

An Improved MOEA/D Algorithm for Multi-objective Multicast Routing with Network Coding

Huanlai-Xing^{a,*}, Zhaoyuan-Wang^a, Tianrui-Li^a, Hui-Li^b, Rong-Qu^c

^a*School of Information Science and Technology,
Southwest Jiaotong University, Chengdu 611756, China*

^b*School of Mathematics and Statistics,
Xi'an Jiaotong University, Xi'an 710049, China*

^c*School of Computer Science,
The University of Nottingham, Nottingham NG8 1BB, UK*

Abstract

Network coding enables higher network throughput, more balanced traffic, and securer data transmission, etc. However, complicated mathematical operations incurs when recombining packets at intermediate nodes, which if not operated properly, leads to very high network resource consumption for the network and unacceptable delay. Therefore, it is of vital importance to minimize various network resources and end-to-end delays while exploiting promising benefits of network coding.

Since multicasting has been used in increasingly more applications, such as video conferencing and remote education, we study the multicast routing problem with network coding. The problem is formulated as a multi-objective optimization problem (MOP), where the coding cost, the link cost and the end-to-end delay are the three objectives to be optimized simultaneously. We adapt multi-objective evolutionary algorithm based on decomposition (MOEA/D) for this MOP by hybridizing it with the population-based incremental learning techniques, which makes use of the global and historical information collected to provide additional guidance to the evolutionary search. Three new schemes

*Corresponding author

Email addresses: hxx@home.swjtu.edu.cn (Huanlai-Xing), wang_zhaoyuan@foxmail.com (Zhaoyuan-Wang), trli@swjtu.edu.cn (Tianrui-Li), lihui10@mail.xjtu.edu.cn (Hui-Li), rong.qu@nottingham.ac.uk (Rong-Qu)

are devised to facilitate the performance improvement, including a probability-based initialization scheme, a problem-specific population updating rule, and a hybridized reproduction operator. Experimental results clearly demonstrate that the proposed algorithm outperforms a number of state-of-the-art MOEAs regarding the solution quality and computational time.

Keywords: Network coding, Multicast, Multi-objective evolutionary algorithm

1. Introduction

Multicast is a one-to-many data delivery method in telecommunications, where information sent from the source is copied and routed to a number of destinations simultaneously. Compared with multiple unicasts, multicast is of high bandwidth-efficiency, especially when there are a large number of receivers [1]. The Internet has witnessed a significant growth in multimedia applications (e.g. video conferencing, IPTV, and remote education), where multicast is a key supporting technique [2]. However, the traditional multicast scheme adopts the store-and-forward data forwarding, where the throughput may not reach to the theoretical maximum [3].

Network coding is a newly emerged communications paradigm, where instead of simply copying and forwarding the incoming data, any intermediate node in the network is allowed to perform mathematical operations (e.g. operations over some finite fields) to recombine different incoming data if necessary [3]. This technique has been reported to be quite effective and helpful in traffic balancing, data security, energy saving, network tomography, and robustness against failures [4, 5, 6, 7]. In particular, with network coding, multicast can always achieve the theoretical maximum multicast data rate [4].

Figure 1 shows a multicast scenario with respect to the multicast data rate, where traditional routing and network coding are employed separately. Figure 1 (a) is the topology of the scenario, where each link is directional and with a capacity of one bit per time unit. Source s wants to multicast two bits, a and b , to two receivers, t_1 and t_2 . According to the Max-Flow Min-Cut theorem,

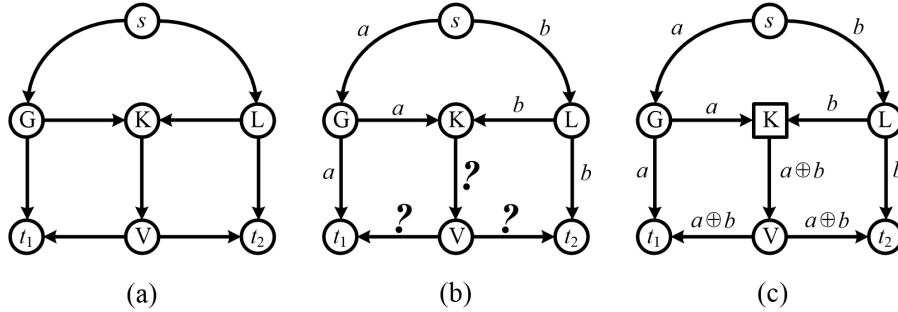


Figure 1: An example multicast scenario. (a) Network topology. (b) Traditional routing. (c) Network coding

the minimum cut between s and t_1 (or t_2) is two bits per time unit, so is the
 25 maximum data rate from s to t_1 and from s to t_2 . Nevertheless, if traditional
 routing is adopted, as can be seen in Figure 1 (b), bottleneck link $K \rightarrow V$ would
 only allow a single bit to be delivered, causing a reduction in the data rate. This
 is because traditional routing is based on the store-and-forward forwarding. On
 the contrary, if node K can perform mathematical operation to recombine a and
 30 b into a single bit, $a \oplus b$, the theoretical maximum data rate to each receiver can
 be obtained at the same time, where in the example of Figure 1 (c) symbol \oplus
 is exclusive-OR operation. Nodes t_1 and t_2 can receive $\{a, a \oplus b\}$ and $\{b, a \oplus b\}$
 and recover the original information a and b after calculating $a \oplus (a \oplus b)$ and
 35 $b \oplus (a \oplus b)$. So, the maximum multicast data rate is equal to the theoretical
 data rate.

As aforementioned, network coding brings benefits to multicast. However, in
 network coding based multicast (NCM), data recombination has to be executed
 at the network layer by performing complicated mathematical operations (called
 coding operations) to combine different incoming information at corresponding
 40 intermediate nodes. Hence, the computational overhead could be extremely high
 and complex coding operations may cause large end-to-end delays. In addition,
 instead of maintaining a single path, NCM employs multiple paths to deliver
 data to any receiver, which enables super-fast data rate, however, at the expense
 of higher link resource consumption. Therefore, research has been conducted

45 to optimize NCM routing from different aspects while utilizing the benefits of network coding.

Main research streams on the NCM routing optimization include coding cost minimization [8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19], link cost minimization [20, 21, 22, 23, 24], delay related optimization [25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35] and multi-objective optimization [36, 37, 38, 39, 40]. Details are given in Subsection 2.2. Towards practical deploying of NCM, it is important to study the trade-off between coding and link costs, as well as satisfying end users with high quality-of-experience (especially delay). However, such issue has received little consideration. Many existing problem models do not take the user experience into account [36]. Some problem models which only concern the minimization of the total cost and end-to-end delay cannot distinguish the trade-off between the coding and link costs [40]. This paper extends the problem models in [36, 40], and establishes a new multi-objective NCM routing model, where all the key factors in NCM data transmission, namely, the coding cost, link cost and the average end-to-end delay, are formulated as three objectives.

Multi-objective evolutionary algorithms (MOEAs) can easily obtain a set of promising solutions in a single run due to their population-based frameworks. They have thus received increasingly more research attention from fields of multi-objective optimization and evolutionary computation. Multi-objective evolutionary algorithm based on decomposition (MOEA/D) is among the highlighted MOEAs [41]. MOEA/D decomposes a multi-objective optimization problem (MOP) into a number of scalar optimization subproblems, each with an aggregated objective. MOEA/D showed to have a better optimization performance with lower computational cost than a number of state-of-the-art MOEAs, e.g. NSGA-II [42] and SPEA2 [43]. In the literature, a number of sophisticated techniques have been incorporated into the MOEA/D framework to further exploit its potential, e.g. estimation of distribution algorithm (EDA) [44, 45, 46, 47, 48, 49, 50], differential evolution (DE) [51, 52, 53, 54, 55, 56], memetic algorithm (MA) [57, 58, 59, 60, 61, 62, 63, 64], ant colony optimization (ACO) [65, 66, 67], particle swarm optimization [68, 69, 70], simulated anneal-

ing [71], and so on. With different techniques integrated, hybrid MOEA/Ds are usually reported to gain decent optimization performance when solving MOPs. Details are reviewed in Subsection 2.2.

As one of the EDAs, population based incremental learning (PBIL) combines GA and machine learning. It manipulates a real-valued probability vector (PV) and extracts statistical information from promising samples to evolve the PV [72]. Unlike other EAs, the evolutionary process of PBIL involves neither explicit population nor complicated operators, such as crossover and mutation, thus incurs much less computational and memory costs while gaining similar or even better optimization performance, compared with traditional EAs [72]. Moreover, PBIL has been reported as an excellent optimizer for solving the NCM-based single-objective optimization problem [14]. We thus explore the potential of integrating PBIL components into MOEA/D to strengthen MOEAs when addressing the three-objective NCM routing problem in this paper.

The contribution of the work includes the formulation of a new multi-objective optimization problem and a hybrid MOEA to address it, as listed below.

- **A NCM routing optimization problem with three objectives.** The computing resource, bandwidth resource and delay are all important factors when considering the practical deployment of NCM. All of them need to be kept as low as possible. In this work we formulate a three-objective optimization problem, simultaneously minimizing three objectives, i.e. the coding cost, the link cost and the average end-to-end delay of NCM.
- To tackle the MOP above, we propose a hybrid MOEA incorporating PBIL into the original MOEA/D framework, with three novel features listed as below.
 - **A probability-based initialization scheme.** The problem concerned in the paper is highly constrained and infeasible solutions dominate the search space. To start with an initial population of feasible individuals, we propose a probability-based initialization

105 scheme, where each individual is created according to the estimated
distribution of feasible solutions. This scheme helps to obtain a set
of promising individuals with high diversity.

– **A problem-specific population updating rule.** Due to the
special features of the proposed problem, when adopted, the orig-
110 inal MOEA/D may reproduce similar individuals in the population,
leading to serious prematurity and thus a deteriorated optimization
performance. To overcome this problem, the paper introduces a
problem-specific population updating rule. Once a promising indi-
vidual is generated, it updates a single individual in the population,
115 if the improvement of the individual quality is the most significant
among this current population. This helps reserve high level of
diversity.

– **A hybridized reproduction operator.** Global exploration and
local exploitation are two important research issues in designing ef-
120 ficient and effective MOEAs. However, they usually contradict with
each other. To address this, we devise a reproduction operator which
combines reproduction techniques in GA and PBIL. A control func-
tion is devised to decide the percentage of individuals generated from
each reproduction technique. Analysis indicates that with this oper-
125 ator, the evolution is able to maintain a relatively high level of global
exploration thus contribute to a balanced optimization performance.

The rest of the paper is organized as follows. Section 2 describes the problem
formulation and related works. Section 3 briefly reviews the original MOEA/D
and PBIL. The proposed algorithm is introduced in Section 4. Simulation results
130 are demonstrated in Section 5. In Section 6, conclusions are provided.

2. Problem formulation and related work

2.1. Problem formulation

The network is represented by a directed graph $G = (V, E)$, where V is the node set, E is the link set and each link $e \in E$ has a unit capacity. In NCM
135 on network G , there are a source node $s \in V$, a set of receivers $T = \{t_1, \dots, t_d\}$,
 $t_k \in V$, and an expected data rate R . The source delivers the same data to each
node $t_k \in T$ at R [4, 6].

Given a NCM request, the task is to find a connected subgraph in G to
support the multicast with network coding [15]. This subgraph is referred to
140 as a NCM subgraph (denoted by G_{NCM}). A NCM subgraph includes R link-
disjoint paths connecting s and each receiver. A coding node is a node which
performs coding operations; a coding link is an outgoing link of a coding node
via which the outgoing data are a combination of the data received by the
coding node. In network G , a *merging node* is a non-receiver intermediate node
145 with multiple incoming links [10, 11]. Only merging nodes can become coding
nodes and perform packet recombination. The number of coding links is used
to estimate the amount of coding operations performed in the NCM [9]. More
descriptions can be found in [15].

Figure 2 illustrates an example NCM scenario, where source s delivers two
150 bits, a and b , to two receivers, t_1 and t_2 , respectively. The data transmission
scheme is shown in Figure 2 (a), where coding node K performs packet recom-
bination $a \oplus b$. Figure 2 (b) shows the four paths originating from source s
to one of the receivers. Note that, paths to the same receiver are link-disjoint
paths. For example, paths $P_1(s \rightarrow t_1)$ and $P_2(s \rightarrow t_1)$ are link-disjoint. All
155 paths transmitting the NCM data form the NCM subgraph, as illustrated in
Figure 2 (b).

The following lists notations used in the paper:

- s : the source node in network $G(V, E)$;
- $T = t_1, t_2, \dots, t_d$: the set of receivers, where $d = |T|$ is the number of
160 receivers;

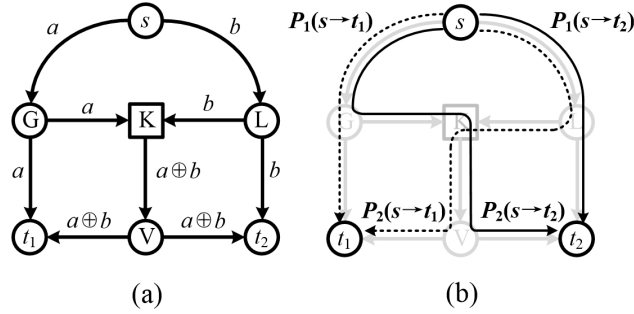


Figure 2: An example NCM scenario. (a) Data delivery. (b) The NCM subgraph.

- R : the data rate (an integer) at which s delivers data to T ;
- $P_i(s, t_k)$: the i -th path from s to t_k , where $t_k \in T$ and $i = 1, \dots, R$;
- $r(s, t_k)$: the achievable data rate from s to receiver $t_k \in T$;
- C_{code} : the number of coding links in $G_{NCM}(s, T)$;
- 165 • $c_{link}(e)$: the cost incurred on link e if $e \in G_{NCM}(s, T)$;
- $delay(P_i(s, t_k))$: the end-to-end delay of path $P_i(s, t_k)$.

The task is to find an appropriate NCM subgraph in $G(V, E)$, which satisfies the NCM data rate requirement, with three objectives minimized, as shown below in Eq. 1.

Minimize:

$$\begin{cases} f_1 = \sum_{\forall e \in G_{NCM}(s, T)} c_{link}(e) \\ f_2 = C_{code} \\ f_3 = \frac{1}{d} \sum_{k=1}^d \max\{delay(P_i(s, t_k)) | i = 1, \dots, R\} \end{cases} \quad (1)$$

Subject to:

$$r(s, t_k) = R, \forall t_k \in T \quad (2)$$

- 170 In Eq. 1, objective f_1 is to minimize the bandwidth resource, i.e. total link cost involved during the NCM; objective f_2 is to minimize the computing resource consumption, i.e. coding cost, in the NCM; objective f_3 is to minimize

the average end-to-end delay along all paths in the NCM subgraph. Constraint 2 restricts that R link-disjoint paths are to be constructed from s to each of the receivers so that the expected data rate R is achievable.

The above three-objective minimization problem belongs to MOPs. Suppose there are two solutions (f_1^*, f_2^*, f_3^*) and (f_1', f_2', f_3') . (f_1^*, f_2^*, f_3^*) dominates (f_1', f_2', f_3') or (f_1', f_2', f_3') is dominated by (f_1^*, f_2^*, f_3^*) only if any of the following three conditions is satisfied: $\{f_1^* < f_1', f_2^* \leq f_2', f_3^* \leq f_3'\}$ or $\{f_1^* \leq f_1', f_2^* < f_2', f_3^* \leq f_3'\}$ or $\{f_1^* \leq f_1', f_2^* \leq f_2', f_3^* < f_3'\}$. Optimal solutions to the problem above constitute the Pareto-optimal Set (PS).

2.2. Related work-an overview

This subsection first reviews the main streams of NCM routing optimization problems in the literature.

(1) Optimization in NCM Routing

- **Coding cost minimization.** Performing coding operations consumes extra computing resources, compared with the traditional store-and-forward based routing. Hence, one research stream focuses on minimizing the amount of coding operations necessarily performed. Early research studied greedy-based optimization approaches [8, 9]. Later on, several genetic algorithms (GAs) were proposed for minimizing coding cost [10, 11, 12]. Recent research adapted estimation of distribution algorithms (EDAs), including quantum-inspired evolutionary algorithms (EAs) [13], population based incremental learning (PBIL) [14] and compact GA [15]. Moreover, EAs hybridized with other techniques, e.g. entropy-based evaluation relaxation and path-oriented encoding, were investigated [16, 17, 18, 19].

- **Link cost minimization.** NCM data are delivered through multiple paths which are made up of links [8]. In real networks, different links, when employed for data transmission, incur different costs, known as link costs. NCM routing plans with less total link costs are thus preferred. Lun et al. formulated a minimum-cost NCM multicast over packet networks,

where wireline and wireless networks were both considered [20, 21, 22]. Cui and Ho studied the least-cost integral network coding problem, where the packet injection rate on each link was constrained to be integral [23]. Researchers also investigated minimum cost subgraph construction in static and dynamic environments [24].

- **Delay related optimization.** Delay is one of the most important metrics evaluating network performance. A considerable amount of applications require guarantees on stringent delay, especially for real-time broadband multimedia applications [25]. However, network coding gains high bandwidth utilization at the expense of consuming extra computational resources at corresponding nodes [4]. Packet recombination (i.e. coding operation) incurs additional processing delay in individual nodes, and cause severely large end-to-end delays if data are not routed appropriately. Delay-related issues, when deploying practical NCM, have thus drawn a great amount of attention. Delay analysis and its minimization have been studied in the context of wireless networks [26, 27], overlay networks [28], broadcast erasure channels with feedback [29, 30], instantly decodable network coding [31, 32, 33], and multicasting [34, 35].
- **Multi-objective optimization.** All the above research concerned single-objective optimization problems. However, both coding and link costs incur in real-world NCM data delivery thus should be both minimized. This problem can be formulated as a bi-objective optimization problem, for which a number of multi-objective evolutionary algorithms (MOEAs) have been proposed [36, 37, 38, 39]. When launching NCM, Network Service Providers (NSP) pay for computing and bandwidth resources they consume. Optimization on the two objectives helps NSPs to find a trade-off between the cost of paying the limited computing resource (coding) and bandwidth resource (link), to gain high profits. In addition, end users usually expect to have decent quality of experience (QoE), especially small end-to-end delay. This is in nature conflicted interests to those NSPs who

prefer less network resource consumption. In [40], the trade-offs between the total cost (i.e. weighted sum of the coding and link costs) and the maximum end-to-end delay of multiple paths were studied.

235 (2) MOEA/Ds integrated with other techniques

Incorporating sophisticated techniques into MOEA/D has become an important direction in the MOEA/D family, including estimation of distribution algorithms, differential evolution, memetic algorithm, ant colony optimization and so on, as the following reviews.

- 240 • **Estimation of distribution algorithm (EDA)**. Recently, EDAs gain good attentions in solving various single-optimization problems (SOPs) [44]. In principle, they are a family of EAs that incorporate machine learning techniques, where statistical information of promising solutions is extracted to build probabilistic models, from which samples are generated. 245 Compared with traditional EAs, EDAs usually obtain better optimization results, with relatively less space and time complexity. A decomposition-based EDA, namely MEDA/D, is proposed to handle the multiobjective knapsack problems [45]. Shim et al. incorporated the restricted Boltzmann machine and the evolutionary gradient search into the MOEA/D framework [46]. This algorithm performs well in addressing the multiobj- 250 jective multiple traveling salesman problem (TSP). Gao et al. investigated a similar problem, i.e. multiobjective TSP, using multiobjective EDA based on decomposition (MEDA/D) [47], where the probabilistic model is built based on priori and learnt information. To gain a balanced per- 255 formance on global exploration and local exploitation, a hybrid adaptive MOEA that synthesizes GA, EDA and DE was presented [48]. Promising solutions generated at an early stage of evolutions by these algorithms are used to produce corresponding proportion of solutions in the next generation. Giagkiozis et al. developed a combination of MOEA/D and EDA for 260 the many-objective optimization problems [49], where a novel generalized

decomposition method unifies different performance objectives. Ray et al. designed a hybridized architecture combining MOEA/D and quantum genetic algorithm for tackling many objective optimization problems [50], where systematic sampling is adopted to establish the reference directions and the evolution of quantum individuals is driven by a simple variation operator.

265

- **Differential evolution (DE)**. DE was integrated into the framework of MOEA/D in [51] to effectively handle complicated Pareto fronts for MOPs, and it is reported to perform much better than NSGA-II. A variant of MOEA/D-DE was presented for the multi-objective analog cell sizing problem [52]. Two performance enhancing mechanisms are incorporated to balance between the diversity and guiding information from neighbors, and to improve the local search ability of DE using a scaling factor. Tan et al. proposed a modified MOEA/D-DE with a uniform design method to generate uniformly distributed scalar optimization subproblems and a simplified quadratic approximation to enhance the local exploitation and the accuracy of aggregation function values [53]. Combined with Gaussian mutation operators, MOEA/D-DE also has a decent performance in devising Yagi-Uda antennas [54]. Another variant of MOEA/D-DE, namely the adaptive DE for multiobjective problem (ADEMO/D), integrated with a number of adaptive strategies, gains evenly distributed solutions well approximating the Pareto front for continuous MOPs [55, 56].

270

- **Memetic algorithm (MA)**. Local search operators (assisted with domain knowledge) have recently been incorporated into MOEA/D. With improved local exploitation, the proposed algorithms (usually called multi-objective memetic algorithms) provide better solutions than pure MOEA/D. Chen et al. enhanced the performance of MOEA/D by integrating guided mutation and priority update [57]. Mei et al. proposed an MA/D with extended neighborhood search, namely D-MAENS, for solving the capacitated arc routing problem [58]. Later on, Shang et al. improved the

285

290

performance of D-MAENS by two novel schemes, one for solution replacement and the other for elitism maintenance [59]. MOEA/D is hybridized with a mathematical programming technique (called Nelder and Mead’s algorithm), where Nelder and Mead’s algorithm serves as the local search mechanism [60]. Alhindi and Zhang investigated how guided local search is used to strengthen MOEA/D in terms of escaping local optima [61]. Mashwani and Salhi presented a hybrid MOEA/D, where particle swarm optimization (PSO) and DE are incorporated. In the algorithm, DE acts as the main evolutionary framework and PSO is in charge of local search [62]. By combining ideas from MOEA/D and Pareto local search, Ke et al. proposed a memetic algorithm based on decomposition (MOMAD) [63], where three populations are initialized by a problem-specific single objective heuristic and evolved by the Pareto local search and single objective local search procedures. Ma et al. developed a MOEA/D with Baldwinian learning for continuous MOPs, where evolving information from the distribution model of the population is extracted by a Baldwinian learning operator [64].

- **Ant colony optimization (ACO).** Inspired by ACO, Li et al. introduced a probabilistic representation based on pheromone trails into MOEA/D and demonstrated its good potential in handling hard MOPs with many local optima [65]. Ke et al. introduced the ACO into MOEA/D for solving multi-objective 0-1 knapsack problem and bi-objective TSP [66]. Instead of using sub-colonies, each ant solves one of the scalar optimization problems obtained. Cheng et al. proposed a hybrid multiobjective optimization framework integrating the ACO into MOEA/D, called MoACO/D [67], where an ant colony is divided into many sub-colonies in an overlapped manner, and each sub-colony addresses a certain SOP decomposed from the original MOP.

- **Other techniques.** A considerable amount of research efforts have also been made to incorporating various optimization techniques into the

MOEA/D framework. These include PSO [68, 69, 70], simulated annealing [71], artificial bee colony optimization [73], fuzzy system [74], Gaussian process model [75], opposition-learning [76], teaching-learning algorithm [73, 77], and so on.

3. Overview of MOEA/D and PBIL

3.1. MOEA/D

In MOEA/D, the fundamental idea is to decompose a MOP into N scalar optimization subproblems (SOSPs) [41]. MOEA/D aims to optimize all SOSPs simultaneously in a collaborative and time-efficient manner. Three decomposition methods are introduced in [41]. This paper considers the Tchebycheff approach, one of the most commonly used. A SOSP achieved by the decomposition of a MOP can be expressed as follows:

$$\text{Minimize : } g(x|\lambda, z^*) = \max_{1 \leq j \leq m} \{\lambda_j |f_j(x) - z_j^*|\} \quad (3)$$

$$\text{Subject to : } x \in \Omega \quad (4)$$

where m is the number of objectives, $\lambda = (\lambda_1, \dots, \lambda_m)$ is a weight vector, i.e. $\lambda_j \geq 0, j = 1, \dots, m$, and $\sum_{j=1}^m \lambda_j = 1$. $z^* = \{z_1^*, \dots, z_m^*\}$ is the reference point, i.e. $z^* = \min\{f_j(x) | x \in \Omega\}$, where Ω is the decision space.

It is assumed that a set of N weight vectors $\lambda^1, \dots, \lambda^N$ should be selected properly so the optimal solutions of those SOSPs will well approximate the Pareto-optimal front (PF). In addition, the neighborhood relationship of SOSPs can be measured by Euclidean distances between the weight vectors. Neighboring SOSPs have similar fitness landscapes and their optimal solutions should be close in the decision space. Information sharing between neighborhoods thus can be exploited to accomplish the optimization task.

The evolutionary procedure of MOEA/D can be described below.

Global structure:

- A population of N points $x_1, \dots, x_N \in \Omega$, where x_i is the current individual to SOSP(i), the i -th SOSP.
- $z = (z_1, \dots, z_m)$, where $z_j, j = 1, \dots, m$, is the best-so-far value for objective f_j .
- An external population (EP), which stores nondominated solutions found during the search.

Input: a given MOP; stopping criteria; N : the number of SOSPs; W : the number of the neighbors for each SOSP; $\lambda^1, \dots, \lambda^N$: uniformly distributed weight vectors; p_c : the crossover rate; p_m : the mutation rate.

MOEA/D Procedure:

Initialization:

- 1: Set $EP = \emptyset$.
- 2: For arbitrary weight vector λ^i , calculate the W closest weight vectors, $\lambda^{i(1)}, \dots, \lambda^{i(W)}$, via Euclidean distance and set $\varphi(i) = \{i(1), \dots, i(W)\}$.
- 3: Generate an initial population x_1, \dots, x_N and evaluate $f_u(x_j)$ for each individual.
- 4: Initialize $z = \{z_1, \dots, z_m\}$.

Repeat:

- 5: **for** $i = 1$ to N **do**
- 6: **Reproduction:** Generate a new solution y by two individuals x_u and x_l using crossover and mutation operators, where $u, l \in \varphi(i)$.
- 7: **Improvement:** Improve y by using a problem-specific improvement repair operator, which is *optional*.
- 8: **Update of z :** For $j = 1, \dots, m$, if $f_j(y) < z_j$, set $z_j = f_j(y)$.
- 9: **Update of neighboring solutions:** For each $k \in \varphi(i)$, if $g(y|\lambda^k, z) \leq g(x_k|\lambda^k, z)$, then set $x_k = y$ and $f_j(x_k) = f_j(y), j = 1, \dots, m$.
- 10: **Update of EP:** Remove those solutions dominated by y from EP and add y to EP if it is not dominated by any member in EP.
- 11: **end for**

Termination:

12: Until stopping criteria are satisfied, output EP.

370 3.2. PBIL

PBIL has been reported to gain promising performance when solving the single-objective network coding resource minimization problem [14]. Instead of using an explicit population, PBIL manipulates a real-valued probability vector (PV). When sampled, PV generates a number of binary solutions and the best
375 one is used to update the PV. By making use of global information, promising solutions are generated with increasingly higher probabilities stored in PV.

Let $\mathbf{P}^{(k)} = \{P_1^k, \dots, P_L^k\}$ be a PV at generation k , where L is the dimension of the solution encoding and P_l^k is the probability of obtaining ‘1’ at the l -th position. Denote $\mathbf{B}^{(k)} = \{B_1^k, \dots, B_L^k\}$ and α the best so far solution during the
380 search and the learning rate, respectively. Figure 3 shows the procedure of the original PBIL. The PV at generation k , $\mathbf{P}^{(k)}$, is updated by Eq. 5.

$$\mathbf{P}^{(k)} = (1.0 - \alpha) \cdot \mathbf{P}^{(k-1)} + \alpha \cdot \mathbf{B}^{(k)} \quad (5)$$

After PV is updated, mutation operation may be used to avoid local optima [72]. Let α be the probability shifting at each position, and P_l^k is to be mutated, Eq. 6 is typically adopted in mutation.

$$P_l^{(k)} = (1.0 - \sigma) \cdot P_l^{(k-1)} + f_{rnd} \cdot \sigma \quad (6)$$

385 where f_{rnd} is either 0.0 or 1.0, randomly generated with probability 0.5.

4. The proposed MOEA/D-PBIL

As known, the individual representation is one of the most important issues in EAs. This section starts with the individual representation used for the problem concerned in this paper. Then, three novel features, i.e. a probability-
390 based initialization scheme, a problem-specific population updating rule and a hybridized reproduction operator, are introduced. At last, the overall procedure of the proposed algorithm is given in detail.

Initialization:

- 1: Set $k = 0$.
- 2: Set $P_l^k = 0.5, l = 1, \dots, L$. So $\mathbf{P}^{(k)}$ is initialized as $\{0.5, \dots, 0.5\}$.
- 3: Sampling a set $\mathbf{S}^{(k)}$ of N individuals from $\mathbf{P}^{(k)}$ and find the best sample and $\mathbf{B}^{(k)}$.

Repeat:

- 4: Set $k = k + 1$.
- 5: Find the best sample $\mathbf{B}^{(k)}$ from $\mathbf{B}^{(k-1)} \cup \mathbf{S}^{(k-1)}$.
- 6: Update $\mathbf{P}^{(k)}$ by Eq. 5.
- 7: Mutate $\mathbf{P}^{(k)}$ by Eq. 6.
- 8: Sampling a set $\mathbf{S}^{(k)}$ of N individuals from $\mathbf{P}^{(k)}$.

Termination:

- 9: Until stopping criteria are satisfied, output $\mathbf{B}^{(k)}$.

Figure 3: Procedure of the original PBIL [72]

4.1. Individual representation and evaluation

Binary link state individual representation (BLS-IR) has been widely used
 395 in network coding related optimization problems, including a number of single-
 objective optimization problems and MOPs [10, 11, 12, 13, 14, 15, 19, 40].
 In particular, BLS-IR is able to facilitate an easy process of estimating the
 consumption of the coding resource during the NCM data transmission. As
 mentioned before, coding operations can be performed at merging nodes only.
 400 BLS-IR is based on the graph decomposition method (GDM) which helps to
 clearly show how information flows are forwarded within each merging node [11].
 The MOP concerned in this paper also involves the estimation of coding resource
 consumption, so it is rationale to utilize BLS-IR to represent individuals. The
 following introduces GDM, BLS-IR and the raw fitness evaluation.

405 In GDM, each merging node M with I_M incoming links and O_M outgoing
 links is decomposed into I_M incoming auxiliary nodes and O_M outgoing aux-
 iliary nodes, connected by all possible routes passing the merging node. Each

link flows into node M is redirected to one of the I_M incoming auxiliary nodes, and each node has only one link flows into it. Similarly, each outgoing link from
410 node M is redirected to one of the O_M outgoing auxiliary nodes and each node has only one outgoing link. Besides, within each decomposed merging node, an auxiliary link connects each incoming auxiliary node with each outgoing auxiliary node. Given an original graph G , every merging node is decomposed by GDM and then a decomposed graph G' is created.

415 In BLS-IR, each individual x is represented by a string of binary variables, each associated with an auxiliary link between auxiliary nodes. Value 1 for a binary variable means the corresponding link in G' is active and information can pass by; otherwise, the corresponding link in G' is inactive and no information is allowed to pass. An individual x thus corresponds to an explicit and unique
420 decomposed graph $G_D(x)$. Based on $G_D(x)$, we determine if a valid NCM subgraph (see Section 2 for details) can be found.

The feasibility is firstly checked when evaluating an individual x . If a NCM subgraph from the corresponding decomposed graph $G_D(x)$ with the expected data rate satisfied can be found, x is feasible; otherwise, it is regarded infeasible.
425 One of the max-flow algorithms, the Goldberg algorithm, is used to calculate the max flow between the source and each receiver within the obtained NCM subgraph [78]. For each feasible individual, three objective values are calculated according to Eq. 1. For infeasible individuals, three sufficiently large objective values are set, ensuring that infeasible individuals are less competitive than
430 feasible ones during the evolutionary search procedure.

4.2. The probability-based initialization scheme

In MOEAs, the initial population generally has a great impact on the optimization performance. Unfortunately, the problem concerned in the paper is highly constrained, and with BLS-IR, infeasible solutions dominate the search
435 space. A random initial population very likely leads to a deteriorated optimization performance. Thus, a probability-based initialization (PBI) scheme is devised in this paper to guarantee that the proposed MOEA begins with a set

```

1: Set initial population  $set_{init} = \emptyset$  and probability vector  $\mathbf{P}^{(init)} =$ 
    $\{P_{init}, P_{init}, \dots, P_{init}\}$ .
2: while  $|set_{init}| < N$  do
3:   Generate a new individual  $x$  by sampling  $\mathbf{P}^{(init)}$  once.
4:   if  $x$  is feasible then
5:     Place  $x$  in  $set_{init}$ .
6:   end if
7: end while

```

Figure 4: Pseudo code of the PBI scheme

of feasible individuals with high level of diversity.

In the literature, to deal with such problem, Kim et al. inserted an all-one
440 individual into the initial population to ensure that the search starts with at least
one feasible solution [11, 12]. However, such method is not effective for MOPs,
as MOEAs require higher level of population diversity than single-objective EAs.
Therefore, our previous work investigated the estimated distribution of feasible
solutions over the entire search space [40]. It was found that the majority of
445 feasible solutions are closer to the all-one individual. Based on this finding, a
smart initialization scheme is proposed to generate an individual pool of multiple
feasible individuals based on the all-one individual. However, such scheme leads
to an initial population of highly similar individuals, which seriously harms the
population diversity.

We use the concept of PV in PBIL to generate the initial population. The
450 distribution of feasible individuals in the search space is estimated to extract
statistical information. Instead of setting each value in the PV to 0.5, we set a
larger probability at each position of PV, to generate feasible individuals with
a higher probability. Figure 4 shows the pseudo code of the PBI scheme. This
455 is in compliance with the finding above, i.e. an individual similar to the all-one
individual is more likely to be feasible.

The PBI scheme generates the initial population with feasible individuals

while considering its diversification. As long as P_{init} is not set too close to 1, the initial population could maintain a certain degree of diversity. A smaller P_{init} is
460 more likely to gain a more diversified initial population, which is of course at the expense of longer computational time. Since diversity is extremely important for MOEAs, it is worth compromising the computational cost.

4.3. A problem-specific population updating rule

In the original MOEA/D, a better individual replaces not only the best-so-
465 far individual of the corresponding SOSP, but also those of neighboring SOSPs. However, the problem concerned in the paper is highly complicated and constrained, and feasible individuals only account for a very small proportion of the population [40]. In addition, the majority of feasible individuals are close to the all-one individual. If we adapt the original population updating rule, where
470 better individual updates every SOSP within the same neighborhood, similar individuals will rapidly dominate the population, cause serious prematurity and deteriorate the optimization performance.

To overcome the above problem, this paper proposes a problem-specific population updating (PSPU) rule, where, instead of multiple SOSPs, only a single
475 SOSP is updated with the newly generated promising individual. Let the i -th SOSP be denoted by $SOSP(i)$, where $i = 1, \dots, N$. Let $SOSPs(i)$ be the set of the neighbors of $SOSP(i)$ including itself, where $SOSPs(i) = \{SOSP(i(1)), \dots, SOSP(i(W))\}$, $i(1), \dots, i(W) \in \varphi(i)$ (see Subsection 3.1 for details). For an arbitrary $SOSP(i)$, an individual y is generated after reproduction, and replaces the
480 individual of a neighboring SOSP with the most significant fitness improvement. Note that it is possible the newly generated individual is worse than all of the current individuals of $SOSPs(i)$. In this case, the new one is discarded.

The fitness improvement of $SOSP(i)$, $\Delta_{SOSP(i)}$, and the most significant improvement regarding the fitness among $SOSPs(i)$, Δ_{max} , are defined in Eq.
485 7 and Eq. 8, respectively.

$$\Delta_{SOSP(i)} = g(x_i|\lambda^i, z) - g(y|\lambda^i, z) \quad (7)$$

$$\Delta_{\max} = \max \Delta_{SOSP(j)}, j \in \varphi(i) \quad (8)$$

where, $\varphi(i)$ contains the indexes of all SOSPs in $SOSPs(i)$.

Compared with the original population updating rule in MOEA/D, the proposed PSPU rule defines that a newly generated individual updates the most appropriate SOSP only. Thus, the search is guided to explore promising areas in the search space while maintaining a diversified population. With the proposed
 490 rule, MOEA/D-PBIL gains better performance as observed in Subsection 5.4.

4.4. A hybridized reproduction scheme

When designing MOEAs, the issue of exploration and exploitation at different stages of the search should be carefully considered to support effective search
 495 over the vast solution landscape. Traditional EA recombination operators, e.g. crossover, recombine at least two individuals selected from the population, making use of the local information only. They perform well at the beginning of the evolution, but get worse due to gradual loss of population diversity, leading to a deteriorated global search performance. PBIL manipulates a PV and generates
 500 new individuals by sampling from it. By making use of the global and historical information, promising regions can be explored in parallel and new regions also have chance to be discovered. An effective global exploration is obtained by the intrinsic memory of PV. The recombination of PBIL can thus act as a complement to the traditional EA recombination.

To achieve a balanced global exploration and local exploitation, MOEA/D-
 505 PBIL utilizes a hybridized reproduction (HR) scheme which uses the genetic operators of MOEA/D and the probabilistic sampling operators of PBIL at different stages of the evolution. By controlling the proportion of the offspring produced by MOEA/D and those by PBIL, the proposed algorithm aims at
 510 striking a balanced global exploration and local exploitation. To be specific, at the early stage of the evolution, MOEA/D reproduction is selected with a higher probability, which helps to explore the search space; at the middle stage of the evolution, both reproductions incur; at the late stage, PBIL-reproduction

is more likely to be selected for concentrating on promising areas in the search
515 space.

According to the HR scheme, we need a controlling parameter $CP(k)$ for generation k to determine how many individuals are generated by PBIL-reproduction. We find that the cumulative distribution function (CDF) of the Cauchy distribution can be adapted for controlling the PBIL offspring proportion, since the
520 CDF curve grows gradually and smoothly from a value close to 0 to a value close to 1 [79]. We thus define $CP(k)$ in Eq. 9. This parameter determines which reproduction method is used to generate a new individual. For the i -th SOSP, if a uniformly distributed random number $rand \leq CP(k)$, then PBIL-reproduction is chosen to produce offspring; otherwise, MOEA/D-reproduction
525 is chosen. If PBIL-reproduction cannot produce a feasible individual after a predefined number of attempts, especially in the early stage of the evolution when building up the probabilistic blocks for the PV, MOEA/D-reproduction is used instead.

$$CP(k) = \frac{1}{\pi} \arctan\left(\frac{k - K/2}{\gamma}\right) + \frac{1}{2} \quad (9)$$

where, parameter γ is a predefined value governing the steepness of the CP
530 curve and parameter K is a predefined number of generations.

Figure 5 illustrates an example curve of CP, where K is set to 200 and γ is set to 4, 11, and 18, respectively. A smaller γ leads to a deeper slope (in the paper, γ is fixed at 11). It is clear that at the early stage ($k = 1-50$), MOEA/D-reproduction is more likely to be chosen; during the middle stage ($k = 51-150$),
535 the probability of selecting PBIL-reproduction gradually increases and becomes higher than that of MOEA/D-reproduction after $k = 100$; at the last stage, individuals generated by PBIL-reproduction dominate the population. Using this controlling parameter, a balanced global exploration and local exploitation is obtained, leading to a decent performance as seen in Subsection 5.5.

540 Figure 6 illustrates the procedure of the proposed HR scheme at generation k . With the calculated $CP(k)$ according to Eq. 9, the HR scheme decides the

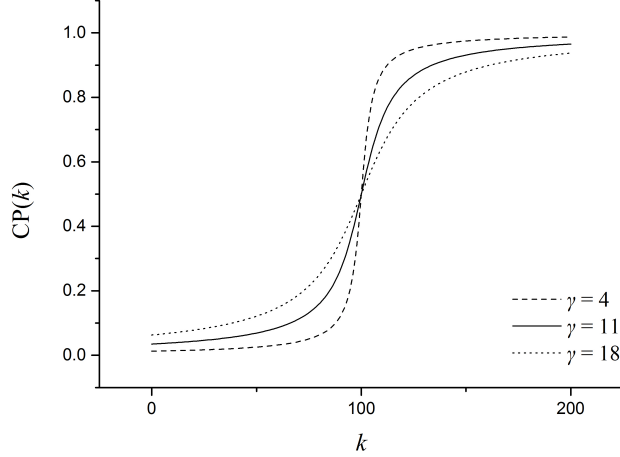


Figure 5: Example CP with $K = 200$ and different values of γ

percentage of individuals each reproduction method generates at a certain generation. Let the PV associated with the i -th SOSp at generation k denoted by $\mathbf{P}^{(i,k)}$. Variable `attempts` is a counter, which records how many times $\mathbf{P}^{(i,k)}$ has
545 been sampled before a feasible individual appears. Variable `isSuccess` records the state whether PBIL-reproduction successfully produces a feasible individual. In the early stage of the search, $\mathbf{P}^{(i,k)}$ focuses on learning probabilistic features from promising samples. During this period, it is very likely that sampling $\mathbf{P}^{(i,k)}$ only results into infeasible individuals. The initial state of `isSuccess`
550 is set to false. Constant Θ stands for the maximum number of attempts tried when sampling $\mathbf{P}^{(i,k)}$. When PBIL-reproduction is chosen, $\mathbf{P}^{(i,k)}$ is repeatedly sampled. This procedure stops when either a feasible individual appears or Θ attempts have been tried. If no feasible individual can be generated, we use MOEA/D-reproduction to produce a new individual.

555 The HR scheme has one significant advantage, i.e. helping to balance the global exploration and the local exploitation during the evolution. By incorporating PBIL reproduction method, the HR scheme is able to enhance the global exploitation ability of the proposed MOEA. This helps to provide a bal-

```

1: Calculate the value of the controlling parameter  $CP(k)$ .           ▷ (Eq. 9)
2: for  $i = 1$  to  $N$  do                                           ▷ (For the  $i$ -th SOSp)
3:   Generate a random number  $rand$ .           ▷ (uniformly distributed in  $[0, 1]$ )
4:   if  $rand \leq CP(k)$  then                                       ▷ (PBIL-reproduction is used)
5:     Set  $attempts = 0$  &  $isSuccess = false$ .
6:     while  $attempts < \Theta$  do                                     ▷ (Attempts a number of times)
7:       Set  $attempts = attempts + 1$ .
8:       Sample an individual  $y$  from PV  $\mathbf{P}^{(i,k)}$ .           ▷ (Subsection 3.2)
9:       if  $y$  is feasible then
10:          $isSuccess = true$ .
11:       Break.
12:     end if
13:   end while
14:   if  $isSuccess == false$  then   ▷ (MOEA/D-reproduction is used)
15:     Generate an individual  $y$  by using crossover and mutation.   ▷
    (Subsection 3.1)
16:   end if
17:   else                                                           ▷ (MOEA/D-reproduction is used)
18:     Generate an individual  $y$  by using crossover and mutation.   ▷
    (Subsection 3.1)
19:   end if
20: end for
Output: the offspring population.

```

Figure 6: Procedure of the HR scheme at generation k

anced performance between global exploration and local exploitation during all
560 stages of the evolution, which is in favor of gaining an excellent optimization
performance.

4.5. The overall procedure of MOEA/D-PBIL

The proposed MOEA/D-PBIL is based on the basic evolutionary frame-
work of MOEA/D (already reviewed in Subsection 3.1). Let k be the current
565 generation of evolution. The following gives the whole evolutionary procedure.

Input:

- the MOP with m objectives and individual length L ; the stopping criteria;
the population size N ; the number of neighbors W ; the N weight vectors
 $\lambda^1, \dots, \lambda^N$; the crossover rate p_c ; the mutation rate p_m (Subsection 3.1)
- 570 • the learning rate α , the probability shifting σ (Subsection 3.2)
- the probability for the PBI scheme P_{init} (Subsection 4.2)
- the predefined number of generations K ; the predefined value for smooth-
ness γ in $CP(k)$; the predefined number of attempts Θ (Subsection 4.4)

MOEA/D-PBIL Procedure:

575 **Initialization:**

- 1: Set $EP = \emptyset$ and $k = 0$.
- 2: Calculate $\varphi(i)$ neighbors for $SOSP(i)$, $i = 1, \dots, N$. ▷ (Subsection 3.1)
- 3: Generate a population x_1, \dots, x_n by the PBI scheme. ▷ (Subsection 4.2)
- 4: Initialize $z = \{z_1, \dots, z_m\}$.
- 580 5: Initialize $\mathbf{P}^{(i,k)} = \{P_1^k, \dots, P_L^k\}$, $i = 1, \dots, N$. ▷ (Subsection 3.2)

Repeat:

- 6: **for** $i = 1$ to N **do**
- 7: **Reproduction:** Produce an individual y by the HR scheme. ▷
(Subsection 4.4)
- 585 8: **Update of z :** For each $j \in \{1, \dots, m\}$, if $f_j(y) < z_j$, set $z_j = f_j(y)$.

- 9: **Update of population:** the PSPU rule is used to update the population. ▷ (Subsection 4.3)
- 10: **Update of PV:** Update $\mathbf{P}^{(i,k)}$ by Eq. 5 and Eq. 6. ▷ (Subsection 3.2)
- 590 11: **Update of EP:** Remove those solutions dominated by y from EP and add y to EP if it is not dominated by anyone in EP. ▷ (Subsection 3.1)
- 12: **end for**
- Termination:**
- 13: If stopping criteria are satisfied, stop and output EP.

595 In Step 3, the PBI scheme is used to generate an initial population, where PV $\mathbf{P}^{(init)}$ is repeatedly sampled in order to guarantee that every individual in the population is feasible. This provides the proposed algorithm a set of promising and diversified individuals to begin with. PBIL reproduction method is integrated into the MOEA/D framework (see Section 4.4). So, in the proposed

600 algorithm, each SOSP is associated with a PV, e.g. $\mathbf{P}^{(i,k)} = \{P_1^k, \dots, P_L^k\}$. In Step 5, each PV is initialized as $\{0.5, 0.5, \dots, 0.5\}$, where value ‘0.5’ is the probability of generating ‘1’ at that position. In Step 7, a new individual is produced by the MOEA/D- or PBIL- reproduction. In Step 9, the PSPU rule first calculates Δ_{max} according to Eq. 7 and Eq. 8. For $j \in \varphi(i)$, if $\Delta_{SOSP(j)} =$

605 $\Delta_{max} > 0$, then set $x_j = y$ and $f_u(x_j) = f_u(y)$, $u = 1, \dots, m$, where $f_u(x)$ is the u -th objective value of individual x . No matter whether the PBIL reproduction method is chosen, $\mathbf{P}^{(i,k)}$ is consistently updated at each generation, where $i = 1, \dots, N$. Step 10 defines the above procedure. In Step 13, the termination condition is that the algorithm evolves a predefined number of generations.

610 The learning rate α defines how quickly PV learns from the best individual, and has a great impact on the convergence of PV [72]. In MOEA/D-PBIL, α is adaptively set during the evolutionary process. At the beginning of the evolution, as the quality of individuals is generally low, a small α is set so that PVs could learn from promising individuals; during the evolution, the value of α

615 is increased gradually until reaching to a maximal threshold (0.1 in this paper),

Table 1: Test Benchmark Networks and Their Parameters [40]

Networks	Parameters			
	nodes	links	receivers	rate
7-copy	57	84	8	2
15-copy	121	180	16	2
Rnd-1	20	37	5	3
Rnd-2	20	39	5	3
Rnd-3	30	60	6	3
Rnd-4	30	69	6	3
Rnd-5	40	78	9	3
Rnd-6	40	85	9	4
Rnd-7	50	101	8	3
Rnd-8	50	118	10	4

which to some extent helps to provide a fine local exploitation.

5. Performance evaluation

This section studies the effectiveness of the three performance-enhancing schemes, i.e. the PBI scheme, the PSPU rule and the HR scheme, respectively, on benchmark test instances using certain performance metrics. The overall performance of MOEA/D-PBIL is then evaluated, comparing against a number of state-of-the-art MOEAs, including NSGA-II [42] and SPEA2 [43].

5.1. Test instances

Ten widely used benchmark instances are considered in this paper, including two fixed networks (7-copy and 15-copy, see details in [18]) and eight randomly generated networks (Rnd-1 to Rnd-8, with network size from 20 to 50, see details in [40]). The associated parameters of the ten instances are given in Table 1.

In the paper, for an arbitrary $e \in E$, its link cost $c_{link}(e)$ and propagation delay is uniformly distributed in the range of $[5, 15]$ and $[2ms, 10ms]$, respectively. The coding cost C_{code} is the number of coding links in the obtained network coding based multicast subgraph $G_{NCM}(s, T)$. We assume any coding operation consumes the same amount of processing time, i.e. 1ms. To

encourage scientific comparisons, the details of all instances can be found at <http://www.cs.nott.ac.uk/~rxq/benchmarks.htm>. The predefined number of generations for all algorithms for comparison is set to 200. All experiments were run on a Windows 8 OS computer with Intel(R) Core(TM) i7-3740QM CPU 2.7 GHz and 8 GB RAM. The results are obtained by running each algorithm 20 times (unless stated otherwise), from which the statistics are collected and analyzed.

5.2. Performance measures

To thoroughly evaluate the performance of the proposed algorithm, we employ five widely recognized performance measuring metrics throughout the experiments.

Let PF_{ref} be a reference set of solutions well approximating the true PF, and PF_{known} be the set of nondominated solutions obtained by an algorithm. Note that we may not know the true PF for highly complex multi-objective optimization problems, including the problem concerned in this work, so we combine the best-so-far solutions obtained by all algorithms after all runs and select the nondominated solutions as the reference set. This method has been widely adopted when evaluating multi-objective algorithms in the literature.

- **Inverted generational distance (IGD)** [41]: IGD is defined as the average distance from each point v in PF_{ref} to its nearest counterpart in PF_{known} , as follows:

$$IGD = \frac{\sum_{v \in PF_{ref}} d(v, PF_{known})}{|PF_{ref}|} \quad (10)$$

where $d(v, PF_{known})$ is the Euclidean distance (in the objective domain) between solution v in PF_{ref} and its nearest solution in PF_{known} and $|PF_{ref}|$ is the number of solutions in PF_{ref} . IGD measures the convergence and diversity of an obtained nondominated solution set. This metric is commonly used to evaluate the overall performance of an algorithm. A lower IGD indicates a better overall performance of an algorithm.

- **Generational distance (GD)** [80]: GD measures the average distance from each point v in PF_{known} to its nearest counterpart in PF_{ref} , as defined below:

$$GD = \sqrt{\frac{\sum_{v \in PF_{known}} d(v, PF_{ref})}{|PF_{known}|}} \quad (11)$$

where $d(v, PF_{ref})$ is the Euclidean distance between v in PF_{known} and its nearest point in PF_{ref} . This metric is used to measure how closely PF_{known} converges to PF_{ref} . A smaller GD indicates the obtained PF is closer to the true PF.

660

- **Maximum spread (MS)** [80]: MS reflects how well the true PF is covered by the points in PF_{known} through the hyperboxes formed by the extreme function values observed in PF_{ref} and PF_{known} , as shown in Eq. 12.

$$MS = \sqrt{\frac{1}{m} \sum_{i=1}^m \left(\frac{\min(f_i^{\max}, F_i^{\max}) - \max(f_i^{\min}, F_i^{\min})}{F_i^{\max} - F_i^{\min}} \right)^2} \quad (12)$$

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.

665

- **Average Computational Time (ACT)**: ACT is the average running time consumed by an algorithm over 20 runs. This metric is a direct indicator of the computational complexity of an algorithm being tested.
- **Student's t -test** [79]: This test is to compare two algorithms 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 results indicate if the performance of one algorithm is significantly better than, significantly worse than, or statistically equivalent to that of another algorithm, respectively.

670

675 5.3. The effectiveness of the PBI scheme

In general, an initial population should contain a considerable amount of diverse and feasible individuals. A PBI scheme is proposed (see Subsection 4.2) to generate a set of feasible initial individuals, by repeatedly sampling an initial PV $\mathbf{P}^{(init)} = \{P_{init}, P_{init}, P_{init}\}$ until a predefined number of feasible individuals are created. The value of P_{init} is of vital importance to the performance of the PBI scheme. Three different settings are compared in the proposed scheme. Besides, the PBI scheme is also compared with two existing initialization schemes, i.e. Kim’s method [11] and Xing’s method [40], as listed below.

- 685 • **Kim’s method** [11]: the initial population is randomly generated. An all-one individual is included into the population to ensure the search start with a feasible search point. It is widely used in the network coding resource minimization problem.
- 690 • **Xing’s method** [40]: one-bit mutation is performed on the all-one individual and its variants to produce a set of feasible individuals that are very closely distributed around the all-one individual. This method has been adopted in MOP with network coding.
- 695 • **PBI**: the proposed initialization scheme. Three settings, i.e. $P_{init} = 0.7, 0.8$ and 0.9 , are tested. For simplicity purpose, we represent them as PBI(0.7), PBI(0.8) and PBI(0.9), respectively. A larger P_{init} leads to a higher probability of generating ‘1’ at the corresponding position. This is in compliance with the research findings in [40], that individuals closer to the all-one individual are more likely to be feasible.

As aforementioned, IGD reflects the overall performance of an algorithm regarding the quality of the obtained PF_{known} . Hence, IGD is also used to evaluate the initial population. We compare the three initialization methods using six 700 test instances, including 2 small instances (Rnd-1 and Rnd-2), 2 medium-sized instances (Rnd-5 and Rnd-7), and 2 large instances (7copy and Rnd-8). Figure

7 illustrates the comparisons among different initialization methods, where horizontal axis represents the IGD of each population and the vertical axis is the computational time consumed by each method.

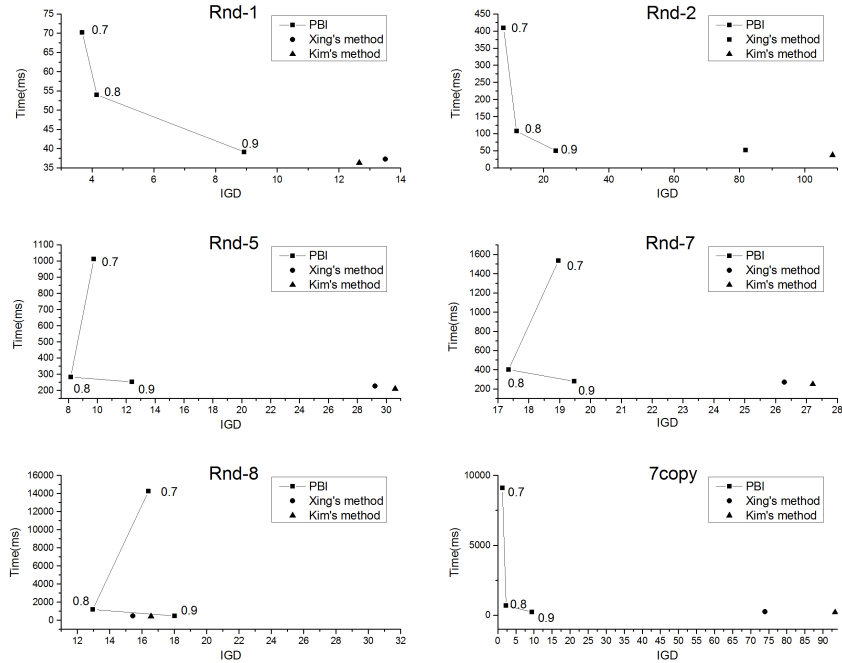


Figure 7: Comparisons among different initialization schemes w.r.t. IGD

It is clearly seen that the PBI scheme performs significantly better than the other two methods in terms of IGD. Kim's method provides an initial population with at least a feasible individual. However, infeasible individuals still account for the majority of the population [40], thus the individuals in the objective domain are far away from PF_{ref} . Xing's method produces a feasible population, however without a good diversification, i.e. all individuals are too close to the all-one individual. The proposed PBI scheme considers not only the feasibility but also the diversity of the population and thus obtains the best performance.

Table 2: Student’s t -test results of A1, A2 and A3

Network	A3 \leftrightarrow A1	A3 \leftrightarrow A2
Rnd-1	+	\sim
Rnd-2	+	\sim
Rnd-5	+	+
Rnd-7	+	+
Rnd-8	+	+
7copy	+	+

Kim’s and Xing’s methods are simple, thus both consume a smaller amount
715 of time in all instances. On the other hand, the computational cost of the PBI
scheme has a wide spread in different instances. The smallest time cost of PBI
is comparable to that of Kim’s and Xing’s methods.

With regard to different settings of P_{init} of the PBI scheme, it is easily seen
that PBI(0.7) and PBI(0.8) are better than PBI(0.9) in all instances. Sampling
720 a $\mathbf{P}^{(init)}$ with larger P_{init} , tends to produce more feasible individuals. There-
fore, to produce the same number of individuals, smaller P_{init} takes more time.
On the contrary, however, it is more likely to form a diversified population (ac-
cording to the studies in [72]), thus PBI(0.7) and PBI(0.8) have better IGD
values than PBI(0.9). Considering not only the quality of the population, but
725 more realistically, also the time efficiency, we hereafter set $P_{init} = 0.9$ in the
experiments.

To further evaluate the superiority of the PBI scheme, we run the MOEA/D-
PBIL with the three different initialization schemes, namely A1 with Kim’s
method, A2 with Xing’s method and A3 with PBI(0.9), on the above six selected
730 instances, results of Student’s t -test shown in Table 2.

In Table 2, symbols ‘+’, ‘-’, and ‘ \sim ’ in column A \leftrightarrow B indicate that algo-
rithm A is significantly better than, significantly worse than, and statistically
equivalent to algorithm B, respectively, in terms of the IGD. It is clear that A3
performs better than A2 and A1 in all instances. This also demonstrates that
735 by providing the algorithm with a diversified and feasible population, the PBI
scheme helps to guide the search towards the true PF.

Table 3: Results of mean(SD) w.r.t. IGD, GD and MS (the best results are in bold) from A3 and A4.

Network	IGD		GD		MS	
	A3	A4	A3	A4	A3	A4
Rnd-1	2.66(1.65)	1.85 (1.39)	0.67(0.84)	0.47 (0.74)	0.86 (0.11)	0.83(0.02)
Rnd-2	3.51(1.91)	1.96 (1.36)	0.16(0.49)	0.08 (0.35)	0.76(0.26)	0.89 (0.10)
Rnd-3	0.06(0.11)	0.02 (0.03)	0.04(0.10)	0.00 (0.00)	0.98(0.41)	0.99 (0.02)
Rnd-4	2.95(1.01)	2.79 (0.69)	1.76(1.23)	1.58 (0.66)	0.69(0.14)	0.78 (0.06)
Rnd-5	1.21(0.23)	0.90 (0.23)	0.73(0.20)	0.72 (0.25)	0.79(0.07)	0.87 (0.08)
Rnd-6	0.12(0.06)	0.11 (0.07)	0.15(0.25)	0.10 (0.21)	1.00 (0.00)	1.00 (0.00)
Rnd-7	1.43(0.30)	1.37 (0.13)	1.12(0.85)	0.62 (0.38)	0.90(0.03)	0.99 (0.01)
Rnd-8	0.89(0.57)	0.69 (0.45)	1.10(0.62)	0.79 (0.63)	0.88(0.05)	0.90 (0.05)
7copy	0.05(0.03)	0.03 (0.02)	0.17(0.07)	0.13 (0.02)	0.99 (0.01)	0.99 (0.01)
15copy	0.16(0.08)	0.12 (0.05)	0.34(0.17)	0.22 (0.04)	0.97 (0.01)	0.97 (0.01)

5.4. The effectiveness of the population updating rule

To evaluate the effectiveness of the PSPU rule (Subsection 4.3), we compare two MOEAs regarding the optimization results obtained, as listed below.

- 740 • **A3**: MOEA/D [41] with the PBI method, where the original population updating rule is utilized
- **A4**: A3 with the PSPU rule

The optimization results in terms of the IGD, GD and MS are reported in Table 3. Not surprisingly, A4 clearly outperforms A3 regarding all measures in almost all instances, indicating the effectiveness of the PSPU rule. The design of the proposed rule is in compliance with nature of the MOP being tackled, and to deal with the issue that the search space is dominated by infeasible individuals. Feasible individuals are difficult to generate during the evolution. So, when promising individuals appear, if no limitation is defined to update the population, their genes could be rapidly spread over the population within a few generations. The population diversity then would be lost quickly. The PSPU rule limits the number of SOSPs to be updated, to maintain a certain level of diversity while optimizing.

Table 4: Student’s *t*-test results of A3 and A4

Network	Rnd-1	Rnd-2	Rnd-3	Rnd-4	Rnd-5	Rnd-6	Rnd-7	Rnd-8	7copy	15copy
A4 ↔ A3	+	+	+	~	~	+	~	+	~	+

Student’s *t*-test is conducted to compare A3 and A4, regarding IGD, GD
755 and MS, and the results are shown in Table 4. A4 performs at least no worse,
often better than A3 (in more than half of the instances). This confirms the
contribution of the PSPU rule to the proposed algorithm.

5.5. The effectiveness of the HR scheme

The HR scheme described in Subsection 4.4 adaptively uses one of the two
760 reproduction methods, from MOEA/D and PBIL, to strike a balance between
global exploration and local exploitation during different stages of the evolution.
The effectiveness of the HR scheme is evaluated by comparing the following two
MOEAs.

- **A4:** Original MOEA/D with the PBI method and the PSPU rule
- 765 • **A5:** A4 with the HR scheme, i.e. the proposed MOEA/D-PBIL in this
paper

The results of IGD, GD, MS in Table 5, indicate that A5 outperforms A4 in
9 instances regarding IGD and GD and 6 instances in terms of MS, respectively.
This demonstrates that the HR scheme in A5 helps to provide a decent global
770 exploration and local exploitation during the evolution. On the other hand, the
global exploration ability gradually decreases in the original MOEA/D which
only utilizes genetic operators for offspring reproduction, thus results into local
optima solutions.

Results of *t*-test are in Table 6. A5 performs outstandingly better than A4,
775 indicating the HR scheme greatly improves the performance of the proposed
MOEA/D-PBIL.

Table 5: Results of mean(SD) (the best are in bold) from A4 and A5

Network	IGD		GD		MS	
	A4	A5	A4	A5	A4	A5
Rnd-1	1.85(1.39)	0.00 (0.00)	0.47(0.74)	0.00 (0.00)	0.83(0.02)	1.00 (0.00)
Rnd-2	1.96(1.36)	0.00 (0.00)	0.08(0.35)	0.00 (0.00)	0.89(0.10)	1.00 (0.00)
Rnd-3	0.02(0.03)	0.00 (0.00)	0.00(0.00)	0.00 (0.00)	0.99(0.02)	1.00 (0.00)
Rnd-4	2.79(0.69)	0.85 (0.26)	1.58(0.66)	1.17 (0.48)	0.78(0.06)	0.81 (0.08)
Rnd-5	0.90 (0.23)	1.05(0.55)	0.72(0.25)	0.54 (0.22)	0.87 (0.08)	0.82(0.01)
Rnd-6	0.11(0.07)	0.00 (0.00)	0.10(0.21)	0.00 (0.00)	1.00 (0.00)	1.00 (0.00)
Rnd-7	1.37(0.13)	1.14 (0.40)	0.62(0.38)	0.42 (0.37)	0.99 (0.01)	0.99 (0.01)
Rnd-8	0.69(0.45)	0.60 (0.43)	0.79(0.63)	0.63 (0.52)	0.90(0.05)	0.91 (0.04)
7copy	0.03(0.02)	0.02 (0.01)	0.13(0.02)	0.11 (0.01)	0.99 (0.01)	0.99 (0.01)
15copy	0.12(0.05)	0.09 (0.02)	0.22 (0.04)	0.22 (0.04)	0.97(0.01)	0.98 (0.01)

Table 6: Student's *t*-test results of A4 and A5

Network	Rnd-1	Rnd-2	Rnd-3	Rnd-4	Rnd-5	Rnd-6	Rnd-7	Rnd-8	7copy	15copy
A5 ↔ A4	+	+	+	+	~	+	~	~	+	~

5.6. The overall performance evaluation

The proposed MOEA/D-PBIL algorithm is finally thoroughly investigated through performance comparisons against the following seven state-of-the-art MOEAs in the literature.

- **NSGA-II**: As one of the classical MOEAs originally proposed by Deb et al [42], NSGA-II is featured with three significant features, a fast non-dominated sorting approach, an elitism approach and a parameter-free diversity preservation scheme. We set the population size $N = 100$, the crossover rate $p_c = 0.9$ and the mutation rate $p_m = 1/L$, where L is the individual length.
- **NSGA-II-Xing**: With two improvements, i.e. Xing's initialization method (in Subsection 5.3) and an individual delegation scheme in favor of diversification, NSGA-II-Xing is able to gain promising optimization performance for the bi-objective MOP with network coding in our previous work [40].

- **SPEA2**: The strength Pareto evolutionary algorithm 2 [43] is another widely recognized MOEA. We denote the archive size by N_{arc} and set $N_{arc} = N = 100$, $p_c = 0.9$ and $p_m = 1/L$, respectively.
- 795 • **MOPSO**: The multiobjective algorithm based on particle swarm intelligence [81] is well-known and widely used for performance comparison. A population of 100 particles is maintained. We set $p_m = 1/L$ and 30 divisions for the adaptive grid.
- 800 • **MOPBIL1**: the multiobjective PBIL proposed by Kim et al. has been reported to outperform a number of GA-based MOEAs when solving MOPs in the context of the robot soccer system [82]. When updating the i -th PV, a solution randomly selected from the archive is used. Let the number of PVs, the learning rate, and the amount of shift in the mutation be denoted by n_{PV} , α , and σ , respectively. We set $N = 100$, $N_{arc} = 50$, $n_{PV} = 100$, $\alpha = 0.15$, $p_m = 0.02$, and $\sigma = 0.2$.
- 805 • **MOPBIL2**: the first multiobjective PBIL presented by Bureerat and Sriworamas [83]. The i -th PV is updated by 5 solutions randomly selected from the archive. We set $N = 100$, $N_{arc} = 50$, $n_{PV} = 100$, $\alpha = 0.15$, $p_m = 0.02$, and $\sigma = 0.2$.
- 810 • **MOEA/D**: The original MOEA/D proposed by Zhang and Li (see Subsection 3.1 for details) [41]. We set $N = 100$, $p_c = 0.9$ and $p_m = 1/L$, respectively.
- **MOEA/D-PBIL**: The improved MOEA/D proposed in this paper. We set $N = 100$, $P_{init} = 0.9$, $r = 11$, $p_c = 0.9$ and $p_m = 1/L$, respectively.

Results of IGD, GD, MS are collected in Tables 7, 8, and 9, respectively. In terms of IGD and GD, MOEA/D-PBIL performs the best, obtaining the minimum IGD values in 8 instances (except Rnd-4 and Rnd-7 in Table 7) and the minimum GD values in 9 instances (except Rnd-4 in Table 8). This indicates that the nondominated solutions of MOEA/D-PBIL are well diversified

Table 7: Results of IGD (Best results are in bold) from the eight algorithms under comparison

Network	Rnd-1	Rnd-2	Rnd-3	Rnd-4	Rnd-5	Rnd-6	Rnd-7	Rnd-8	7copy	15copy
NSGA-II	0.00	0.00	0.00	2.10	1.19	0.01	1.27	1.07	0.03	2.70
	(0.00)	(0.00)	(0.00)	(1.06)	(0.30)	(0.01)	(0.29)	(0.18)	(0.01)	(0.60)
NSGA-II-Xing	0.00	0.00	0.00	0.85	1.14	0.00	1.21	1.09	0.03	2.56
	(0.00)	(0.00)	(0.00)	(0.26)	(0.22)	(0.00)	(0.39)	(0.20)	(0.01)	(0.53)
SPEA2	5.27	5.68	0.92	3.97	1.73	2.07	3.57	1.46	0.08	2.20
	(0.54)	(0.86)	(0.24)	(3.39)	(0.49)	(1.13)	(1.49)	(0.36)	(0.03)	(0.61)
MOPSO	3.08	3.51	0.82	2.20	1.97	1.77	2.65	3.78	0.54	7.30
	(0.54)	(1.87)	(0.36)	(0.39)	(0.21)	(0.34)	(0.44)	(0.33)	(0.08)	(1.39)
MOPBIL1	0.00	0.44	0.00	0.69	1.15	1.08	1.11	1.07	0.05	3.15
	(0.00)	(0.17)	(0.00)	(0.59)	(0.71)	(0.48)	(0.63)	(0.58)	(0.02)	(2.63)
MOPBIL2	0.42	0.63	0.00	0.78	1.24	0.82	1.54	1.07	0.06	3.02
	(0.31)	(0.23)	(0.00)	(0.65)	(0.67)	(0.26)	(0.81)	(0.59)	(0.01)	(1.87)
MOEA/D	3.26	6.94	0.24	4.64	3.71	2.52	3.78	5.23	0.50	2.86
	(0.34)	(0.78)	(0.09)	(0.44)	(0.43)	(0.59)	(0.76)	(0.23)	(0.41)	(2.53)
MOEA/D-PBIL	0.00	0.00	0.00	0.83	1.05	0.00	1.14	0.60	0.02	0.09
	(0.00)	(0.00)	(0.00)	(0.41)	(0.55)	(0.00)	(0.40)	(0.43)	(0.01)	(0.02)

and converged closer to the true PF, compared with those of other algorithms.

820 MOEA/D-PBIL also achieves the best coverage of the true PF, and obtains the highest MS in 8 instances (except Rnd-4 and Rnd-5 in Table 9). According to these three performance indicators, MOEA/D-PBIL gains the best performance in all instances due to the PBI scheme, PSPU rule, and HR scheme, which leads to a balanced trade-off between the global exploration and local exploitation, 825 achieving better diversity and convergence at the same time.

Student’s t -test between the 8 algorithms in Table 10 indicates that MOEA/D-PBIL is the best algorithm among all algorithms, performing no worse, and usually better than the others in most of the instances.

The results of ACT in Table 11 show that compared with the others, the original MOEA/D and MOEA/D-PBIL achieve the smallest ACTs in all instances. 830 This is because the computational complexity of addressing multiple SOSPs simultaneously is usually much lower than solving a MOP directly [41]. The original MOEA/D is faster than MOEA/D-PBIL in most of the instances, since the latter with a number of complicated mechanisms requires relatively longer

Table 8: Results of GD (Best results are in bold) from the eight algorithms under comparison

Network	Rnd-1	Rnd-2	Rnd-3	Rnd-4	Rnd-5	Rnd-6	Rnd-7	Rnd-8	7copy	15copy
NSGA-II	0.00	0.00	0.00	0.87	0.81	0.01	1.59	1.49	0.27	2.02
	(0.00)	(0.00)	(0.00)	(0.98)	(0.16)	(0.01)	(0.82)	(0.19)	(0.07)	(0.35)
NSGA-II-Xing	0.00	0.00	0.00	0.81	0.88	0.00	1.36	1.29	0.21	1.82
	(0.00)	(0.00)	(0.00)	(0.79)	(0.19)	(0.00)	(0.89)	(0.25)	(0.07)	(0.27)
SPEA2	0.31	0.40	0.10	2.75	0.94	1.53	0.80	0.41	0.37	2.02
	(0.53)	(0.68)	(0.14)	(2.54)	(0.23)	(1.04)	(0.99)	(0.27)	(0.14)	(0.37)
MOPSO	0.56	0.71	0.19	2.93	1.54	1.72	1.96	2.56	1.24	3.26
	(0.62)	(0.79)	(0.13)	(1.00)	(0.35)	(0.22)	(0.63)	(0.53)	(0.16)	(0.22)
MOPBIL1	0.00	0.02	0.00	0.35	0.82	0.84	2.20	1.05	0.30	1.92
	(0.00)	(0.02)	(0.00)	(0.33)	(0.39)	(0.57)	(1.52)	(0.75)	(0.24)	(1.02)
MOPBIL2	0.30	0.05	0.00	0.60	0.87	0.75	2.05	1.03	0.27	1.88
	(0.37)	(0.02)	(0.00)	(0.53)	(0.41)	(0.32)	(1.79)	(0.86)	(0.19)	(1.15)
MOEA/D	2.33	2.13	1.62	3.85	1.30	1.84	3.07	2.30	2.22	3.01
	(0.42)	(1.11)	(0.37)	(0.22)	(0.18)	(0.71)	(0.34)	(0.18)	(0.20)	(0.14)
MOEA/D-PBIL	0.00	0.00	0.00	1.17	0.54	0.00	0.42	0.63	0.11	0.22
	(0.00)	(0.00)	(0.00)	(0.48)	(0.22)	(0.00)	(0.37)	(0.52)	(0.01)	(0.04)

Table 9: Result of MS (Best results are in bold) from the eight algorithms under comparison

Network	Rnd-1	Rnd-2	Rnd-3	Rnd-4	Rnd-5	Rnd-6	Rnd-7	Rnd-8	7copy	15copy
NSGA-II	1.00	1.00	1.00	0.81	0.85	0.98	0.90	0.63	0.99	0.54
	(0.00)	(0.00)	(0.00)	(0.08)	(0.07)	(0.01)	(0.04)	(0.02)	(0.01)	(0.04)
NSGA-II-Xing	1.00	1.00	1.00	0.92	0.90	1.00	0.91	0.86	0.99	0.55
	(0.00)	(0.00)	(0.00)	(0.04)	(0.04)	(0.00)	(0.03)	(0.02)	(0.01)	(0.03)
SPEA2	0.99	0.99	0.99	0.86	0.87	0.77	0.90	0.84	0.99	0.57
	(0.04)	(0.04)	(0.01)	(0.06)	(0.07)	(0.09)	(0.02)	(0.03)	(0.01)	(0.06)
MOPSO	0.69	0.74	0.82	0.63	0.53	0.71	0.68	0.63	0.47	0.71
	(0.13)	(0.03)	(0.08)	(0.01)	(0.06)	(0.03)	(0.03)	(0.02)	(0.04)	(0.03)
MOPBIL1	1.00	0.97	1.00	0.98	0.89	0.81	0.94	0.85	0.98	0.50
	(0.00)	(0.01)	(0.00)	(0.02)	(0.06)	(0.09)	(0.03)	(0.08)	(0.02)	(0.13)
MOPBIL2	1.00	0.95	1.00	0.96	0.87	0.84	0.93	0.85	0.98	0.51
	(0.00)	(0.02)	(0.00)	(0.03)	(0.08)	(0.12)	(0.03)	(0.09)	(0.01)	(0.11)
MOEA/D	0.67	0.72	0.80	0.63	0.52	0.71	0.51	0.63	0.47	0.69
	(0.11)	(0.03)	(0.07)	(0.01)	(0.05)	(0.04)	(0.01)	(0.02)	(0.04)	(0.04)
MOEA/D-PBIL	1.00	1.00	1.00	0.96	0.82	1.00	0.99	0.91	0.99	0.98
	(0.00)	(0.00)	(0.00)	(0.03)	(0.10)	(0.00)	(0.01)	(0.04)	(0.01)	(0.01)

Table 10: Student’s *t*-test results of the eight algorithms under comparison

Network	Rnd-1	Rnd-2	Rnd-3	Rnd-4	Rnd-5
MOEA/D-PBIL \leftrightarrow NSGA-II	~	~	~	+	+
MOEA/D-PBIL \leftrightarrow NSGA-II-Xing	~	~	~	+	+
MOEA/D-PBIL \leftrightarrow SPEA2	+	+	+	+	+
MOEA/D-PBIL \leftrightarrow MOPSO	+	+	+	+	+
MOEA/D-PBIL \leftrightarrow MOPBIL1	~	+	~	~	+
MOEA/D-PBIL \leftrightarrow MOPBIL2	+	+	~	~	+
MOEA/D-PBIL \leftrightarrow MOEA/D	+	+	+	+	+
	Rnd-6	Rnd-7	Rnd-8	7copy	15copy
MOEA/D-PBIL \leftrightarrow NSGA-II	~	+	+	+	+
MOEA/D-PBIL \leftrightarrow NSGA-II-Xing	~	+	+	~	+
MOEA/D-PBIL \leftrightarrow SPEA2	+	+	+	+	+
MOEA/D-PBIL \leftrightarrow MOPSO	+	+	+	+	+
MOEA/D-PBIL \leftrightarrow MOPBIL1	+	~	+	+	+
MOEA/D-PBIL \leftrightarrow MOPBIL2	+	+	+	+	+
MOEA/D-PBIL \leftrightarrow MOEA/D	+	+	+	+	+

835 time to improve the optimization performance. MOEA/D-PBIL compared with other MOEAs (except the original MOEA/D) demonstrates its efficiency.

6. Conclusions

This paper formulated a multi-objective optimization problem in the context of multicasting with network coding, where the three objectives, namely the coding cost, link cost and the end-to-end delay are minimized simultaneously. By analyzing the property of the search space, population-based incremental learning (PBIL) components were incorporated into the evolutionary framework, and a modified multi-objective evolutionary algorithm based on decomposition (MOEA/D-PBIL) was proposed. Three performance-enhancing schemes were developed, namely, the probability-based initialization scheme, the hybridized reproduction operator and the problem-specific population updating rule. The first scheme is able to offer a feasible and diversified initial population; the second one is in favor of avoiding the prematurity effect; and the last one helps to balance the global exploration and local exploitation during the search. The ex-

845

Table 11: Result of ACT (*Sec.*) (Best results are in bold) from the eight algorithms under comparison

Network	Rnd-1	Rnd-2	Rnd-3	Rnd-4	Rnd-5
NSGA-II	15.51	23.83	60.52	60.93	92.33
NSGA-II-Xing	15.08	23.53	58.85	60.95	90.50
SPEA2	9.66	17.73	34.01	48.87	77.55
MOPSO	28.38	25.94	27.38	178.60	228.54
MOPBIL1	9.21	17.22	36.71	62.72	80.05
MOPBIL2	9.36	18.90	36.86	64.37	86.03
MOEA/D	5.02	6.88	17.37	16.59	31.05
MOEA/D-PBIL	5.64	7.06	17.15	34.23	34.34
	Rnd-6	Rnd-7	Rnd-8	7copy	15copy
NSGA-II	67.55	195.37	265.49	95.57	909.79
NSGA-II-Xing	67.37	195.24	276.85	94.22	884.97
SPEA2	44.33	223.00	239.36	87.50	938.81
MOPSO	26.84	501.15	782.50	182.00	1603.45
MOPBIL1	45.95	148.50	289.39	91.11	869.67
MOPBIL2	48.61	146.36	298.66	93.80	846.94
MOEA/D	18.03	37.31	66.59	29.79	210.79
MOEA/D-PBIL	17.93	41.35	89.05	40.12	711.45

850 perimental results demonstrated that with the three new schemes, the proposed
MOEA/D-PBIL algorithm gained the best optimization performance in terms of
performance indicators, namely the inverted generational distance, generational
distance, maximum spread, t -test results, compared with seven state-of-the-art
MOEAs in the literature on the problem under consideration.

855 **References**

- [1] L. Harte, Introduction to data multicasting, Althos Publishing, 2008.
- [2] A. Benslimane, Multimedia multicast on the internet, ISTE, 2013.
- [3] R. Ahlswede, N. Cai, S.-Y. Li, R. W. Yeung, Network information flow,
IEEE Transactions on Information Theory 46 (4) (2000) 1204 – 1216.
- 860 [4] S.-Y. Li, R. W. Yeung, N. Cai, Linear network coding, IEEE Transactions
on Information Theory 49 (2) (2003) 371–381.
- [5] N. Cai, R. W. Yeung, Secure network coding, in: Proceedings of IEEE
International Symposium on Information Theory (ISIT), IEEE, 2002, p.
323.
- 865 [6] C. Fragouli, J.-Y. Le Boudec, J. Widmer, Network coding: an instant
primer, ACM SIGCOMM Computer Communication Review 36 (1) (2006)
63.
- [7] A. E. Kamal, 1 + n network protection for mesh networks: Network coding-
based protection using p-cycles, IEEE/ACM Transactions on Networking
870 18 (1) (2010) 67–80.
- [8] S. E. Fragouli, Christina, Information flow decomposition for network cod-
ing, IEEE Transactions on Information Theory 52 (3) (2006) 829.
- [9] M. Langberg, A. Sprintson, J. Bruck, The encoding complexity of network
coding, IEEE/ACM Transactions on Networking (TON) 14 (SI) (2006)
875 2386–2397.

- [10] M. Kim, C. W. Ahn, M. Mdard, M. Effros, On minimizing network coding resources: An evolutionary approach, in: *Second Workshop on Network Coding, Theory, and Applications (NetCod2006)*, Citeseer, Boston, 2006.
- [11] M. Kim, M. Médard, V. Aggarwal, U.-M. O'Reilly, W. Kim, C. W. Ahn, M. Effros, Evolutionary approaches to minimizing network coding resources, in: *IEEE International Conference on Computer Communications (INFOCOM)*, IEEE, 2007, pp. 1991–1999.
- [12] M. Kim, V. Aggarwal, U.-M. O'Reilly, M. Médard, W. Kim, Genetic representations for evolutionary minimization of network coding resources, in: *Applications of Evolutionary Computing*, Springer, 2007, pp. 21–31.
- [13] H. Xing, Y. Ji, L. Bai, Y. 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) (2010) 1105–1113.
- [14] H. Xing, R. Qu, A population based incremental learning for network coding resources minimization, *IEEE Communications Letters* 15 (7) (2011) 698–700.
- [15] H. Xing, R. Qu, A compact genetic algorithm for the network coding based resource minimization problem, *Applied Intelligence* 36 (4) (2012) 809–823.
- [16] C. W. Ahn, Fast and adaptive evolutionary algorithm for minimum-cost multicast with network coding, *Electronics Letters* 47 (12) (2011) 700.
- [17] H. N. Luong, H. T. T. Nguyen, C. W. Ahn, Entropy-based efficiency enhancement techniques for evolutionary algorithms, *Information Sciences* 188 (2012) 100–120.
- [18] H. Xing, R. Qu, G. Kendall, R. Bai, A path-oriented encoding evolutionary algorithm for network coding resource minimization, *Journal of the Operational Research Society* 65 (8) (2013) 1261–1277.

- [19] H. Xing, R. Qu, L. Bai, Y. Ji, On minimizing coding operations in network coding based multicast: an evolutionary algorithm, *Applied Intelligence* 41 (3) (2014) 820–836.
- 905
- [20] D. S. Lun, M. Médard, T. Ho, R. Koetter, Network coding with a cost criterion, in: *Proceedings of International Symposium on Information Theory and its Applications (ISIT)*, Citeseer, 2004, pp. 1232–1237.
- [21] D. S. Lun, N. Ratnakar, R. Koetter, M. Médard, E. Ahmed, H. Lee, Achieving minimum-cost multicast: A decentralized approach based on network coding, in: *Proceedings of International Symposium on Information Theory and its Applications (ISIT)*, Vol. 3, IEEE, 2005, pp. 1607–1617.
- 910
- [22] D. S. Lun, N. Ratnakar, M. Médard, R. Koetter, D. R. Karger, T. Ho, E. Ahmed, F. Zhao, Minimum-cost multicast over coded packet networks, *IEEE Transactions on Information Theory* 52 (6) (2006) 2608–2623.
- 915
- [23] T. Cui, T. Ho, Minimum cost integral network coding, in: *Proceedings of International Symposium on Information Theory and its Applications (ISIT)*, IEEE, 2007, pp. 2736–2740.
- [24] F. Zhao, M. Médard, D. Lun, A. Ozdaglar, Minimum-cost subgraph algorithms for static and dynamic multicasts with network coding, *New Directions in Wireless Communications Research* (2009) 317–349.
- 920
- [25] A. Striegel, G. Manimaran, A survey of qos multicasting issues, *IEEE Communications Magazine* 40 (6) (2002) 82–87.
- [26] M. Alvandi, M. Mehmet-Ali, J. Hayes, Delay optimization in multi-hop wireless networks with network coding, in: *Wireless Communications and Networking Conference (WCNC)*, IEEE, 2013, pp. 1381–1386.
- 925
- [27] H. Li, X. Liu, W. He, J. Li, W. Dou, End-to-end delay analysis in wireless network coding: A network calculus-based approach, in: *31st International Conference on Distributed Computing Systems (ICDCS)*, IEEE, 2011, pp. 47–56.
- 930

- [28] N. Cleju, N. Thomos, P. Frossard, Network coding node placement for delay minimization in streaming overlays, in: *IEEE International Conference on Communications (ICC)*, IEEE, 2010, pp. 1–5.
- [29] P. Sadeghi, R. Shams, D. Traskov, An optimal adaptive network coding scheme for minimizing decoding delay in broadcast erasure channels, *EURASIP Journal on Wireless Communications and Networking* 2010 (2010) 4.
- [30] E. Drinea, L. Keller, C. Fragouli, Real-time delay with network coding and feedback, *Physical Communication* 6 (2013) 100–113.
- [31] S. Sorour, S. Valaee, Completion delay minimization for instantly decodable network coding with limited feedback, in: *IEEE International Conference on Communications (ICC)*, IEEE, 2011, pp. 1–5.
- [32] A. Douik, S. Sorour, M.-S. Alouini, T. Y. Al-Naffouri, Delay reduction in lossy intermittent feedback for generalized instantly decodable network coding, in: *International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, IEEE, 2013, pp. 388–393.
- [33] S. Sorour, S. Valaee, On minimizing broadcast completion delay for instantly decodable network coding, in: *IEEE International Conference on Communications (ICC)*, IEEE, 2010, pp. 1–5.
- [34] W.-L. Yeow, A. T. Hoang, C.-K. Tham, Minimizing delay for multicast-streaming in wireless networks with network coding, in: *IEEE International Conference on Computer Communications (INFOCOM)*, IEEE, 2009, pp. 190–198.
- [35] H. Zhang, J. Zhou, J. Li, Minimising delay for video conference with network coding, *International Journal of Information and Coding Theory* 2 (1) (2011) 38–58.

- [36] M. Kim, M. Médard, V. Aggarwal, U.-M. O'Reilly, On the coding-link cost tradeoff in multicast network coding, in: Military Communications Conference, IEEE, 2007, pp. 1–7.
- 960 [37] M. Kim, M. Médard, U.-M. O'Reilly, Network coding and its implications on optical networking, in: Optical Fiber Communication Conference, Optical Society of America, 2009, pp. 1–3.
- [38] C. W. A. Yoo, Multi-objective evolutionary approach to coding-link cost trade-offs in network coding, *Electronics Letters* 48 (25) (2012) 1.
- 965 [39] K. Hao, B. Wang, Y. Luo, Multi-objective network coding optimization based on nsga-ii algorithm, in: International Conference on Control Engineering and Communication Technology, IEEE Computer Society, 2012, pp. 843–846.
- [40] H. Xing, R. Qu, A nondominated sorting genetic algorithm for bi-objective network coding based multicast routing problems, *Information Sciences* 233 (2013) 36–53.
- 970 [41] Q. Zhang, H. Li, Moea/d: A multiobjective evolutionary algorithm based on decomposition, *IEEE Transactions on Evolutionary Computation* 11 (6) (2007) 712–731.
- 975 [42] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multi-objective genetic algorithm: Nsga-ii, *IEEE Transactions on Evolutionary Computation* 6 (2) (2002) 182–197.
- [43] E. Zitzler, M. Laumanns, L. Thiele, Spea2: Improving the strength pareto evolutionary algorithm, *Evolutionary Methods for Design, Optimization, and Control* (2002) 95 – 100.
- 980 [44] M. Pelikan, D. E. Goldberg, F. G. Lobo, A survey of optimization by building and using probabilistic models, *Computational Optimization and Applications* 21 (1) (2002) 5 – 20.

- [45] Y. Li, A. Zhou, G. Zhang, A decomposition based estimation of distribution
985 algorithm for multiobjective knapsack problems, in: Eighth International
Conference on Natural Computation (ICNC), IEEE, 2012, pp. 803–807.
- [46] V. A. Shim, K. Tan, K. K. Tan, A hybrid estimation of distribution algo-
rithm for solving the multi-objective multiple traveling salesman problem,
in: IEEE Congress on Evolutionary Computation (CEC), IEEE, 2012, pp.
990 1–8.
- [47] F. Gao, A. Zhou, G. Zhang, An estimation of distribution algorithm based
on decomposition for the multiobjective tsp, in: Eighth International Con-
ference on Natural Computation (ICNC), IEEE, 2012, pp. 817–821.
- [48] V. A. Shim, K. Tan, K. K. Tan, A hybrid adaptive evolutionary algorithm
995 in the domination-based and decomposition-based frameworks of multi-
objective optimization, in: IEEE Congress on Evolutionary Computation
(CEC), IEEE, 2012, pp. 1–8.
- [49] I. Giagkiozis, R. C. Purshouse, P. J. Fleming, Generalized decomposition
and cross entropy methods for many-objective optimization, *Information*
1000 *Sciences* 282 (2014) 363 – 387.
- [50] T. Ray, M. Asafuddoula, A. Isaacs, A steady state decomposition based
quantum genetic algorithm for many objective optimization, in: IEEE
Congress on Evolutionary Computation (CEC), IEEE, 2013, pp. 2817–
2824.
- 1005 [51] H. Li, Z. Qingfu, Multiobjective optimization problems with complicat-
ed pareto sets, moea/d and nsga-ii, *IEEE Transactions on Evolutionary*
Computation 13 (2) (2009) 284 – 302.
- [52] B. Liu, F. V. Fernández, Q. Zhang, M. Pak, S. Sipahi, G. Gielen, An
enhanced moea/d-de and its application to multiobjective analog cell sizing,
1010 in: IEEE Congress on Evolutionary Computation (CEC), IEEE, 2010, pp.
1–7.

- [53] Y.-Y. Tan, Y.-C. Jiao, H. Li, X.-K. Wang, A modification to moea/d-de for multiobjective optimization problems with complicated pareto sets, *Information Sciences* 213 (2012) 14–38.
- 1015 [54] R. Carvalho, R. R. Saldanha, B. N. Gomes, A. C. Lisboa, A. X. Martins, A multi-objective evolutionary algorithm based on decomposition for optimal design of yagi-uda antennas, *IEEE Transactions on Magnetics* 48 (2) (2012) 803 – 806.
- [55] S. M. S. Venske, R. A. Goncalves, M. R. Delgado, Ademo/d: adaptive
1020 differential evolution for multiobjective problems, in: *Brazilian Symposium on Neural Networks (SBRN)*, IEEE, 2012, pp. 226–231.
- [56] S. M. Venske, R. A. Gon Alves, M. R. Delgado, Ademo/d: Multiobjective optimization by an adaptive differential evolution algorithm, *Neurocomputing* 127 (2014) 65 – 77.
- 1025 [57] C.-M. Chen, Y.-p. Chen, Q. Zhang, Enhancing moea/d with guided mutation and priority update for multi-objective optimization, in: *IEEE Congress on Evolutionary Computation (CEC)*, IEEE, 2009, pp. 209–216.
- [58] Y. Mei, K. Tang, X. Yao, Decomposition-based memetic algorithm for multiobjective capacitated arc routing problem, *IEEE Transactions on Evolutionary Computation* 15 (2) (2011) 151–165.
1030
- [59] R. Shang, J. Wang, L. Jiao, Y. Wang, An improved decomposition-based memetic algorithm for multi-objective capacitated arc routing problem, *Applied Soft Computing Journal* 19 (2014) 343.
- [60] S. Zapotecas Martinez, C. A. Coello Coello, A direct local search mechanism for decomposition-based multi-objective evolutionary algorithms, in:
1035 *IEEE Congress on Evolutionary Computation (CEC)*, IEEE, 2012, pp. 1–8.
- [61] A. Alhindi, Q. Zhang, Moea/d with guided local search: Some preliminary experimental results, in: *Computer Science and Electronic Engineering Conference (CEEC)*, IEEE, 2013, pp. 109–114.

- 1040 [62] W. K. Mashwani, A. Salhi, Multiobjective memetic algorithm based on decomposition, *Applied Soft Computing* 21 (2014) 221–243.
- [63] L. Ke, Q. Zhang, R. Battiti, Hybridization of decomposition and local search for multiobjective optimization., *IEEE transactions on cybernetics* 44 (10) (2014) 1808.
- 1045 [64] X. Ma, F. Liu, Y. Qi, L. Li, L. Jiao, M. Liu, J. Wu, Moea/d with baldwinian learning inspired by the regularity property of continuous multiobjective problem, *Neurocomputing* 145 (2014) 336–352.
- [65] H. Li, D. Landa-Silva, X. Gandibleux, Evolutionary multi-objective optimization algorithms with probabilistic representation based on pheromone trails, in: *IEEE Congress on Evolutionary Computation (CEC)*, IEEE, 2010, pp. 1–8.
- 1050 [66] L. Ke, Q. Zhang, R. Battiti, Moea/d-aco: A multiobjective evolutionary algorithm using decomposition and antcolony, *IEEE Transactions on Cybernetics* 43 (6) (2013) 1845–1859.
- 1055 [67] J. Cheng, G. Zhang, Z. Li, Y. Li, Multi-objective ant colony optimization based on decomposition for bi-objective traveling salesman problems, *Soft Computing* 16 (4) (2012) 597 – 614.
- [68] W. Peng, Q. Zhang, A decomposition-based multi-objective particle swarm optimization algorithm for continuous optimization problems, in: *IEEE International Conference on Granular Computing (GrC)*, IEEE, 2008, pp. 534–537.
- 1060 [69] N. Al Moubayed, A. Petrovski, J. McCall, A novel smart multi-objective particle swarm optimisation using decomposition, in: *Parallel Problem Solving from Nature, PPSN XI*, Springer, 2010, pp. 1–10.
- 1065 [70] N. Al Moubayed, A. Petrovski, J. McCall, D 2 mopso: multi-objective particle swarm optimizer based on decomposition and dominance, in: *Evo-*

lutionary Computation in Combinatorial Optimization, Springer, 2012, pp. 75–86.

- [71] H. Li, D. Landa-Silva, An adaptive evolutionary multi-objective approach
1070 based on simulated annealing, *Evolutionary Computation* 19 (4) (2011)
561–95.
- [72] S. Baluja, Population-based incremental learning. a method for integrat-
ing genetic search based function optimization and competitive learning,
Tech. rep., Technical Report CMU-CS-94-163, Carnegie Mellon University
1075 (1994).
- [73] M. A. Medina, S. Das, C. Coello Coello, J. M. Ramirez, Two
decomposition-based modern metaheuristic algorithms for multi-objective
optimization: a comparative study, in: *IEEE Symposium on Computational
Intelligence in Multi-Criteria Decision-Making (MCDM)*, IEEE, 2013, pp.
1080 9–16.
- [74] M. Nasir, A. K. Mondal, S. Sengupta, S. Das, A. Abraham, An im-
proved multiobjective evolutionary algorithm based on decomposition with
fuzzy dominance, in: *IEEE Congress on Evolutionary Computation (CEC)*,
IEEE, 2011, pp. 765–772.
- 1085 [75] A. Zhou, Q. Zhang, G. Zhang, A multiobjective evolutionary algorithm
based on decomposition and probability model, in: *IEEE Congress on Evo-
lutionary Computation (CEC)*, IEEE, 2012, pp. 1–8.
- [76] X. Ma, F. Liu, Y. Qi, M. Gong, M. Yin, L. . J. L. Li, Moea/d with
opposition-based learning for multiobjective optimization problem, *Neu-
rocomputing* 146 (2014) 48 – 64.
1090
- [77] M. A. Medina, J. M. Ramirez, C. A. C. Coello, A novel multi-objective op-
timizer for handling reactive power, in: *IEEE Grenoble PowerTech (POW-
ERTECH)*, IEEE, 2013, pp. 1–6.

- [78] A. V. Goldberg, A new max-flow algorithm, Laboratory for Computer Science, Massachusetts Institute of Technology, 1985.
- 1095
- [79] R. E. Walpole, R. H. Myers, S. L. Myers, K. Ye, Probability and statistics for engineers and scientists, Pearson Education, 2007.
- [80] K. C. Tan, Y. J. Yang, C. K. Goh, A distributed cooperative coevolutionary algorithm for multiobjective optimization, IEEE Transactions on Evolutionary Computation 10 (5) (2006) 527 – 549.
- 1100
- [81] C. A. Coello Coello, Lechuga, Mopso: a proposal for multiple objective particle swarm optimization, in: IEEE Congress on Evolutionary Computation (CEC), Vol. 2, IEEE, 2002, pp. 1051 – 1056.
- [82] J.-H. Kim, Y.-H. Kim, S.-H. Choi, I.-W. Park, Evolutionary multi-objective optimization in robot soccer system for education, IEEE Computational Intelligence Magazine 4 (1) (2009) 31 – 41.
- 1105
- [83] S. Bureerat, K. Sriwornamas, Population-based incremental learning for multiobjective optimisation, in: Soft Computing in Industrial Applications, Springer Berlin Heidelberg, 2007, pp. 223–232.