doi:10.3772/j.issn.1006-6748.2017.01.005

Network coding resources optimization with transmission delay constraint in multicast networks[①]

Qu Zhijian (曲志坚)[②*], Fu Jia[**], Liu Xiaohong[*], Li Caihong[*]
([*] School of Computer Science and Technology, Shandong University of Technology, Zibo 255049, P. R. China)
([**] School of Information and Communication Engineering, Beijing University of Posts
and Telecommunications, Beijing 100876, P. R. China)

Abstract

Minimizing network coding resources of multicast networks, such as the number of coding nodes or links, has been proved to be NP-hard, and taking propagation delay into account makes the problem more complicated. To resolve this optimal problem, an integer encoding routing-based genetic algorithm (REGA) is presented to map the optimization problem into a genetic algorithm (GA) framework. Moreover, to speed up the search process of the algorithm, an efficient local search procedure which can reduce the searching space size is designed for searching the feasible solution. Compared with the binary link state encoding representation genetic algorithm (BLSGA), the chromosome length of REGA is shorter and just depends on the number of sinks. Simulation results show the advantages of the algorithm in terms of getting the optimal solution and algorithmic convergence speed.

Key words: network coding, genetic algorithm (GA), search space, multicast network

0 Introduction

Network coding provides numerous advantages over store-and-forward based routing solution[1]. In network coding research community, the network coding operations are always assumed that it has to be conducted in all coding-possible nodes to achieve the required throughput. However, it is usually the case that network coding is required only at a subset of the coding-possible nodes[2]. For example, a source node *s* wishes to transmit multicast messages to all of the destinations at the multicast rate of two. In order to guarantee the required multicast rate, a multicast tree may contains one to three coding nodes according to different multicast tree establishing algorithm. The problem of determining such a minimum coding nodes subset has been proved to be NP-hard[3].

Many scholars have concentrated on this problem and worked for minimizing the number of coding nodes or links. For example, Wang, et al.[4] modified the ant colony algorithm to optimize network coding resources, and Xing, et al.[5-7] proposed some evolutionary algorithms. Qu, et al.[8] proposed an adaptive quantum-inspired evolutionary algorithm based on Hamming distance to optimize the network coding resources in multicast networks. However, their work did not take delay or delay difference constraints into account.

Obviously, network coding will lead to increase the transmission delay and the computational overhead when combining packets in intermediate nodes[9]. To solve this problem, Wu, et al.[10] developed a joint coding and feedback scheme to improve the throughput and reliability in wireless multicast. The results showed that the delay, encoding, and decoding complexities were low even for a large number of receivers. To reduce the packets delay and minimize packet dropping probability, Yu, et al.[11] considered multiple transmission methods and integrated packet scheduling with adaptive network coding method selection, and presented a dynamic coding-aware routing metric, which could increase potential coding opportunities. Those literatures above show the necessity of researching on network coding with delay needs.

In real-time interactive applications, the efficien-

[①] Supported by the National Natural Science Foundation of China (No. 61473179), Shandong Province Higher Educational Science and Technology Program (No. J16LN20), Natural Science Foundation of Shandong Province (No. ZR2016FM18) and the Youth Scholars Development Program of Shandong University of Technology.

[②] To whom correspondence should be addressed. E-mail: zhijianqu@sdut.edu.cn

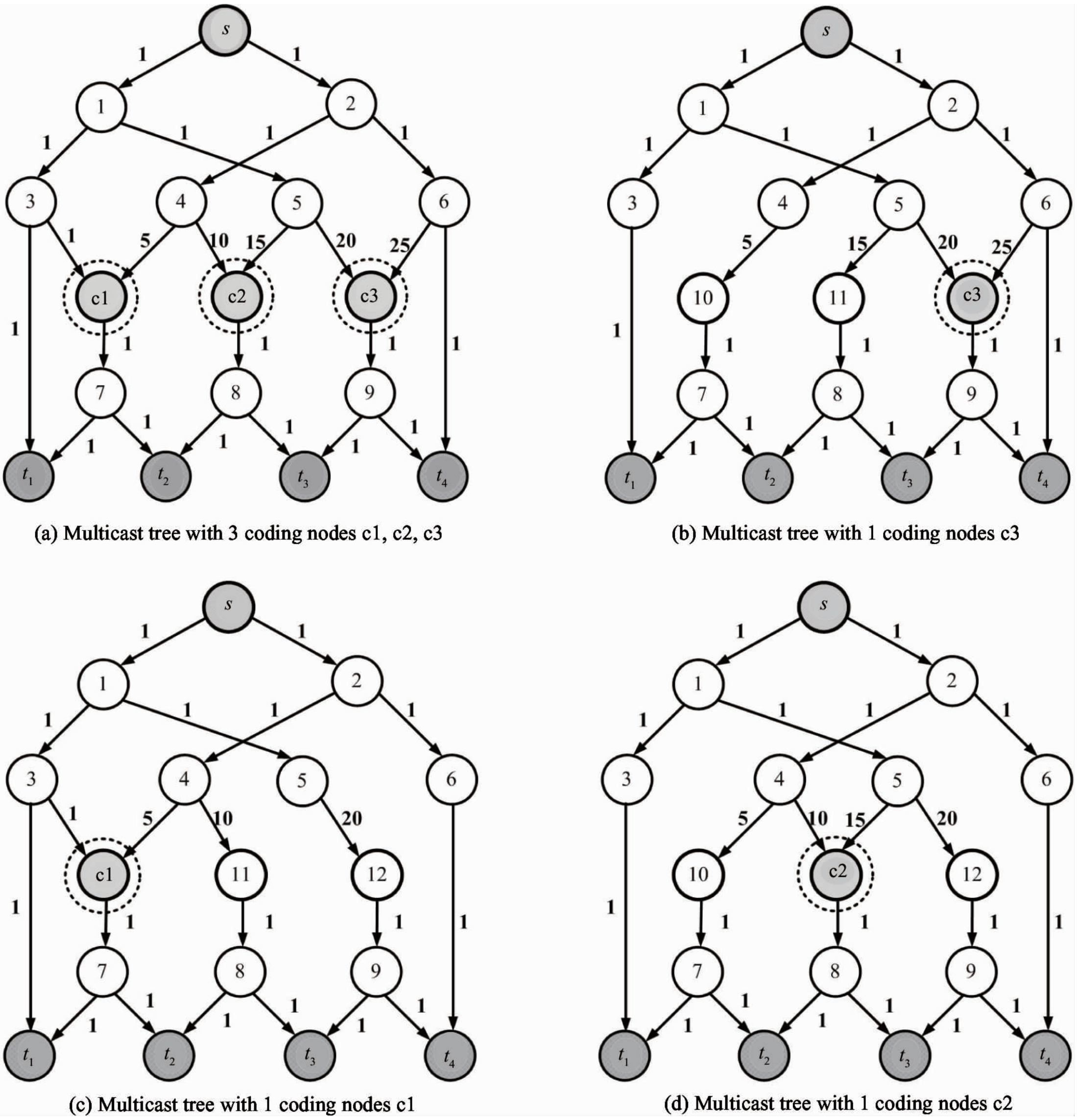
Received on Feb. 22, 2016

cy of the message transmission and the fairness among receivers should be considered as a must one. The network coding resource optimization problems under the restriction of the end-to-end delay and the delay difference among the source-destination paths (see Ref. [12] for details) will be considered. The investigation of combining network coding resource optimization with delay needs can render realistic significance.

1 Problem formulation

The illustration of transmission delay constraints in network coding based multicast network is shown as follows: Considering the same network scenarios in Fig. 1(a), where each link has a unit capacity and source s expects to send data at rate of 2 units to the sink t_1, t_2, t_3, and t_4, respectively. The transmission delay for each link is labeled in Fig. 1 and both the coding and decoding time is assumed to be 1. Note that the coding-waited time and decoding-waited time should also be highly considered.

Figs1(a), (b), (c), and (d) are 4 kinds of data transmission schemes with different amount of coding nodes. Note that the delay from source s to the sinks just depends on the path with maximum delay among all the source-destination paths. Namely, the original information cannot be recovered until all necessary messages received by a sink are collected. For example, the delays of two paths from s to sink t_1 are 3 and 10, respectively in Fig. 1. Hence, the practical



(a) Multicast tree with 3 coding nodes c1, c2, c3

(b) Multicast tree with 1 coding nodes c3

(c) Multicast tree with 1 coding nodes c1

(d) Multicast tree with 1 coding nodes c2

Fig. 1 Network coding sub-graph with different coding nodes and path delay property

delay between s and sink t_1 is 10 and the inter-destination delay difference is 20 which is the difference between 30 and 10. Here, the concept of delay difference is defined as follows:

Delay Difference: The maximum sink delay subtracts the minimum sink delay.

For example, in Table 1, the maximum sink delay is 30 (t_3), the minimum sink delay is 10 (t_1), thus the delay difference of the multicast scenario is 20. So a triple (1, 30, 20) is got, here the first represents the coding node number, the second represents the maximum sink delay, and the third represents the delay difference of the given multicast scenario.

Table 1 The path from s to each sink with end-to-end delay and inter-destination delay difference for Fig. 1(b)

Source	Sink	Paths	Path delay	Delay difference
s	t_1	s-1-3-t_1;	3	20
		s-2-4-c_1-10-t_1	10	
	t_2	s-1-3-c_1-7-t_2	6	
		s-2-4-c_2-8-t_2	15	
	t_3	s-1-5-c_2-8-t_3	20	
		s-2-6-c_3-9-t_3	30	
	t_4	s-1-5-c_3-9-t_4	25	
		s-2-6-t_4	3	

The triples of Figs1(b) ~ (d) are (1, 30, 20), (1, 24, 14), and (1, 24, 15). It is easy to understand that, even though the number of coding nodes is the same, the performance of delay is quite different in different transmission scenarios. If the maximum end-to-end delay and delay difference bounds are confirmed, the feasible transmission scenarios will be determined. This is a very important problem, but the algorithm proposed by Kim, et al. and Xing, et al. just dedicated their efforts to minimize coding nodes/links without considering the delay requirements. So their coding sub-graph may contain paths severely violating the delay constraints, which deteriorates the benefits that network coding is characterized to a certain extent.

2 Mathematic model

The problem is to find the minimal set of coding links where the network coding scenario is required to 1) achieve multicast rate r, 2) meet the end-to-end delay requirements, and 3) meet the inter-destination delay difference constraints. The given multicast network is represented by a directed acyclic graph $G = (V, E)$, where V denotes the set of vertices and E is the set of edges. The capacity of each edge $e \in E$ is unit, and if there has an edge exceeding unit capacity, it is represented by multiple unit edges. The multicast problem is considered with one source $s \in V$ and a sink set $T \subseteq V - \{s\}$, a link delay function $d: E \rightarrow R^+$, delay tolerance Θ, delay difference Ω, and the rate r. Here rate r is an integer and it can be achievable if a transmission scheme can guarantee all $|T|$ sinks receiving packets from the source at least rate r. More specifically, to achieve rate r, there should be at least r link-disjoint paths to each sink subject to:

$$P_i(s,v_k) \cap P_j(s,v_k) = \varnothing$$
$$\forall v_k \in T; \forall i \neq j; \ i, j = 1,2,\cdots,r; \qquad (1)$$
$$k = 1,2,\cdots,|T|;$$

where $P_i(s,v_k)$ denotes the i-th path from source s to the k-th sink v_k. This multicast problem can be converted into a multi-objective optimization problem, and stated mathematically as

Minimize:

$$c = \sum\nolimits_{i=1}^{n} x_i \qquad (2)$$

where c represents the amount of coding links employed in network coding.

Subject to:

$$\min\{R(s,v_i) \mid v_i \in T, \ i = 1,2,\cdots,|T|\} \geqslant r \qquad (3)$$

$$\sum\nolimits_{l \in P_j(s,v_i)} d(l) \leqslant \Theta$$
$$\forall v_i \in T; \ j = 1,2,\cdots,r; \ i = 1,2,\cdots,|T| \qquad (4)$$

$$\max\{\delta_{v_i}\} - \min\{\delta_{v_k}\} \leqslant \Omega$$
$$\forall i, \ k = 1,2,\cdots,|T|; \ v_i,v_k \in T; \qquad (5)$$

$$x_i \in \{0, 1\} \ i = 1,2,\cdots,n \qquad (6)$$

where symbol x including n elements, denotes the output links of an intermediate node with multiple input links. If $x_i = 1$, it means that the i-th output link must be linearly coded; otherwise, it means that no coding operation is required over this link. Here the concept of merging node is presented.

Merging Node: The intermediate nodes which have multiple input links are called merging nodes.

All the output links of a merging node are named as potential coding links.

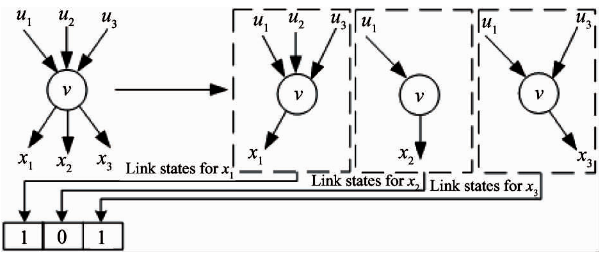


Fig. 2 An example of determining the output link states of a merging node

Fig. 2 gives an illustration for Eq. (2). Node v is an intermediate node which has 3 input links and 3 output links x_1, x_2, and x_3. The state of each output link depends on whether this link has to carry the coded message. Here, the link states of x_1, x_2, and x_3 are '101' according to their own transmission. Supposing that the number of merging nodes in a multicast network is m, hence, $n = \sum_{i=1}^{m} w_i$, where n is the summation of output degree of all m merging nodes and w_i denotes that the i-th node has w_i output streams. If a multicast tree is established, set $X(x_i \in X)$ will be determined. Correspondingly, coding operations c will be obtained by counting the number of '1' in set X.

In Eq. (3), $R(s, v_i)$ represents the rate from source s to sink v_i under the current multicast tree. To achieve the desired multicast rate r, the minimum $R(s, v_i)$ among all the sinks should observe Eq. (3). Namely, for each sink, at least r link-disjoint paths from source to each one must be guaranteed. Eqs(4) and (5) are referred to as source-destination delay constraint and inter-destination delay difference constraint, respectively. The end-to-end delay for sink v_i actually depends on the one that has the maximum delay among r link-disjoint paths and is calculated by $\delta_{v_i}(i = 1,2,\cdots,|D|;\ v_i \in D)$ for each sink v_i, which is defined as

$$\delta_{v_i} = \max\{\sum_{l \in P_j(s,v_i)} d(l) \mid j = 1,2,\cdots,r\}$$
$$\forall v_i \in T; i = 1,2,\cdots,|T| \qquad (7)$$

here $P_j(s, v_i)$ means the j-th path from s to sink v_i. After computing the end-to-end delay of each sink, the inter-destination delay variation can be obtained by Eq. (5). Therefore, the objective is to minimize Eq. (2) subjected to Eqs(3), (4), (5) and (6). Note that in Eqs(4) and (5), the coding-waited time is not considered, which will be explained and calculated in the following REGA.

3 Solutions

A routing-based encoding representation approach[13] is employed to map the optimization problem to a GA framework. The given multicast topology is decomposed to a secondary graph (see more details in the Ref. [8]), and then the path array $P_i(i = 1,2,\cdots,|T|)$ which contains all paths from s to each sink t_i subject to Eq. (4) is gotten. For the i-th path array P_i, all possible combinations are selected each of which consists of r link-disjoint paths. If the number of the selected combinations is m_i, the combinations are numbered sequentially from 1 to m_i, and then P_i is updated by $m_i$combinations. Therefore, array P_i consists of $r \cdot m_i$ paths. Table 2 shows all possible r link-disjoint paths in array P_i.

Table 2 Array P_i includes all possible r link-disjoint paths

Number	Corresponding paths
1	r link-disjoint paths combination 1
2	r link-disjoint paths combination 2
…	…
m_i	r link-disjoint paths combination m_i

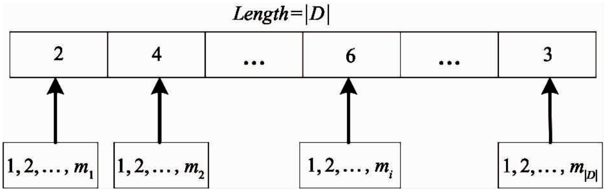


Fig. 3 Integer encoding representation

A clear integer encoding representation is depicted in Fig. 3. The length of integer encoding string is equal to the number of sinks. Each chromosome has $|T|$ genes and the value of the i-th position in the string varies from 1 to m_i. This encoding approach is called routing-based encoding (RE). According to this integer encoding representation, the search space size is $\prod_{i=1}^{|T|} m_i$. The solution space will grow with the number of sinks and the number of r link-disjoint paths combinations in each P_i. Therefore, the RE-based computational complexity could still be increased exponentially if the number of them keeps rising.

An integer encoding routing-based genetic algorithm (REGA) is presented to search the optimal solution. The flowchart of REGA is shown in Fig. 4.

The algorithm starts from an initial population Q_{pop} containing k chromosomes. The fitness function is given in Eq. (8), where α is the number of coding links, λ and β are constants, d_n is the number of source-destination paths, d_v denotes the maximum inter-destination delay variation and $p(d_v, \Omega)$ is a penalty function which is defined in Eq. (9).

$$f = \frac{\lambda}{\alpha + d_n + \beta \cdot p(d_v, \Omega) + 1} \qquad (8)$$

$$p(d_v, \Omega) = \begin{cases} d_v - \Omega & \text{if } d_v > \Omega \\ 0 & \text{otherwise} \end{cases} \qquad (9)$$

When performing the algorithm, the fittest chromosome is directly copied into the new population. This elitism strategy shows great improvement for the algorithmic performance in our simulation. Roulette selection and single crossover are carried out and followed by a random mutation which helps escaping from the local optimal. Moreover, it is widely demonstrated that the infeasible chromosomes also help population evolve rapidly. Hence, according to the level of violation, different scale penalties are imposed on the infeasible solutions while determining their fitness, which speed up search from not only feasible domain but also infeasible domain.

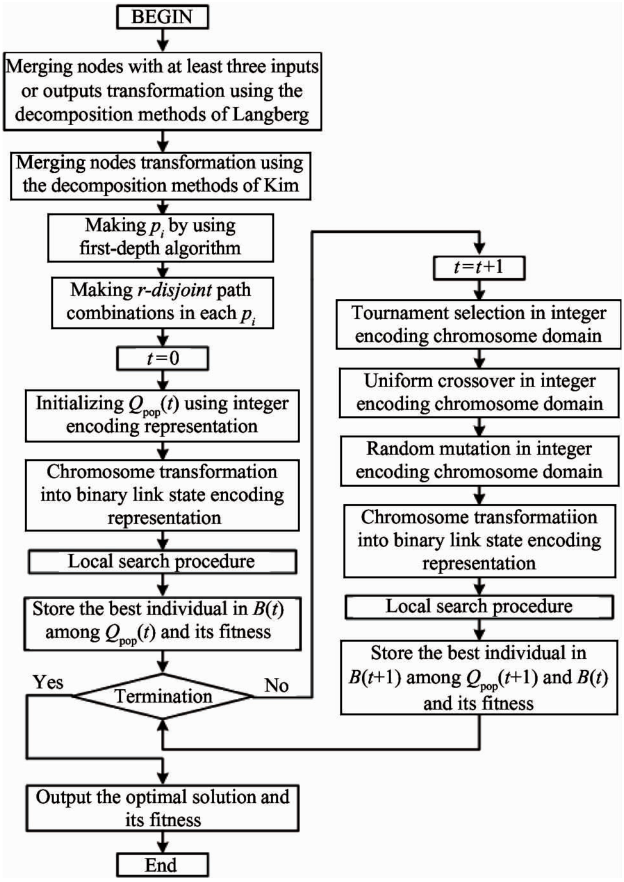

BEGIN
Merging nodes with at least three inputs or outputs transformation using the decomposition methods of Langberg
Merging nodes transformation using the decomposition methods of Kim
Making p_i by using first-depth algorithm
Making *r-disjoint* path combinations in each p_i
t=0
Initializing $Q_{pop}(t)$ using integer encoding representation
Chromosome transformation into binary link state encoding representation
Local search procedure
Store the best individual in $B(t)$ among $Q_{pop}(t)$ and its fitness
Yes
Termination
No
t=t+1
Tournament selection in integer encoding chromosome domain
Uniform crossover in integer encoding chromosome domain
Random mutation in integer encoding chromosome domain
Chromosome transformatiion into binary link state encoding representation
Local search procedure
Store the best individual in $B(t$+1) among $Q_{pop}(t$+1) and $B(t)$ and its fitness
Output the optimal solution and its fitness
End

Fig. 4 The flowchart of REGA

4 Simulation and analysis

To evaluate the performance of REGA, comparisons have been carried out with BLSGA[3] over the 2 fixed multicast networks and 1 random multicast network. The 2 fixed networks are 3-copy and 7-copy networks which have been used in Ref. [3]. Fig. 5 illustrates an example of n-copy network, where Fig. 5(a) is the original network and Fig. 5(b) is a 3-copy network constructed by cascading 3 copies of the original network. For the n-copy network, the source node is at the top, and the sinks are at the bottom. According to Fig. 5, it is known that the n-copy network contains $n+1$ sinks.

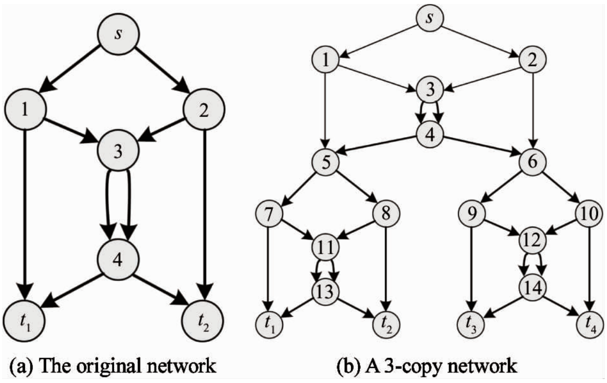


Fig. 5 Illustration the n-copy network

The parameters of the 3 test networks shown in Table 3.

Table 3 Network parameters

Networks	Nodes	Links	Sinks	Rate
3-copy	19	30	4	2
7-copy	43	70	8	2
random	40	85	10	4

The parameters for REGA are set as following: $delay \in (0,1]$ with arbitrary unit which is uniformly distributed, crossover operator $p_c = 0.8$, mutation operator $p_m = 0.1$, and $\lambda = 10$, $\beta = 100$. In addition, the size of population (*Popsize*) and the number of iteration (*NI*) shown in following result tables vary with the size of solution space. Moreover, all simulations are run on a Windows XP computer with Intel(R) Core (TM) 2 Quad CPU, 2.40GHz, 3.25G RAM

With regard to BLSGA, note that it is developed for a non-delay-constrained network coding resource optimization problem. Hence to provide an apple-to-apple comparison, some adjustments for BLSGA should be enforced to introduce link delay. For each individual in the population, it represents a multicast tree which is checked whether the current transmission scheme can guarantee the given multicast rate achievable. If the rate can be achieved, Ford-Fulkson algorithm is used to compute the end-to-end paths for each sink, and then the end-to-end delay for each source-destination path can be calculated. Correspondingly, the inter-destination delay variation is obtained. Due to the introduction of delay requirements, fitness function has to be adjusted and re-defined as

$$F(y) = \begin{cases} \text{number of blocks with two or more ones} \\ \qquad\qquad\qquad\qquad \text{if } y \text{ is feasible;} \\ n_{\max} + d_n + \beta \times p(d_v, \Omega) \\ \qquad\qquad\qquad\qquad \text{if } y \text{ is infeasible;} \end{cases} \tag{10}$$

where $n_{\max}$ is the number of potential coding links. The definition of d_n and $p(d_v, \Omega)$ is identical with that of REGA and $\beta = 100$. If anyone of the constraints Eqs(3), (4), and (5) is violated, chromosome y is infeasible and corresponding penalty is added in the process of fitness evaluation.

Before comparing the searching result under the delay restriction, the convergence speed of the 2 algorithms are tested firstly. Table 4 and Table 5 list the algorithmic parameters used in this test. Fig. 6 and Fig. 7 show the convergence speed over 15-copy and 31-copy networks. The index called mean network coding operation times (*MNCO*) is employed in the simulation, and it is defined as Eq. (11). Here n_{trials} indicates the times of algorithmic trial and t is the number of evolutionary generation. $b(t,i)$ denotes the best fitness (namely, the number of minimal coding links) so far when the population evolves to the t-th generation in the i-th trial. In the simulation $n_{\text{trials}} = 30$.

Table 4 Algorithmic parameters of REGA

Algorithmic parameter	Value
Population size	20
Tournament size	6
Maximum generation	20
Run times	30
Mutation probability	0.1
Crossover probability	0.8

Table 5 Algorithmic parameters of BLSGA

Algorithmic parameter	Value
Population size	150
Tournament size	10
Maximum generation	1000
Run times	30
Mutation probability	0.006
Crossover probability	0.8

$$MNCO = \frac{\sum_{i=1}^{n_{\text{trials}}} b(t,i)}{n_{\text{trials}}} \quad t = 1,2,\cdots,m; \quad (11)$$

In Fig. 6 only 13 generation evolutions are required to hit the best solution. However, in order to get a best solution, BLSGA needs a huge population 150 and a big evolutionary generation 1000. From Fig. 7, it can be seen that even using such prominent parameters, however, the performance is still worse than that of REGA. It is clear that the convergence speed of REGA is better, especially in the larger multicast networks (such as 31-copy network). The result provides a solid demonstration about the efficiency of REGA.

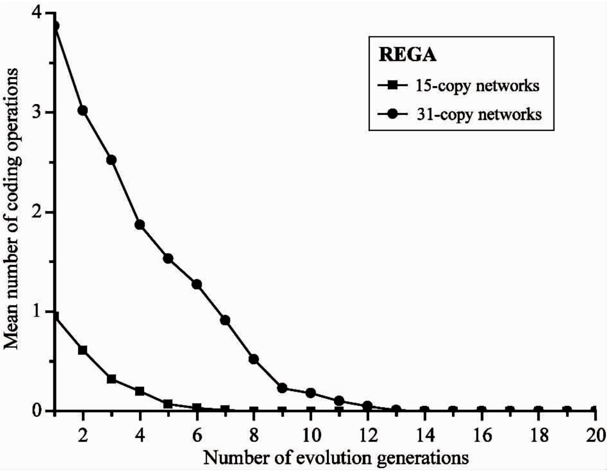


Fig. 6 Convergence speed of REGA

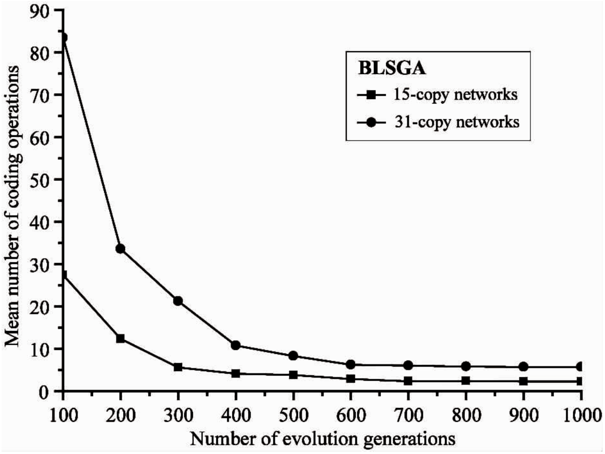


Fig. 7 Convergence speed of BLSGA

Moreover, the simulation results about searching for the best solution under the delay restriction are shown in Table 6, Table 7, Table 8. The numeric results in a row in parentheses indicate coding number (CN), maximum end-to-end delay (MD), maximum delay difference (MDD), computational time (CT, the unit is second), *Popsize*, and *NI*, respectively. Therefor the numbers in the brackets from Table 6 to Table 8 indicate the numerical result of (CN, MD, MDD, CT, *Popsize*, *NI*) respectively. Actually, each algorithm in tables has been run for 30 times, and the best result among those 30 trials is chosen.

From the numerical results it can be known that BLSGA even cannot find a satisfied solution. Several reasons may be responsible for this. First, due to the binary link state representation in BLSGA, the size of solution space is too large to handle, especially in such strict constraints. Thereby an increment for BLSGA on both *Popsize* and *NI* has to be imposed to promote the search ability, which extremely enhance CT. Besides, according to the chromosome encoding principle, the length of the string in REGA just depends on the number of sinks and is evidently much shorter than that of BLSGA, which can reduce the computing complexity and speed up the algorithm operation. On the other hand, compared with the constitution of solution space of REGA, BLSGA search space has too many infeasible solutions while in REGA only the solutions subject to Eqs(3) and (4) can be covered.

Table 6 Numerical results of 3-copy network

Algorithms	3-copy network(19nodes, 30links, 4sinks, 2rate)			
	$\Theta=3$, $\Omega=0.3$	$\Theta=3.3$, $\Omega=0.08$	$\Theta=3$, $\Omega=0.1$	$\Theta=2.7$, $\Omega=1$
REGA	(0, 2.75, 0.25, 6.78, 50, 100)	(2, 3.21, 0.07, 11.67, 50, 150)	(3, 2.63, 0.09, 6.75, 50, 100)	(1, 2.63, 0.13, 2.91, 50, 50)
BLSGA	(0, 2.75, 0.25, 244.51, 150, 200)	–	–	(1, 2.63, 0.65, 244.34, 150, 200)

Table 7 Numerical results 7 copy network

Algorithms	7-copy network(43nodes, 70links, 8sinks, 2rate)			
	$\Theta=10$, $\Omega=2$	$\Theta=8$, $\Omega=1.5$	$\Theta=8$, $\Omega=1$	$\Theta=8$, $\Omega=0.4$
REGA	(0, 8.99, 1.66, 315.06, 150, 250)	(2, 7.85, 1.28, 125.03, 100, 200)	(4, 7.70, 0.86, 131.69, 100, 200)	(5, 7.70, 0.39, 135.81,100,200)
BLSGA	(0, 8.37, 1.81, 2389.6, 200, 1000)	–	–	–

Table 8 Numerical results of random network

Algorithms	Random network (40nodes, 85links, 10sinks, 4rate)
	$\Theta=8$, $\Omega=3$
REGA	(0, 4.22, 2.84, 4.27, 20, 50)
BLSGA	(0, 4.22, 2.84, 28.5, 20, 50)

Moreover, the simulation results also indicate that the coding numbers will be influenced by different delay constraints in a same multicast network. When the QoS level becomes severer, the encoding cost will increase, hence it should play a trade-off between the encoding cost and delay constraints. As for the performance of the algorithms, thanks to the RE and some excellent evolutionary schemes, such as fitness penalty and elitism strategy, REGA outperforms significantly over the BLSGA in terms of the ability to get a satisfied solution and the computational time.

5 Conclusions

In this study, the problem of minimizing the coding cost with required data rate and delay constraints in network coding based multicast networks is studied. An algorithm named REGA, has been proposed to address this problem. In REGA, an efficient and problem-specific local search scheme is designed to help the algorithm to obtain the optimality more quickly. Then it is demonstrated that even under the severe delay constraints, the proposed REGA can still render a feasible solution for real application in such coding resources optimization area. The distinguished suboptimal solutions obtained by REGA prove the capability and efficiency of the utility in this field.

References

[1] Ahlswede R, Cai N, Li S, et al. Network information flow. *IEEE Transactions on Information Theory*, 2000, 46(4): 1204-1216

[2] Minkyu K, Varun A, Una-May O, et al. Genetic representations for evolutionary minimization of network coding resources. *Computer Science*, 2007, 4448:21-31

[3] Minkyu K, Médard, Aggarwal V, et al. Evolutionary approaches to minimize network coding resources. In: Proceedings of the 26th IEEE International Conference on Computer Communications, Anchorage, USA, 2007. 1991-1999

[4] Wang Z, Xing H, Li T, et al. A modified ant colony optimization algorithm for network coding resource minimization. *IEEE Transactions on Evolutionary Computation*, 2016, 20(3): 325-342

[5] Xing H, Qu R, Bai L, et al. On minimizing coding operations in network coding based multicast: an evolutionary algorithm. *Applied Intelligence*, 2014, 41(3): 820-836

[6] Xing H, Qu R. A nondominated sorting genetic algorithm for bi-objective network coding based multicast routing problems. *Information Sciences*, 2013, 233(6): 36-53

[7] Xing H, Qu R, Kendall G, et al. A path-oriented encoding evolutionary algorithm for network coding resource

minimization. *Journal of the Operational Research Society*, 2014, 65(8): 1261-1277

[8] Qu Z, Liu X, Zhang X, et al. Hamming-distance-based adaptive quantum-inspired evolutionary algorithm for network coding resources optimization. *The Journal of China Universities of Posts and Telecommunications*, 2015, 22 (3): 92-99

[9] Keshavarz-Haddad A, Riedi R. Bounds on the benefit of network coding for wireless multicast and unicast. *IEEE Transactions on Mobile Computing*, 2014, 13(1), 102-115

[10] Wu F, Sun Y, Yang Y, et al. Constant-delay and constant-feedback moving window network coding for wireless multicast: design and asymptotic analysis. *IEEE Journal on Selected Areas in Communications*, 2015, 33 (2): 127-140

[11] Yu Y, Peng Y, Li X, et al. Distributed packet-aware routing scheme based on dynamic network coding. *China Communications*, 2016, 13(10): 20-28

[12] Rouskas G, Baldine I. Multicasting routing with end-to-end delay and delay variation constraints. In: Proceedings of the 15th Annual Joint Conference of the IEEE Computer Societies, Networking the Next Generation, San Francisco, USA, 1996. 353-360

[13] Hadj-Alouane A, Bean J. A genetic algorithm for the multiple-choice integer program. *Operations Research*, 1997, 45(1): 92-101

Qu Zhijian, born in 1980. He received his Ph. D degrees in Information and Communication Engineering Department of Beijing University of Posts and Telecommunications in 2011. He is currently an associate professor in the School of Computer Science and Technology, Shandong University of Technology. His research interests include network coding, intelligence algorithm, and optical multicast.