Optical Packet Switching over Arbitrary Physical Topologies using the Manhattan Street Network: An Evolutionary Approach

Olufemi Komolafe*, David Harle*, David Cotter^
** Dept. of Electronic & Electrical Engineering, Strathclyde University, Glasgow G1 1XW, U.K.*
^ Corning Research Centre, Adastral Park, Martlesham Heath, Ipswich IP5 3RE, U.K.
Tel.: (+44) 141 248 2090. Fax.: (+44) 141 552 4968. Email: femi@comms.eee.strath.ac.uk.

Key words: Optical Packet Switching, Virtual Topology Design, Manhattan Street Network, Genetic Algorithms, Optical Processing

Abstract: Optical packet switching over arbitrary physical topologies typically mandates complex routing schemes and the use of buffers to resolve the likely contentions. However, the relatively immature nature of optical logic devices and the limitations with optical buffering provide significant incentive to reduce the routing complexity and avoid optical domain contentions. This paper examines how the Manhattan Street Network (MSN) and a particular routing scheme may be used to facilitate optical packet switching over arbitrary physical topologies. A novel approach, genetic algorithms (GA), is applied to the problem of deploying the MSN (near) optimally in arbitrary physical topologies. A problem encoding is proposed and different implementations of GA described. The optimum GA parameters are empirically selected and GA is successfully used to deploy the MSN in physical topologies of up to 100 nodes. Favourable results are obtained. GA are also seen to out-perform other heuristics at deploying the MSN in arbitrary physical topologies for optical packet switching.

1. INTRODUCTION

The proliferation of the Internet has provided significant incentive and impetus to develop and implement improved packet switching architectures. It is widely accepted that optical packet switching will be one of the ways the demand for bandwidth is met. One approach to optical packet switching is to assume that the switching may be controlled electronically, although the data packets will remain in the optical domain throughout the switching. However, situations may arise in which it is desirable to control the switching of packets in the optical domain. Optical logic devices will read the packet's header, process the information and switch the packet accordingly [1, 2]. An advantage of optically controlling the switching of packets is that optical processing devices are faster than their electronic counterparts. A disadvantage, however, is the fact that the algorithmic processing undertaken must be extremely simple given the relatively undeveloped nature of optical logic devices. Regardless of whether the header processing is implemented optically or electronically, another important constraint is the need to minimise or avoid optical domain contention due to the non-availability of static random access optical memory devices and the limited scalability of travelling-wave optical buffers. Nevertheless, it is often the case that real-life (hence arbitrary) physical topologies necessitate complex routing schemes and the need for buffers to resolve the likely contentions.

This paper investigates how the Manhattan Street Network (MSN) and a particular routing scheme may be used to realise optical packet switching over arbitrary physical topologies, avoiding complex routing operations and contentions in the optical domain. Section 2 introduces the MSN and the routing scheme. The MSN is deployed as a virtual topology over the arbitrary physical topology. Section 3 presents the fundamentals of the technique used to attempt to deploy the MSN optimally in the physical topologies; genetic algorithms (GA). Section 4 discusses the alternative problem-specific implementations of GA considered in this paper. Section 5 presents the results obtained when selecting the optimum parameters for the different GA, compares the results obtained for each GA implementation, and compares the performance of GA against other heuristics. Section 6 contains the conclusions.

2. THE MANHATTAN STREET NETWORK WITH THE *CLOCKWORK ROUTING* SCHEME

2.1 *Clockwork Routing* Scheme Operation

The Manhattan Street Network [3] (MSN) with the *Clockwork Routing* scheme [4, 5] has been suggested for use in optical packet switched networks for a number of reasons. The routing processing is extremely simple and suitable for optical implementation, no optical domain buffering is required, no resequencing is needed at the destination nodes and throughput is comparable with conventional store-and-forward packet switching [4, 5].

Clockwork Routing requires that all the nodes in an *NxN* MSN be synchronised to a global clock. The timeslots are arranged in a modulo-*N* sequence of frames. Each node consists of a simple 2x2 cross-bar switch. All nodes are in the cross state for the first *N*-1 timeslots in each timeframe, and in the bar state for the last timeslot in the timeframe. Figure 1 shows a 4x4 MSN. By correctly inserting a packet into a timeslot on a particular output link, it is automatically routed to the destination. No additional processing is required at intermediate nodes – these nodes need only determine whether the packet has reached its destination or not. Such a simple "for me or not for me" evaluation may be readily implemented in the optical domain [1, 2]. Optical domain contentions are avoided through the use of peripheral electronic buffers.

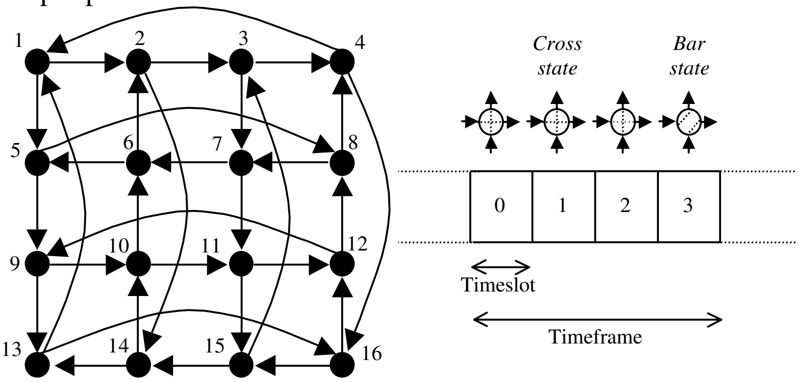

Figure 1. 4x4 MSN

2.2 Deployment of the MSN with Clockwork Routing in Arbitrary Physical Topologies

Outside the local area network, the MSN may be deployed (or embedded) as a virtual topology in WDM networks. Nodes of the MSN are mapped onto those of the physical topology. Links in the MSN are implemented as lightpaths between the corresponding nodes [6, 7]. The length of these lightpaths, in terms of concatenated physical topology hops, will vary according to which MSN nodes are mapped onto which physical topology nodes. A shortest path routing algorithm is used to route the lightpaths over the physical topology. Each node contains wavelength selective devices, in particular an optical cross-connect (OXC), that route and manipulate the lightpaths as desired. Packets are optically switched at the terminal of lightpaths with each node implementing the *Clockwork Routing* scheme.

There are several issues associated with deploying the MSN in arbitrary physical topologies; this paper focuses on one of the main issues - the (near) optimal embedding of the MSN in the physical topology. The virtual topology design problem and several of its sub-problems are known to be *NP*-complete [8] and so heuristics must be adopted to find (near) optimal solutions in reasonable time.

There are up to $N^2!$ ways to deploy an *NxN* MSN in a given physical topology - for a 8x8 MSN this corresponds to up to 1.27x10^{89} different embeddings! The quality of these embeddings will differ significantly. The cost considered in this paper is the mean lightpath length. (Throughout this paper, distance is in terms of concatenated hops over a topology, and not in terms of physical length.) The lightpath length is important for several reasons: it affects the number of wavelengths needed, impacts the dimensions of the wavelength selective devices, indicates the number of OXCs a packet has to traverse between any two adjacent nodes in the MSN and would impact the deployment of optical amplifiers and the consumption of other network resources. Furthermore, lightpath length corresponds to a figure of merit often used in graph embedding theory [9].

Packets will typically need to traverse more than one lightpath between source and destination. The aggregate number of physical hops traversed by a packet corresponds to the number of OXCs crossed. It is desirable to minimise the length of lightpaths and, hence, the aggregate number of OXCs that packets encounter between source and destination. The number of physical hops a packet can traverse is limited by the characteristics of the fibre coupled with distortions such as crosstalk and insertion loss introduced by the OXCs. In [10] it was observed that the aggregate source to destination distance packets encounter is simply the product of the mean lightpath length and the mean inter-nodal distance over the MSN - the latter quantity is known [4, 5] and this paper proposes the novel use of genetic algorithms to minimise the former.

3. GENETIC ALGORITHM THEORY

Genetic algorithms (GA) are a family of adaptive computational models inspired by evolution in nature. The *survival of the fittest* criterion provides evolutionary pressure for the population to develop increasingly fit individuals with time. GA were originally proposed in [11] and although several different nuances have been developed, they all adhere to the same basic principles. The first step in an implementation of GA is to create a randomly generated initial *population*. The *individuals* in the population each represent a possible solution to the problem being tackled. The individuals are normally coded as a binary string. A *fitness* (or cost) function needs to be defined and evaluated for each individual in the population. Individuals are copied from the population into what is known as the *mating pool*. The mating pool will be the same size as the population. However, the number of times an individual is copied into the mating pool is correlated to its fitness. Pairs of individuals from the mating pool are randomly selected to *reproduce*. Reproduction or mating typically yields two new *offspring*. The offspring ought to have inherited attributes from both *parents*. It can be seen that by ensuring that fit individuals appear more frequently in the mating pool, good properties ought to propagate down the *generations*. The offspring produced become the new generation and the whole process in repeated. The process is repeated until ideally all the individuals in the population converge into an extremely fit individual [12, 13]. GA work because of what is termed *implicit parallelism*. Fitness evaluation, selection and reproduction ensure that numerous areas of the search space are sampled simultaneously.

4. APPLICATION OF GENETIC ALGORITHMS TO DEPLOYING THE MSN IN ARBITRARY PHYSICAL TOPOLOGIES

GA have been applied to a plethora of complex mathematical problems [13] and to numerous telecommunications problems [14]. Nevertheless, other heuristics are typically applied to the virtual topology design problem [8]. This paper applies GA to virtual topology embedding, and compares the results obtained with some other approaches which may be adopted.

4.1 Problem Encoding

The problem encoding adopted in this paper is based on approaches taken for the application of genetic algorithms to the (in)famous Travelling Salesman Problem (TSP). The TSP problem is to find the visitation order of a list of cities that the salesman should choose in order to minimise the aggregate distance covered. Several of the issues encountered when attempting to apply GA to the TSP are pertinent when using GA to deploy the MSN in arbitrary physical topologies.

GA may be readily applied to problems that are accurately encodable as a population of individuals represented by binary strings in which the frequency and position of values in individuals is inconsequential. This is evidently not the case with the TSP or the problem of deploying the MSN in arbitrary physical topologies. The search space for the TSP may be considered to be the permutation of the set of cities. Intuitively, the cost or fitness of each individual will be the distance that results from touring the cities in the order that the individual specifies. In the problem encoding adopted in this paper, the position of an element in an individual in the population is indicative of which MSN node is mapped onto which physical topology node. Consider the example of an individual when a 4x4 MSN is to be deployed in a 16-node physical topology:

4 7 6 12 8 1 9 3 13 16 10 14 15 2 11 5

This particular individual implies that node 4 in the MSN is mapped onto node 1 in the physical topology, MSN node 7 is mapped onto node 2 in the physical topology and so on. The fitness or cost of this particular individual is the resulting mean lightpath length when all the MSN nodes are mapped onto the prescribed physical topology nodes. Note that this encoding places a slightly different emphasis compared to the TSP. In the TSP, it is the order of the elements that matter since the tour is cyclic. However, when deploying the MSN in arbitrary physical topologies, it is the absolute position of the elements that is most significant.

Several techniques exist for selecting individuals from the population to be placed in the mating pool so that fitter individuals have a greater chance of being selected than less fit individuals. The technique used in this paper is *Binary Tournament Selection* [13]. In this method, pairs of individuals are randomly selected from the population and the fitter individual is copied into the mating pool. Both individuals are replaced into the original population. This process continues until the mating pool is full.

Mutation is an operation that is often applied in GA. The premise is that it is advantageous to introduce some controlled randomness in the formation of offspring in order to allow a better exploration of the search space. Thus some elements in an offspring are randomly perturbed with a certain probability. Mutation is not explicitly applied in this paper, however it will be seen that mutation occurs implicitly in several of the GA implementations considered.

4.2 Crossover Techniques

The method used to produce offspring from two parents in the mating pool is known as *crossover*. Simple crossover consists of selecting a random point in the individuals and then swapping their "tails". Thus the resulting offspring will inherit features from both parents. Such a method of crossover is well suited for the traditional applications of genetic algorithms where the problem is readily encodable as a binary string in which the frequency and position of values is unimportant. This is evidently not the case when GA are applied to MSN deployment. Consider the crossover of Parents A and B to give Offspring P and Q illustrated below. The crossover point is randomly chosen.

A	4	7	6	12	8	1	9	3	13	16	10	14	15	2	11	5
B	3	4	11	15	14	1	13	12	7	2	6	10	9	5	8	16

P	4	7	***6***	12	***8***	1	***9***	3	13	***16***	***6***	10	***9***	5	***8***	***16***
Q	3	4	***11***	***15***	***14***	1	13	12	7	***2***	10	***14***	***15***	***2***	***11***	5

It may be readily observed that the both Offspring P and Offspring Q are invalid. The duplications and omissions that exist in both offspring have been highlighted. Recalling the fact that the elements and position indicate the mapping of MSN nodes onto physical topology nodes, it can be seen that duplications and omissions are a significant problem. Four alternative crossover techniques are considered in this paper to prevent the formation of invalid offspring.

4.2.1 Crossover and Correct (XC)

Crossover and Correct is the simplest crossover technique considered in this paper. In XC, a randomly chosen crossover point is selected and the tails of the parents are swapped. Any duplication or omissions in the two offspring produced are detected and corrected. The correction of the offspring is done in such a manner that the elements and positions from one parent are left unaltered - this will hopefully ensure better propagation of characteristics down the generations compared to the alternative of correcting arbitrarily from the portion copied from both parents. Using the example previously given, it can now be seen that both Offspring P and Offspring Q contain neither duplications nor omissions. The elements that have been changed in order to correct the offspring have been highlighted. It can see that all the correction is done only on elements from one parent. The correction operation may be considered as implicitly introducing mutation into the offspring produced.

A	4	7	6	12	8	1	9	3	13	16	10	14	15	2	11	5
B	3	4	11	15	14	1	13	12	7	2	6	10	9	5	8	16

P	4	7	***14***	12	***11***	1	***15***	3	13	***2***	6	10	9	5	8	16
Q	3	4	***8***	***9***	***6***	1	13	12	7	***16***	10	14	15	2	11	5

4.2.2 Partially Mapped Crossover (PMX)

Partially Mapped Crossover [15] is implemented by selecting two random crossover points. The elements of both parents between both parents are exchanged. In each offspring, elements that have been displaced in the crossover region are exchanged with elements that have been introduced into the crossover region. Alternatively, PMX may be considered as specifying a number of swap operations to be executed in each parent to yield the offspring. In the example below, PMX implies elements 6 & 11, 12 & 15, 8 & 14, and 9 & 13 are swapped in both parents to produce new offspring. Again it may be seen that mutation is being implicitly applied by the swapping operations.

A	4	7	6	12	8	1	9	3	13	16	10	14	15	2	11	5
B	3	4	11	15	14	1	13	12	7	2	6	10	9	5	8	16

P	4	7	11	15	14	1	13	3	9	16	10	8	12	2	6	5
Q	3	4	6	12	8	1	9	15	7	2	11	10	13	5	14	16

4.2.3 Order Crossover (OX)

Order Crossover [16] is so called as it attempts to preserve the order that elements appear in parents without producing illegal offspring. Two crossover points are randomly selected. The elements between the crossover points are copied from Parent A into Offspring Q, and from Parent B into Offspring P. The remaining elements in Offspring P are obtained by copying the elements from Parent A. Starting just after the crossover region, the elements are copied in a cyclic manner in the order they appear in Parent A. Any elements already in the crossover region are skipped in order to produce legitimate offspring. Offspring Q is formed analogously.

A	4	7	6	12	8	1	9	3	13	16	10	14	15	2	11	5
B	3	4	11	15	14	1	13	12	7	2	6	10	9	5	8	16

P	8	9	3	16	14	1	13	12	7	2	6	10	15	11	5	4
Q	15	12	7	2	6	1	9	3	13	16	10	14	5	8	4	11

4.2.4 Cycle Crossover (CX)

Cycle Crossover [16] ensures that each position in an offspring is occupied by a corresponding element from one of the parents. Therefore, in order to produce legitimate offspring, groups of elements that must be taken from the same parent have to be found. Essentially, selecting an element from a particular parent compels other elements to be chosen from that parent. These groups of elements that must be selected from the same parent are called *cycles*. Since every element in the offspring must occupy a position it occupies in either parent, in the example below it is only possible to have either 4 or 3 in position 1 in the offspring. For Offspring P, say 4 is selected from Parent A, this means that 7 must be selected from Parent A for position 2, since selecting 4 from Parent B implies that the offspring will contain 4 in two positions. Furthermore, 13 must be selected in position 9 in order to prevent duplication of 7. Consequently, positions 1,2 and 9 must be chosen from the same parent, hence are part of the same cycle. The other elements in this cycle and any other cycles must be found. The cycles obtained from the exemplar parents are given below. *X* denotes the occurrence of the same element in the same position in both parents. Offspring are formed by choosing cycles from different parents. In this paper, Offspring P always chooses the elements in Cycle 1 from Parent A and the other cycles' elements from Parent B. Offspring Q selects elements in Cycle 1 from Parent B and the remaining elements are chosen from Parent A. It can be observed that each element in both offspring occurs in a corresponding position in one of the parents.

A	4	7	6	12	8	1	9	3	13	16	10	14	15	2	11	5
B	3	4	11	15	14	1	13	12	7	2	6	10	9	5	8	16
Cy	1	1	2	1	2	X	1	1	1	3	2	2	1	3	2	3
P	4	7	11	12	14	1	9	3	13	2	6	10	15	5	8	16
Q	3	4	6	15	8	1	13	12	7	16	10	14	9	2	11	5

5. RESULTS

Cases in which there is a mismatch in the number of nodes in the physical and virtual topologies are outside the scope of this paper. This paper considers examples of embedding a 4x4, a 6x6, an 8x8 and a 10x10 MSN in exemplar arbitrary physical topologies of 16, 36, 64 and 100 nodes respectively. The network with 16 nodes is a topology of the UK [17]. The other networks were chosen to illustrate the difference, if any, that the network size makes to the best embeddings found, and to investigate the impact problem size has on GA performance. The networks were chosen so that they all have comparable mean degrees and are all illustrated in Figure

2. It can be seen that packet switching over such topologies will require the storage of large and complex routing tables. Furthermore, the arbitrariness of the connectivity implies that contentions are likely to occur. Embedding the MSN is a possible way to alleviate both these problems and allow optical packet switching.

A conventional approach when designing virtual topologies for wavelength-routed WDM networks is to use the traffic matrix as an input in the optimisation process - for example [8]. It may be argued that this is an *ad hoc* approach since the traffic matrix is dynamic. The viewpoint adopted in this paper is that it is better to deploy the virtual topology using invariable parameters (such as the connectivity of the physical and virtual topologies) as the inputs to the optimisation process. Furthermore, it is assumed all fibres have equal physical length, i.e. unity. This is done in order to realise a direct correlation between the mean lightpath length and the number of OXCs encountered by a packet or lightpath.

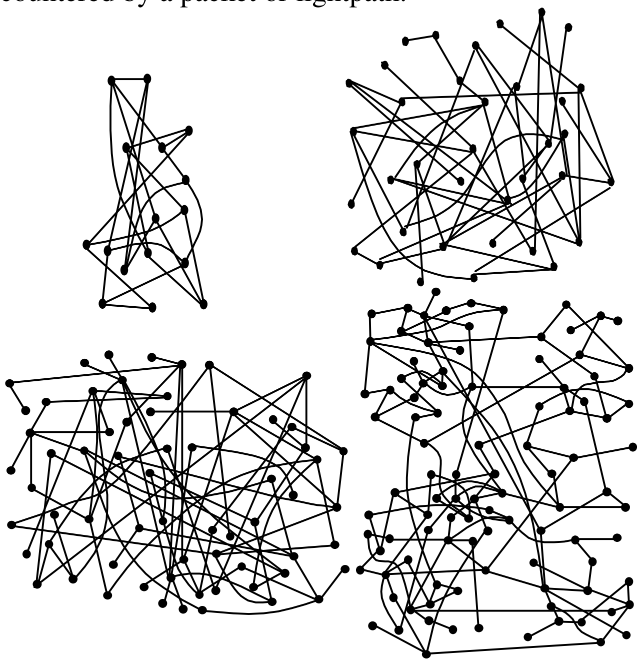

Figure 2. Exemplar arbitrary physical topologies of 16, 36, 64 and 100 nodes

5.1 Selection of Optimal GA Parameters

There are several critical discretionary parameters in the implementation of GA. The different crossover techniques and selection method have been discussed. In this paper, the population size and the probability of crossover are two parameters that were selected empirically. The population size is the number of individuals that are considered in each generation. Intuitively it seems desirable to have as large a population as possible. However, this must be balanced by the need to obtain results relatively quickly and memory limitations. Allowing the probability of crossover to be less than unity means that it is possible for two parents to survive intact into the next generation - once selected from the mating pool, the two parents are copied directly into the next generation. Population sizes of 250, 500, 1000 and 2000 were considered. The probability of crossover was either 0.5 or 1. These eight alternatives were all investigated for each crossover technique. Figures 3 to 6 show results for the example of deploying the 6x6 MSN in the 36-node physical topology. The mean of the mean lightpath lengths for the population for different generations is shown. Note that in this section, emphasis is placed not on the performance of the different crossover techniques relative to each other, but rather on the results obtained when certain parameters are varied for each crossover technique.

For Crossover and Correct (XC) and Cycle Crossover (CX) (Figures 3 and 6 respectively) it is evident that a population size of 2000 and a probability of crossover of 1 gives the best results. For XC convergence occurs just after 100 generations. It takes around 70 generations for convergence to occur with CX. For the Partially Mapped Crossover (PMX), it can be observed in Figure 4 that having a probability of crossover of 0.5 results in significantly faster convergence than always carrying out crossover. However, having the probability of crossover as 1 gives the better results. The impact of different population sizes is less conclusive and appears slightly surprising. Further investigation was carried out and it was found that a population size of 1000 gives the best results. Note that PMX takes significantly longer to converge than the other crossover techniques. For the Order Crossover (OX) from Figure 5 it is evident that no improvement to the mean lightpath length is observed when the probability of crossover is 1. When the probability of crossover is 0.5, a population size of 2000 gives the best results.

Consequently, the population sizes and probability of crossover have been determined and are used for the results presented in the rest of the paper. The population size for XC, CX and OX is 2000. PMX has a population of 1000. The probability of crossover is always 1, with the exception of OX where the probability of crossover is 0.5.

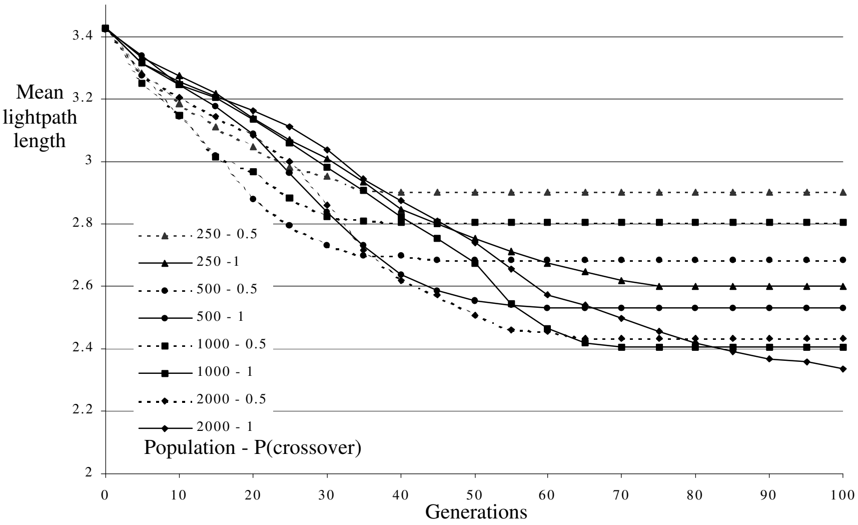

Figure 3. Effect of varying population size and crossover probability on performance of Crossover and Correct (XC)

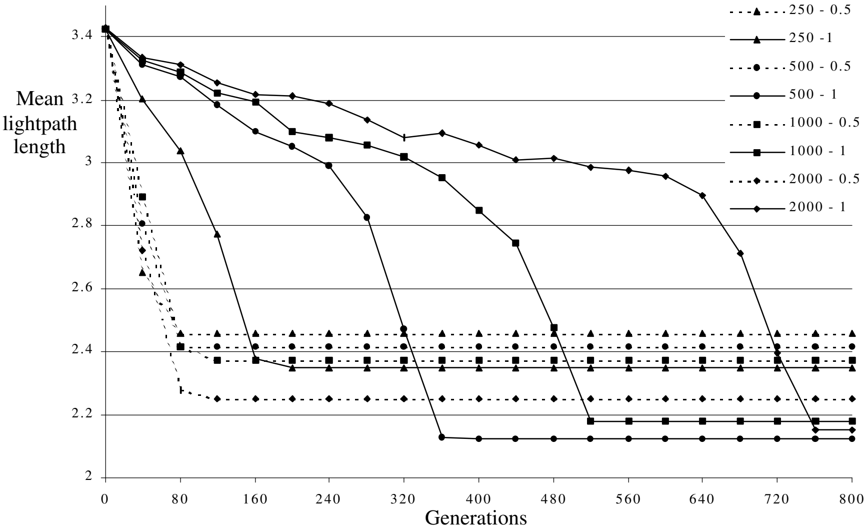

Figure 4. Effect of varying population size and crossover probability on performance of Partially Mapped Crossover (PMX)

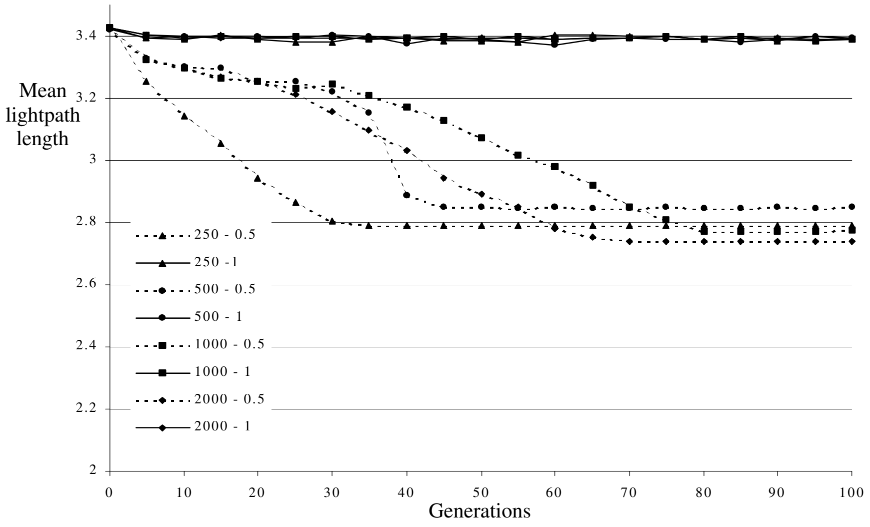

Figure 5. Effect of varying population size and crossover probability on performance of Order Crossover (OX)

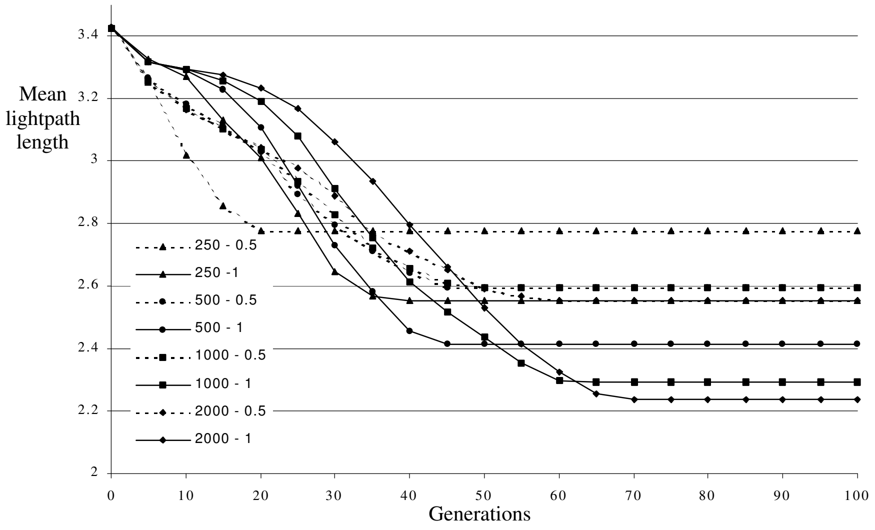

Figure 6. Effect of varying population size and crossover probability on performance of Cycle Crossover (CX)

5.2 Performance of Different Crossover Techniques

5.2.1 Results for Different Network Sizes

The different crossover techniques with the appropriate parameters were used to deploy the MSN in the physical topologies of different sizes. XC, OX and CX all have a population size of 2000 and for fairness the same randomly generated initial population was used. PMX's population is 1000, nevertheless, the implementation ensured that its initial population is a subset of the initial population for the other crossover techniques. In all cases, the overall mean lightpath length of the initial population were very similar.

Figure 7 shows the results for embedding the 4x4 MSN (Figure 1) in the 16-node physical topology shown in Figure 2. It may be observed that PMX gives the best overall result. OX gives the worst result - a final mean lightpath length is 1.78. PMX, CX and XC obtain a mean lightpath length of 1.56, 1.59 and 1.63 respectively. It is encouraging to observe that the different GA implementations all reduce the mean lightpath length significantly, making the prospect of using the MSN to achieve optical packet switching over this arbitrary topology feasible.

The results for the deployment of the 6x6 MSN in the 36-node physical topology are shown in Figure 8. It may be seen that Partially Mapped Crossover (PMX) gives the best overall results - a mean lightpath length of 2.15 after 500 generations. Cycle Crossover (CX) obtains a mean lightpath length of 2.19 after just 69 generations - this corresponds to a reduction in the mean lightpath length of almost 40% from the initial result by considering less than 140,000 different embeddings. Crossover and Correct (XC) obtains a mean lightpath length of 2.28 after 100 generations. Order Crossover (OX) yields a mean lightpath length of 2.74 after roughly 60 generations.

Once again, OX gives the worst result when deploying the MSN in the 64-node physical topology. From Figure 9, it is apparent that OX yields a mean lightpath length of 3.83 after 70 or so generations. XC comes next, obtaining a mean lightpath length of 3.15 after 140 generations. The mean lightpath length that CX obtains is 2.66, obtained after less than 100 generations. The lowest mean lightpath length obtained was achieved by PMX - 2.45 after approximately 600 generations. PMX almost halves the mean lightpath length by considering only 600, 000 different embeddings out of a search space of up to 1.27×10^{89}.

A similar pattern is observed when the 10x10 MSN is embedded in the 100 node physical topology. The results are shown in Figure 10. The mean lightpath length in the initial population is 4.96. OX manages to reduce this to 4.38 after approximately 75 generations. XC achieves a mean lightpath length of 3.63 after 160 generations. CX reduces the mean lightpath length to 2.88 - a reduction of over 40% in only 120 generations. Once again, PMX obtains the best result but takes the longest to converge. PMX's final mean lightpath length is 2.56. The mean lightpath length has been reduced by almost 50% in approximately 1700 generations.

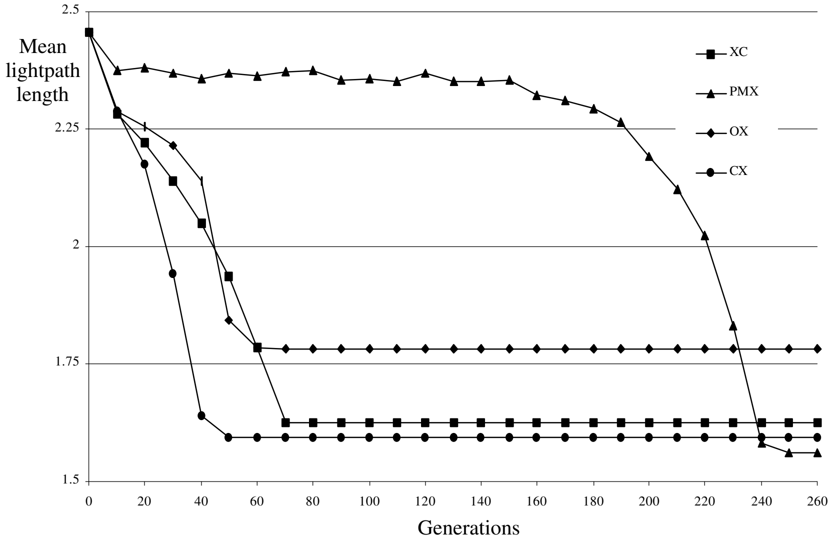

Figure 7. Deployment of 4x4 MSN in physical topology using different crossover techniques

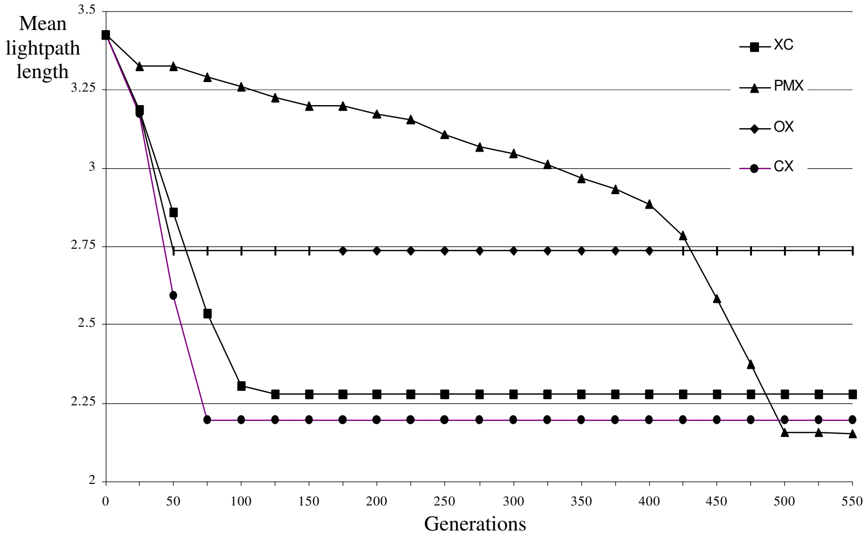

Figure 8. Deployment of 6x6 MSN in physical topology using different crossover techniques

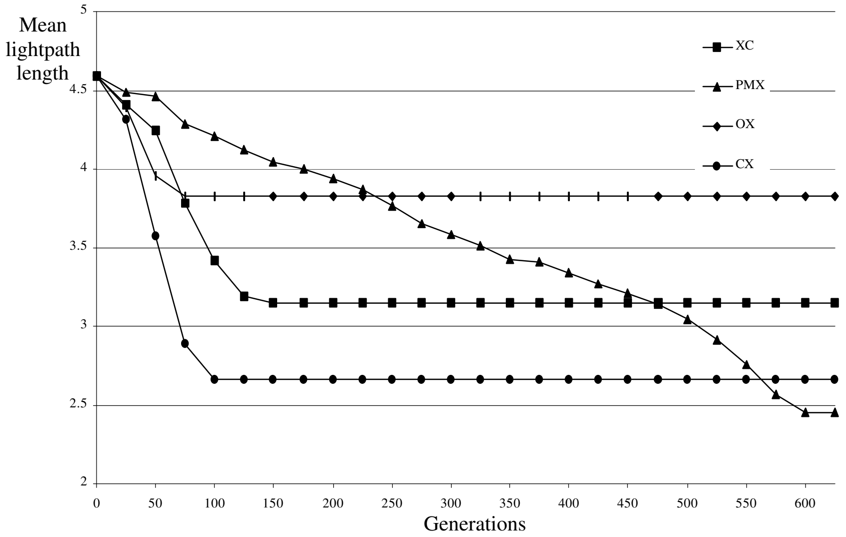

Figure 9. Deployment of 8x8 MSN in physical topology using different crossover techniques

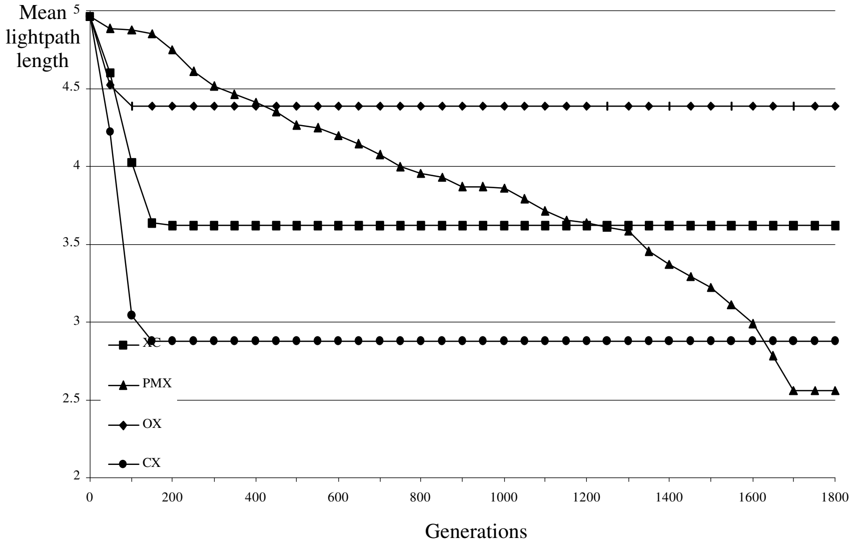

Figure 10. Deployment of 10x10 MSN in physical topology using different crossover techniques

5.2.2 Comments on Results

Several patterns emerge from the results obtained. The meritocratic ordering of the different crossover techniques is PMX, CX, XC and OX. PMX always took significantly longer to converge than the other crossover techniques but always gave the best results. The variation in the performance of the different crossover techniques appears to increase with increasing network size. For example for a 4x4 MSN, PMX is only 12% better than OX but the difference is greater than 40% for a 10x10 MSN.

The reason for the discrepancy in the performance of the different crossover techniques may be understood in terms of the two fundamental operations of optimisation or search techniques - *exploration* and *exploitation* [13]. A good search algorithm should explore the search space whilst exploiting the information gained.

Order Crossover (OX) consistently gives the worst result of all the crossover techniques considered in this paper. From the description of OX given in Section 4.2.3 it is evident that OX does a lot of exploration but very little exploitation - or to be more accurate exploits the wrong information. OX seeks to preserve the ordering of the nodes in the individual - this is valid information for the Travelling Salesman Problem (TSP) for which OX was originally proposed. However, in the problem encoding described in this paper, it is the absolute position of elements in the individual that is most important, and OX does not exploit this information adequately. This hypothesis is buttressed by the observation from Figure 5 that a crossover probability of 0.5 and not 1 gives the better results. This indicates that if crossover is always carried out, relevant characteristics of parents are not passed onto the offspring. However, adopting a crossover probability less than one allows the preserving of important characteristics of parents since the parents have a chance of appearing unaltered in the subsequent generation.

Cycle Crossover (CX) is arguably the best crossover technique at preserving and exploiting the important characteristics of the parents - each position in the offspring is filled with elements from the same position in one of the parents. CX gave results that were only slightly worse than PMX but required significantly fewer generations. For example, for the 8x8 MSN, CX gave a result that was 8.6% worse than PMX but did so in over 500 generations less. This suggests that CX is extremely good at exploitation. It would be interesting to try to improve the amount of exploration done, by allowing the offspring to contain a few elements not occurring the same position in the parent. The impact of introducing mutation to CX will be extremely interesting to investigate.

Partially Mapped Crossover (PMX) obtained the best results. From this it may be concluded that PMX best combined exploration and exploitation synergistically. In [15] it was proven that PMX is extremely good at exploring different areas of the search space for the TSP with implicit parallelism. Crossover and Correct (CX) was an implementation of the simple crossover with the creation of any invalid individuals being prevented. Consequently, it should not be expected to perform as well as PMX and CX which have been specially formulated and shown to tackle the TSP excellently. Nevertheless, CX still achieves reductions on the starting mean lightpath length - an impressive feat considering its simplicity.

5.3 Comparison of GA with Other Heuristics

The different implementations of genetic algorithms have been compared to each other for varying network sizes. There are other techniques that may be used to attempt to deploy the MSN in arbitrary physical topologies, and this section compares the best GA results with the results for alternative optimisation techniques. The three methods to which GA are compared are simulated annealing [18], hill climbing and random search.

SA has been successfully used for deploying hypercubes in NSFnet [19], deploying the MSN in different physical topologies [10] and successfully implemented for similar combinatorial optimisation problems. The SA parameters that were found to obtain good results in [10] are used in this paper. The hill climb was started at a random point in the search space and the number of hill-climbing steps executed is comparable to the aggregate number of individuals considered in the GA implementation that achieved the best result. Random searches of 5 million points were conducted.

The results obtained for the different heuristics is shown in Table 1. The lowest mean lightpath length obtained for each network size has been highlighted. From Table 1 it can be observed that GA out-performs simulated annealing, hill climbing and random search for all network sizes with the exception of a 6x6 where SA and hill-climbing give the best result. (The factors that affect the performance of the heuristics are currently under investigation.) The difference between the GA result and the other techniques' results increases with the network size, suggesting that GA are well-suited to finding high quality embeddings as the search space increases.

Evaluating the cost function is relatively time consuming, consequently the number of solutions examined by each heuristic is indicative of the time taken. For say $N = 4$, GA and HC consider a similar number of solutions, whereas SA and the random search consider approximately 10 and 20 times more solutions respectively. Consequently, not only does GA produce better results than the other heuristics, it does so significantly quicker.

Table 1. Comparisons of GA with other heuristics

N	Genetic algorithms	Simulated annealing	Hill climbing	Random search
4	***1.56***	1.63	1.63	1.72
6	2.15	***2.11***	***2.11***	2.81
8	***2.45***	2.57	2.70	3.89
10	***2.56***	3.11	3.00	4.45

6. CONCLUSIONS

The Manhattan Street Network and a particular routing scheme have been seen to avoid complex routing algorithms and prevent optical domain contentions making it attractive for optical packet switching for some applications. This paper has examined how the MSN may be deployed as a virtual topology in several exemplar physical topologies. A new technique, genetic algorithms, has been successfully applied to the virtual topology design problem. Different implementations of GA have been introduced and their optimum parameters empirically determined. These GA implementations have been compared to each other in terms of the lowest mean lightpath they yield when used to deploy the MSN in the physical topologies. GA are also seen to out-perform other heuristics at deploying the MSN in arbitrary physical topologies for optical packet switching.

REFERENCES

[1] "Nonlinear Optics for High-Speed Digital Information Processing", D. Cotter *et al*, Science, Volume 286, November 1999.

[2] "Ultra-High-Bit-Rate Networking: From the Transcontinental Backbone to the Desktop," D. Cotter, J. K. Lucek, D. D. Mercenac, IEEE Communications Magazine, April 1997.

[3] "The Manhattan Street Network," N. F. Maxemchuk, Proceedings of IEEE Globecom '85, December 1985.

[4] "A New Packet Routing Strategy for Ultra-Fast Photonic Networks," F. Chevalier, D. Cotter, D. Harle, Proceedings of IEEE Globecom '98, November 1998.

[5] "Routing and Fairness Issues Associated with a Self Routing Control Strategy in Regular Mesh Photonic Networks," F. Chevalier, D. Cotter, D. Harle, Proceedings of the International Teletraffic Conference (ITC 16), June 1999.

[6] "Lightnets: Topologies for High Speed Optical Networks," I. Chlamtac, A. Ganz, G. Karmi, IEEE/OSA Journal of Lightwave Technology, Volume 11, Number 5/6, May/June 1993.

[7] "Deployment of Architectures for Ultrafast Optical Packet Switching," O. Komolafe, D. Harle, D. Cotter, Proceedings of IEE UKTS 17, May 2000.

[8] "A Survey of Virtual Topology Design Algorithms for Wavelength Routed Optical Networks," R. Duttra, G. N. Rouskas, TR-99-06, Dept. of Computer Science, N. Carolina State University.

[9] "Data Encodings and Their Costs," A. L. Rosenberg, Acta Informatica, Vol. 9, 1978.

[10] "Ultrafast Optical Packet Switching over Arbitrary Physical Topologies using the Manhattan Street Network," O. Komolafe, D. Harle, D. Cotter, Submitted to IEEE ICC 2001.

[11] "Adaptation in Natural and Artificial Systems," J. H. Holland, University of Michigan Press, 1975.

[12] "A Genetic Algorithm Tutorial," D. Whitley, CS-93-103, Dept. of Computer Science, Colorado State University.

[13] "An Overview of Genetic Algorithms: Part 1, Fundamentals," D. Beasley, D. R. Bull, R. R. Martin, University Computing, Volume 15, Number 2, 1993.

[14] "Evolutionary Telecommunications: A Summary," M.C. Sinclair, Proceedings GECCO '99, July 1999.

[15] "Alleles, Loci, and the Travelling Salesman Problem," D. E.Goldberg, R. Lingle, in Genetic Algorithms & their Applications: Proc. of an International Conf. on Genetic Algorithms, Grefenstette (Ed), L. Erlbaum Associates, 1983.

[16] "A Study of Permutation Crossover Operators on the Travelling Salesman Problem," I. M. Oliver, D. J. Smith, J. R. C. Holland, in Genetic Algorithms & their Applications: Proc. of the 2nd International Conf. on Genetic Algorithms, Grefenstette (Ed), L. Erlbaum Associates, 1987.

[17] "Dynamic Topology Approach to Network Resilience," M. Jorgenson, B Turton, A. Goode, and A. Hume, Proceedings of IEE UKTS 15, March 1998.

[18] "Optimization by Simulated Annealing," S. Kirkpatrick, C. D. Gerlat, M. P. Vecchi, Science, Volume 220, Number 4598, 1983.

[19] "Some Principles for Designing a Wide-Area Optical Network," B. Mukherjee, S. Ramamurthy, D. Banerjee, A. Mukherjee, Proceedings of IEEE Infocom 1994.