Meta-heuristic Algorithms for Optimized Network Flow Wavelet-based Image Coding

Huseyin Kusetogullari[1,*], Mark S. Leeson[2], Burak Kole[2] and Evor L. Hines[2]

[1]Department of Computer Engineering, Gediz University, Izmir, Turkey

[2]School of Engineering, University of Warwick, Coventry, CV4 7AL, UK

***Abstract*:** Optimal multipath selection to maximize the received multiple description coding (MDCs) in a lossy network model is proposed. Multiple Description Scalar Quantization (MDSQ) has been applied to the wavelet coefficients of a color image to generate the MDCs which are combating transmission loss over lossy networks. In the networks, each received description raises the reconstruction quality of an MDC-coded signal (image, audio or video). In terms of maximizing the received descriptions, a greater number of optimal routings between source and destination must be obtained. The Rainbow Network Flow (RNF) collaborated with effective meta-heuristic algorithms is a good approach to resolve it. Two meta-heuristic algorithms which are Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) have been utilized to solve the multi-objective optimization routing problem for finding optimal routings each of which is assigned as a distinct color by RNF to maximize the coded descriptions in a network model. By employing a local search based priority encoding method, each individual in GA and particle in PSO is represented as a potential solution. The proposed algorithms are compared with the multipath Dijkstra algorithm (MDA) for both finding optimal paths and providing reliable multimedia communication. The simulations run over various random network topologies and the results show that the PSO algorithm finds optimal routings effectively and maximizes the received MDCs with assistance of RNF, leading to reduce packet loss and increase throughput.

Keywords: Genetic algorithm (GA), particle swarm optimization (PSO), rainbow network flow (RNF), multi-objective multipath optimization, routing representation, multiple description image coding, image transmission.

1. Introduction

The transmission of multimedia information over communication channels/paths has become a challenging problem with the increased usage of multimedia services in networks. Transmitting original source (information) naturally requires a significant amount of bandwidth and storage. This has been a strong motivation to examine and develop an efficient optimization method in order to use less bandwidth as well as finding the optimum network routings.

In multimedia transmission, original source image should be compressed for reducing redundancy in the image as well as efficient usage of bandwidth. There are two different approaches that have been applied to compress source image [1]. The first is lossless, where a compact representation of the source coding can be decoded to reconstruct

* Corresponding Author Tel.: +905423599211
E-mail address: huseyin.kusetogullari@gediz.edu.tr

the original signal without error. The second is lossy, which causes distortion in the original signal and the exact reconstruction of the original source cannot be achieved [1]. The purpose of using both techniques is to encode the source into a compressed digital representation that can be used for transmission. However, packet transmission problems such as packet dropping or congestion may occur over lossy transmission networks. For instance, there may be encounters with low capacity links, network congestion or excessive delay to deliver packets. To combat multimedia packet loss transmission problem, multiple description coding (MD coding or MDC) transmitting through multipath is preferred because even if one MDC packet is lost over a path, the lost MDC packet may be received via another path [1, 2]. Thus, using this approach maximizes MDC packets over lossy networks since the probability of receiving packets at the destinations increases [3].

The descriptions which carry similar information of the original source can be efficiently generated by using various quantization techniques (e.g. multiple description scalar and vector quantization) [6, 7] and sampling methods (orthogonal, quincunx) [8]. Since the quantization methods are used, transformation methods such as Discrete Wavelet Transform (DWT) [3], Discrete Cosine Transform (DCT) [4], and Embedded Zero Tree Wavelet Transform (EZW) [5] provide significant improvements in terms of preserving the important information of the multimedia source. In this work, a wavelet-based multiple description scalar quantization method has been applied to generate the MDCs of the color images since such methods are known to provide excellent rate-distortion performance [3]. Thus, the important information or energy in the sub-bands of the transformed image will be protected. The generated MDCs with acceptable quality are transmitted over multipath in lossy networks but finding the optimal paths and providing enough bandwidth capacity from source to destination are the two complex problems because of the many potential intermediate destinations an MDC packet might traverse before reaching its final destination [9]. To find an optimum solution, various algorithms have been proposed to provide greater and efficient performance of communication. For instance, Jiazi et al. [39] proposed Multipath Dijkstra Algorithm (MDA) to obtain multipath and they show that the algorithm gains great flexibility by employing different link metrics and cost functions. Furthermore, Genetic Algorithms (GAs) and Particle Swarm Optimization (PSO) are significant approaches to resolve the communication problems [10, 11]. They are used for solving different NP-hard network problems such as K-shortest paths [12], constrained shortest-path [13], multi-objective shortest path [14] and network flow [15]. In the most of routing optimization problems, only one weight or cost associated with each network link has been considered to find the optimum solution e.g., delay or length [10,11,17,18]. Begen et al. [19, 45] examined multimedia transmission over optimized lossy networks. They state that each network link has more than one cost parameter such as packet loss rate, length and bandwidth as it makes the network routing optimization problem even harder. However, they neither provide an optimization method to solve the multi constrained network routing problem nor a path selection method. In this paper, a new multi-objective cost function and an enhanced path representation are explained to solve these open problems and the performances of meta-heuristic algorithms are examined to find optimal multipath in the multi constrained network problems.

The problem of simultaneously optimizing multiple weights and costs is defined as one of multi-objective network optimization. In the simulated network models, three different cost variables will be considered associated with each network link that will be used to optimize path length, bandwidth consumption and packet loss rate for receiving an

acceptable quality of transmitted image. The strategies employed in [18] and [19] are used to select the numerical weights for each edge to optimize the shortest paths and packet loss rate. In this problem, the significant goal is to find optimal multi paths with minimum packet loss rate and path length as well as maximizing the average reconstruction quality of a received MDC-coded signal (image, audio or video) at the sink nodes. Meta-heuristic algorithms such as GA and PSO are employed in this work to provide an optimal trade-off between the cost values and the performances of them are compared with the MDA [39]. Three different fitness functions which are single objective shortest path, single objective minimal packet loss rate and multi-objective cost functions are employed for efficient result estimation in both finding optimal multipath and providing multimedia transmission. The simulations described in this paper consider undirected path graphs of a given node size n and edge size m. The optimization will employ a given set of MDC packet subsets and utilize the RNF algorithm, a GA and PSO. The main contribution and strength of the paper are as follows;

- A new multi-objective cost function is proposed and compared with the previously used single objective cost functions in terms of finding optimum multipath as well as increasing the probability of receiving multimedia packets at the destination nodes in lossy network problems [18, 19].
- Two meta-heuristic algorithms which are GA and PSO algorithms have been adapted with both proposed multi-objective cost function and RNF algorithm, and they are compared with MDA [39].
- Average quality of received images is estimated in terms of optimized network resources and statistics based on three fitness functions.
- The effectiveness of the proposed multi-objective cost function has been shown in transmitting the multimedia information through the optimized network. Besides, creating descriptions using the wavelet based image coding has been further improved to produce MDCs of color image for the available bandwidth between the source and destination nodes.
- Priority encoding method is enhanced to reduce invalid paths for corresponding multi-cost problem in terms of reducing computational time for finding multipath in networks.

The rest of the paper is organized as follows: work related to MDC generation is given in section II, the model and analysis of the network problem is presented in section III, the details of meta-heuristic algorithms for solving the multi-objective network routing problem are explained in section IV, results and discussions are provided in section V and the paper is concluded in Section VI.

2. Work related to Multiple Description Coding (MDC) Generation

MDC provides a good quality of received images if losses are inevitable in the network and many works have been examined to design practical MDC systems. MD scalar quantization (MDSQ) is one of the most popular techniques which have been discussed in [6]. The improvement of MDSQ which is MD vector quantization is studied and reported in [7]. The quantization methods cause loss of the information of the original image. However, there are

lossless MD image coding generation methods which are implemented by using source coding and transformation methods. The wavelet and discrete cosine transformation methods are most used techniques which have been widely discussed in the literature. For example, one of the most popular algorithms called Set Partitioning in Hierarchical Trees (SPIHT) used the wavelet approach [5]. Furthermore, Servetto et al. [3] applied the discrete wavelet transform (DWT) and resulting wavelet coefficients quantized by MDSQ to generate MDCs. Wang et al. [4] used the pairwise correlating transformation to generate multiple correlated descriptions in the framework of standard DCT-based image coding. Splitting an image into the descriptions was discussed by Zhang et al. [8].

This paper is focused upon two different problems, namely (i) optimizing network routings in order to get acceptable quality of received images in the lossy network and (ii) the simple generation of MDCs. Wavelets are attractive in image coding problems due to a tradition of excellent rate-distortion performance, so, we have applied MDSQ on a wavelet based colored image to generate MDCs.

2.1 Generating MDCs for Proposed Method

Even though the RGB color space can be used for pixel transmission, it has the disadvantage of illumination dependence. This means that there is a significant amount of correlation between the RGB components. If the illumination of an image changes because of packet losses in a lossy network, the achievement of high reconstructed images will be compromised.

Furthermore, the chrominance coefficients can be used for enhancement of the received image rather than modeling its intensity and can be neglected for larger changes without affecting our perception of the image. So it is necessary to transform the RGB color space to one of the color spaces where the separation between intensity and chrominance is more discriminate. Because of the linear conversion between RGB and YCbCr color spaces, we use the YCbCr color space to model the transmitted image. Poynton [20] discussed the conversion from RGB to YCbCr color space and from YCbCr to RGB color space.

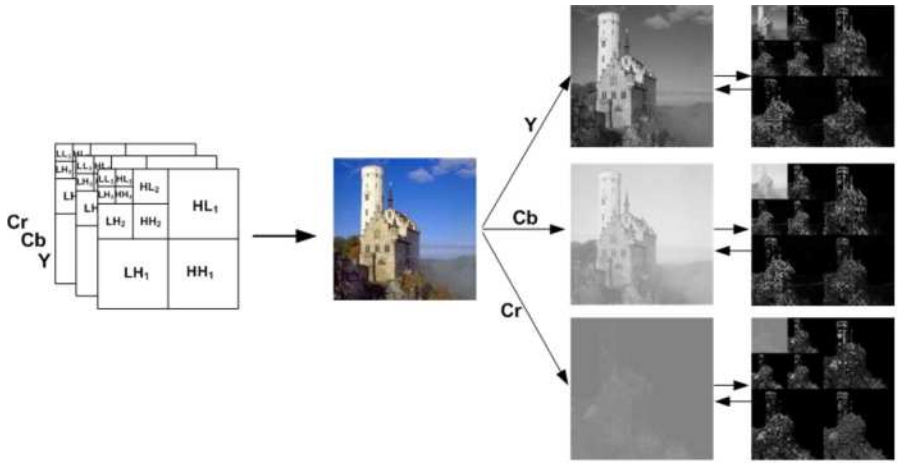


Fig. 1. The conversion of RGB to YCbCr color space and implementation of DWT with two levels of decomposition.

After the *RGB* to *YCbCr* color space conversion shown in Fig. 1, a transformation method has been applied to each color space of the given image to find the energy distribution of its coefficients. This determination of the

image energy distribution may be approached via several transformation approaches [3, 4]. However, wavelets have gained popularity in recent years for feature extraction [21], denoising [22], compression [23], face recognition [24] and packet image transmission [25] because they offer high coding efficiency [26]. The discrete wavelet transform (DWT) procedure uses a low pass filter and a high pass filter, chosen such that they divide the frequency range equally between them and is thus a suitable tool for packetizing the image in our application.

Briefly, the DWT may be explained as follows: The 2-D wavelet decomposition of an image is performed by applying a 1-D DWT along the rows of the image first and then the results are decomposed along the columns. This operation results in four decomposed sub-band images referred to as low–low (LL), low–high (LH), high–low (HL), and high–high (HH). The frequency components of those sub band images cover the frequency components of the original image. The LL band can be decomposed once again in the same manner, thereby producing even more sub bands. This can be repeated to any level as shown in Fig. 1. Loss of low level sub-band coefficients over the network effects significant distortion when image is reconstructed at the destinations. Therefore, MDSQ based MDC generation will be an efficient technique for the proposed method because most of the signal energy in the lower resolution subbands can be protected when the MDCs are generated. Further details of MDSQ for wavelet based image coding can be found in [26]. In this work, we use i.i.d. Gaussian source of variance one.

The achievable rates of MDCs are denoted as:

$$R = \sum_{i=1}^{L_i} \sum_{j=1}^{L_j} \left(n_j^{(Y)} R_{i,j}^{(Y)} + n_j^{(Cb)} R_{i,j}^{(Cb)} + n_j^{(Cr)} R_{i,j}^{(Cr)} \right) \quad (1)$$

where L_i is the number of descriptions, L_j is the number of subbands. $R_{i,j}^{(Y)}$, $R_{i,j}^{(Cb)}$ *and* $R_{i,j}^{(Cr)}$ are the bit rate of related subband of luminance, Chrominance Blue and Chrominance Red components, respectively, $n_j^{(Y)}$,$n_j^{(Cb)}$and $n_j^{(Cr)}$ are the ratios of number of samples in layer j over that of the full resolution layer. In the case of less available bandwidth, $n_j^{(Cb)}$ and $n_j^{(Y)}$ are indicated as small as possible to reduce the redundancy.

The achievable bit rates of the generated MDCs of color image can be different according to the capacity of the optimal paths obtained. Let k be the number of optimal paths from source to each sink.

Case 1: k=1: Single description coding (SDC) has been considered to transmit throughout only one optimal path. In SDC, each source sample is encoded with an average rate of "R" bits per sample and each codeword is placed in SDC packets to transmit through the network. Upon delivery, the SDC packets are then received by the decoder and after appropriate transformation the source sample is transferred to their destination. Furthermore, because of random or congestion losses on the path, SDC packets may be lost before reaching their final destination. Some losses may involve important information that will cause substantial distortion in the received image. In order to decrease the distortion of received color image, image interpolation methods have been applied to improve quality of received image [27]. The rate R of generated SDC will depend on the

bandwidth *bw* of the obtained optimal path and will be denoted as follows:

$$R = \frac{bw}{H \times W \times d}\ (bpp) \tag{2a}$$

$$D(R) = e^{-2R} \tag{2b}$$

Where *H*, *W* and *d* are the height, width and dimension of the image, respectively, and $D(R)$ denotes the achieved average distortion at the bit rate *R*.

Case 2: k=2: This assumption is different from the SDC approach because two optimal paths can transmit the descriptions generated. This problem is related to a two channel/path with three receivers' problem that multiple descriptions (MDs) can be transmitted over these channels/paths [1, 3]. Even if a receiver gets only part of the descriptions, it can still reconstruct an image with acceptable quality. In the MDC model, source information is encoded by several encoders where multiple description scalar quantization (MDSQ) and index assignment methods were used in this work [1, 3]. Furthermore, rates of "R_1"and "R_2"bits per samples are allocated to first and second encoder, respectively. Totally, a rate of "R_0" bits per sample is decoded at the destination. The total rate of the generated MDC will depend on the bandwidths of the optimal paths and is estimated as follows:

$$R_0 = \sum_{i=1}^{2} \frac{bw_i}{H \times W \times d}\quad (bpp) \tag{3a}$$

$$D_i(R) = e^{-2R_i} \qquad , \mathrm{i} = 0,1,2 \tag{3b}$$

Case 3: k>2: This problem is similar to the MDC problem but the received description is more likely than the cases of SDC and MDC but increasing number of descriptions will cause data redundancy transmission through the network. In addition, this problem is related to generation of *k* different descriptions of an image and is customarily referred to *n*-channel MDC generation; further details can be found in [28]. In this case, multiple description image coding using several multiple description scalar quantizers have been applied to the wavelet coefficients of YCbCr to create *k* descriptions [3, 28, 29]. This coding method is suitable for this case because more than two descriptions can be generated for the corresponding problem. The MDCs can be generated with achievable bit rates according to obtained bandwidths (bw) of the optimized paths. Consequently, the total rate of generated MDC and the average distortion will be:

$$R = \sum_{i=1}^{k} \frac{bw_i}{H \times W \times d}\,(\,bpp) \tag{4a}$$

$$D(bw_i) = e^{-2\sum_{i=1}^{k}\frac{bw_i}{HxWxd}} \tag{4b}$$

Where bw_i is the bandwidth of the optimized path *i,* and *H*, *W* and *d* are the height, width and dimension of the image, respectively. The quality and distortion estimation of the received images at the destination nodes

are developed in the following section.

2.2 Image Quality Estimation

The quality of the received images is estimated using several methods. Let the transmitted high resolution (HR) image I_o and reconstructed HR image I_r be of size HxW pixels and consist of three spectral bands, i.e., Y, Cb and; Cr which are first converted to R,G and; B and then, the following quantitative metrics are used to compare I_o and I_r:

Correlation Coefficient (CC) [30]*:* the correlation between each band of the reference image and the reconstructed image:

$$CC = \frac{1}{N_B}\sum_{i=1}^{N_B}\frac{\sum_{x,y}\left(\upsilon_o^{(i)}(x,y) - \overline{\upsilon}_o^{(i)}\right)\left(\upsilon_r^{(i)}(x,y) - \overline{\upsilon}_r^{(i)}\right)}{\sqrt{\sum_{x,y}\left(\upsilon_o^{(i)}(x,y) - \overline{\upsilon}_o^{(i)}\right)^2 \sum_x\left(\upsilon_r^{(i)}(x,y) - \overline{\upsilon}_r^{(i)}\right)^2}}, \quad (5)$$

where N_B is the number of spectral bands, i.e., N_B=3, (x, y) is spatial pixel coordinate, $\upsilon_o(x, y) = [\upsilon_o{}^{(1)}(x,y)\ \upsilon_o{}^{(2)}(x, y)\ \upsilon_o{}^{(3)}(x, y)] = [I_o^{(R)}(x, y)\ \ I_o^{(G)}(x,y)\ \ I_o^{(B)}(x, y)]$ *denotes* the spectral vectors of the pixel (x, y) in the transmitted original image and $\upsilon_r(x, y) = [\upsilon_r{}^{(1)}(x, y)\ \upsilon_r{}^{(2)}(x, y)\ \upsilon_r{}^{(3)}(x, y)] = [I_r^{(R)}(x, y)\ I_r^{(G)}(x, y)\ I_r^{(B)}(x, y)]$ denotes the vector obtained after reconstruction. In order to clearly analyze image quality in the optimization problem, it is beneficial to use the CC equation where the estimated values are bounded between 0 and 1. As a result, the CC value should be as close to 1 as possible in the maximization problem.

Root Mean Square Error (RMSE): the root mean square error between the original image and the reconstructed image, i.e.,

$$RMSE = \frac{1}{N_B}\sqrt{\sum_{i=1}^{N_B}\Delta\left(I_o^{(i)}, I_r^{(i)}\right)}, \quad (6)$$

Where

$$\Delta\left(I_o^{(i)}, I_r^{(i)}\right) = \sqrt{\frac{1}{HxW}\sum_{x,y}\left(X_o^{(i)}(x,y) - X_r^{(i)}(x,y)\right)^2} \quad (7)$$

The RMSE value should be as close to 0 as possible.

3. Modeling and Analysis

The links or edges have associated costs that could be based on their distance, capacity, and transmission medium quality. Let $G = (V,E,W,Q,B)$ be a connected weighted undirected graph with $n = |V|$ nodes and $m = |E|$ edges, sets $\omega_{i,j} \in W$, $p_{i,j} \in Q$ and $bw_{i,j} \in B$ independent cost variables from each other and represent the weights of each edge $(i,$

$j) \in E$ where the weight is restricted to be a nonnegative real number [10, 11, 18, 19].

3.1 Utilization of Rainbow Network Flow (RNF) and Network Modeling

The RNF problem collaborating with the efficient generation of MDCs to maximize the quality of received images for all sinks is an NP-hard problem [31]. RNF gives a distinct color to each optimized path to prevent receiving the same description or packet twice at a destination node because this will not increase the reconstruction quality of the received image but will increase network bandwidth consumption. RNF is a convenient way for solving NP hard network flow problems such as finding multipath between multi-source and destinations (an NP hard flow problem that has been discussed in [16]) in addition to its bandwidth saving properties. RNF was first discussed to maximize the flow in a network model and can be greatly simplified if the flow of MDC descriptions from diversity servers is optimized with respect to a single sink or multi-sinks [16]. Furthermore, for each optimized path which is colored with i, a new description is generated and colored with i that can be sent from corresponding colored path i. Assume that a number of disjoint paths k from a source to a destination can be optimized, then, there are k^2 colors achieved between k number of sources and destinations. Here, a GA and PSO collaborating with RNF have been used to optimize the routings across the overall lossy network so that a different number of paths from source to destination can be obtained which was discussed with the three cases in previous section.After optimizing the routing problem, the MDCs should be transmitted through the optimized paths in order to determine the acceptable quality of reconstructed images at the sink nodes. Therefore, the network flow of MDC that achieves the maximum fidelity at the sinks must have flow paths of distinct colors and maximizing the network flow of MDC [31].

A version of the RNF that is particularly relevant to our problem is defined with following inputs:

1) $G=<V,E,W,Q,B>$, an undirected graph with a node set V, an edge set E and set of costs W, Q *and* B.
2) $S = \{s_1\}$, $T = \{t_1, t_2, t_3, \ldots t_r\}$ two subsets of V representing the set of source and sink nodes respectively.
3) A function $R: E \rightarrow R+$ representing the capacity of each link considered an integer in G.
4) A set $\omega_{i,j} \in W$, $\zeta_{i,j} \in Q$ and $bw \in B$ called the independent costs of the links.
5) A set $D \subset X$ called the description packet set in source.
6) The set $DR \subset Y$ called received description packet set in T.

The descriptions are sent over the optimal colored paths and reconstructed at the destination nodes as illustrated in Fig. 2. If the network graph is reliable, the image will be the same as the transmitted image. However, if losses are present (which is inevitable) then the quality of received image will be decreased because of the smaller number of packets arriving at the destination nodes.

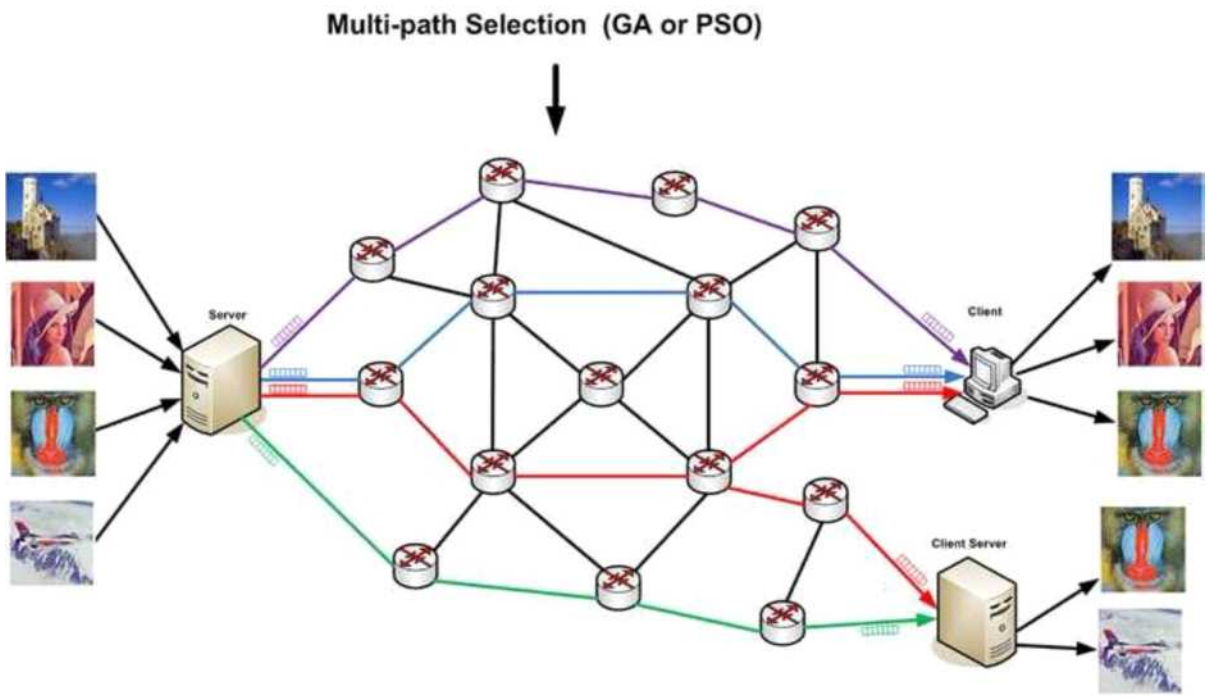
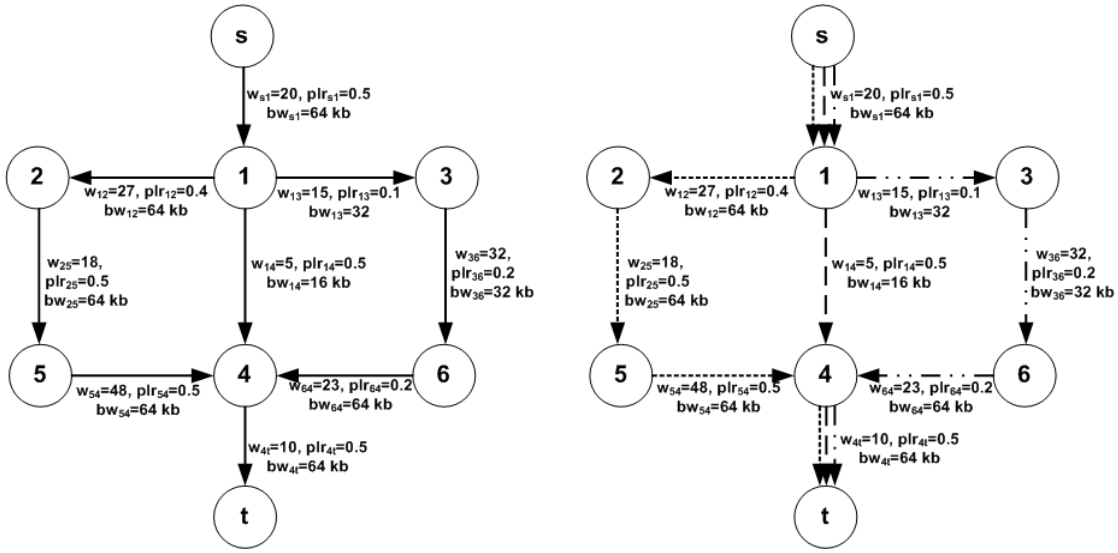


Fig. 2. Graphical representation of optimal multipath selection for proposed multimedia communication model.

Figs. 3 (a) and (b) illustrate a small direct network with 8 nodes to discuss and analyze the RNF algorithm with independent cost values associated with each network link. The server node (s) has a description (SDC) to transmit to the sink node t. Nodes 1, to 6 are used as intermediate nodes. Each edge is assigned three independent cost values such as path length ($\omega_{i,j}$), packet loss rate ($\zeta_{i,j}$) and available bandwidth ($bw_{i,j}$). Based on the explanation of RNF, Fig. 3 (b) demonstrates the different colored optimal routing solutions (shown by solid, pointed and dashed lines in the figure) to transmit the description in a lossy network of eight nodes. Each optimal routing is giving different advantages and disadvantages while the description is sent to the sink node. For example, the greatest rate of description is generated and transmitted over the path (s-1-2-5-4-t) but it is the longest optimal path. The shortest path is (s-1-4-t) but the lowest rate of description can be sent over this path. In addition, packet loss rates over the optimal paths (defined as the difference between the number of packets sent and packets received per time unit) will affect the process by determining the quality of the image at the sink node. Estimation of packet loss rate will be briefly explained in the next section. The third optimal path (s-1-3-6-4-t) provides a low packet loss rate as well as a trade-off between the cost values of bandwidth and path length. Here, to solve the complex network problem, RNF enhanced by GA and PSO methods has been applied to obtain optimal routings to reach a specific sink. Furthermore, three different fitness functions have been discussed in the GA and PSO implementations to obtain optimal routings.



(a) (b)

Fig. 3. An example of routing problem for RNF. (a) The directed network with 8 nodes and 8 edges assigned with path lengths (*w*), packet loss rates (*plr*) and bandwidths (*bw*). (b) Three optimal routings for an MDC, marked by solid, pointed and dashed icons and edges for RNF.

3.2 Packet Loss Models

Packets are randomly dropped in the network and generally there will be no opportunity to look inside the packets and discriminate in terms of which are lost. The usual way to handle a packet loss is to retransmit the lost packet in a data network. Retransmission protocols such as TCP, whereby the receiver tells the sender whether a packet arrived or not facilitate end-to-end reliability despite unreliable links. However, retransmitting a lost packet will cause a network cost and will entail delivery delay. MDC can be utilized to avoid dropping a packet without retransmission. In lossy networks, packet losses can occur for various reasons which are mainly classified into three types: 1) Random losses, caused by poor channel conditions; 2) Burst losses, mainly due to link or node failure in wired systems; 3) Congestion, due to limited buffer, bandwidth and processing capability at network routers. The proposed algorithm attempts to find the optimal disjoint paths. This means that lost packets which are caused by bursting can be neglected because they generally occur on joint paths.

Random Losses: Transmitting packets over an individual link can be described using Bernoulli trials with a single packet loss probability over the first link is p_1 and loss-free probability $1\text{-}p_1$. Let the set of link loss rates to be estimated be denoted as $p = \{p_1, p_2, \ldots, p_L\}$, where p_L is the packet loss rate at the link L. For the individual links of a given path, we assume that, the independent packet losses are defined by a Bernoulli loss process. The loss probability of the *k-th* path is evaluated as follows:

$$P_k = 1 - \prod_{i=1}^{L} (1 - p_i) \tag{8}$$

Let M be the number of transmitted packets from the source node and $\mu_{1:L}$ is the expected packet loss over the

transmitted links. In this case, the probability of successful received packets will depend on the packet loss rate of the corresponding link, i.e.,

$$\mu_1 + \mu_2 + \cdots + \mu_L = M\left[1 - \prod_{i=1}^{L}(1 - p_i)\right] \tag{9}$$

where

$$\mu_n = \begin{cases} Mp_1 & n = 1 \\ M\left\{\prod_{i=1}^{n-1}(1 - p_i)\right\}p_n & n > 1 \end{cases} \tag{10}$$

where L is the number of links of the corresponding path. Furthermore, the Bernoulli model can also be used for the congestion problem [32]. Packet loss probabilities over the network links are discussed in [19] and estimated in the range of 10%-20%.

4. Meta-heuristic Algorithms for Multi-objective Network Problem

4.1 Overview of Algorithms for Network Routing Problems

Variations of routing problems have to be solved to achieve high throughput communication in a wide variety of network problems such as *K* shortest- paths [12], constrained shortest-path [13], multi-objective shortest path [14] and dynamic shortest path routing discovery [10]. Most of these problems are NP-hard and one way of solving such problems is to assign a cost metric (weight) for each link in the network [10, 11, 19]. To date, a range of deterministic algorithms such as Dijkstra's algorithm [46], breadth-first search algorithm [47] and the Bellman-Ford algorithm [48] have been developed to find the lowest cost or routing from a specific source to specific destination through a network. Jiazi et al. [39] improved Dijkstra's shortest path algorithm and proposed multipath Dijkstra algorithm (MDA) to find optimal multipath by using different cost functions in network topologies. They show that the proposed algorithm is flexible in obtaining multiple shortest paths. Recently, Evolutionary Algorithms (EAs) have drawn considerable attention as routing selection problem solvers as they provide robust and efficient approach for solving complex network routing problems [10]. EAs such as genetic algorithms (GAs) [10, 49], genetic programming [50] and evolutionary programming [51] have been previously proposed for various network problems. Amongst the EAs, GA provides the best performance for optimization problems [10]. Besides the GA, Particle Swarm Optimization (PSO) is another efficient optimization algorithm, first discussed by Kennedy et al. [40], which may be employed to search for the optimum solution based on social agent behavior inspired swarm intelligence.

In this paper, two meta-heuristic algorithms are used with three different fitness functions, i. e., shortest routing path [10, 11], minimum packet loss rate [19] and proposed optimal multipath selection, to find optimal multipath in lossy networks to provide effective multimedia transmission. The algorithms are compared with the MDA method [39] to obtain optimal multipath in various lossy network models to provide efficient multimedia transmission.

4.2 Genetic Algorithm

Genetic algorithms (GAs) are becoming more widely used in many areas to find high quality solutions in optimization problems [33-37]. In the implementation, the GA starts with an initial set of chromosomes (typically random) represented as binary numbers to offer a solution for the problem. By applying the evolution operators, a new generation is formed by firstly applying Tournament Selection Operator (TSO) which considers the fitness values of the parents, and rejection of the others so as to keep the population size constant. To perform it, new chromosomes are formed by merging two parents from the current generation using the two-point crossover operator and modifying a chromosome using a uniform mutation operator. Let's consider two different individuals with *n* dimensions of the given population, i. e.

$$\begin{aligned} \overrightarrow{x_i} &= (x_{i1}, \dots, x_{ir}, \dots, x_{is}, \dots, x_{in}) \\ \overrightarrow{x_j} &= (x_{j1}, \dots, x_{jr}, \dots, x_{js}, \dots, x_{jn}), \end{aligned} \tag{11}$$

where *r* and *s* denote two randomly selected crossover points, and $r \leq s$. After the crossover operator employed, the produced individuals or new candidate solutions are:

$$\begin{aligned} \overrightarrow{x_i} &= \left(x_{i1}, \dots, x_{jr}, \dots, x_{js}, \dots, x_{in}\right) \\ \overrightarrow{x_j} &= (x_{j1}, \dots, x_{ir}, \dots, x_{is}, \dots, x_{jn}) \end{aligned} \tag{12}$$

Let us suppose $\vec{x} = (x_1 x_2, \dots, x_{ik}, \dots, x_n)$ where *k* denotes the randomly selected point in the chromosome. Thus, the *k-th* gene of the individual can be changed or mutated to produce another individual. The best chromosome is selected to be transferred in the new population using elitism in GA. After some generations, the algorithms converge to the best chromosome, which represents the optimum or suboptimal solution to the problem. Moreover, two different stopping criteria are used, one being the achievement of the optimal fitness value and the other the reaching of a previously set iteration limit. The pseudo-code of GA is as follows:

Table 1

Pseudo-code of GA.

```
Procedure GA
{
    set the generation index g = 1;
    initialize population;
    while termination condition not satisfied do
{
    evaluate current population;
    select parents using tournament selection method [10];
    apply two-point crossover and uniform mutation operators to the selected parents to create offspring;
    select the best individual to be a member of the next generation by using elitism [10];
    Insert new offspring in the new population;
```

set $g \leftarrow g + 1$

}

stop and return the best solution in the current population

}

4.2.1 Proposed Fitness Function for the Multimedia Communication Routing Problem

In GA and PSO algorithms, the quality of a represented potential solution is estimated by a fitness function. The purpose of using fitness functions is to map a chromosome and a particle representation into a scalar value or a cost value. For the multi-objective network problem, the fitness function estimates a cost value for each individual that is close to global optimum or not. Thus, the strong and weak candidates can be evaluated according to the fitness values and the new population is produced based on them. The objective functions or fitness functions that involve computational efficiency and accuracy are defined with constraints as follows:

$$f_1 = min\left(\sum\nolimits_{i,j\in E}^{n}(\omega_{i,j})I_{i,j}\right) \tag{13}$$

$$f_2 = min\left(1 - \prod_{i,j\in E}^{n}(1-\zeta_{i,j})I_{i,j}\right) \tag{14}$$

$$f_3 = min\|y\|_2 = min\left(\sqrt{y_1^2 + y_2^2}\right) \tag{15}$$

Where,

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \left(\frac{\sum_{i,j\in E}^{n}(\omega_{i,j})I_{i,j}}{c}\right) \\ \left(1 - \prod_{i,j\in E}^{n}(1-\zeta_{i,j})I_{i,j}\right) \end{bmatrix}$$

$$s.t. \qquad \sum\nolimits_{j=1}^{n} I_{i,j} - \sum\nolimits_{j=1}^{n} I_{j,i} = \begin{cases} 1, & i = 1 \\ 0, & i = 2,3,\dots,n-1 \\ -1, & otherwise \end{cases}$$

$$(bw_{i,j})I_{i,j} \geq \varepsilon$$

$$I_{j,i} \in \{0,1\}, \quad \forall\,(i,j)$$

where c is the maximum value of weights $\omega_{i,j} \in W$, $bw_{i,j} \in B$ is the bandwidth of the link connection indicator $I_{i,j}$ and f_3 is a proposed multi-objective function for finding the optimal multi paths; ε denotes the capacity of the transmitted data or image. Furthermore, the single objective functions f_1 and f_2 are used to find shortest path and minimum packet loss rate of a path in the given network topologies, respectively. Let $I_{i,j}$ be the link connection indicator between nodes i and j which plays the role of a chromosome and particle map providing information on whether the link from node i to node j is included in a routing path, i.e.,

$$I_{i,j} = \begin{cases} 1, & \text{if the link from node i to node j exists in the routing path} \\ 0, & \text{otherwise} \end{cases} \quad (16)$$

4.3 Particle Swarm Optimization (PSO)

The Particle Swarm Optimization (PSO) algorithm is a new optimization algorithm which is inspired by the social behavior of colony of animals in environment [40]. Like GAs, PSO is a population-based optimization method that searches multiple solutions [41-43] but it employs a different "competitive" strategy. Position and velocity of each particle *i* is updated if the fitness value does not provide the minimum optimum solution and the position of the following properties of an individual particle *i* are updated at each iteration. To find the fitness value of particle *i*, the PSO uses the fitness cost functions defined in Eqs. (13), (14) and (15) to find the minimum optimum result or multipath in the lossy networks. A current position with D dimensions in search space, $\vec{x}^{(i)}$, a current velocity with D dimensions, $\vec{v}^{(i)}$, personal best position and global best position in search space, f_{best}^{i} , f_{best}^{g}, respectively. The personal best position, f_{best}^{i}, corresponds to the position in search space where particle *i* had the smallest value as determined by the objective function *f*, aiming to minimize the task. The global best position denoted by f_{best}^{g}, represents the position yielding the current global best position amongst the entire particles best positions f_{best}^{i}. Eqs. (17) and (18) define how the *velocities* and *locations of* swarm particles are updated at iteration *k*, respectively. In summary, PSO performs the following steps:

1) Initialize the number of particles *p*, generation index *k=0*, constants c_1 *=2.0* and c_2 *=2.0;*
2) Randomly initialize positions of all particles $\vec{x}^{(i)} = \{ x_1^i, x_2^i, \dots, x_D^i \}\ for\ i = 1, \dots, p$;
3) Randomly initialize velocities of all particles $\vec{v}^{(i)} = \{ v_1^i, v_2^i, \dots, v_D^i \};\ 0 \le v_1^i \le v_{max}$, $for\ i = 1, \dots, p$;
4) While stopping criteria are not reached do:
5) For each particle *i*:
6) **Evaluate the fitness value** $f^i\ for\ i = 1, \dots, p$;
   If $f^i < f_{best}^i$ then $f_{best}^i = f^i$, $\overrightarrow{px}^{(i)} = \vec{x}^{(i)}$
   Else $f^i < f_{best}^g$ then $f_{best}^g = f^i$, $\overrightarrow{gx}_{k+1}^{(i)} = \vec{x}_k^{(i)}$
7) End For
8) For each particle *i* in the swarm:
9) **Update velocity:** Estimate velocity $\vec{v}^{(i)}$ using Eq. (17)
10) **Update position:** Update position $\vec{x}^{(i)}$ using Eq. (18)
11) End For
12) Set *k* = *k*+1
13) End While

In this work, PSO explores a D-dimensional space, using a population of particles which are initially provided with random velocities and positions which are updated by using the following equations:

$$\vec{v}_{(k+1)}^{(i)} = \omega \vec{v}_{(k)}^{(i)} + c_1 r_{(k)}^{(i)} \left(\overrightarrow{px}_{(k)}^{(i)} - \vec{x}_{(k)}^{(i)} \right) + c_2 r_{(k)}^{(i)} \left(\overrightarrow{gx}_{(k)}^{(i)} - \vec{x}_{(k)}^{(i)} \right) \quad (17)$$

$$\vec{x}_{(k+1)}^{(i)} = \vec{x}_{(k)}^{(i)} + \vec{v}_{(k+1)}^{(i)} \quad (18)$$

where c_1 and c_2 are two learning factors that control the influence of personal best and global best in the search process and ω is the inertia weight which is used to update velocity for increasing the convergence speed. In the PSO algorithm, both c_1 and c_2 are chosen between [0, 4.0] to perform good results. Inertia weight ω is selected randomly between [0, 1] at each generation [11]. Furthermore, a priority encoding method is also applied in PSO implementation to represent the valid paths.

4.4 Enhanced Priority-based Encoding for Path Representation

Each node in the network topology is given a unique integer index value from $1,\ldots,n$ where n is the number of nodes. In addition, each index node is a gene of the chromosome for the GA and a position of the particle for the PSO and the number of nodes in the network routing topology may differ from individual to individual. An example of a fixed-length individual (e.g. chromosome or particle) and its decoded path is shown in Fig. 4. The length of the individual which is represented as a string variable in Fig. 4 should not exceed the maximum length of n. In the GA implementation, a gene in a chromosome is characterized by two factors: the position of the gene located within the structure of chromosome, and the costs between two connected genes. In the encoding method, the position of a gene is used to represent the node number and the costs of two connected genes are used to evaluate the fitness values. In the PSO implementation, the positions located within the structure of particles are used to represent the valid path.

In the routing problem, representing a valid path is a critical and typical complexity problem because of the many potential intermediate nodes that can be selected as a node ID in a solution path. If the individual (chromosome or particle) decoding strategy is not well employed, the obtained path may not terminate at the destination node leading to an invalid path. Such an invalid path clearly does not provide any benefit for solving the problem because it cannot be decoded to a solution. The paths represented must aim to find a potential solution for solving the routing problem efficiently. Priority-based encoding is a very effective method to represent a path that has been applied in its basic and enhanced forms to solve shortest path routing problems [11, 38]. In this work, an improved priority-based encoding method or local search based priority encoding method has been applied to represent a potential routing solution.

In the GA and PSO algorithms, characterizations of individuals have been used to represent a valid path. For instance, a gene in a chromosome is characterized by two factors: locus, i.e., the position of the gene located within the structure of chromosome, and allele, i.e., the value taken by the gene. In the proposed encoding method, the position of a gene in the GA and the priority of a particle in the PSO are used to represent a candidate path which can be uniquely determined from this encoding.

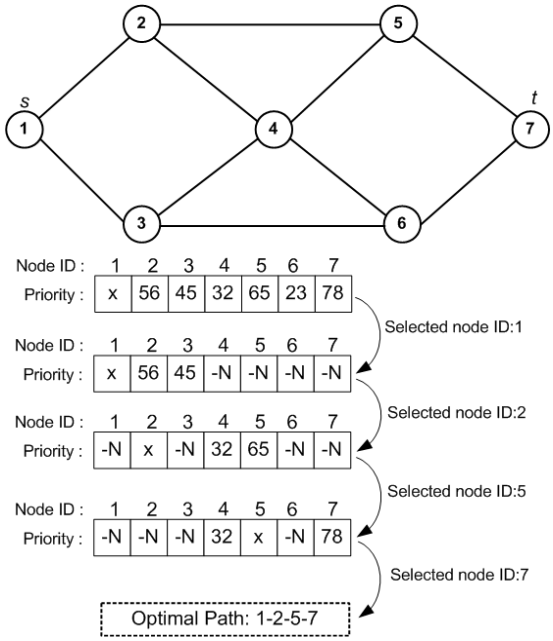


Fig. 4. An example of undirected graph with 7 nodes and 12 edges and the path representation according to local search method with priority-based encoding.

In terms of an undirected connection network, Fig. 4 illustrates the local search based priority encoding method and its decoding. Initially, the global priority is randomly generated and then, an attempt is made to find a connected node ID to the source node 1. Nodes 2 and 3 are eligible for the next possible node which can be easily fixed according to adjacency relations among the nodes. The node priorities are 56 and 45, respectively so node 2 has the highest priority and is incorporated into the path. The possible nodes next to node 2 are nodes 1, 4 and 5 but the first must be removed to be searched as a next node in the path by inserting a value of $-N$ because it was chosen as a node ID in previous steps. In this case, node 5 has the largest priority value and is placed in the path. Then, we form the set of nodes available for the next position and select node 7 with the highest priority among them. These steps are repeated until we obtain a valid path from source to destination (1-2-5-7).

5. Results and Discussions

In general, we are interested in two different problems, i.e., multipath optimization in lossy networks and receiving acceptable quality of an image. Therefore, GA and PSO have been first applied to find optimal multipath which are colored by RNF in the network topology and then, MDSQ is applied to the wavelet coefficients of YCbCr to generate the MDCs with achievable rates to transmit through the optimized network topology. The proposed algorithms are compared with the multipath Dijkstra algorithm (MDA) for both finding optimal paths and providing reliable multimedia communication. All the simulations were performed with MATLAB on Intel Core i3 processor (2.40-GHz) and 3-GB (2.86 usable) RAM.

The implementation results are part of the simulation approach to understand our network problem. To verify the results of different topologies, randomly generated networks with different number of nodes with randomly assigned

link costs were investigated for optimal multipath solution. In this work, network simulator Sinalgo [68] is used to randomly generate five different network models as case studies each having a different number of nodes and links. Each network model containing four different sinks (e.g., client, and server) is as follows: case I: 20 nodes, 86 links; case II: 40 nodes, 183 links; case III: 80 nodes, 324 links; case IV: 120 nodes, 567 links; case V: 150 nodes, 953 links. Generated cost values on each link in a network topology will make the corresponding problem solvable. Therefore, the cost values on each link were randomly selected to analyze the simulation results [10, 11, 19]. In the network topologies, the available bandwidth on each link ($bw_{i,j}$) was chosen randomly in the range of 32kB-512kB. The packet loss probabilities ($\zeta_{i,j}$) were randomly assigned in the range of 10% - 20% for all links [19] and the link length ($\omega_{i,j}$) was randomly chosen between [25, 125]. This approach follows that adopted previously in a variety of optimization problems [10, 11, 44].

In the GA and PSO implementations, population size (K) was set to two different values, e.g. 30 and 60 and a number of generations (NG) is selected to find all optimum paths. These algorithms attempt to find all optimal paths until all of them are determined in the network model. In order to keep the number of fitness function evaluations ($NF = NG*K$) fixed and give the same chance for the meta-heuristic algorithms to solve the optimization problem, we run GA-60 and PSO-60 with 500 generations, and GA-30 and PSO-30 with 1000 generations. In the GA implementation, the fitness value of each chromosome was computed and then, the Tournament Selection is applied for parent selection [34]. *Two* point uniform crossover with a crossover rate Cr of 0.8 is used together with a uniform mutation rate Mr of 0.01 to evolve the initial population. To keep the fittest individual in next generation, the best individual is selected using elitist selection in the GA. Thus, the GA algorithm tried to converge to a solution according to the initial population generated, reducing the chance that the program got stuck in local minima. In the PSO implementation, the particle positions and velocities were initialized with random real numbers in the ranges of [0, 100] and [0, 50], respectively. The learning factors c_1 and c_2 set to 2 and inertia weight ω is selected randomly between [0, 1] in each generation [11].

Our goal is to find the optimum colored multipath routings using GA or PSO collaborating with the RNF algorithm in multi-objective network models. In order to analyze results clearly, the routings between source and multi-destination nodes (four nodes) are optimized and then the MDCs generated are transmitted to the destination nodes. In addition, we suppose that each sink node demands different image source from a source node.

5.1 Performance Comparisons of Algorithms

The evolution of algorithms provides great improvement to create individuals to find the optimum cost value in the optimization problems. There are various parameters in GA and PSO to affect the evolution of population and optimum cost value estimation. The most important parameters for GA are population size K, crossover rate Cr and mutation rate Mr. However, the most significant parameters for PSO are population size K, learning factors c_1 and c_2 and inertia weight ω. In order to demonstrate the best cost function value (global or local optimum) of the population at each generation for the fitness function f_1, the estimated cost values are recorded for different K values.

Figs. 5 (a), (b), (c) and (d) depict the cost values of the population computed using the implemented algorithms

according to Eq. (13). Different values of population size K and generations g are used in the meta-heuristic algorithms to give fair chance to achieve the optimum cost value. For instance, GA and PSO are employed with K= 30 and g =1000 to get the results as shown in Figs. 5 (a), (c) and (d) and they illustrate the best cost fitness values, the average fitness values and the worst fitness values, respectively. In Figs. 5 (a) and (b), multipath Dijkstra algorithm (MDA) [39] provides a reference point to compare the meta-heuristic algorithms. Fig. 5 (b) illustrates the best cost fitness values estimated by using PSO and GA with K=60 and g =500. According to the best cost fitness values in Figs. 5 (a) and (b), increasing the population size K decreases the number of iterations to reach the global optimum or suboptimal solution. For example, PSO-60 estimates the best cost fitness value at the shortest iteration number as shown in Fig. 5 (b). Furthermore, averages of the cost fitness values, shown in the Fig. 5 (c), demonstrate that the evolution continues to improve the solution until all individuals reach the best cost fitness result (global or local optimum).

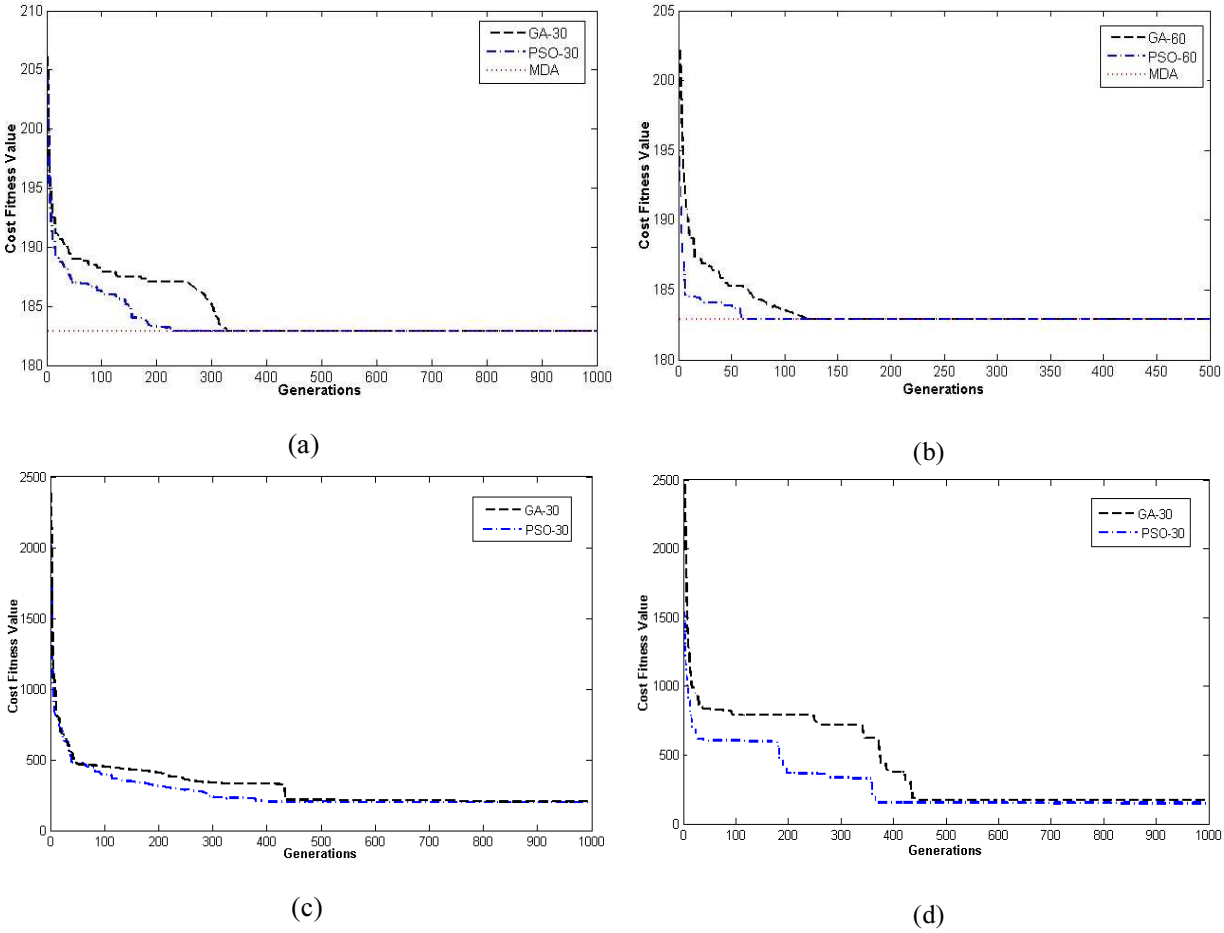


Fig. 5. The performance analysis of GA, PSO and MDA algorithms for the fitness function f_1 under the different population sizes K=30 and K=60, Cr = 0.8 and Mr = 0.01, c_1 and c_2 set to 2.(a) The best cost fitness values computed by using GA-30, PSO-30 and MDA, (b) The best cost fitness values computed by using GA-60, PSO-60 and MDA, (c) The average cost fitness values computed by using GA-30 and PSO-30, (d) The worst fitness values computed by using GA-30 and PSO-30.

Table 2 shows the comparison of the average CPU time required to execute the algorithms to find all optimal paths between randomly selected one source and four sink nodes in two different network models. To obtain the results, each algorithm was executed 100 times. For the smallest network model, multipath Dijkstra algorithm (MDA) provides better average execution time compared to the other algorithms and it takes average of 17.36 seconds for execution. However, the execution time is dramatically increased for the larger network models. It is clearly seen that the PSO-30 has the best average execution time compared to the other algorithms for the larger network model as an extreme case and it needs average of 28.03 seconds for execution. As a result, from this perspective, MDA requires less execution time for the smallest network models whereas the PSO-30 takes less execution time for the growing network models.

Table 2
Comparison of execution time of the algorithms.

	Execution Time (seconds)	**Execution Time (seconds)**
Algorithm	**20 Nodes**	**150 Nodes**
MDA [39]	17.36	34.32
GA-30	18.64	29.12
GA-60	21.43	35.15
PSO-30	17.59	28.03
PSO-60	19.65	29.71

5.2 Routing Optimization and Performance Analyses in Lossy Networks

The GA and PSO methods implemented with RNF algorithm are optimizing the multipath routing problem in the constructed network models and three different fitness cost functions (13)-(15) were utilized for efficient routing estimation. In addition, path links which have been found were not considered in the next path estimation and therefore, the links over the obtained routings were removed to find a next disjoint path efficiently. This means that the adjacent matrix of the network model was dynamically changed for efficient results.

Each network model was simulated 20 times to analyze and understand the performance of the implemented algorithms and to perform it, different population sizes and different number of generations were used in both GA and PSO algorithms. The results evaluated clarify that the implemented algorithms perform well to resolve the NP hard RNF problem. The estimated path costs are illustrated in Figs. 6 and 7 for the three different fitness functions. Optimization was performed using each of these fitness functions in the implemented algorithms, i.e., f_1 for Figs. 6 (a) and (c); f_2 for Figs. 6 (b) and (d); f_3 for Fig. 7 (a) and (b), which show the performance of five different algorithms to find the optimal multipath in the randomly generated network models. Figs. 6 (a) and (d) indicate that multipath Dijkstra algorithm (MDA) provides slightly better performance than PSO-60 and much better performance than the other algorithms to resolve the NP-hard routing problem using the single objective functions. On the other hand, the GA-30 performs relatively poor for all network topologies in terms of solution quality estimation. Figs. 6 (b) and (c) show that the PSO-60 performs well compared to the other algorithms.

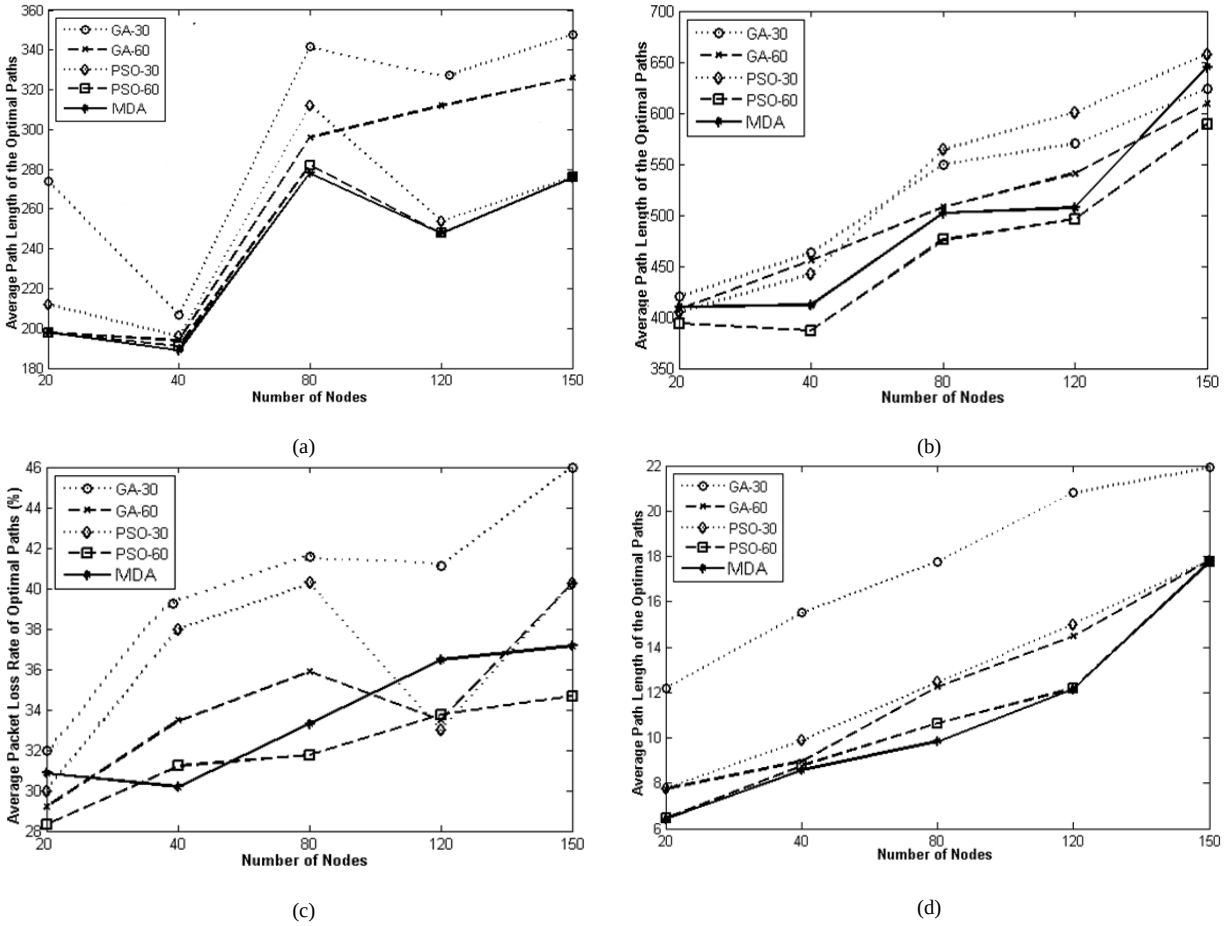
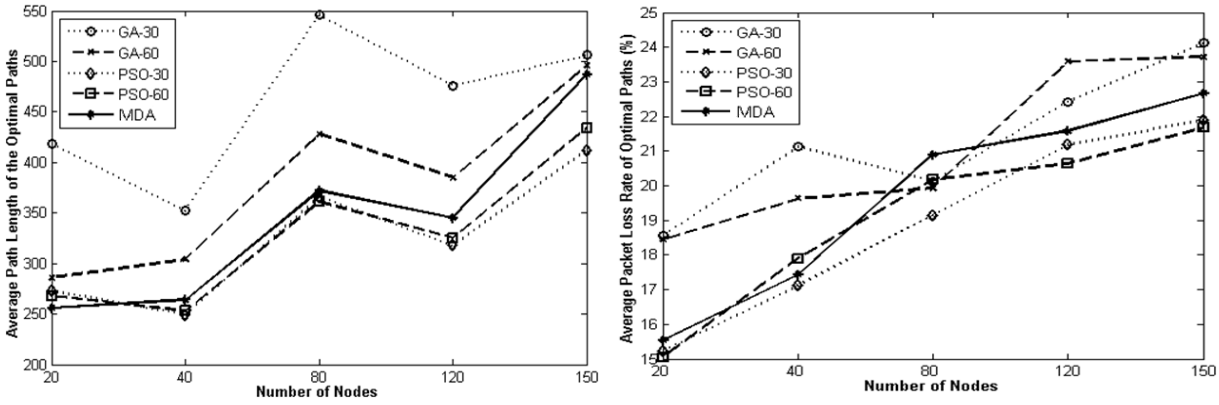


Fig. 6. Comparison of the optimization results obtained by using GA and PSO with K = 30 and 60 and Dijkstra. (a) , (b) Comparison of the results of average path lengths estimated by using f_1 and f_2, respectively.(c), (d) Comparison of the results of average packet loss rates (%) estimated by using f_1 and f_2, respectively.

Fig. 7 (a) and (b) denote the comparison of results between average path length and average packet loss rate of the optimal paths when the third fitness function f_3 is used in the algorithms. These results clarify that f_3 provides optimal trade-off between the cost values in multipath optimization. Moreover, PSO-30 and PSO-60 perform well among the all algorithms in optimizing the multipath in the lossy networks. Furthermore, meta-heuristic algorithms achieve great solution to optimize multi-objective network model by using the proposed multi-objective fitness function whereas MDA provides better results by using the single objective functions.



(a) (b)

Fig. 7. Comparison results of the average packet loss rates (%) and path lengths of optimal paths estimated by using f_3 in GA, PSO and Dijkstra.

Eventually, the first fitness function f_1 estimates the shortest paths effectively compared to the other fitness functions but does not perform well for optimizing packet loss rates. However, the second fitness function f_2 optimizes the packet loss rate of the paths better than f_1. Furthermore, the modified fitness function f_3 offers superior performance than f_1 and f_2 in terms of optimizing two independent cost variables simultaneously. In order to further evaluate and compare the fitness functions, i. e. shortest path, minimum packet loss rate and optimal multipath selection, the following equation has been developed:

$$W_{i,j,k} = \frac{C_{i,j,k}}{\max(C)} + \frac{P_{i,j,k}}{\max(P)} \tag{19}$$

Where W is a total cost matrix and, i, j and k are index of algorithms, index of network model and used fitness functions, respectively. C and P are the optimized path length and optimized packet loss rate matrices estimated by using all algorithms as shown in Figs. 6 and 7. Table 3 illustrates the total cost matrix W of the three different fitness functions employed in the GA-30, GA-60, PSO-30, PSO-60 and MDA. The total cost matrix indicates that PSO-60 implemented with the optimal multipath selection function f_3 gives the minimum result with 0.734 and GA-30 implemented with the shortest path function f_1 gives the maximum result with 1.468. Furthermore, PSO-60 implemented with the optimal multipath selection function f_3 is the best performing approach of the comparisons to compute the minimum optimal results of average cost of 0.897. As a result, PSO-60 implemented with the optimal multipath selection function provides the greatest performance compared to the other algorithms in terms of finding optimal multipath in networks.

Table 3
Comparison results of algorithms using three different cost functions.

	Shortest Path f_1					*Minimum Packet Loss Rate f_2*					*Optimal Multipath Selection f_3*				
# Nodes	**GA-30**	**GA-60**	**PSO-30**	**PSO-60**	**MDA**	**GA-30**	**GA-60**	**PSO-30**	**PSO-60**	**MDA**	**GA-30**	**GA-60**	**PSO-30**	**PSO-60**	**MDA**
20	1.112	0.935	0.974	0.917	0.972	0.903	0.789	0.785	0.740	0.785	1.039	0.836	0.746	0.734	0.727
40	1.085	1.022	1.124	0.987	0.943	1.041	0.888	0.887	0.779	0.843	0.994	0.889	0.751	0.773	0.783
80	1.389	1.230	1.349	1.119	1.146	1.222	1.038	1.128	0.955	1.012	1.267	1.084	0.973	0.987	1.019
120	1.326	1.202	1.103	1.111	1.170	1.318	1.137	1.240	1.019	1.071	1.210	1.098	0.944	0.942	0.992
150	1.468	1.370	1.295	1.173	1.227	1.425	1.314	1.387	1.281	1.428	1.293	1.271	1.102	1.130	1.229
Min.	1.085	0.935	0.974	0.917	0.943	0.903	0.789	0.785	0.740	0.785	0.994	0.836	0.746	0.734	0.775
Max.	1.468	1.370	1.349	1.173	1.227	1.425	1.314	1.387	1.281	1.428	1.293	1.271	1.102	1.130	1.298
Ave.	1.276	1.152	1.169	1.061	1.092	1.182	1.033	1.085	0.955	1.028	1.161	1.035	0.903	0.897	0.950

5.3 Performance Comparison of Algorithms using Non-parametric Tests

The main idea of using the non-parametric tests is that they can deal with probabilistic and non-probabilistic methods without any limitation. In this section, non-parametric test results are presented and examined for comparing the applied meta-heuristic algorithms. In order to achieve the test results in Table 4, the Friedman, Friedman Aligned and Quade non-parametric tests are applied to the estimated results shown in Figs. 6 and 7. The purpose of using Friedman, Friedman Aligned and Quade non-parametric tests is to determine whether there are significant differences among the algorithms considered over given sets of data. These tests obtain the ranks of the algorithms for each individual data set, i.e., the best performing algorithm receives the rank of 1, the second best rank 2, etc. The equations and further details of the non-parametric procedures of Friedman, Friedman Aligned and Quade can be found in [52-55]. Furthermore, statistical analysis of the results of experiments was performed using the available software [2] and the open source JAVA program calculates multiple comparison procedures: the Friedman, Iman–Davenport, Bonferroni–Dunn, Holm, Hochberg, Holland, Rom,Finner, Li, Shaffer, and Bergamnn-Hommel tests as well as adjusted *p*-values [56-64].

Table 4 depicts the average ranks computed using Friedman, Friedman Aligned and Quade non-parametric tests. Based on the results, PSO-60 is the best performing algorithm of the comparison, with average rank of 1.0, 3.2, and 1.0 for the Friedman, Friedman Aligned, and Quade tests, respectively. The *p*-values computed through the statistics of each of the tests considered (18.560, 4.028, and 13.418). The Iman Davenport statistic and *p*-value are computed $9.588*10^{-4}$, 0.4021 and $5.528*10^{-5}$, respectively.

Table 4
Average rankings of the algorithms using the non-parametric statistical procedure, statistics and *p*-values.

Algorithm	Friedman	Friedman Aligned	Quade
GA-30	5.00	22.99	4.99
GA-60	3.80	17.20	3.70
PSO-30	3.00	13.20	3.23
PSO-60	1.00	3.20	1.00
MDA	2.20	8.40	2.06
Statistics	18,560	4.028	13.418
p-value	$9.588*10^{-4}$	0.4021	$5.528*10^{-5}$

[2] http://sci2s.ugr.es/sicidm

In the second statistical analysis tests, multiple comparison post-hoc procedures (eight) are used to compare the control algorithm PSO-60 with the rest of algorithms. The results are shown by computing *p*-values for each comparison. Tables 5-7 show the *p*-values obtained, using the ranks computed by the Friedman, Friedman Aligned, and Quade tests, respectively [52-55]. Based on the computed results, all tests show significant improvements of the PSO-60 over GA-30, PSO-30, GA-60 and MDA for all the post-hoc procedures considered. Besides this, the Li's procedure is the most conclusive to give the clearest results for reaching the lowest *p*-values in the comparisons.

Table 5

Adjusted *p*-values for Friedman (PSO-60 is the control method).

Algorithm	Unadjusted	Bonferroni	Holm	Hochberg/ Hommel /Holland	Rom	Finner	Li
GA-30	6.33*10-5	$2.53*10^{-4}$	$2.53*10^{-4}$	$2.53*10^{-4}$	$2.41*10^{-4}$	$2.53*10^{-4}$	$8.27*10^{-5}$
GA-60	0.00511	0.02044	0.01533	0.01533	0.01525	0.01019	0.00659
PSO-30	0.04550	0.18201	0.09101	0.09101	0.08891	0.06020	0.05580
MDA	0.23013	0.92055	0.23013	0.23013	0.23013	0.23013	0.23011

Table 6

Adjusted *p*-values for Friedman Aligned (PSO-60 is the control method).

Algorithm	Unadjusted	Bonferroni	Holm	Hochberg/ Hommel /Holland	Rom	Finner	Li
GA-30	$2.11*10^{-5}$	$8.41*10^{-5}$	$8.41*10^{-5}$	$8.41*10^{-5}$	$8.01*10^{-5}$	$8.41*10^{-5}$	$2.85*10^{-5}$
GA-60	0.00263	0.01053	0.00789	0.00789	0.00787	0.00525	0.00356
PSO-30	0.03168	0.12675	0.06337	0.06337	0.06236	0.04202	0.04127
MDA	0.26393	1.05573	0.26393	0.26393	0.26393	0.26393	0.26392

Table 7

Adjusted *p*-values for Quade (PSO-60 is the control method).

Algorithm	Unadjusted	Bonferroni	Holm	Hochberg/ Hommel /Holland	Rom	Finner	Li
GA-30	0.01051	0.04206	0.04206	0.04206	0.04011	0.04140	0.02041
GA-60	0.08418	0.33672	0.25254	0.25254	0.25254	0.16127	0.14289
PSO-30	0.15316	0.61265	0.30632	0.30632	0.30632	0.19881	0.23274
MDA	0.49508	1.98034	0.49508	0.49508	0.499508	0.49508	0.49054

Table 8 presents 10 hypotheses of equality among the five algorithms and the *p*-values achieved. Using different levels of significance, i.e. $\alpha = 0.05$ and $\alpha = 0.1$, first three hypotheses are rejected by all the post-hoc procedures. For instance, let us compare PSO-60 vs. GA-30 by Nemenyi, Holm, Shaffer and Bergmann procedures [65-67] and the *p*-values are estimated for each post-hoc procedure which are less than both $\alpha = 0.05$ and $\alpha = 0.1$. Therefore, we reject the hypothesis for the PSO-60 vs. GA-30. The same process is applied for all hypotheses so first three hypotheses are rejected. On the other hand, the rest of hypotheses are not rejected because the computed *p*-values by Nemenyi, Holm, Shaffer and Bergmann procedures are greater than both $\alpha = 0.05$ and $\alpha = 0.1$. Consequently, these hypotheses show the improvement of PSO-60 over GA-30 and GA-60, and that of MDA over GA-30.

Table 8

Adjusted *p*-values for tests for multiple comparisons among the all methods.

Index	Hypothesis	Unadjusted	Nemenyi [65]	Holm[58]	Shaffer[66]	Bergmann[67]
1	**PSO-60 vs. GA-30**	**$6.33*10^{-5}$**	**$6.33*10^{-4}$**	**$6.33*10^{-4}$**	**$6.33*10^{-4}$**	**$6.33*10^{-4}$**
2	**GA-30 vs. MDA**	**0.00511**	**0.04710**	**0.04599**	**0.03066**	**0.03066**
3	**PSO-60 vs. GA-60.**	**0.00511**	**0.04710**	**0.04599**	**0.03066**	**0.03066**
4	GA-30 vs. PSO-30	0.05551	0.44510	0.31850	0.27300	0.18200
5	PSO-30 vs.PSO-60	0.05551	0.44510	0.31850	0.27300	0.18200
6	GA-60 vs. MDA	0.10959	1.08598	0.54799	0.43839	0.32879
7	GA-30 vs.GA-60	0.23013	2.28139	0.92055	0.92055	0.46027
8	PSO-60 vs. MDA	0.23013	2.28139	0.92055	0.92055	0.46027
9	GA-60 vs.PSO-30	0.42371	4.22717	0.92055	0.92055	0.46027
10	PSO-30 vs. MDA	0.42371	4.22717	0.92055	0.92055	0.46027

5.4 Performance of Generated Descriptions

After finding the distinct colored optimal paths in the multi-objective networks, the problem of generating descriptions with achievable bit rates needs to be resolved. This problem was addressed separately for the source and each sink node because descriptions with different bit rates will be generated according to the capacity of the routings and number of routings which reach the destinations. For example, when one colored path reaches the first sink and two reach the second sink, the description can be divided by two at the source node for delivery to the second sink because even if one description fails to reach the destination, the other description may arrive. However, the same solution cannot be offered for sink one because if a packet is lost over the path, the quality of received image will be reduced at the destination. To avoid this or decrease the effect of the loss at the sink one, interpolation methods have been applied for SDC [27].

The rates of descriptions are based on the estimated capacity or the available bandwidth of each optimal path. In order to estimate required capacity of the optimal paths, the constraint ε has been used in the meta-heuristics employed. Suppose four different optimal paths are determined between a source and sink node with available bandwidths. In this case, four descriptions with achievable bit rates will be generated according to the available bandwidths. However, the number of colored paths obtained between source and destination nodes may be different based on the constraint ε when routings are optimized by GA and PSO. This means that the creation of descriptions is based on the number of optimal paths obtained and the capacities of each optimal path. The number of optimal colored paths reaching each sink node is determined by using following constraint values $\varepsilon = 32{,}64{,}96$ and 128 kB, and tabulated in the Tables 9-12, respectively.

Table 9

Number of paths obtained based on $\varepsilon = 32kB$

Net.	n, m, sinks	# of paths between			
		$s\text{-}t_1$	$s\text{-}t_2$	$s\text{-}t_3$	$s\text{-}t_4$
1	20, 86, 4	3	4	3	1
2	40, 183, 4	4	3	5	3
3	80, 324, 4	5	3	5	4
4	120,567 , 4	3	6	4	6
5	150, 953, 4	6	4	5	7

Table 10

Number of paths obtained based on $\varepsilon = 64\ kB$

Net.	n, m, sinks	# of paths between			
		$s\text{-}t_1$	$s\text{-}t_2$	$s\text{-}t_3$	$s\text{-}t_4$
1	20, 86, 4	2	3	3	1
2	40, 183, 4	2	2	3	2
3	80, 324, 4	3	2	3	2
4	120,567 , 4	2	5	3	4
5	150, 953, 4	4	2	4	5

Table 11
Number of paths obtained based on $\varepsilon = 96\ kB$

Net.	n, m, sinks	# of paths between			
		s-t_1	s-t_2	s-t_3	s-t_4
1	20, 86, 4	2	2	2	1
2	40, 183, 4	2	2	3	1
3	80, 324, 4	3	1	3	2
4	120,567 , 4	2	3	2	4
5	150, 953, 4	3	2	2	3

Table 12
Number of paths obtained based on $\varepsilon = 128kB$

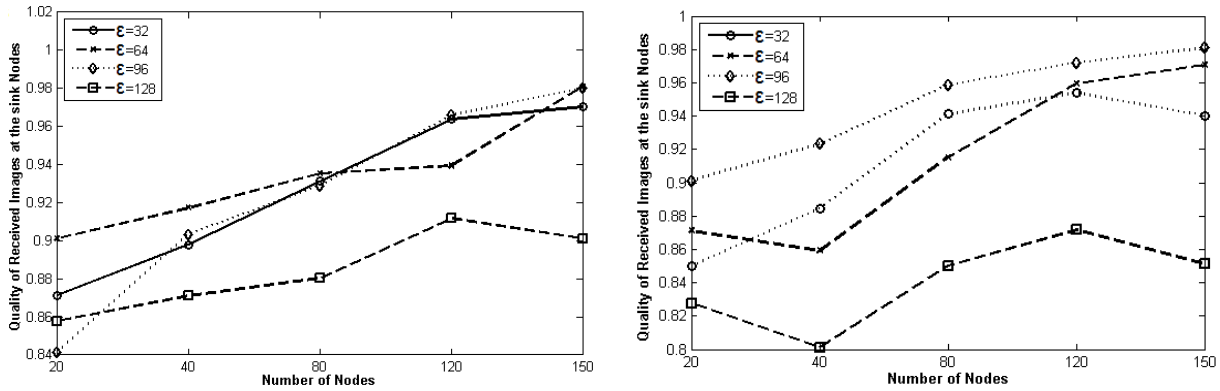
Net.	n, m, sinks	# of paths between			
		s-t_1	s-t_2	s-t_3	s-t_4
1	20, 86, 4	1	1	1	1
2	40, 183, 4	2	1	1	1
3	80, 324, 4	1	1	2	1
4	120,567 , 4	1	2	2	2
5	150, 953, 4	2	1	1	2

Here, the color conversion and MDC generation algorithm shown in Fig. 1 has been tested by employing a color image of dimension 512x512 pixels with three description levels of the decomposition for the wavelet transform. Multiple descriptions were generated by applying Multiple Description Scalar Quantization (MDSQ) to the wavelet coefficients of the colour image models. Experiments have been performed on four popular images: “Castle”, “Lena”, “Baboon”, and “Aircraft” as shown in Fig. 8, to test the overall system. In addition, each sink node demands a different image from the source node; therefore, each color image was simultaneously transmitted to different sink nodes through the distinct colored optimal paths.

Fig. 8. Test Images used in the experiments.

Different descriptions of the generated image (D_1^t, D_2^t, ..., D_k^t) of an MD coded source can be transmitted to the destination node t via different paths. As discussed in the previous section, PSO and GA implementations are giving superior cost results to find the optimal paths in different network models. Image transmission was analyzed in the networks optimized by the third fitness function because it balances the cost values of packet loss rate and path length on the links. Furthermore, different values of constraint ε were employed to obtain the required bandwidth of each path.

Descriptions generated with achievable bit rates at the source node were simultaneously sent through the optimal lossy paths to the clients. Figs. 9 (a) and (b) show how the descriptions are affected by lossy paths in the networks. To reveal the trends, CC was selected to measure the quality results since this is commonly employed in image processing. After performing 10 simulation runs for each network model, the quality of received images at all sink nodes was ascertained and the average qualities are illustrated in Figs. 9 (a) and (b). From these results, the best quality of transmitted image is received at ε= 64 kB and ε= 96 kB. These results also clarify that the transmitted SDC causes more received image distortion. Therefore, poorer quality images are received at ε= 128 kB even if there is high bandwidth to transmit the data. The reason is because fewer optimal paths are determined at ε= 128 kB and SDCs are mostly transmitted as tabulated in Table 10.

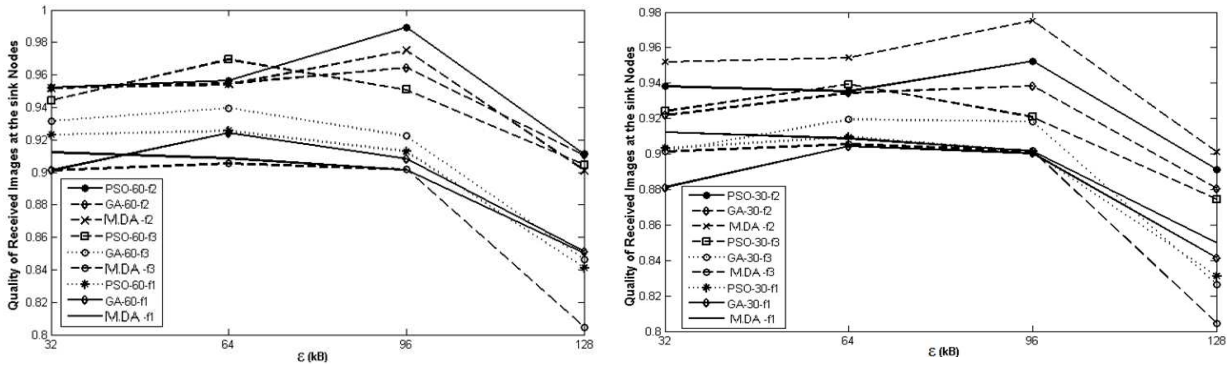


(a) (b)

Fig. 9. Comparison results of quality of received images in CC. (a) Obtained quality results in the optimized networks by PSO -60. (b) Obtained quality results in the optimized networks by GA -60.

5.5 Reliable Transmission Optimization for Image Quality Enhancement

Although multipath routing can increase the reliability of transmission, using too many paths may increase data redundancy and energy consumption. To provide the reliable transmission in a given lossy network, the meta-heuristic algorithms are employed to find the optimal paths as discussed in previous section. The number of obtained optimal paths are based on the demanding bandwidth as tabulated in Tables 9-12. For instance, three MDCs are created with available bit rates according to the number of obtained paths shown in second row and third column of Table 9. To analyze and understand the performance of MDCs in the optimized networks, qualities of received images are estimated by using the equation of correlated coefficients (CC).



(a) (b)

Fig. 10. Comparison results for the average qualities of received images which are transmitted through the optimized network models by using (a) GA-60, PSO-60 and MDA, (b) GA-30, PSO-30 and MDA.

Based on a corresponding constraint value of ε, descriptions are generated and transmitted over the obtained lossy paths from a specific source to destination nodes for all optimized lossy networks. Thereafter, the average quality of received images is estimated for the corresponding value of ε. Figs. 10 (a) and (b) illustrate the average quality of received images in order to provide reliable multimedia transmission. Among the three fitness functions, f_2 provides the highest quality received images at the clients when $\varepsilon = 96\,kB$ is used,

whereas for the other fitness functions, the corresponding value is $\varepsilon = 64\ kB$. The results indicate that using fitness functions f_1 and f_3 with $\varepsilon = 64\ kB$ and f_2 with $\varepsilon = 96\ kB$, will avoid redundancy in data transmission and reception, thus, decreasing energy consumption of network. Besides this, transmitting MDCs over optimized paths improves the reliable multimedia transmission in lossy networks.

In order to understand and analyze the results statistically, we have applied Friedman, Friedman Aligned and Quade non-parametric tests to the data shown in Figs. 10 (a) and (b). Table 13 illustrates the average ranks computed using Friedman, Friedman Aligned and Quade non-parametric tests [52-55]. It compares the average ranks of five different implementations, i.e. GA-30, GA-60, PSO-30, PSO-60 and MDA using with three different fitness functions, i. e. shortest path (SP), minimum packet loss (MPL) and optimal multipath (OMP). Based on the results, PSO-60 implemented with the minimum packet loss fitness function (PSO-60-MPL) gives the best performance, with the average rank of 1.25, 6.25, and 1.10 for the Friedman, Friedman Aligned, and Quade tests, respectively. The *p*-values computed through the statistics of each of the tests considered ($7.19*10^{-6}$, 0.9968, and $5.03*10^{-8}$). However, the worst average ranking results are computed for the GA-30-SP, MDA-OMP and PSO-30-SP.

Table 13
Average rankings of the algorithms using the non-parametric statistical procedure, statistics and *p*-values.

Algorithm	Friedman	Friedman Aligned	Quade
PSO-60-SP	10.00	38.00	10.60
GA-60-SP	11.00	41.00	10.61
PSO-30-SP	12.50	50.50	12.80
GA-30-SP	14.00	54.00	13.40
MDA -SP	11.25	44.75	10.79
PSO-60-MPL	1.25	6.25	1.10
GA-60-MPL	2.75	9.00	2.59
PSO-30-MPL	5.25	18.75	4.90
GA-30- MPL	7.25	25.25	6.80
MDA- MPL	3.00	9.00	3.00
PSO-60-OMP	3.25	11.25	3.599
GA-60- OMP	7.00	29.75	7.80
PSO-30- OMP	7.00	27.00	7.20
GA-30- OMP	11.50	44.00	11.80
MDA - OMP	13.00	49.00	13.00
Statistics	49.575	3.742	8.182
p-value	$7.19*10^{-6}$	0.9968	$5.03*10^{-8}$

In the second statistical analysis tests, multiple comparison post-hoc procedures (eight) are used to compare the control algorithm PSO-60-OMP with the rest of algorithms. The results are shown by computing *p*-values for each comparison. Tables 14-16 show the *p*-values obtained, using the ranks computed by the Friedman, Friedman

Aligned, and Quade tests, respectively. Based on the computed results, all tests show significant improvements of the PSO-60-MPL over the rest of algorithms for all the post-hoc procedures considered.

Table 14

Adjusted *p*-values for Friedman (PSO-60-MPL is the control method).

Algorithm	Unadjusted	Bonferroni	Holm	Hochberg/ Hommel	Holland	Rom	Finner	Li
GA-30-SP	$5.53*10^{-5}$	$7.74*10^{-4}$	$7.74*10^{-4}$	$7.74*10^{-4}$	$7.74*10^{-4}$	$7.36*10^{-4}$	$7.74*10^{-4}$	$1.51*10^{-4}$
MDA-OMP	$2.02*10^{-4}$	0.0028	0.0026	0.0026	0.0026	0.0025	0.0014	$5.55*10^{-4}$
PSO-30-SP	$3.74*10^{-4}$	0.0052	0.0044	0.0044	0.0044	0.0042	0.0017	0.0010
GA-30-OMP	0.0011	0.0166	0.0130	0.0130	0.0130	0.0124	0.0041	0.0032
MDA -SP	0.0015	0.0219	0.0156	0.0156	0.0155	0.0148	0.0043	0.0042
GA-60-SP	0.0021	0.0286	0.0184	0.0184	0.0182	0.0175	0.0047	0.0055
PSO-60-SP	0.0056	0.0792	0.0452	0.0452	0.0443	0.0430	0.0112	0.0152
GA-30-MPL	0.0577	0.8089	0.4044	0.3450	0.3407	0.3281	0.0989	0.1367
GA-60-OMP	0.0690	0.9662	0.4141	0.3450	0.3488	0.3281	0.1052	0.1591
PSO-30-OMP	0.0690	0.9662	0.4141	0.3450	0.3488	0.3281	0.1052	0.1591
PSO-30-MPL	0.2059	2.8826	0.8236	0.6352	0.6023	0.6352	0.2542	0.3608
PSO-60-OMP	0.5271	7.3792	1.5812	0.6352	0.8942	0.6352	0.5825	0.5910
MDA -MPL	0.5799	8.1198	1.5812	0.6352	0.8942	0.6352	0.6071	0.6139
GA-60-MPL	0.6352	8.8935	1.5812	0.6352	0.8942	0.6352	0.6352	0.6352

Table 15

Adjusted *p*-values for Friedman Aligned (PSO-60-MPL is the control method).

Algorithm	Unadjusted	Bonferroni	Holm	Hochberg/ Hommel	Holland	Rom	Finner	Li
GA-30-SP	$1.10*10^{-4}$	0.0015	0.0015	0.0015	0.0015	0.0014	0.0015	$6.25*10^{-4}$
PSO-30-SP	$3.39*10^{-4}$	0.0047	0.0044	0.0044	0.0044	0.0041	0.00237	0.0019
MDA -OMP	$5.36*10^{-4}$	0.0075	0.0064	0.0064	0.0064	0.0061	0.0025	0.0030
MDA -SP	0.0018	0.0255	0.0201	0.0201	0.0198	0.0190	0.0063	0.0102
GA-30-OMP	0.0022	0.0313	0.0223	0.0223	0.0221	0.0212	0.0063	0.0125
GA-60-SP	0.0048	0.0685	0.0440	0.0440	0.0431	0.0418	0.0113	0.0270
PSO-60-SP	0.0101	0.1419	0.0811	0.0811	0.0782	0.0771	0.0202	0.0544
GA-60-OMP	0.0570	0.7986	0.3993	0.3993	0.3371	0.3796	0.0977	0.2445
PSO-30-OMP	0.0929	1.3006	0.5574	0.5574	0.4429	0.5300	0.1407	0.345
GA-30-MPL	0.1239	1.7347	0.6195	0.6195	0.4838	0.5891	0.1690	0.4128
PSO-30-MPL	0.3114	4.3600	1.2457	0.8237	0.7752	0.8237	0.3780	0.6386
PSO-60-OMP	0.6851	9.5978	2.056	0.8237	0.9689	0.8237	0.7407	0.7955
GA-60-MPL	0.8237	11.532	2.056	0.8237	0.9689	0.8237	0.8458	0.8237
MDA -MPL	0.8237	11.532	2.056	0.8237	0.9689	0.8237	0.8458	0.8237

Table 16

Adjusted *p*-values for Quade (PSO-60- MPL is the control method).

Algorithm	Unadjusted	Bonferroni	Holm	Hochberg/ Hommel	Holland	Rom	Finner	Li
GA-30-SP	0.1795	2.5141	2.5141	0.8699	0.9374	0.8699	0.9374	0.5801
MDA -OMP	0.1941	2.71811	2.5239	0.8699	0.9395	0.8699	0.9374	0.5989
PSO-30-SP	0.2017	2.8245	2.5239	0.8699	0.9395	0.8699	0.9374	0.6081
GA-30-OMP	0.2430	3.4023	2.6732	0.8699	0.9532	0.8699	0.9374	0.6514
MDA -SP	0.2898	4.0584	2.8989	0.8699	0.9673	0.8699	0.9374	0.6903
GA-60-SP	0.2999	4.1993	2.8989	0.8699	0.9673	0.8699	0.9374	0.6976
PSO-60-SP	0.2999	4.1993	2.8989	0.8699	0.9673	0.8699	0.9374	0.6976
GA-60-OMP	0.4647	6.5066	3.2533	0.8699	0.9874	0.8699	0.9374	0.7814
PSO-30-OMP	0.5056	7.0796	3.2533	0.8699	0.9874	0.8699	0.9374	0.7954
GA-30-MPL	0.5339	7.4759	3.2533	0.8699	0.9874	0.8699	0.9374	0.8042
PSO-30-MPL	0.6784	9.4979	3.2533	0.8699	0.9893	0.8699	0.9374	0.8391
PSO-60-OMP	0.7850	10.991	3.2533	0.8699	0.9893	0.8699	0.9374	0.8579
MDA -MPL	0.8357	11.700	3.2533	0.8699	0.9893	0.8699	0.9374	0.8653
GA-60-MPL	0.8699	12.179	3.2533	0.8699	0.9893	0.8699	0.9374	0.8699

In order to compare the combinations of algorithms with different fitness functions used, 105 hypotheses of equality have been created among the 15 different approaches. Table 17 presents the first 8 hypotheses with the *p*-values achieved. The first four and six hypotheses are rejected by all the post-hoc procedures for the levels of significance $\alpha = 0.05$ and $\alpha = 0.1$, respectively. However, the rest of hypotheses are not rejected by all the post-hoc procedures. Based on the level of significance $\alpha = 0.1$, the first six hypotheses show the improvement of PSO-60-MPL over GA-30-SP, MDA-OMP and PSO-30-SP, that of GA-60-MPL over GA-30-SP, that of MDA-MPL over GA-30-SP, and that of PSO-60-OMP over GA-30-SP. According to the level of significance $\alpha = 0.05$, the first four hypotheses show the improvement of PSO-60-MPL over GA-30-SP, MDA-OMP and PSO-30-SP, and that of GA-60-MPL over GA-30-SP.

Table 17

Adjusted *p*-values for tests for multiple comparisons among all the methods.

Index	Hypothesis	Unadjusted	Nemenyi[65]	Holm[58]	Shaffer[66]
1	**PSO-60-MPL vs. GA-30-SP.**	**$5.53*10^{-5}$**	**0.0058**	**0.0058**	**0.0058**
2	**PSO-60-MPL vs. MDA-OMP**	**$2.02*10^{-4}$**	**0.0212**	**0.0210**	**0.0184**
3	**PSO-30-SP vs. PSO-60-MPL**	**$3.74*10^{-4}$**	**0.0393**	**0.0385**	**0.0341**
4	**GA-30-SP vs. GA-60-MPL**	**$3.74*10^{-4}$**	**0.0393**	**0.0385**	**0.0341**
5	**GA-30-SP vs. MDA -MPL**	**$5.04*10^{-4}$**	**0.0529**	**0.0509**	**0.0503**
6	**GA-30-SP vs. PSO-60-OMP**	**$6.75*10^{-4}$**	**0.0708**	**0.0675**	**0.0614**
7	PSO-60-MPL vs. GA-30-OMP	0.0011	0.1249	0.1177	0.1082
8	GA-60-MPL vs. MDA-OMP	0.0011	0.12492	0.1177	0.1082

Fig. 11 demonstrates the effect of the information losses to the RMSE and CC in a lossy network. To estimate

the quality results and show the effect of bit rate change, a compressed image with different bit rates between 16-128 kB was sent from source to a sink node. To clarify the simulation results, each compressed image is sent 40 times. The average qualities of RMSE and CC of received images are used for demonstration. In Fig. 11, the upper plane depicts the average qualities of RMSE and the lower plane illustrates the average qualities of CC. In the upper plane, it shows that when some of the packets have been lost, the slope is positive and the quality of the image has been decreased. The lower plane shows that when CC has positive slope, the quality of the image has been increased.

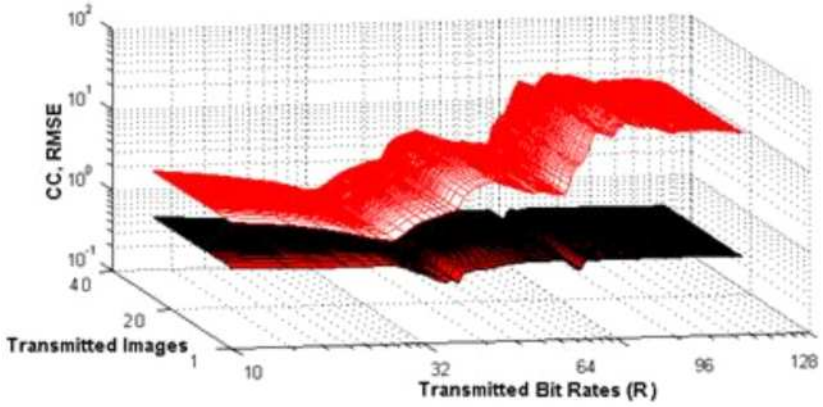


Fig. 11. Quality estimation of received color images at a sink node and demonstrating the effect of bit rate to the image quality in the lossy network.

6. Conclusion

In this paper, we have presented optimal multipath selection by using Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) collaborating with the Rainbow Network Flow (RNF) method and the generation of MDCs with achievable rates. For maximizing the received MDCs, the GA and PSO are applied to find optimal routings and RNF is used to maximize the routing efficiency between source and clients. To find the optimal multipath in multi-objective network models efficiently, a new multi-objective fitness function is defined based on the three different cost types of the links and compared with two single objective fitness functions when they are employed in the GA, PSO and multipath Dijkstra algorithm (MDA). The obtained results indicate that proposed multi-objective fitness function provides superior trade-off between the multi-cost values over the network links to obtain the optimal multipath since other fitness functions are well to optimize only one cost value over the links. GA and PSO algorithms using with two different population sizes are compared with MDA to find the optimal multipath for each fitness function. After finding optimal multipath, the number of MDCs are generated based on the number of optimal paths and amount of available bandwidth by using the wavelet based multiple description scalar quantization. The generated MDCs are sent through the optimal multipath to compare the performances of algorithms employed with three different fitness functions for providing reliable multimedia transmission. PSO performs better than GA and MDA for finding efficient routing paths and providing reliable multimedia transmission in lossy networks. Consequently, using PSO with the proposed fitness function solves the multi-

objective network routing problem efficiently and this solution provides maximal performance of the usage of routings so that each description can be transmitted through the optimal routings.

Acknowledgment

The first author is thankful to EPSRC for funding the part of project.

References

[1] K. Goyal, "Multiple Description Coding: Compression Meets The Network," IEEE Signal Processing Magazine, vol. 18, no. 5, pp. 74-93, 2001.

[2] D. Jurca and P. Frossard, "Distributed media rate allocation in multi-path networks," Signal Processing: Image Communication, vol. 23, no. 10, pp. 754–768, Nov 2005.

[3] S .D. Servetto, K. Ramachandran, V.Vaishampayan and K. Nahrstedt, "Multiple description wavelet based image coding," IEEE Trans. on Image Processing , vol. 9, no. 5, pp 818-826, 2000.

[4] Y. Wang, M. T. Orchard, and A. R. Reibman, "Multiple description coding for noisy channels by pairwise correlating transforms," in Proc. Workshop Multimedia Signal Process., 1997, pp. 419–424.

[5] A. Said and W. Pearlman, "A New, fast and Efficient Image Codec based on Set Partitioning in Hierarchical Trees," IEEE Trans. on Circuits and Systems for Video technology, vol. 6,no. 3, pp. 243 – 250, June 1996.

[6] V. A. Vaishampayan, "Design of multiple description scalar quantizers," IEEE Trans. Inf. Theory, vol. 39, no. 3, pp. 821–834, Mar. 1993.

[7] M. Fleming and M. Effros, "Generalized multiple description vector quantization," in Proc. Data Compression Conf., 1999, pp. 3–12.

[8] N. Zhang, Y. Lu, F. Wu, X. Wu, and B. Yin, "Efficient multiple description image coding using directional lifting-based transform," IEEE Trans. on Circuit. Syst. Video Technol., vol. 18, no. 5, pp. 646–656, May 2008.

[9] U. Black, IP Routing Protocols, RIP, OSPF, BGP, PNNI & Cisco routing protocols, Prentice Hall, 2000.

[10] W. A. Chang and R. S. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations," IEEE Trans. Evol. Comput., vol. 6, no. 6, pp. 566–579, Dec. 2002.

[11] A. W. Mohemmed , N. C. Sahoo and T. K. Geok, "Solving shortest path problem using particle swarm optimization," Applied Soft Computing, vol.8, no.4, pp. 1643-1653, Sep. 2008.

[12] N. Shi, "K constrained shortest path problem," IEEE Transactions on Automation Science and Engineering, vol. 7, no. 1, pp. 15 –23, Jan. 2010.

[13] Y. Xiao, K. Thulasiraman, and G. Xue, "Constrained shortest linkdisjoint paths selection: a network programming based approach," IEEE Transactions on Circuits and Systems, vol. 53, no. 5, pp. 1174 – 1187, May. 2006.

[14] X. Sun, X. You, and S. Liu, "Multi-objective ant colony optimization algorithm for shortest route problem," in Proc. International Conference on Machine Vision and Human-Machine Interface, Apr. 2010, pp. 796 –798.

[15] X. Wu, B. Ma, and N. Sarshar, "Rainbow network flow of multiple description codes," IEEE Transactions on Information Theory, vol. 54, no. 10, pp. 4565 –4574, Oct. 2008.

[16] X. Wu, B. Ma, N. Sarshar, "Rainbow Network Flow of Multiple Description Codes," IEEE Trans. Inf. Theory, vol. 54, no. 10, pp. 4565 – 4574, Oct. 2008.

[17] G. Retvari, J. J.Biro, T. Cinkler, "On Shortest Path Representation," IEEE/ACM Trans. on Networking, vol.15, no.6, pp.1293-1306, 2007.

[18] G. Zhou, and M. Gen, "Genetic algorithm approach on multi-criteria minimum spanning tree problem," European Journal of Operational Research, vol. 114, no.1, pp. 141-152, Apr. 1999.

[19] Ali C. Begen, Yucel Altunbasak and Ozlem Ergun, "Multi-path selection for multiple description encoded video streaming," in Proc. IEEE Int. Conf. Communications (ICC), Anchorage, AK, May 2003.

[20] C.A. Poynton, A Technical Introduction to Digital Video, Wiley, New York (1996).

[21] C. C. Liu, D. Q. Dai, and H. Yan, "Local discriminant wavelet packet coordinates for face recognition," J. Mach. Learn. Res., vol. 8, pp. 1165–1195, 2007.

[22] A. Vosoughi, A. Vosoughi, and M. B. Shamsollahi, "Nonsubsampled higher-density discrete wavelet transform for image denoising," in Proc. IEEE international Conference on Acoustics, Speech and Signal Processing, Washington, DC, Apr. 2009, pp. 1173-1176.

[23] D. Q. Dai and H. Yan, "Wavelet and face recognition," In Face Recognition, K. Delac and M. Grgic, Eds. Vienna, Austria: I-Tech Edu. Publ., 2007, ch. 4, pp. 59–74.

[24] K. S. Shanmugan and A. M. Breipohl, Random Signals: Detection, Estimation and Data Analysis. Hoboken,

NJ: Wiley, 1988.
[25] T. Frajka and K. Zeger, “Robust packet image transmission by wavelet coefficient dispersement,” in Proc. IEEE international Conference on Acoustics, Speech and Signal Processing, 2001, pp.1745-1748.
[26] S.D. Servetto, K. Ramchandran, V.A. Vaishampayan, K. Nahrstedt , "Multiple description wavelet based image coding," IEEE Transactions on Image Processing, vol.9, no.5, pp.813-826, May 2000.
[27] J. Ridge, F. W. Ware,J. D. Gibson , "Permuted smoothed descriptions and refinement coding for images," IEEE Journal on Selected Areas in Communications, , vol.18, no.6, pp.915-926, June 2000.
[28] R. Venkataramani, G. Kramer, V.K. Goyal,"Multiple description coding with many channels," IEEE Transactions on Information Theory, vol.49, no.9, pp. 2106- 2114, Sep. 2003.
[29] M. Majid, C. Abhayaratne, “Multiple Description Scalar Quantization With Successive Refinement,” in Proc. 17th European Signal Processing Conf. (EUSIPCO 2009), Aug. 2009, pp. 2268-2272.
[30] L. Alparone, S. Baronti, A. Garzelli, and F. Nencini, “A global quality measurement of pan-sharpened multispectral imagery,” IEEE Geo. and Rem.Sensing Letters, vol. 1, no. 4, pp. 313–317, Oct 2004.
[31] X. Wu, B. Ma, N. Sarshar, "Rainbow network problems and multiple description coding," in Proc. IEEE International Symposium on Information Theory (ISIT 2005), Sep. 2005, pp. 268-272.
[32] M. Alasti, K. Sayrafian-Pour,A. Ephremides, and N. Farvardin, "Multiple description coding in networks with congestion problem," IEEE Trans. Infor. Theory, vol.47, no.3, pp.891-902, Mar 2001.
[33] K.-T. Ko and K.-S. Tang, “Using Genetic Algorithms to Design Mesh Networks,” Computer, vol. 30, no. 8, pp. 56-61, Aug. 1997.
[34] Y.-K. Kwok and I. Ahmad, “Efficient scheduling of arbitrary task graphs to multiprocessors using a parallel genetic algorithm,” Journal of Parallel and Distributed Computing , vol. 47, no. 1, pp. 58-77,Nov. 1997.
[35] K.F. Man, K.-S. Tang and S. Kwong, “Genetic algorithms: Concepts and Applications,” IEEE Transactions on Industrial Electronics, vol. 43, no. 5, pp. 519–534, Oct. 1996.
[36] Y.-S. Yen, Y.-K. Chan, H.-C. Chao, J. H. Park, “A genetic algorithm for energy-efficient based multicast routing on MANETs,” Computer Communications, vol. 31, no. 4, pp. 858-869, Mar 2008.
[37] S. Pierre and G. Legault,”A genetic algorithm for designing distributed computer network topologies,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 28, no. 2, pp. 249–257, Apr 1998.
[38] M. Gen, R. Cheng, and D. Wang, “Genetic algorithms for solving shortest path problems,” in Proc. IEEE International Conference on Evolutionary Computation, Apr. 1997, pp. 401 –406.
[39] J. Yi, A. Adnane, S. David,and B. Parrein, “Multipath optimized link state routing for mobile ad hoc networks”, Ad Hoc Networks, vol. 9, no. 1, pp. 28-47, 2011.
[40] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in Proc. IEEE International Conference on Neural Networks, vol. 4, Nov. 1995, pp. 1942 –1948 vol.4.
[41] S. Das, A. Abraham, U. Chakraborty, and A. Konar, “Differential evolution using a neighborhood-based mutation operator,” IEEE Transactions on Evolutionary Computation, vol. 13, no. 3, pp. 526 –553, Jun. 2009.
[42] J. Zhang and A. Sanderson, “Jade: Adaptive differential evolution with optional external archive,” IEEE Transactions on Evolutionary Computation, vol. 13, no. 5, pp. 945 –958, Oct. 2009.
[43] S. Kiranyaz, T. Ince, A. Yildirim, and M. Gabbouj, “Evolutionary artificial neural networks by multi-dimensional particle swarm optimization,” Neural Networks, vol. 22, no. 10, pp. 1448 – 1462, 2009.
[44] T. Ye, S. Kalyanaraman, “A recursive random search algorithm for network parameter optimization,” ACM SIGMETRICS Performance Evaluation Review, vol. 32, no. 3, pp. 44-53, 2005.
[45] A. C. Begen, Y. Altinbasak, , O. Ergun, M. H. Ammar, “Multi-path selection for multiple description video streaming over overlay networks”, Signal Processing: Image Communication, vol. 20, no.1, pp. 39-60, 2005.
[46] E. W. Dijkstra, “A note on two problems in connexion with graphs,” Numerische Mathematik, pp. 269–271 1959.
[47] B. Awerbuch, “Distributed shortest paths algorithms,” in Proc. 21st Annual ACM Symposium on Theory of Computing (STOC), pp. 490 -500, 1989.
[48] R. E. Bellman, “On a routing problem,” Quarterly of Applied Mathematics, pp. 87-90, 1958.
[49] S. Yang, H. Cheng, and F. Wang, “Genetic algorithms with immigrants and memory schemes for dynamic shortest path routing problems in mobile ad hoc networks,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 40, no. 1, pp. 52 –63, Jan. 2010.
[50] G. Yan, “Design of qos anycast network cluster balance based on genetic algorithm,” in Proc. International Conference on Signal Processing Systems, pp. 610-614, 2009.
[51] 12. J. Arabas and S. Kozdrowski, “Applying an evolutionary algorithm to telecommunication network design,” IEEE Transactions on Evolutionary Computation, vol. 5, pp. 309-322, 2001.

[52] D. J. Sheskin, “Handbook of Parametric and Nonparametric Statistical Procedures,” 4th ed., Chapman & Hall/CRC, 2006.
[53] W. Kruskal, W. Wallis, “Use of ranks in one-criterion variance analysis,” Journal of the American Statistical Association,vol. 47, pp. 583–621, 1952.
[54] D. Quade, “Using weighted rankings in the analysis of complete blocks with additive block effects,” Journal of the American Statistical Association, vol. 74, no. 367, pp. 680–683, 1979.
[55] W. Daniel, “Applied Nonparametric Statistics,” 2nd ed., Duxbury Thomson Learning, 2000.
[56] S.Y.P.H. Westfall, “Resampling-based Multiple Testing: Examples and Methods for p-Value Adjustment,” John Wiley and Sons, 2004.
[57] O. Dunn, “Multiple comparisons among means, Journal of the American Statistical Association,” vol. 56, pp. 52–64, 1961.
[58] S. Holm, “A simple sequentially rejective multiple test procedure, Scandinavian Journal of Statistics,” vol. 6, pp. 65–70, 1979.
[59] M.C.B.S. Holland, “An improved sequentially rejective Bonferroni test procedure,” Biometrics, vol. 43, pp. 417–423, 1987.
[60] H. Finner, “On a monotonicity problem in step-down multiple test procedures,” Journal of the American Statistical Association, vol. 88, pp., 920–923, 1993.
[61] Y. Hochberg, “A sharper Bonferroni procedure for multiple tests of significance,” Biometrika, vol. 75, pp. 800–803, 1988.
[62] G. Hommel, “A stagewise rejective multiple test procedure based on a modified Bonferroni test,” Biometrika, vol. 75, pp. 383–386, 1988.
[63] D. Rom, “A sequentially rejective test procedure based on a modified Bonferroni inequality, Biometrika,” vol. 77, pp. 663–665, 1990.
[64] J. Li, “A two-step rejection procedure for testing multiple hypotheses, Journal of Statistical Planning and Inference,” vol. 138, 1521–1527, 2008.
[65] P.B. Nemenyi, “Distribution-free Multiple comparisons,” Master’s thesis, Princeton University, 1963.
[66] J. Shaffer, “Modified sequentially rejective multiple test procedures,” Journal of American Statistical Association, vol. 81, pp. 826–831, 1986.
[67] G. Bergmann, G. Hommel, “Improvements of general multiple test proceduresfor redundant systems of hypotheses,” in: P. Bauer, G. Hommel, E. Sonnemann (Eds.), Multiple Hypotheses Testing, Springer, pp. 100–115, 1988.
[68] Sinalgo: a simulation framework for manets. Availaible on:dcg.ethz.ch/projects/sinalgo.