Received: 30 June 2016 | Revised: 9 August 2016 | Accepted: 7 September 2016
DOI: 10.1002/cpe.3985

WILEY

SPECIAL ISSUE PAPER

The packing problem of uncertain multicasts

Bangbang Ren[1] | Deke Guo[1,3] | Junjie Xie[1] | Wenxin Li[2] | Bo Yuan[1] | Yunfei Liu[1]

[1]Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha, China
[2]School of Computer Science and Technology, Dalian University of Technology, Dalian, China
[3]Guangxi Cooperative Innovation Center of Cloud Computing and Big Data, Guangxi Colleges and Universities Key Laboratory of Cloud Computing and Complex Systems, Guilin University of Electronic Technology, Guilin, China

Correspondence
Deke Guo, School of Information System and Management, National University of Defense Technology, Changsha, China. Guangxi Cooperative Innovation Center of Cloud Computing and Big Data, Guangxi Colleges and Universities Key Laboratory of Cloud Computing and Complex Systems, Guilin University of Electronic Technology, Guilin 541004, China.
Email: dekeguo@nudt.edu.cn

Funding information
National Natural Science Foundation, Grant/Award Number: 61422214; National Basic Research Program, Grant/Award Number: 2014CB347800; Hunan Provincial Natural Science Fund for Distinguished Young Scholars, Grant/Award Number: 2016JJ1002; Distinguished Young Scholars of NUDT, Grant/Award Number: JQ14-05-02; Preliminary Research Funding of NUDT, Grant/Award Number: ZDYYJCYJ20140601

Summary

Multicast performs better than unicast in delivering the same content from a fixed single source to a set of destinations. Many efforts have been made to optimize such kind of deterministic multicast, such as minimizing the transmission cost of each multicast session. In practice, it is not necessary that the source of each multicast session has to be in a specific location, as long as certain constraints are satisfied. Accordingly, applications usually meet a novel multicast with uncertain sources, ie, uncertain multicast. That is, multiple nodes have the responsibility to act as the root node of a multicast session. Prior proposals have addressed an uncertain multicast by constructing the minimum cost forest. However, it is still unknown how to efficiently share the network resources, when a set of uncertain multicast occupies the network simultaneously. To tackle such a challenging issue, we present the packing problem of uncertain multicasts (MPU) to minimize the total transmission cost, under the constraint of link capacity. We prove that the MPU problem is NP-hard. An intrinsic solution is constructing the minimum cost forest for each uncertain multicast individually. This method, however, is inefficient and may be infeasible because of the constraint of link capacity. Thus, we design 2 dedicated greedy methods, named priority-based and adjusting congested link, to approximate the optimal solution. The comprehensive results indicate that both of our 2 methods can find a feasible solution for the MPU problem. Moreover, given a set of uncertain multicasts, the adjusting congested link method can generate a desired transmission structure for each uncertain multicast and achieve the least total cost when packing them.

KEYWORDS

packing problem, routing algorithm, uncertain multicast

1 | INTRODUCTION

Multicast is designed to jointly deliver the same content from a single source to a set of destinations. Compared with unicast, multicast can save bandwidth efficiently because it can significantly reduce unnecessary duplicated transmissions among a series of independent unicast paths toward different destinations.[1] Given a multicast session, it is critical to construct a desired tree structure to span the source node and all destination nodes. There are 2 different optimization goals when constructing a multicast tree. The first one is to minimize the transmission cost or delay for each destination, while the second one focuses on minimizing the total transmission cost or delay for all destinations.[2] There are several available methods to accomplish the first goal by employing the shortest path routing method.[3] The behind insight is to derive the shortest path from the common source to each destination independently and combine such shortest paths as a tree. The second goal can be achieved through the well-known Steiner minimum tree (SMT) problem in any general graph.[4] Many efficient methods have been proposed to approximate the optimal solution of this NP-hard problem.[5–7] A common feature of the aforementioned methods is that they always consider the deterministic multicast session with 1 fixed source node.

As pointed in one study,[8] it is not necessary that the source of a multicast session has to be in a specific location as long as certain constraints are satisfied. A major reason is that the widely used content replica designs for improving the robustness and efficiency in various networks, such as the content distribution network, Internet Protocol television networks,[9] enterprise networks,[10,11] and data center networks.[12,13] When a multicast session has multiple potential sources, each destination may have opportunity to select any replica node as its source in theory. In this setting, we call such kind of multicast problem as the uncertain multicast. Compared with the traditional deterministic multicast, the resultant routing structure of an uncertain multicast is a minimum cost forest (MCF). It consists of multiple disjoint trees, which root at different sources.[8]

For a single deterministic or uncertain multicast, the tree or forest building method does not consider the constraint of each link capacity. In more general scenarios, it is very usual that many multicast sessions occur simultaneously. If the capacity of each occupied link is always adequate, each multicast session can achieve the desired SMT or MCF for each deterministic or uncertain multicast. However, in practice the capacity of each link along the desired path between the source and 1 destination may be insufficient to accommodate the flow. That is, some links may be blocked when constructing the desired SMT or MCF. Accordingly, the building method should find another routing path to accommodate the flow.

In this general scenario, we must systematically schedule a series of multicast sessions to find an available routing for each multicast session. This requirement is formulated as the packing problem of multicast sessions, which has been well studied for deterministic multicast sessions.[14] However, it is still unknown how to carefully schedule a set of uncertain multicast sessions. It is impossible to satisfy all uncertain multicast sessions with their optimal MCFs.

In this paper, we formally present the packing problem of uncertain multicasts (MPU) to minimize the total transmission cost of all multicast sessions, under the constraint of link capacity. For a single uncertain multicast, our earlier work has designed an efficient method E-MCF to construct an MCF, spanning some source nodes and all destination nodes.[8] However, the connectivity of network may change as 1 multicast request be satisfied, and such a method cannot be directly used to address the MPU problem. For the MPU problem of multiple uncertain multicasts, we need to construct a reasonable forest for each multicast session and minimize the total cost of all resultant forests under the constraint of link capacity. We prove that the MPU problem is NP-hard. For this reason, we design 2 dedicated greedy methods, named priority-based (BP) and adjusting congested link (ACL), to approximate the optimal solution. We further conduct large-scale simulations to evaluate the performance of different solutions, under different number of uncertain multicast sessions and average bandwidth of each link. We then evaluate the impact of session size and network size on the MPU solutions and the impact of the number of sources. The comprehensive results indicate that both of our 2 methods can find a feasible solution for the MPU problem. Moreover, the ACL method can generate a desired forest structure for each uncertain multicast and achieve the least total cost for the MPU problem.

The rest of this paper is organized as follows. In Section 2, we describe the MPU problem using an illustrative example and formally model the MPU problem. In Section 3, 2 efficient methods are presented to approximate the optimal solution for the NP-hard MPU problem. The evaluation results are presented in Section 4. Sections 5 and 6 summarize the related work and conclude the whole paper, respectively.

2 | PROBLEM STATEMENT

We start with the important observation about the packing problem of uncertain multicasts and then give the notations of the problem and the formulation model.

2.1 | Observations

It is well known that the multicast protocol is an efficient way to deliver the same content to a group of destinations. It benefits in efficiently conserving the bandwidth and reducing the amount of caused network traffic, compared with the unicast protocol.[1] Recently, an important finding about multicast is that the source of a multicast session is not necessary to be in specific location, as long as certain constraints are satisfied. A major reason is that the widely used content replica designs for improving the robustness and efficiency in various networks. Thus, when addressing the routing problem of an uncertain multicast with multiple sources, the resultant routing structure is usually a forest, not a tree.

Additionally, several sessions of uncertain multicasts will usually occur simultaneously in many networks. In this scenario, each multicast session usually needs a multicast forest to accomplish requests. If the involved network resource is adequate, we can construct the desired minimal cost forest for each uncertain multicast. Otherwise, such multicast forests must cooperate with each other to satisfy the bandwidth constraint of each link. This fact brings a new optimization problem. That is, given a network and the capacity constraint on each link, we need to construct a multicast forest for each uncertain multicast session while reducing the total cost of all forests, instead of minimizing the cost of each multicast forest.

Figure 1 plots an illustrative example of the packing problem of uncertain multicast with multiple potential sources (MPU). Assume that all nodes can serve as different roles, such as source nodes, destination nodes, and even intermediate nodes. In Figure 1, the network topology is a random graph with 10 nodes. A 2-tuples(c_{ij}, b_{ij}) represents the cost and capacity of each link, respectively. Three uncertain multicast sessions, δ_1, δ_2, and δ_3, are injected into the network. In the session δ_1, a node set $\{v_5\}$ records the source nodes, and a node set $\{v_{10}, v_3, v_2, v_9, v_7\}$ records the destination nodes. In the session δ_2, a node set $\{v_6, v_9\}$ represents the source nodes, and a node set $\{v_2, v_4, v_1, v_{10}, v_7\}$ represents the destination nodes. In the session δ_3, a node set $\{v_2, v_8\}$ represents the source nodes, and a node set $\{v_9, v_4, v_1\}$ represents the destination nodes.

Given an uncertain multicast session, all potential source nodes and all destination nodes are usually insufficient to form a tree or forest

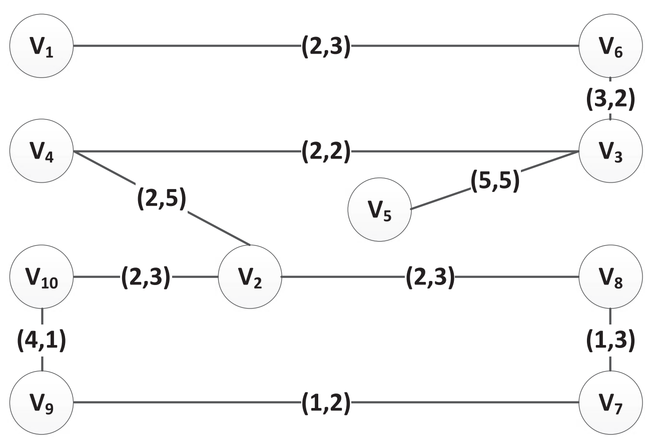


FIGURE 1 The initial network with the link weight (c_{ij}, b_{ij})

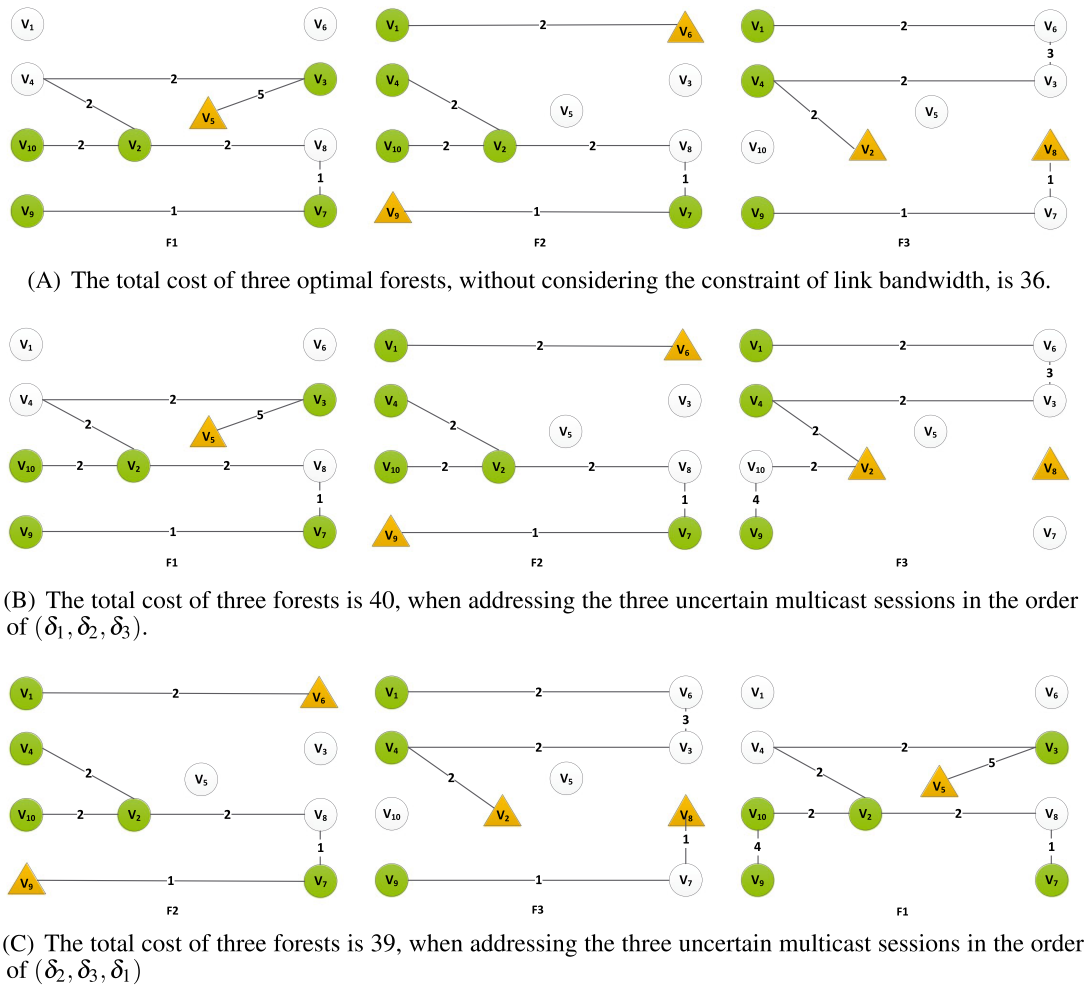


FIGURE 2 An illustrative example of the packing problem of uncertain multicasts. A, The total cost of 3 optimal forests, without considering the constraint of link bandwidth, is 36. B, The total cost of 3 forests is 40, when addressing the 3 uncertain multicast sessions in the order of $(\delta_1, \delta_2, \delta_3)$. C, The total cost of 3 forests is 39, when addressing the 3 uncertain multicast sessions in the order of $(\delta_2, \delta_1, \delta_3)$

structure to span them. It often needs to employ other intermediate nodes to span them. In Figure 2, an ellipse with shadow represents a destination node, a triangle represents a source node, and all ellipse nodes without shadow represent intermediate nodes in the resultant forest. If we calculate the MCF for each uncertain multicast session using our prior E-MCF method,[8] the total cost of these 3 forests is 36. However, because of the capacity constraint of links (v_7, v_9), such forests cannot be accommodated by the network simultaneously. If we persist to accomplish such uncertain multicast requests, we must design different strategy for constructing the required forest structure, which may exhibit high cost than that forest derived from our prior E-MCF method. In Figure 2B, we pack these uncertain multicasts in the order of $\delta_1, \delta_2, \delta_3$. The total cost of 3 updated forests is 40. Additionally, in Figure 2C, we pack these uncertain multicasts in the order of $\delta_2, \delta_1, \delta_3$. Accordingly, the total cost of 3 new forests is 39.

Such observations motivate us to resolve the packing problem of uncertain multicasts. A simple method for this problem is to address all uncertain multicasts in the initial sequence, abbreviated as the One-by-One method. That is, we first calculate the optimal forest with as low cost as possible for the first multicast session and then calculate the reasonable forest for other multicast sessions under the available bandwidth of each link one by one. We can conclude from Figure 2 that tackling a set of uncertain multicasts cooperatively will result in a better packing solution. In Figure 2C, we jointly tackle the 3 uncertain multicasts in the order of δ_2, δ_3, and δ_1. Accordingly, the total transmission cost of all uncertain multicasts is 39. Thus, the processing order among multiple uncertain multicasts is essential to the performance of the packing problem.

2.2 | Notations

We denote the physical network by $G = (V, E, \mathbf{c}, \mathbf{b})$, where V is the set of nodes, E is the set of network links, while $\mathbf{c} : E \rightarrow R^+$ and $\mathbf{b} : E \rightarrow Z^+$ denote the cost function and capacity function of the network, respectively. Assume that there are r multicast streams in the network, and each stream needs an uncertain multicast session to deliver. For each session, let w_k denotes the stream, S_k denotes the source nodes that can supply the stream, while D_k denotes destination nodes that request the stream. Let $|S_k|$ and $|D_k|$ be the cardinality of S_k and D_k, respectively. For an uncertain multicast, it is clear that $|S_k| \geq 1, |D_k| \geq 1, 1 \leq k \leq r$. We use $\delta_k = (w_k, S_k, D_k)$ to denote an uncertain multicast session. Let P denotes the subgraph of G, and then the cost of P is denoted by $\mathbf{c}(P) = \sum_{(i,j) \in E(P)} c_{ij}$.

2.3 | Problem modeling

The packing problem of uncertain multicasts, abbreviated as the MPU problem, can be formulated as follows. Given a set of uncertain multicast sessions in the network $G = (V, E, \mathbf{c}, \mathbf{b})$, the MPU problem involves 2 design rationales. First, it needs to find a multicast forest F_k for each multicast session in δ_k to deliver a stream w_k from any of the sources in S_k to all destinations in D_k. Second, the total cost of all resultant forests $\sum_{k=1}^{r} c(F_k)$ should be minimized. As aforementioned, when considering the routing problem of an uncertain multicast, the routing structure we got is usually a forest, not a tree. Accordingly, a multicast forest F_k consists of some isolated trees and satisfies the following constraints:

1. Any tree $T_i^k \in F_k$ is rooted at a single source in S_k.
2. Any pair of source nodes in F_k are not reachable with each other.
3. Any destination node connects with only 1 source node.

We assume that any multicast stream will consume 1 unit of the link capacity when it passes a link. Thus, the number of multicast streams carried by each link cannot exceed its link capacity. For example, Figure 1 denotes an undirected network. Each link (i, j) is associated with a 2-tuples (c_{ij}, b_{ij}), where c_{ij} and b_{ij} represent the communication cost and the available capacity of that link, respectively. Consider 3 multicast sessions, $\delta_1, \delta_2, \delta_3$, which appear at the same time. Without considering the constraint of link capacity, we can realize each of such uncertain multicast sessions with the minimal cost forest, as shown in Figure 2A. However, the capacity of each involved link is not always sufficient to accommodate the 3 multicast sessions. In this scenario, Figure 2B,C reports 2 feasible solutions with different total cost. The packing problem of uncertain multicasts can be modeled as follows:

$$\text{Min} \sum_{k=1}^{r} c(F_k) \tag{1}$$

Subject to:

$$\sum_{k=1}^{r} x_{ij}^k \le b_{ij}, \forall (v_i, v_j) \in E(G) \tag{2}$$

$$D_k \subseteq V(F_k), 1 \le k \le r, \tag{3}$$

$$S_k \bigcap V(F_k) \ne \emptyset, 1 \le k \le r, \tag{4}$$

$$x_{ij}^k = 0 \text{ or } 1, \forall i, j, k, \tag{5}$$

where

$$x_{ij}^k = \begin{cases} 1 & \text{if } (i,j) \in E(F_k) \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

Constraint 2 ensures that the total number of multicast streams over each link does not exceed the link capacity. Constraint 3 ensures that each forest for an uncertain multicast session must span all destination nodes. Constraint 4 ensures that each forest for an uncertain multicast session must span some source nodes to deliver its stream, which may be 1 node or even all potential source nodes. Constraints 5 and 6 indicate whether a link is used by a multicast forest.

2.4 | NP-hardness

When $r = 1$, it means that there is just 1 stream in the network. In this setting, the MPU problem is just equivalent to the minimal cost forest problem of an uncertain multicast proposed in our prior work.[8] For a traditional multicast with a single source, the SMT problem[4] is NP-hard in a general graph. It has been proved that the uncertain multicast is more difficult than the traditional deterministic multicast because of the flexible use of multiple potential sources. Therefore, it is easy to derive that the MPU problem with $r \ne 1$ is harder than the MCF problem of a single uncertain multicast. Thus, the MPU problem of a set of uncertain multicasts is also an NP-hard problem.

3 | DESIGN OF APPROXIMATION METHODS

Consider that finding the MCF for an uncertain multicast with multiple sources is NP-hard, thus, there exists no polynomial-time method to find the exact optimal solution for it. However, it is reasonable to approximate the optimal solution using any approximation method. In this paper, we select the approximation method E-MCF.[8] Let $F_{MCF}(G, \delta)$ denotes the resultant forest after applying the E-MCF method, given a multicast session δ in the network G. According to the design insight of the E-MCF method, we further propose the following methods for the MPU problem.

One intrinsic method is to satisfy all uncertain multicast sessions one by one abbreviated as the One-by-One method. This method first applies the E-MCF method to solve a multicast session δ_i with the network G and then apply the E-MCF method to the next multicast session δ_{i+1} with the residual network $R = G - F_{MCF}(G, \delta_i)$. As shown in Figure 2, the processing order of multicast sessions has crucial impact on the total cost of all resultant forests. For this reason, we propose an efficient method to solve the MPU problem with the consideration of priority.

3.1 | Approximation method based on priority

Given 2 multicast forests, it is not reasonable to simply say that the forest with the lower cost is better than the other, without considering the size of each uncertain multicast. To be able to decide which uncertain multicast should be satisfied first, we must normalize the cost of each forest after considering the characteristics of related uncertain multicast. In content distribution network,[15] a service provider will deploy more servers in different areas to satisfy more requests and reduce the number of congested links and overloaded servers. That means that an uncertain multicast session with more sources will perform better in finding new route and using the network resources. Thus, we propose a normalized metric to evaluate each multicast forest, and the definition is as follows:

Definition 1. The final priority of forest for uncertain multicast is calculated by $\mathbf{c}(F_i) \times |D_i|/|S_i|$.

Definition 1 shows that we prefer to satisfy those multicast requests with more destination nodes and less source nodes. When packing a set of uncertain multicasts, we prefer to first satisfy such multicast sessions with high priority. The major idea of our BP method is given as follows. At first, we invoke the E-MCF method to derive k MCFs, $F_{MCF}(G, \delta_1)$, $F_{MCF}(G, \delta_2)$, …, $F_{MCF}(G, \delta_k)$, for k uncertain multicast sessions, δ_1, δ_2, …, δ_k, respectively. Then we calculate the priority for each multicast session. Without loss of generality, we assume that $F_{MCF}(G, \delta_i)$ has the highest priority; hence, $F_{MCF}(G, \delta_i)$ is put into the solution set. After arranging the δ_i, we get the residual network $R = G - F_{MCF}(G, \delta_i)$ with updated network capacity. In the second iteration, we apply the E-MCF method to find $k-1$ multicast forests, $F_{MCF}(R, \delta_1)$, $F_{MCF}(R, \delta_2)$, …, $F_{MCF}(R, \delta_{i-1})$, $F_{MCF}(R, \delta_{i+1})$, …, and $F_{MCF}(R, \delta_k)$ for the remaining $k-1$ multicast sessions in the residual network R. Similarly, we find a multicast forest with the highest priority among such $k-1$ forests and then add this forest into the solution set and update the residual network. The above process repeats until k multicast forests are all determined. The derived solution set must be a reasonable solution because every solution is feasible under the current residual network. The method details are shown in Algorithm 1.

Theorem 1. The time complexity of the BP method is $O(r^2 \times |V|^4)$.

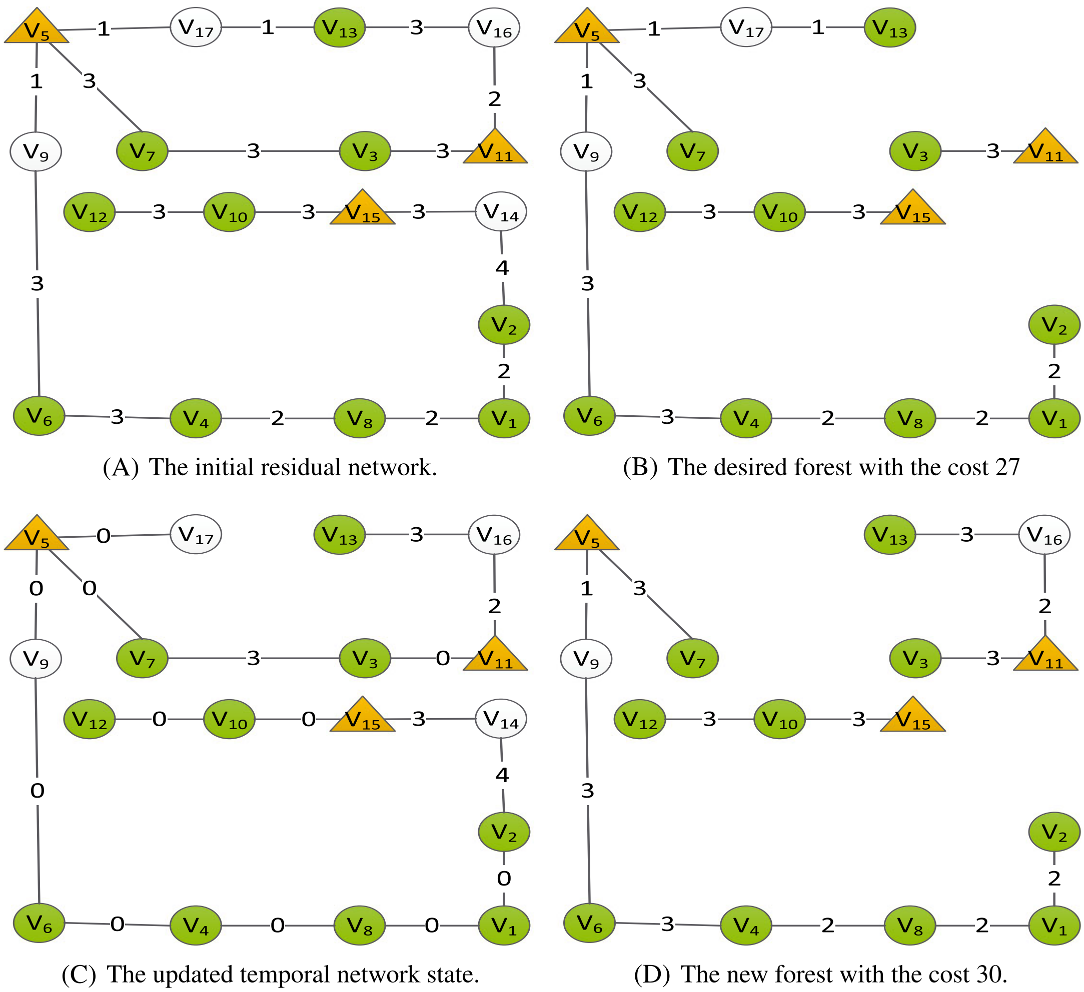
Proof. In each iteration in Algorithm 1, we select 1 optimal multicast forest with the highest priority. After selecting 1 multicast forest, the related multicast session should be ignored in the next iteration. Thus, we have to execute the E-MCF algorithm $\left(\frac{1+2+\ldots+r}{2}\right)$ iterations. Note that the time complexity of E-MCF is $O\left(\left(|S||D| + \frac{|D||D-1|}{2}\right) \times |V|^2\right)$.[8] In our packing problem, it is obviously that the number of source and destination nodes cannot exceed $|V|$. Accordingly, we can conclude that the time complexity of our BP method is $O\left(\left(|S||D| + \frac{|D||D-1|}{2}\right) \times \frac{r(r+1)|V|^2}{2}\right) = O(r^2|V|^4)$. Hence, Theorem 1 is proved. □

Algorithm 1 The priority-based method for the MPU problem

Require: An undirected network $G = (V, E, c, b)$ and r multicast sessions $\delta_1, \delta_2, \ldots, \delta_r$.

Ensure: A feasible solution set $\{F_1, F_2, \ldots, F_n\}$

Let R denotes the residual network, and $R = G$ initially.
Let $F_1, F_2, \ldots, F_r$ denote the resultant forests for the MPU problem, while $F'_1, F'_2, \ldots, F'_r$ denote the temporary solution.
Let $I = \{1, 2, \ldots, r\}$ denotes the indication of unsolved multicast sessions.
for $m = 1$ to r **do**
  for all $i \in I$, $F'_i = F_{MCF}(R, \delta_i)$.
  F_i is equal to F'_i which has the highest priority using $\mathbf{c}(F'_i) \times |D_i|/|S_i|$.
  $R = R - F_i$, $I = I - i$.
end for



(A) The initial residual network. (B) The desired forest with the cost 27

(C) The updated temporal network state. (D) The new forest with the cost 30.

FIGURE 3 An example of replacing a congest link with another shortest path. A, The initial residual work; B, The desired forest with the cost 27; C, The updated temporal network state; D, The new forest with the cost 30

3.2 | Approximation method by adjusting congested links

In the field of traffic engineering,[16] Internet service provider will provide load balance to use the network resource efficiently.[17] Section 2.1 shows that the result of simply combing all forests resulting from the E-MCF method may be infeasible, because of the constraint of link capacity. This fact motivates us to pay more attention to those congested links to generate a feasible solution for the MPU problem. Thus, we design an approximation method by ACLs. The main idea is that we initially apply the E-MCF method to calculate the MCF for each multicast session and get $F_{MCF}(G,\delta_1), F_{MCF}(G,\delta_2), \dots, F_{MCF}(G,\delta_r)$. For simplicity, we let F_i denotes $F_{MCF}(G,\delta_i)$. If the network accommodates those uncertain multicast sessions according to the above forests, some congested links would overload. To be able to get a feasible solution, we must make efforts to balance the traffic load on such congested links. Thus, we propose the second approximation method, which adjusts the use of those congested links (ACL). That means for those forests containing congested links, we delete a congested link and reconnect the 2 end points with a new shortest path.

For example, if link e_{ij} is congested, it is necessary to find a path ($v_i, \dots, v_k, \dots, v_j$) to replace the original link e_{ij}. Note that, it has to satisfy the 3 constraints for a multicast forest in Section 2.3 after introducing the new path and removing the original link. Figure 3A shows a residual network, in which the capacity of each link is sufficient, and the weight of each link denotes the link cost. Assume that there exists a multicast session δ in the network. The set of source nodes is $\{v_5, v_{11}, v_{15}\}$, and the set of destination nodes is $\{v_1, v_2, v_3, v_4, v_6, v_7, v_8, v_{10}, v_{12}, v_{13}\}$. Figure 3B reports an desired forest for the uncertain multicast session δ.

To ease the presentation, we assume that link $e_{13,17}$ is congested. Accordingly, the destination node v_{13} cannot get its request from source node v_5. Figure 3D reports a new solution, where node v_{13} reaches source node v_{11} along another path (v_{13}, v_{16}, v_{11}). Thus, the new forest spans all destination nodes and node v_{17}, which is an intermediate node in the original desired forest. The node v_{17} should be removed from the forest to reduce the cost of the MPU problem because it is neither a destination node nor an intermediate node in the new forest. Moreover, we must ensure that each tree in the new forest has just 1 source node, and every pair of source nodes is not reachable.

To get the solution as shown in Figure 3D, we must take steps as follows. First, after deleting the congested link $e_{13,17}$, we will get 4 connected components in the initial desired forest, rooted at nodes v_5, v_{11}, v_{15}, respectively. Node v_{13} is an single component with itself, and other 3 components are trees, each of which roots at a distinct source node. Second, to reconnect destination node v_{13} to a source node, we just need to find a shortest path from node v_{13} to any other connected component. Thus, we need to calculate the shortest path between v_{13} and each node in the connected component. This step would execute the Dijkstra algorithm many times. However, if we regard a connected component as a whole and set the cost of links in component as 0, then we can calculate the shortest path between v_{13} and any one node in connected components. In this example, we will get 3 shortest paths to the 3 connected components. Because the paths inside each connected component will not bring extra cost, the shortest paths are also the shortest paths connected corresponding source nodes. In the process of finding shortest paths, the state of some involved links needs to be updated temporarily, as shown in Figure 3C. The cost of link in original forest is 0, the cost of other links with sufficient capacity keeps unchange, and the cost of blocked links becomes $+\infty$. The updated temporal state is just used to calculate the shortest path; it does not present the real state of links. Finally, after getting all shortest paths between node v_{13} and all independent connected components, we select the shortest path with the minimum cost and add it into the new forest and delete those leaf nodes that are not destination nodes. The details are shown in Algorithm 2.

Algorithm 2 The ACL method for our packing problem

Require: An undirected network $G = (V, E, c, b)$ and r uncertain multicast sessions, $\delta_1, \delta_2, \dots, \delta_r$.
Ensure: A feasible solution set $F = \{F_1, F_2, \dots, F_r\}$

```
Calculate the MCF for each uncertain multicast session using the
  E-MCF method, denoted as F = {F_1, F_2, ..., F_r}.
Get the sum of all initial forests, and then subtract (G, b) to get
  the residual network R and those congested links, noted as CL =
  {CL_1, CL_2, ..., CL_j}. Here, b_{CL_m} denotes the capacity of a congested
  link CL_m.
for m = 1 to j do
    Find all forests covering the link CL_m, and denote these forests as
      set Φ = {Φ_1, Φ_2, ..., Φ_l}.
    for n = 1 to |Φ| − b_{CL_m} do
        Delete link CL_m in Φ_n, update the residual network state R with
          new temporal links cost c'.
        Find the shortest path P_n using c' toward one source node to
          replace the CL_m in Φ_n
        Add path P_n to Φ_n.
        Delete leaf nodes that are not destination nodes, and update
          the correspond forest in the solution set {F_1, F_2, ..., F_n} and
          residual network R
    end for
end for
```

Theorem 2. The time complexity of our ACL method for the MPU problem is $O(r \times |V|^4)$.

Proof. In Algorithm 2, we first calculate the optimal forest for each uncertain multicast session with the time complexity of $O(r \times ((|S||D| + \frac{|D||D-1|}{2}) \times |V|^2))$. Note that $|S|$ and $|D|$ for each uncertain multicast session cannot exceed $|V|$. Thus, the total time complexity of this step is $O(r \times |V|^4)$. To further find a substituted path for each congested link, we need to execute the Dijkstra algorithm at most $|CL| \times r \times c$ rounds. Here, c denotes the number of connected components. Consider that $|CL| < |E| < |V|^2$ and $c < |V|$, so the time complexity of this step is $O(r \times |V|^3)$. Finally, deleting leaf nodes that are not destination nodes incurs the time complexity of $O(|V|^2)$. Thus, the time complexity of the whole process is $O(r \times |V|^4)$; hence, Theorem 2 is proved. □

4 | PERFORMANCE EVALUATION

In this section, we evaluate our 2 approximation methods for the MPU problem through extensive simulations. Unless otherwise specified, the evaluation settings are as follows. A total of 200 nodes are uniformly allocated in a 100 × 100 grid, while any pair of nodes are established a link with a given probability. The cost of each link is calculated by the distance between the 2 nodes. The link bandwidth follows the normal distribution, with the expected value B_m and the standard deviation σ. We assume that each uncertain multicast session will consume 1 unit bandwidth when it passes through a link.

4.1 | Impact of the number of multicast sessions

Figure 4 shows the changing trend of the total cost of our MPU problem when the number of uncertain multicast sessions grows up. In our evaluations, the average bandwidth of links B_m is set as 40 units, while the standard deviation σ is 20. Each multicast session shares the same settings, including 5 sources and 45 destinations. We report the average result after 10 rounds of simulations for each setting of the number of multicast sessions in Figure 4.

We can see that the curve of our ACL method is always below the curve of BP and One-by-One methods. Additionally, the gap between our 2 curves of ACL and BP methods grows up, as the increasing number of uncertain multicast sessions. The behind reasons are as follows. First, the BP and One-by-One methods exhibit almost same level of performance. The cause is that our BP method does not improve the MCF for each uncertain multicast session directly. Actually, it focuses on finding a strategy to derive a reasonable processing sequence for all uncertain multicast sessions. The strategy we take in this evaluation is to select the MCF with the lowest cost among all multicast sessions. Thus, the overall cost of our BP method will be smaller than that of our One-by-One method, while the difference between them may not obvious when the bandwidth of each involved link is sufficient. Second, the performance of our ACL method is better than that of other 2 methods, because the ACL method focuses on adjusting all congested links in the original forests with low-cost alternative paths. Thirdly, the performance difference between our ACL and BP methods grows up as the increasing number of uncertain multicast sessions. The reason is that more multicast sessions may lead to more congested links. As shown in Table 1, the number of congested links increases as the number of multicast sessions grows up. Table 2 reports the execution time of 3

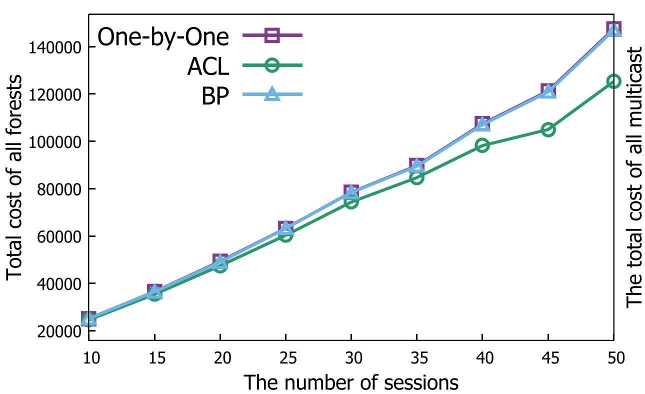


FIGURE 4 The total cost of forests vs the number of uncertain multicast sessions. ACL indicates adjusting congested link; BP, priority-based

TABLE 1 The number of congest links under varied number of uncertain multicast sessions

Sessions	10	15	20	25	30	35	40	45	50
Congested links	7	10	13	16	20	30	31	42	50

TABLE 2 Execution time(s) of different methods under varied number of uncertain multicast sessions

Number of Sessions	One-by-One	BP	ACL
10	1.66	9.17	1.10
15	2.40	19.20	2.04
20	3.21	33.828	3.20
25	3.99	51.52	5.83
30	4.82	74.03	8.08
35	5.60	101.48	13.43
40	6.42	131.39	18.41
45	7.14	166.07	26.97
50	8.13	205.13	36.97

Abbreviations: ACL, adjusting congested link; BP, priority-based.

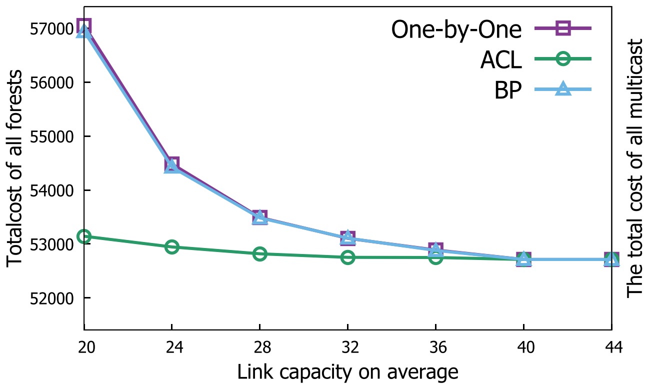


FIGURE 5 The total cost of forests vs the average link capacity. ACL indicates adjusting congested link; BP, priority-based

methods. We can see that our BP method consumes more time as the time complexity indicates.

4.2 | Impact of the link capacity on average

Given a set of uncertain multicasts, Figure 5 shows the changing trend of the total cost as the average link capacity increases. In this experiment, the average link capacity gradually increases, while other parameters are fixed. We can see that the gap between the BP and ACL methods is high when the average link capacity is not sufficient, and the gap decreases as the average link capacity becomes sufficient. The behind reason is that the network can satisfy more uncertain multicast sessions when it exhibits more sufficient link capacity. To the extreme, all uncertain multicast sessions can be satisfied with the optimal multicast forests because each link no longer suffers the capacity constraint.

4.3 | Impact of the session size and network size

We further evaluate the execution time of our ACL method under different size of a multicast session and varied size of the network.

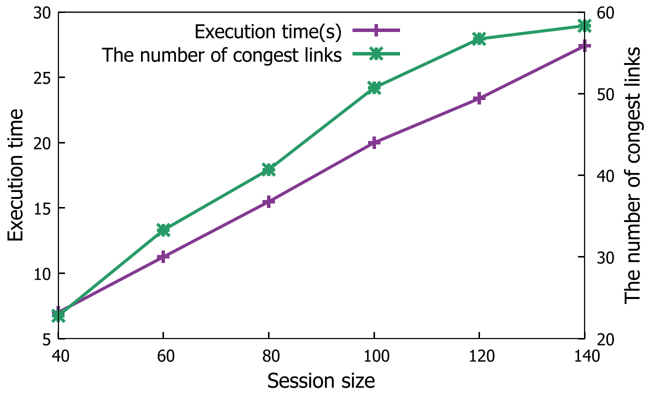


FIGURE 6 The impact of the session size under our adjusting congested link method

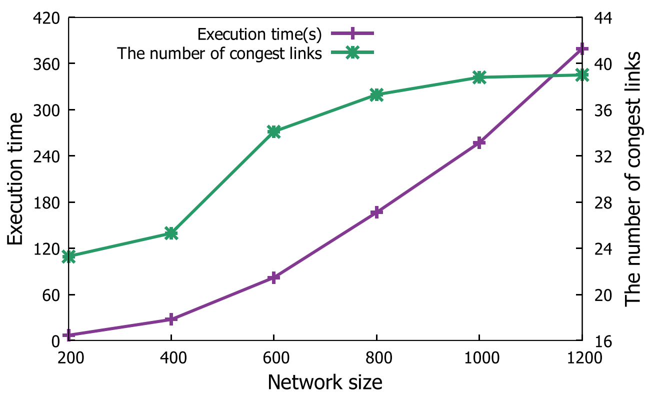


FIGURE 7 The impact of the network size under our adjusting congested link method

The multicast session size is measured by the total number of source nodes and destination nodes. Note that the settings of other parameters do not change. Figures 6 and 7 show that the number of congested links and the execution time always grow up along with the increase of the session size and network size. Thus, it will occupy more links to enable a large multicast session. This will improve the congestion probability for each link. Figure 6 also shows that the execution time grows up along with the increasing number of congested links. We further evaluate the impact of the network size, while the settings of other parameters are also fixed. Figure 7 reports the evaluation result. As aforementioned, the time complexity of our ACL method is $O(r|V|^2)$. This conclusion indicates that the execution time will grow along with the increase of network size in theory. Note that the evaluation results report the similar conclusion.

4.4 | Impact of the average number of sources in uncertain multicasts

We evaluate the impact of the average number of sources in uncertain multicasts under our ACL method. We can see from Figure 8 that the total cost of all uncertain multicasts and the execute time decreases along with the increasing number of sources. For an uncertain multicast, it is obvious that more sources can decrease the cost of the multicast forest, because of the high probability to pick the best source.

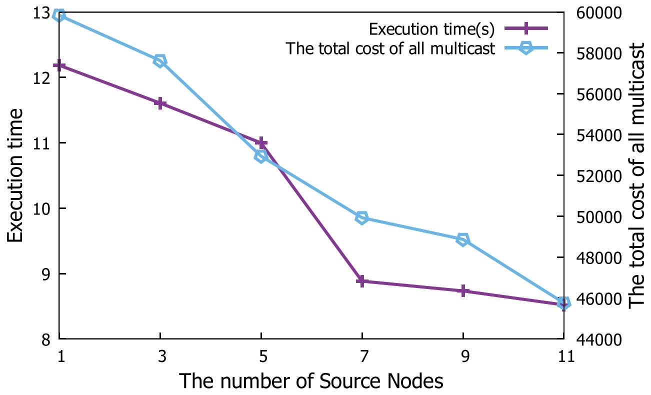


FIGURE 8 The impact of the number of sources under our adjusting congested link method

There may exist less number of congested links because each multicast forest may use less number of links. This result provides evidence to reveal that more sources can save the network resource significantly.

5 | RELATED WORK

Given a group of nodes, the group multicasting[18] allows each member node to multicast a stream to all other members. That is, each member node can serve as a source node, which launches a multiple stream in the group. Some proposals have investigate the problem of multicast tree packing under different optimization objectives.

For example, several literatures motivate to minimize the total cost of all multicast trees,[19-21] while others try to avoid the network bottlenecks, so as to support as many applications as possible with the guaranteed quality of service.[2,14,22,23] Chen et al considered the multicast tree packing problem with group multicasting.[14] They assume that each multicast tree requires the same amount of bandwidth and devotes to minimize the maximum congestion. Literature[24] analyzed the similar problem while each multicast tree has different bandwidth requirement. In addition, some researchers reconsider the tree packing problem under different optimization objectives. For example, they aim to reduce the total cost of trees,[19-21] where the cost can be represented as the communication cost or delay.

Routing algorithm is an efficient method to solve network resource through reducing redundance in transmission. From information theory, we know that we can also save resource by compressing information in source node with proper coding. The network coding[25] can considerably save the network bandwidth by employing the dedicated coding strategy at involved network nodes. For such reasons, literature[26] tried to combine the benefits of multicast and network coding. More precisely, they proposed the multiple multicast sessions with intra-session network coding.

The above related works assume that each of all multicast sessions to be packed has just 1 single source node. This makes that the spanning subgraph for each multicast session is a tree structure, rooted at 1 source node. In real network, it is not necessary that the source of a multicast session has to be in specific location, as long as certain constraints are satisfied. Chen et al consider the packing problem with multisource multicast;[23] the MMForest algorithm can establish a multicast forest to span all destinations and sources under the constraint of achieving the maximal residual bandwidth. MMForest makes each destination select 1 path with the largest residual bandwidth toward all sources and then merges those selected paths from all destinations. Obviously, this method loses nontrivial opportunities to maximize the number of shared links among individual paths.

6 | CONCLUSION

The uncertain multicast can significantly reduce the incurred transmission cost, compared to the traditional deterministic multicast. In the real network, more than 1 uncertain multicast sessions usually share the common network simultaneously. In this paper, we propose the packing problem of uncertain multicasts (MPU). The major motivation is to reduce the total transmission cost resulting from supporting a set of uncertain multicast at the same time under the constraint of link capacity. To tackle such a challenging issue, we first report an intrinsic One-by-One method. We then design other 2 efficient methods, BP and ACL, to approximate the optimal solution of the MPU problem. The BP method focuses on evaluating a multicast forest in each round and selecting the best one. The ACL method focuses on tackling congested links by partially updating involved routing paths. The evaluation results demonstrate that our BP and ACL methods can find a reasonable solution for the MPU problem. Moreover, our ACL method achieves better performance than the other 2 methods for the overall transmission cost, because of a set of uncertain multicasts.

ACKNOWLEDGMENTS

This work is supported in part by the National Natural Science Foundation for Outstanding Excellent young scholars of China under grant no. 61422214, National Basic Research Program (973 program) under grant no. 2014CB347800, the Program for New Century Excellent Talents in University, the Hunan Provincial Natural Science Fund for Distinguished Young Scholars under grant no. 2016JJ1002, the Distinguished Young Scholars of NUDT under grant no. JQ14-05-02, the Preliminary Research Funding of NUDT under grant no. ZDYYJ-CYJ20140601, the Guangxi Cooperative Innovation Center of cloud computing and Big Data, and the Guangxi Colleges and Universities Key Laboratory of cloud computing and complex systems.

REFERENCES

1. Malli R, Zhang X, Qiao C. Benefits of multicasting in all-optical networks. Photonics East (ISAM, VVDC, IEMB). *Int Soc Opt Photonics*. 1998;3531:209–220.
2. Noronha C, Tobagi F. Optimum routing of multicast streams. *In Proc IEEE Infocom: Networking Global Commun*. 1994;2:865–873.
3. Deering SE, Cheriton DR. Multicast routing in datagram internetworks and extended LANs. *ACM Trans Comput Syst (TOCS)*. 1990;8(2):85–110.
4. Smith WD, Shor PW. Steiner tree problems. *Algorithmica*. 1992;7(1):329–332.
5. Mehlhorn K. A faster approximation algorithm for the Steiner problem in graphs. *Inf Process Lett*. 1988;27(3):125–128.
6. Winter P, Smith MG. Path-distance heuristics for the Steiner problem in undirected networks. *Algorithmica*. 1992;7(1-6):309–327.
7. Koch T, Martin A. Solving Steiner tree problems in graphs to optimality. *Networks An Int J*. 2012;32(3):207C232.
8. Hu ZY, Guo DK, Xie JJ, Ren BB. Multicast routing with uncertain sources in software-defined network. *Proceedings of the IEEE conference on Symposium on Quality of Service(IWQoS)*, Beijing; 2016:319–324.
9. Mahimkar AA, Ge Z, Shaikh A. et al. Towards automated performance diagnosis in a large IPTV network. *ACM SIGCOMM Comput Commun Rev*. 2009;39(4):231–242.
10. Wu K, Li H, Wang L. et al. hJam: attachment transmission in WLANs. *IEEE Trans Mobile Comput*. 2013;12(12):2334–2345.
11. Wang G, Zhang S, Wu K, Zhang Q, Ni L. TiM: fine-grained rate adaptation in WLANs. *IEEE Trans Mobile Comput*. 2016;15(3):748–761.
12. Li D, Li Y, Wu J, et al. ESM: efficient and scalable data center multicast routing. *Networking IEEE/ACM Trans*. 2012;20(3):944–955.
13. Li D, Xu M, Liu Y, et al. Reliable multicast in data center networks. *IEEE Trans Comput*. 2014;63(8):2011–2024.
14. Chen S, Gunluk O, Yener B. Optimal packing of group multicastings. *Proc Int Conf Comput Commun*. 1998;3:980–987.
15. Furht B. Content distribution network. Research Proficiency Exam Report; 2004.
16. Awduche D, Malcolm J, Agogbua J, et al. Requirements for traffic engineering over MPLS. *J Evidence-Based Soc Work*. 1999;10(10):63–72.
17. Pham PP, Perreau S. Increasing the network performance using multi-path routing mechanism with load balance. *Ad Hoc Networks*. 2004;2(4):433–459.
18. Fei A, Duan Z, Gerla M. Constructing shared-tree for group multicast with QoS constraints. *Proceedings of the IEEE Conference on Global Telecommunications, San Antonio*; 2001;4:2389–2394.
19. Wang CF, Liang CT, Jan RH. Heuristic algorithms for packing of multiple-group multicasting. *Comput Oper Res*. 2002;29(7):905–924.
20. Jia X, Wang L. A group multicast routing algorithm by using multiple minimum Steiner trees. *Comput Commun*. 1997;20(9):750–758.
21. Low CP, Wang N. An efficient algorithm for group multicast routing with bandwidth reservation. *Computer Communications*, 2000, 23(18):1740–1746.
22. Chen S, Gnlk O, Yener B. The multicast packing problem. *IEEE/ACM Trans Networking*. 2000;8(8):311–318.
23. Chen YR, Radhakrishnan S, Dhall S, et al. On multi-stream multi-source multicast routing. *Comput Networks*. 2013;57(15):2916–2930.
24. Lee CY, Cho HK. Multiple multicast tree allocation in IP network. *Comput Oper Res*. 2004;31(7):1115–1133.
25. Gruska DP. Network information flow. *Fundamenta Informaticae*. 2006;72(1):167–180.
26. Ho T, Viswanathan H. Dynamic algorithms for multicast with intra-session network coding. *Inf Theory IEEE Trans*. 2009;55(2):797–815.

How to cite this article: Ren B, Guo D, Xie J, Li W, Yuan B, Liu Y. The packing problem of uncertain multicasts. *Concurrency Computat: Pract Exper*. 2017;29:e3985. https://doi.org/10.1002/cpe.3985