Genetic Algorithm Approach to Solve the Shortest Disjoint Path Problem for Optimized Rainbow Network Flow of Multiple Description Codes

H Kusetogullari, M S Leeson and E L Hines

University of Warwick

Abstract: The network flow of multiple description codes (MDCs) is considered and the idea of routing as finding paths for a rainbow of colours (rainbow network flow or RNF) incorporated. An optimized RNF of MDCs that utilises a genetic algorithm (GA) is presented. The GA is used to find the shortest disjoint paths between source node and the destination node to optimize the amount of outgoing MDC packets or provide maximal performance of the bandwidth capacity of every single route for image transmission across the network. The GA based method finds the optimum solution and the effects of variations in GA parameters are investigated.

1. Introduction.

Multiple description coding (MDC) breaks a single media stream into a number of substreams (the descriptions), whose packets are routed over different, preferable disjoint paths. MDC provides resilience to network packet loss because when this occurs the presence of multiple streams means that the receiver experiences a loss of quality rather than service interruption. MDC may be applied to many transmission scenarios including speech, audio, image, video, and volumetric data [1].

Given a network graph G = <V,E>, the aim is to deliver good reconstruction quality. In a reliable network, we must maximize the number of received distinct packets; when packet losses are inevitable routing must be such that all of the received packets are useful. Routing is a fundamental engineering task on the Internet, finding a path from a source to destination host. It is a complex problem to understand and implement within networks because of the many potential intermediate destinations a packet might traverse before reaching its destination [2]. In this work, our aim is to find shortest disjoint paths to optimize the amount of outgoing MDC packets. This provides the best bandwidth capacity performance for every single route thus maximizing the number of distinct packets received and hence maximizing the reconstruction quality of an MDC-coded signal.

Suppose that there are k disjoint paths connections from source to destination. In this case, there will be k routes and each packet route may be envisaged as a “colour” [3]. This abstraction leads to application of the Rainbow Network Flow (RNF) algorithm [3] to maximize the number of distinct packets received. Subsequently, a genetic algorithm (GA) may be used to find the shortest disjoint paths between source and destination nodes to appropriately order the outgoing packets for each shortest disjoint path.

Here, we are interested in finding the shortest routing paths to maximize the reconstruction quality of a received MDC-coded signal (image, audio or video) given in the network graph G = <V,E> with a given set of MDC packet subsets. To go from the source to the destination, a packet in this type of network may have to first visit one or more intermediate machines. Often multiple routes, of different lengths, are possible so finding good ones is important in point-to-point networks. In this paper, the single-source network problem is considered as it is better understood than the more demanding multi-source problem.

The rest of the paper is organized as follows: the methodology employed in network optimisation will be explained in section two, the results will be discussed in section three, and the paper will be finalized with conclusions and future work directions.

2. Methodology

The shortest path problem is the problem of finding the minimum-length (cost) path between a given pair of nodes [4]. This problem was first discussed by Dijkstra in 1959 [5] and has been widely researched. The Dijkstra algorithm is considered to be the most efficient method to find the shortest paths between the source and destination nodes. However, it becomes inefficient if the network graph is very large so that in this work, a GA was used to prevent the inefficiencies in the proposed network graph.

Figure 1 shows the block diagram of the optimized RNF of MDCs using a GA. First, one of the available compression and packetization techniques is applied [6] to an image for transmission through the network. Next a GA is applied to find lowest cost path between source and destination nodes. Finally packets on the destination node are combined and converted to produce the received image. If the network graph is reliable, the image will be the same as the original 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 node.

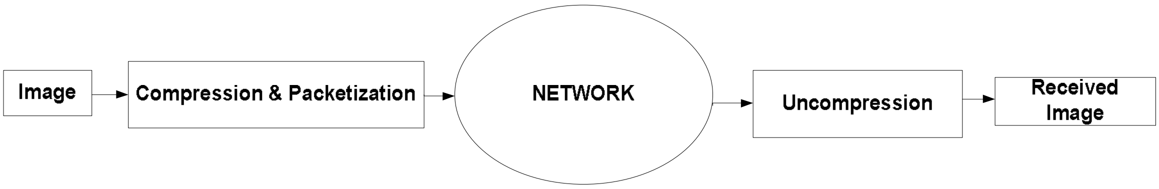


Figure 1. Block Diagram of the whole system.

Each node in the network topology is given a unique integer value index from 1,…….N, where, N is the number of nodes in the network graph. In addition, each node index is a gene of the chromosome for the GA and the number of nodes in the network routing topology may differ from individual to individual which as illustrated in Figure 2.

1	2	3	4	5	6	7	8		N-1	N
1	4	5	9	3	12	-	-		-	-

Figure 2. An example chromosome for a route from node '1' to '12' via nodes 4,5,9,3.

An implementation of a GA begins with a population of (typically random) chromosomes represented as integer numbers. One then evaluates these structures and allocates reproductive opportunities in such a way that those chromosomes which represent a better solution to the target problem are given more chances to reproduce than those chromosomes which are poorer solutions. The quality of a solution is typically defined with respect to the current population. Figure 3 shows the flowchart of the GA.

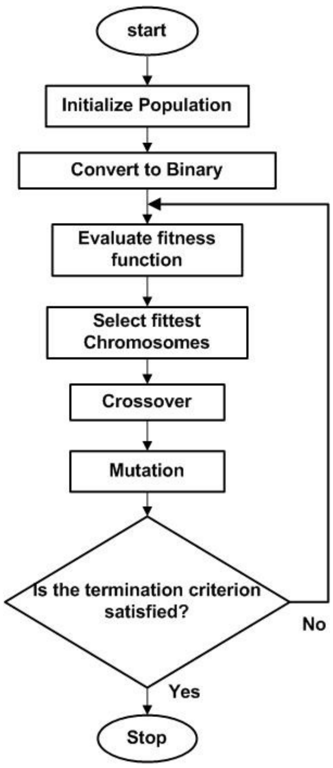


Figure 3. Flowchart of the GA.

The essential GA elements employed comprise:

1) The fitness function, defined here as:

$$\min \sum_{i=1}^{N} w_i \qquad (1)$$

where w_i is the cost of each edge and the fitness value is updated at the end of each generation.

2) Elitism: the retention of a percentage of the population based on fitness, here the top 10% of individuals are retained for the next generation.

3) Population size: the number of individuals there are in each generation. In this study, population sizes of 20, 40, 60 and 80 were utilised to produce the performance comparison shown in Figure 4(a).

4) Crossover operator: a multi point crossover is used to breed new children. Two selected parents are crossed over at every point with equal node indices according to a probability P_c=0.2, 0.4, 0.6 and 0.8 with results shown in Figure 4(b).

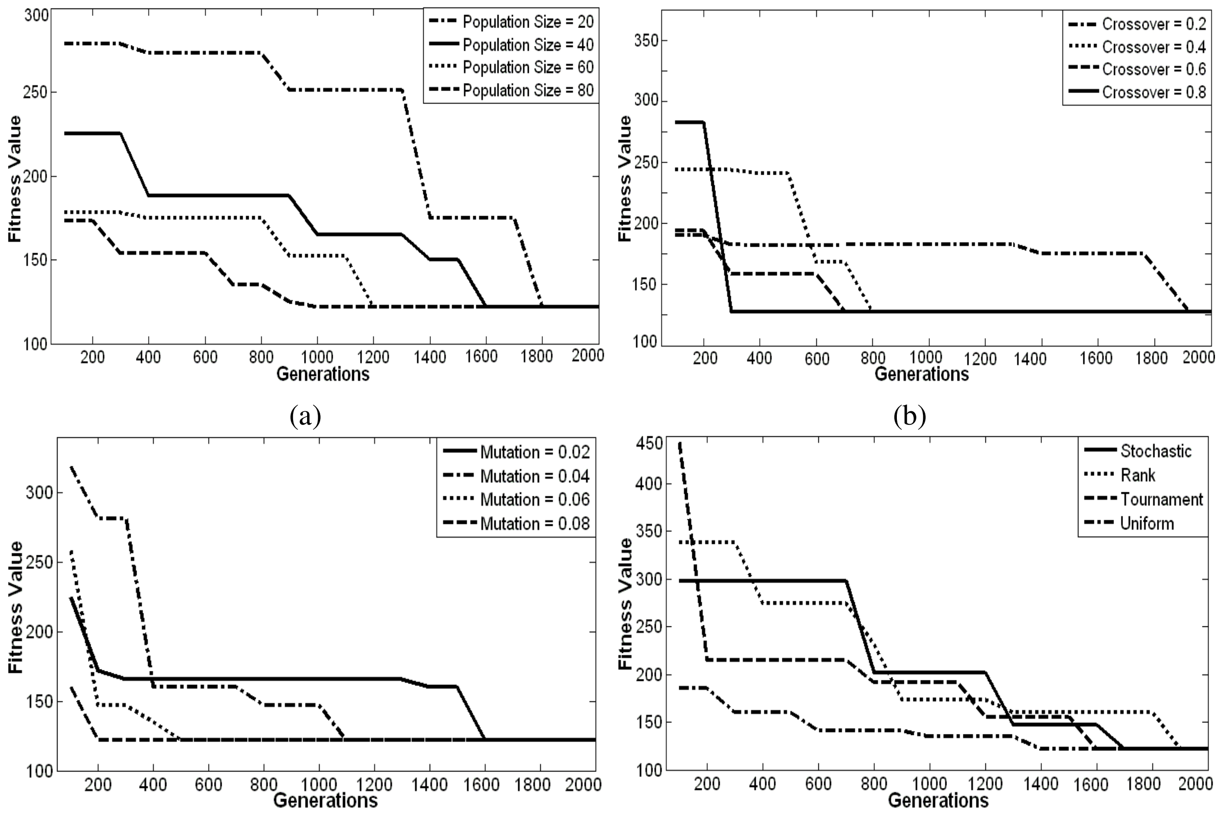
5) Mutation operator: results of mutation probabilities 2%, 4%, 6%, and 8% are illustrated in Figure 4 (c) applied to the population individuals. A random position P is selected in the individual and then a path from P to the destination is regenerated.

6) Selection operator: four methods of choosing parent individuals for crossover are investigated as in Figure 4 (d). These are roulette rank, stochastic, tournament and uniform with the first being employed for the test shown in the previous figures.

7) Termination criteria: extensive (20 or so) repetitions of the individual with best fitness causes algorithm termination.

3. Results and Discussions

A network graph with 20 nodes having edges connecting randomly selected nodes was considered.



(c) (d)

Figure 4. The performance of the genetic algorithm for different feature methods.

The figures shown above were determined by applying different GA features to find one shortest disjoint path between the selected source and destination nodes in the proposed network topology. The problem was small enough for the exact fitness value to be calculated by hand as 122. The network topology based on GA was implemented for four cases; first using population sizes of 20, 40, 60 and 80. Figure 4(a) shows that the performance of the GA improves with an increasing population size as would be expected. In a GA implementation, one of the most important tasks is to breed new children from two individuals. Figure 4(b) shows the performance of crossover using probabilities of P_c=0.2, 0.4, 0.6 and 0.8 with a lower value producing superior convergence. Figure 4(c) shows the performance for mutation probabilities between 0.02 and 0.08, with higher values producing improved performance. Figure 4(d) compares different fitness selection methods.

Using the GA, it is possible to continue to find another set of possible solutions so that the destination can be reached. There may be reasons to ignore the current best path and choose another path, which is similar to the existing one, by ignoring some nodes on the network. The algorithm tries to converge to a solution according to the generated initial population, reducing the chance that the program will get stuck. This would make the program run within a large search space with small space complexity. Sometimes individuals may rapidly come to dominate the population, causing convergence to a local minimum as it is common for many GAs. To avoid this, a larger population with modest crossover rate but relatively high mutation rate will cause rapid convergence to the known optimum solution.

4. Conclusion & Future Work

GAs are flexible and robust as tools for global optimization with variable features such as selection, crossover and mutation making them suitable for many applications. They can deal with combinatorial problems with NP hardness as well as problems with multiple local optima. They are also readily suitable to parallel implementation, which renders them usable in real-time. The primary drawback of GAs results from their flexibility. The design should come up with encoding schemes that allow the GA to take advantage of the underlying building blocks. One has to make sure that the evaluation function assigns meaningful fitness measures to the GA. It is not always clear how the evaluation function can be formulated for the GA to produce an optimal solution. Here a new method to solve the shortest disjoint path problem for optimized rainbow network flow of MDCs using a GA is proposed. This solution aims to achieve an increased maximal performance of the bandwidth capacity of every single route. The experimental results show that this algorithm finds more than one possible solution for a given source and destination. Consequently, the shortest disjoint path can be determined within these experimental results. In future, the number of selected sources and destinations will be increased and the shortest disjoint paths between sources and destinations will be optimized.

Acknowledgments

The first author is thankful to Turgay Celik from University of Warwick for his valuable comments in implementing this work.

References

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

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

[3] X. Wu, Bin Ma, N. Sarshar, "Rainbow Network Problems and Multiple Description Coding", IEEE ISIT05, Pages:268 – 272,2005.

[4] 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.

[5] E. W. Dijkstra, “A note on two problems in connexion with graphs”, Numerische Mathematik, vol. 1, no. 1, pp. 269–271, 1959.

[6] C. W. Lee, C. S. Yang, Y. C. Su, “Adaptive UEP and packet size assignment for scalable video transmission over burst-error channels”, EURASIP Journal on Applied Signal Processing, vol. 2006, no. 1, pp. 256-256,2009.