An Improved MOEA/D for QoS Oriented Multimedia Multicasting with Network Coding

Zhaoyuan Wang
School of Information Science
and Technology
Southwest Jiaotong University
Chengdu 611756, China
wang_zhaoyuan
@foxmail.com

Huanlai Xing[*]
School of Information Science
and Technology
Southwest Jiaotong University
Chengdu 611756, China
hxx@home.swjtu.edu.cn

Tianrui Li
School of Information Science
and Technology
Southwest Jiaotong University
Chengdu 611756, China
trli@swjtu.edu.cn

Yan Yang
School of Information Science
and Technology
Southwest Jiaotong University
Chengdu 611756, China
yyang@swjtu.edu.cn

Rong Qu
School of Computer Science
The University of Nottingham
Nottingham, NG8 1BB, UK
rong.qu@nottingham.ac.uk

ABSTRACT

Recent years witness a significant growth in multimedia applications. Among them, a stream of applications is real-time and requires one-to-many fast data transmission with stringent quality-of-service (QoS) requirements, where multicast is an important supporting technology. In particular, with more and more mobile end users requesting real-time broadband multimedia applications, it is of vital importance to provide them with satisfied quality of experience. As network coding can offer higher bandwidth to users and accommodate more flows for networks than traditional routing, this paper studies the multicast routing problem with network coding and formulates it as a multi-objective optimization problem. As delay and packet loss ratio (PLR) are two important performance indicators for QoS, we consider them as the two objectives for minimization. To address the problem above, we present a multi-objective evolutionary algorithm based on decomposition (MOEA/D), where an all population updating rule is devised to address the problem of lacking feasible solutions in the search space. Experimental results demonstrate the effectiveness of the proposed algorithm and it outperforms a number of state-of-the-art algorithms.

Keywords

multicast, multimedia, multi-objective optimization, network coding, quality-of-service

---

[*]Corresponding author

1. INTRODUCTION

In recent years, a tremendous growth in multimedia applications has been witnessed. Statistical data show that, in a minute, around 150,000,000 GB data are transmitted in the Internet. Among them, about 90% belong to multimedia applications [1, 2]. With the rapid development in mobile Internet, increasingly more people enjoy their media time on mobile devices, which accounts for about 20% of the total media time. Hence, to efficiently support those multimedia applications, data transmission services with high quality-of-service (QoS) are in urgent need by end users, especially those accessing Internet via mobile devices. Parameters e.g. data transmission rate, delay, and packet loss ratio are important performance indicators for QoS. In nature, end users look for multimedia applications with satisfied QoS guarantees. Multicast is a one-to-many communications technology with multiple receivers simultaneously requesting the same information sent from a single source. This technology can well support multimedia applications with multiple end users involved, such as video conferencing, IPTV, interactive online games. However, multicast employing store-and-forward forwarding cannot guarantee the theoretically maximized throughput [3].

Fortunately, network coding can always help the multicast achieve the theoretical maximum throughput, which is particularly suitable to support real-time broadband data transmission, i.e. this technology is ideal for real-time multimedia multicast scenarios with stringent QoS requirements [4]. On the other hand, when employed in multicast, network coding involves coding operations performed at a subset of intermediate nodes in the network. As coding operations are complicated mathematical operations, e.g. calculations over some finite field, network coding based multicast (NCM) not only consumes additional computing and buffering resources, but also could cause serious network performance deterioration [5], including the end-to-end delay and packet loss ratio. Hence, it is a nature way to think of how to consume as less network resources as possible while exploiting

all benefits the NCM brings to the existing network infrastructure.

A considerable amount of research efforts have been dedicated to optimize the NCM routing problem. For example, the network coding resource minimization problem, which is for minimizing the involved computing resource, has drawn a lot of research attention [6, 7, 8, 9, 10, 11, 12]. Moreover, as coding and link costs both incur during the NCM, some researchers study the trade-off between them using multiobjective optimization approaches [13, 14]. In [15], Xing and Qu minimize the total cost and end-to-end delay simultaneously, where the cost is a weighted sum of the coding and link costs. However, to the best of our knowledge, there has not been any research carried out from the perspective of supporting NCM with multiple QoS metrics met. Since end users are in urgent need for satisfied QoS multimedia applications, this paper investigates a QoS oriented bi-objective NCM scenario, where the average end-to-end delay and the average packet loss ratio are the two objectives for minimization, respectively.

Evolutionary algorithms (EAs) usually obtain promising solutions in a single run within a short time due to the population-based parallel computing framework. Hence, when used for addressing network routing selection problems with multiple (often conflicting) objectives, EAs are often considered as an efficient candidate optimizer. Among those multi-objective EAs, multi-objective evolutionary algorithm based on decomposition (MOEA/D) [16] has received a considerable amount of research attention because of its excellent optimization performance. MOEA/D decomposes a given MOP into a number of scalar sub-problems by conventional aggregation approaches, and the sub-problems are solved simultaneously by evolving a population of solutions. It has been proved that MOEA/D obtains promising optimization performance with much lower computational complexity than well-known MOEAs, e.g., NSGA-II [17] and SPEA2 [18], which makes it an ideal optimizer for handling the problem concerned.

In this paper, a QoS oriented bi-objective optimization problem in the context of NCM is studied. We adapt MOEA/D for the proposed problem, where a new scheme is integrated into the basic MOEA/D framework and thus able to enhance the searching procedure. This feature is referred to as all population updating rule, where a newly generated solution is used to update the most suitable one among all sub-problems. The experimental results illustrate the superiority of the proposed MOEA/D over several of the state-of-the-art MOEAs.

2. PROBLEM FORMULATION

A communications network can be modeled as a directed graph $G = (V, E)$, where V and E are the node and link sets, respectively. Assume each link $e \in E$ has a unit capacity. Those with larger capacity are represented by parallel links, each with a unit capacity. In a NCM scenario, there is a source node $s \in V$, a set of receivers $T = \{t_1, ..., t_d\}$, $t_k \in V$, and an expected data rate R. We need to find a subgraph consisting of multiple link-disjoint paths (i.e. paths without common link), where each path is originated from the source s and terminated at one of the receivers, e.g. t_k. In this

subgraph, different data flows may pass through different areas. Such subgraph is referred to as NCM subgraph [10].

The following lists some notations used in the paper:

- s: the source node in $G(V, E)$;
- $T = \{t_1, ..., t_d\}$: set of receivers, where $d = |T|$ is the number of receivers;
- R: the expected data rate (an integer) at which s expects to transmit to T;
- $r(s, t_k)$: the practical data rate from the source to receiver $t_k \in T$ within the NCM subgraph;
- $p_i(s, t_k)$: the i-th path from s to t_k within the NCM subgraph, where $t_k \in T$ and $i = 1, ..., R$;
- $delay(p_i(s, t_k))$: the end-to-end transmission delay on $p_i(s, t_k)$;
- $plr(p_i(s, t_k))$: the packet loss ratio (PLR) on $p_i(s, t_k)$.

The task in the paper is to construct a feasible NCM subgraph with the data rate requirement satisfied and two objectives simultaneously minimized, as follows.

$$\text{Minimize} \begin{cases} D_{avg} = \frac{1}{d} \sum_{k=1}^{d} \max\{delay(p_i(s, t_k)) | i = 1, 2, ... R\} \\ P_{avg} = \frac{1}{d} \sum_{k=1}^{d} \max\{plr(p_i(s, t_k)) | i = 1, 2, ... R\} \end{cases}$$
(1)

$$\text{s.t.} \quad r(s, t_k) \geq R, \forall t_k \in T \tag{2}$$

where, D_{avg} is the average transmission delay of the NCM subgraph; P_{avg} is the average PLR of the NCM subgraph. Constraint 2 defines that the obtained data rate from source s to each receiver must be no less than R. Optimal solutions to the problem concerned are a set of nondominated solutions, known as the Pareto-optimal front (PF) [19]. Given any two solutions x_1 and x_2, we say x_2 is dominated by x_1 if either $\{D_{avg}(x_1) < D_{avg}(x_2), P_{avg}(x_1) \leq P_{avg}(x_2)\}$ or $\{D_{avg}(x_1) \leq D_{avg}(x_2), P_{avg}(x_1) < P_{avg}(x_2)\}$ holds.

3. THE PROPOSED MOEA/D

First of all, the chromosome representation and objective evaluation are introduced. Then, a performance enhancing scheme, i.e. the all population updating rule, is described. Finally, the overall structure of the proposed MOEA/D is given in detail.

3.1 Chromosome representation and objective evaluation

Chromosome representation is one of the most important issues when designing MOEAs. As aforementioned, coding operations have to be performed if necessary. To explicitly see how data-flows pass through each of the intermediate nodes, the binary link state (BLS) chromosome representation is adopted. BLS-based representation has been widely

used in NCM routing problems [9, 10, 11, 12, 15]. More details can be found in [15].

To evaluate a solution in all objectives, the first task is the feasibility checking and the second one is the calculation of objective values. As we know, each solution x corresponds to a certain subgraph. Goldberg algorithm [20] is used to verify if the associated subgraph of x meets the data rate requirement. If yes, x is feasible; otherwise, it is infeasible. For a feasible solution, we compute D_{avg} and P_{avg} of the corresponding subgraph, according to formula 1.

3.2 All population updating rule

In MOEA/D, the traditional way of updating the population is that, when a promising solution is generated, it is used to replace not only the best-so-far solution to the corresponding sub-problem but also those to the neighboring sub-problems. As the problem concerned in the paper is of hard constraint, there are many infeasible solutions in the search space. Although feasible solutions are urgently needed to guide the search towards the true PF, it is quite difficult to obtain them. If a better solution is generated and adopted to update only the neighboring sub-problems, all associated best-so-far solutions could be replaced, which would lead to premature convergence and deteriorated optimization performance. Moreover, due to the problem nature, coding operations could seriously affect the delay and/or PLR. In other word, solutions with a few different links, whose subgraphs may be quite similar, might be far from each other in the search space. Therefore, a solution generated from one sub-problem may be more suitable for a far-away sub-problem. Inspired by this idea, we propose the all population updating rule, extending the neighborhood range to all single-objective optimization problems (SOPs). In this way, a newly generated solution can be used to update the most appropriate SOP and the search is thus well guided to explore promising areas in the search space. When a promising solution is generated, instead of multiple SOPs, a single SOP where the fitness quality improvement is the most significant is updated. With the features above, MOEA/D gains better performance as observed in Section 4. Moreover, details of the proposed solution updating rule can be found in Step 2.3, Subsection 3.3.

3.3 Overall procedure

In MOEA/D, the first step is to convert a MOP into a number of scalar optimization sub-problems (each with a single objective) by using different decomposition methods [16]. This paper adopts Tchebycheff approach for decomposition as it is the most commonly-used and able to help MOEA/D gain decent optimization performance. To be specific, a MOP is decomposed into multiple SOPs as the following shows:

$$\text{Minimize}: g(x|\lambda, z^*) = \max_{1 \le i \le m}\{\lambda_i | f_i(x) - z_i^*|\} \quad (3)$$
$$\text{Subject to}: x \in \Omega$$

where m is number of objectives, $\lambda = (\lambda_1, ..., \lambda_m)$ is a weight vector and $\sum_{i=1}^{m} \lambda_i = 1$, and $f_i(x)$ is the the i-th objective value of x. $z^* = \{z_1^*, ..., z_m^*\}$ is the reference point, where z_i^* is the best-so-far value of the i-th objective.

Let $\lambda^1, ..., \lambda^N$ be N weight vectors, each associated with a SOP. If all of them are properly selected, the obtained best solutions can provide a good approximation to the true PF. In addition, Euclidean distances among those weight vectors are used to define the neighborhood relationship for the SOPs. For two SOPs, if the Euclidean distance between their weight vectors is small, it indicates the optimal solutions to one SOP and those to the other form similar PFs in the objective space (i.e. decision space). Hence, SOPs with closer Euclidean distances are regarded neighbors. In MOEA/D, SOPs are solved in a collaborative manner, where useful information is shared within neighborhoods.

The procedure of the proposed MOEA/D is described below.

Input:

- The multi-objective optimization problem.

- A stopping criterion.

- W: the number of the weight vectors in the neighborhood of each weight vector.

- p_c: the crossover probability.

- p_m: the mutation probability.

Global structure:

- A population of N search points $x_1, ..., x_N \in \Omega$, where x_j is the solution of the j-th sub-problem.

- $z = \{z_1, ..., z_m\}$.

- An external population (EP), which stores nondominated solutions found during the search.

Procedure:

Step 1: Initialization:
  Step 1.1: Set EP $= \Phi$.
  Step 1.2: Compute the Euclidean distances between any two weight vectors and then work out the W closest weight vectors to each weight vector. For $j = 1, ..., N$, set $B(j) = \{j_1, ..., j_W\}$. $\lambda^{j_1}, ..., \lambda^{j_W}$ are the W closest weight vectors to λ^j.
  Step 1.3: Employ initialization scheme in [15] to generate initial population $x_1, ..., x_N$.
  Step 1.4: Initialize $z = \{z_1, ..., z_m\}$, where $z_i = \min(f_i(x_j))$, $1 \le i \le m, 0 \le j \le N$.
Step 2: Update:
For $j = 1, ..., N$
  Step 2.1: Reproduction:
  Randomly select two indexes k and l from $B(j)$ and generate a new solution y by performing crossover and mutation to x_k and x_l.
  Step 2.2: Update of reference point:
  For $i = 1, ..., m$ If $f_i(y) < z_i$, set $z_i = f_i(y)$.
  Step 2.3: All population updating rule:
  Find an index h so that $\max\{g(x_h|\lambda^h, z) - g(y|\lambda^h, z)\}$ has the largest value. Then set $x_h = y$.
                    ▷ (see subsection 3.2)

Step 2.4: Update of EP:

Remove those solutions from EP that are dominated by $f(y)$. Add $f(y)$ to EP if no vector of EP dominates $f(y)$.

Step 3: Stopping Criteria:

If stopping criteria is met, stop the search and output EP. Otherwise, go to **Step 2**.

4. EXPERIMENTS AND ANALYSIS

In this section, we first introduce the test instances and performance metrics for evaluating the proposed MOEA/D. After that, we study the effectiveness of the proposed all population updating rule. Finally, we compare the proposed MOEA/D with several state-of-the-art MOEAs in terms of the optimization performance.

4.1 Test instances

We evaluate the performance of the proposed algorithm on 8 random instances which are widely used in the literature [9, 10, 11, 12, 15]. These instances are all available at http://www.cs.nott.ac.uk/ rxq/benchmarks.htm and more details can be found in [11]. In addition, the delay and the PLR on each link are random numbers, uniformly distributed in the range [2ms, 10ms] and $[1 \times 10^{-5}, 5 \times 10^{-5}]$, respectively. All experiments are run on a computer with Windows 8 OS, Intel(R) Core(TM) i7-3740QM CPU 2.7 GHz and 8 GB RAM.

4.2 Performance measures

Let PF_{ref} be a reference set of nondominated solutions of the true PF and PF_{known} be the set of nondominated solutions obtained by an algorithm. Solutions in PF_{ref} are expected to be uniformly distributed in the objective space along the true PF. Note that true PFs are usually not known for highly complex multi-objective optimization problems including the problem concerned in this work. To determine a reference set PF_{ref}, we combine the best-so-far solutions obtained by all algorithms in all runs and select the nondominated solutions as the reference set. This has been widely adopted in evaluating multi-objective algorithms in the literature.

To thoroughly evaluate the performance of the proposed algorithm, the following performance measuring metrics are employed throughout the experiments.

(1) Generational distance (GD): GD measures the average distance from the obtained nondominated solution set PF_{known} to the reference set PF_{ref}, defined as:

$$GD = \sqrt{\frac{\sum\limits_{v \in PF_{known}} d(v, PF_{ref})}{|PF_{known}|}} \qquad (4)$$

where $d(v, PF_{ref})$ is the Euclidean distance between solution v in PF_{known} and its nearest solution in PF_{ref}. A smaller GD indicates the obtained PF is closer to the true PF.

(2) Inverted generational distance (IGD): IGD is defined distance (in the objective domain) between solution v in PF_{ref}

and its nearest solution in PF_{known}, defined as:

$$IGD = \frac{\sum\limits_{v \in PF_{ref}} d(v, PF_{known})}{|PF_{ref}|} \qquad (5)$$

This metric measures both the diversity and the convergence of an obtained nondominated solution set. A lower IGD indicates a better overall performance of an algorithm.

(3) Maximum spread (MS): this metric reflects how well the true PF is covered by the nondominated solutions in PF_{known} through the hyperboxes formed by the extreme function values observed in PF_{ref} and PF_{known}, as shown in

$$MS = \sqrt{\frac{1}{m} \sum_{i=1}^{m} \left(\frac{\min(f_i^{\max}, F_i^{\max}) - \max(f_i^{\min}, F_i^{\min})}{F_i^{\max} - F_i^{\min}} \right)^2} \qquad (6)$$

where m is the number of objectives; f_i^{max} and f_i^{min} are the maximum and minimum values of the i-th objective in PF_{known}, respectively; and F_i^{max} and F_i^{min} are the maximum and minimum values of the i-th objective in PF_{ref}, respectively. A larger MS shows the obtained PF has a better spread.

(4) Average Computational Time (ACT) consumed by an algorithm over 20 runs. This metric is a direct indication of the computational time of an algorithm.

(5) Student's t-test [11, 21] to compare two algorithms (A and B) in terms of the IGD values obtained in 20 runs. In this paper, two-tailed t-test with 38 degrees of freedom at a 0.05 level of significance is used. The t-test result can show if the performance of A is better than, worse than, or equivalent to that of B from the aspect of statistics.

4.3 The effectiveness of the all population updating rule

We evaluate the effectiveness of the proposed solution updating rule by running two variants of MOEA/D, i.e. traditional MOEA/D (A1) and MOEA/D with the proposed solution updating rule (A2).

Table 1 shows the experimental results collected in terms of GD, IGD and MS. It is observed that A2 algorithm performs better than A1, in terms of the GD, IGD and MS. The results clearly show that the proposed updating rule can significantly improve the optimization performance of MOEA/D. With the new updating rule, one reproduced solution is used to update the most suitable SOP among all SOPs. This rule rationally utilizes new solutions and helps to avoid prematurity. Hence, A2 outperforms A1 in all instances.

4.4 The overall performance evaluation

We evaluate the overall performance of the proposed MOEA/D by comparing it with several distinguished MOEAs, i.e. S-PEA2 and NSGA-II. The following list the parameter settings for each algorithm.

- SPEA2: $N = 50, p_c = 0.9$ and $p_m = 1/l$.

Table 1: Experimental results in terms of GD, IGD and MS (best results are in bold)

Network	GD		IGD		MS	
	A1	A2	A1	A2	A1	A2
Rnd-1	**0.000**	**0.000**	**0.000**	**0.000**	**1.000**	**1.000**
Rnd-2	**0.000**	**0.000**	**0.000**	**0.000**	**1.000**	**1.000**
Rnd-3	**0.000**	**0.000**	**0.000**	**0.000**	**1.000**	**1.000**
Rnd-4	0.112	**0.073**	2.031	**0.173**	0.416	**0.829**
Rnd-5	0.294	**0.190**	0.934	**0.195**	0.687	**0.914**
Rnd-6	0.068	**0.000**	0.159	**0.000**	0.919	**1.000**
Rnd-7	0.027	**0.006**	0.099	**0.008**	0.827	**0.958**
Rnd-8	0.649	**0.249**	1.003	**0.187**	0.700	**0.811**

- NSGA-II: $N = 50, p_c = 0.9$ and $p_m = 1/l$.

- Proposed MOEA/D: $N = 50, p_c = 0.9$ and $p_m = 1/l$.

To make a fair comparison, each algorithm runs 200 generations. The results of GD, IGD, MS and ACT are shown in Table 2.

It is clearly seen that, the proposed MOEA/D gains the best performance regarding GD, IGD and MS. The nondominated solutions obtained by our MOEA/D are closer to the optimal solutions along the true PF and are more diversified in terms of their locations in the objective space. Besides, smaller ACT also indicates that the proposed MOEA/D has lower computational complexity than NSGA-II and SPEA2.

Figure 1 illustrates the best PFs obtained in a single run by NSGA-II, SPEA2 and the proposed MOEA/D, respectively, where the true PF is marked by small and solid dots. Due to space limitation, we select two instances, Rnd4 and Rnd8 as an example. Clearly, compared with NSGA-II and SPEA2, the proposed MOEA/D owns better PF which is a nice approximation to the true PF. On the other hand, as we can see, there is a significant gap between the true PF and those of NSGA-II and SPEA2. Therefore, Figure 1 also demonstrates that the proposed MOEA/D has promising optimization performance regarding the PF obtained.

To further support our observation, we compare the IGD values of the three algorithms by using Student's t-test. Table 3 shows that our algorithm outperforms the others in most of the instances.

5. CONCLUSIONS

This paper formulates a QoS oriented multicast routing problem based on network coding so as to well support real-time broadband multimedia applications. We consider three important performance metrics in the problem, where the average transmission delay and the average packet loss ratio are two objectives to be minimized simultaneously and the data rate (i.e. bandwidth) is a hard constraint. We adapt MOEA/D for the problem above and propose a problem-specific solution updating rule, i.e. the all population updating rule, to improve the optimization performance. In this rule, each new solution is used to update the most suitable sub-problem. The experimental results demonstrate that the proposed MOEA/D performs significantly better than traditional MOEA/D, SPEA2 and NSGA-II with respect to
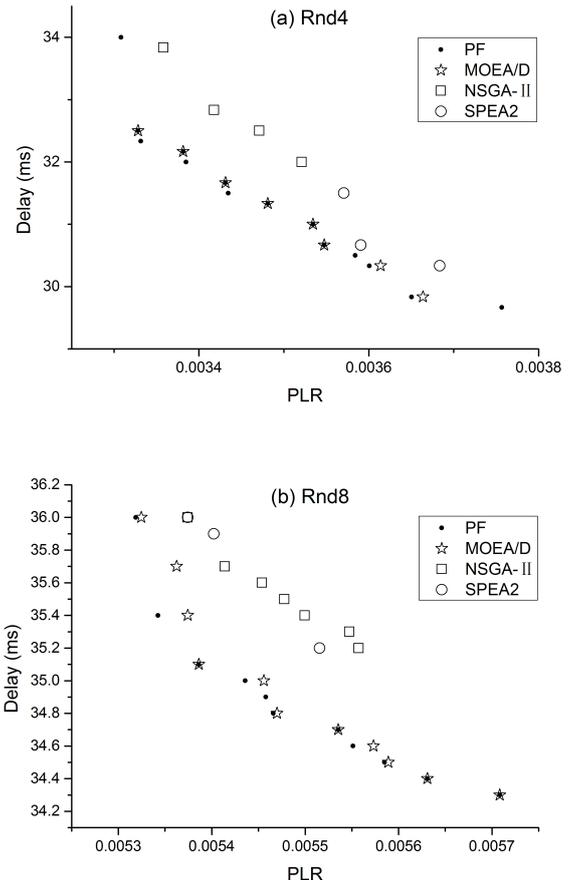
Figure 1: Example of the PFs obtained by different algorithms. (a) Rnd4 network. (b) Rnd8 network.

a number of widely used performance measures, i.e. inverted generational distance, generational distance, maximum spread and average computation time.

6. ACKNOWLEDGMENTS

7. REFERENCES

[1] Xiao Wu, Chong-Wah Ngo, Yi-Ming Zhu, and Qiang Peng. Boosting web video categorization with contextual information from social web. *World Wide Web*, 15(2):197–212, 2012.

[2] Xiao Wu, Yi-Jie Lu, Qiang Peng, and Chong Wah NGO. Mining event structures from web videos. *IEEE Multimedia*, 18(1):38–51, Jan 2011.

[3] S-YR Li, Raymond W. Yeung, and Ning Cai. Linear network coding. *IEEE Transactions on Information Theory*, 49(2):371–381, 2003.

[4] Rudolf Ahlswede, Ning Cai, S-YR Li, and

Table 2: Comparisons of different algorithms (best results are in bold)

Algorithm	GD								IGD							
	Rnd1	Rnd2	Rnd3	Rnd4	Rnd5	Rnd6	Rnd7	Rnd8	Rnd1	Rnd2	Rnd3	Rnd4	Rnd5	Rnd6	Rnd7	Rnd8
SPEA2	**0.000**	0.038	**0.000**	0.802	0.294	0.088	0.053	0.989	**0.000**	0.016	**0.000**	3.625	2.791	2.767	0.254	2.440
NSGA-II	**0.000**	**0.000**	0.015	0.277	0.110	0.035	0.035	1.125	**0.000**	**0.000**	0.009	0.530	0.279	0.023	0.099	2.149
MOEA/D	**0.000**	**0.000**	**0.000**	**0.073**	**0.059**	**0.000**	**0.006**	**0.249**	**0.000**	**0.000**	**0.000**	**0.173**	**0.195**	**0.000**	**0.008**	**0.187**

Algorithm	MS								ACT(sec.)							
	Rnd1	Rnd2	Rnd3	Rnd4	Rnd5	Rnd6	Rnd7	Rnd8	Rnd1	Rnd2	Rnd3	Rnd4	Rnd5	Rnd6	Rnd7	Rnd8
SPEA2	**1.000**	0.987	**1.000**	0.461	0.820	0.919	0.889	0.693	3648	6819	11249	25891	43177	21422	98731	126541
NSGA-II	**1.000**	**1.000**	0.991	0.798	0.889	0.935	0.910	0.673	6302	10418	22219	27539	46811	33156	81218	122446
MOEA/D	**1.000**	**1.000**	**1.000**	**0.829**	**0.914**	**1.000**	**0.958**	**0.811**	**2643**	**3766**	**8651**	**8240**	**18653**	**9031**	**21808**	**50186**

Table 3: Result of t-test

Network	Rnd-1	Rnd-2	Rnd-3	Rnd-4	Rnd-5	Rnd-6	Rnd-7	Rnd-8
MOEA/D↔SPEA2	∼	+	∼	+	+	+	+	+
MOEA/D↔NSGA-II	∼	∼	+	+	∼	+	+	+

Raymond W. Yeung. Network information flow. *IEEE Transactions on Information Theory*, 46(4):1204–1216, 2000.

[5] Michael Langberg, Alexander Sprintson, and Jehoshua Bruck. The encoding complexity of network coding. *IEEE/ACM Transactions on Networking (TON)*, 14(SI):2386–2397, 2006.

[6] Minkyu Kim, Muriel Médard, Varun Aggarwal, U-M O'Reilly, Wonsik Kim, Chang Wook Ahn, and Michelle Effros. Evolutionary approaches to minimizing network coding resources. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pages 1991–1999. IEEE, 2007.

[7] Minkyu Kim, Varun Aggarwal, Una-May O Reilly, Muriel MÃ l'dard, and Wonsik Kim. Genetic representations for evolutionary minimization of network coding resources. In *Applications of Evolutionary Computing*, pages 21–31. Springer, 2007.

[8] Huanlai Xing, Yuefeng Ji, Lin Bai, and Yongmei Sun. An improved quantum-inspired evolutionary algorithm for coding resource optimization based network coding multicast scheme. *AEU-International Journal of Electronics and Communications*, 64(12):1105–1113, 2010.

[9] Huanlai Xing and Rong Qu. A population based incremental learning for delay constrained network coding resource minimization. In *Applications of Evolutionary Computation*, pages 51–60. Springer, 2011.

[10] Huanlai Xing and Rong Qu. A compact genetic algorithm for the network coding based resource minimization problem. *Applied Intelligence*, 36(4):809–823, 2012.

[11] Huanlai Xing, Rong Qu, Graham Kendall, and Ruibin Bai. A path-oriented encoding evolutionary algorithm for network coding resource minimization. *Journal of the Operational Research Society*, 65:1261–1277, 2013.

[12] Huanlai Xing, Rong Qu, Lin Bai, and Yuefeng Ji. On minimizing coding operations in network coding based multicast: an evolutionary algorithm. *Applied Intelligence*, 41(3):820–836, 2014.

[13] Minkyu Kim, Muriel MÍedard, Varun Aggarwal, and Una-May O'Reilly. On the coding-link cost tradeoff in multicast network coding. In *Military Communications Conference, 2007. MILCOM 2007. IEEE*, pages 1–7, Oct 2007.

[14] Minkyu Kim, Muriel MÍedard, and Una-May O'Reilly. Network coding and its implications on optical networking. In *Optical Fiber Communication - incudes post deadline papers, 2009. OFC 2009. Conference on*, pages 1–3. Optical Society of America, March 2009.

[15] Huanlai Xing and Rong Qu. A nondominated sorting genetic algorithm for bi-objective network coding based multicast routing problems. *Information Sciences*, 233:36–53, 2013.

[16] Qingfu Zhang and Hui Li. Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731, 2007.

[17] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[18] Eckart Zitzler, Marco Laumanns, and Lothar Thiele. Spea2: Improving the strength pareto evolutionary algorithm. Technical report, 2001.

[19] Kalyanmoy Deb. Introduction to evolutionary multiobjective optimization. pages 59–96. Springer, 2008.

[20] Andrew V. Goldberg. *A new max-flow algorithm*. Laboratory for Computer Science, Massachusetts Institute of Technology, 1985.

[21] Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, and Keying Ye. *Probability and Statistics for Engineers and Scientists*, volume 5. Macmillan New York, 1993.