans. Netw.*, vol. 15, no. 1, pp. 173−186, 2007. Article (CrossRef Link)

[17] G. Tan and S. A. Jarvis, "Improving the fault resilience of overlay multicast for media streaming," *IEEE Trans. on Paral. and Distr. Systems*, vol. 18, no.6, pp. 721-734, 2007. Article (CrossRef Link)

[18] J. Baek, and J. F. Pâris, "A Scalable Recovery Tree Construction Scheme Considering Spatial Locality of Packet Loss," *KSII Transactions on Internet and Information Systems*, vol. 2, no. 2, pp.82-102, 2008. Article (CrossRef Link)

[19] M. Guo, M. Ammar. "Scalable live video streaming to cooperative clients using time shifting and video patching," in *Proc. of INFOCOM*, vol. 3, pp.1501-1511, 2004. Article (CrossRef Link)
[20] Y. Tian, D. Wu, G. Sun and K. W. Ng, "Improving stability for peer-to peer multicast overlays by active measurements," *Journal of System Architecture*, vol.54, no.1-2, pp. 305-323, 2008. Article (CrossRef Link)
[21] J. H. Lee, C. Y. Lee, "Maximization of Path Reliabilities in Overlay Multicast Trees for Real time Internet Service," *http://heuristic.kaist.ac.kr/paper/max path reliabilities in overlay multicast-korms.pdf*, 2007.
[22] G. Iannaccone, C. Chuah, R. Mortier and S. Bhattacharyya, etc., "Analysis of link failures in an IP backbone," in *Proc. of Internet Measurement Workshop*, IMW, Marseille, France, Nov. 2002. Article (CrossRef Link)
[23] E. Veloso, V. Almeida, W. Meira and A. Bestavros, etc., "A hierarchical characterization of a live streaming media workload," *IEEE/ACM Trans. on Networking*, vol. 14, no.1, pp. 133-146, 2006. Article (CrossRef Link)

Jianguang Weng received his PhD degree from Zhejiang University, China, 2005. After that, He worked in College of Computer Science and Technology, Zhejiang University as Postdoctor. From 2007 to now, he worked in Zhejiang Institute of Media and Communications. During this time, his research was mainly focusing on P2P and Live streaming. He has published more than 15 papers and applied for 2 Chinese patents. His Research interests include Multimedia, Nature computing and Visualization.

Xuelan Zou is a lecturer in Zhejiang Institute of Media and Communications, Hangzhou, P.R. China. She received the Ph.D. degree in communications from Nanjing University of Posts and Telecommunications in 2008. Her research interests include error-control coding, iterative decoding, wireless communication networks and the applications of evolutionary techniques to network design and optimization.

Minhong (Maggie) Wang is an Associate Professor in the Faculty of Education, The University of Hong Kong. She has been involved in multiple disciplinary research in the areas of technology-enhanced learning, complex problem solving and learning, knowledge management, adult learning and human performance, and artificial intelligence. She has published papers in Computers & Education, IEEE Transactions on Information Theory, Information & Management, IEEE Transactions on Education, Educational Technology & Society, Innovations in Education & Teaching International, Expert Systems with Applications, Knowledge-based Systems, Journal of Knowledge Management, among others. She is the Editor-in-Chief of Knowledge Management and E-Learning: an International Journal, and the Editor of KSII Transactions on Internet and Information Systems. She also serves on the editorial board of several international journals including Educational Technology Research and Development, and Educational Technology & Society.