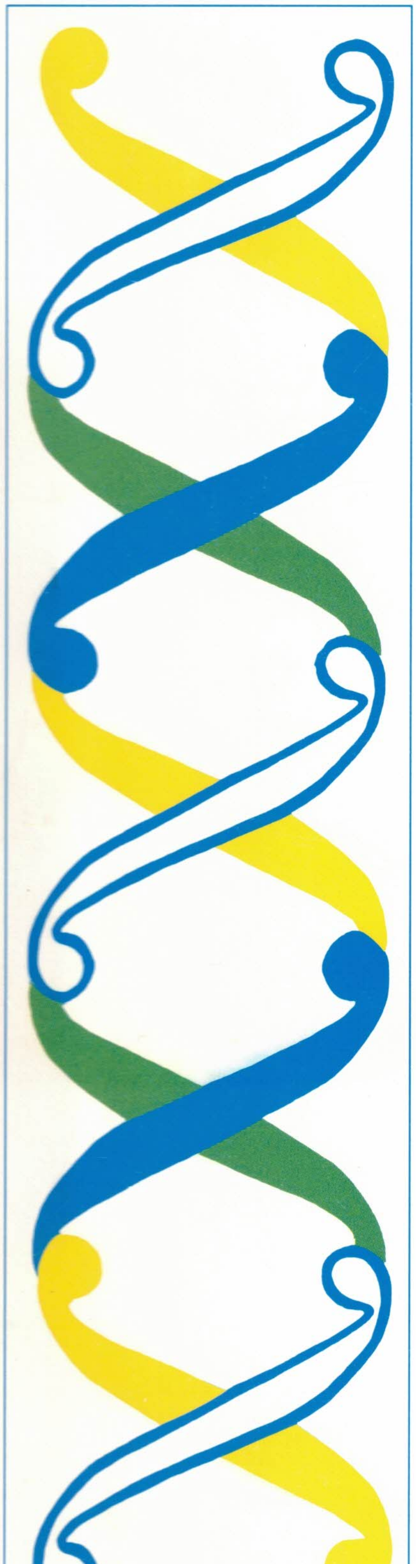
Conference Publication No. 414

IEE



First IEE/IEEE INTERNATIONAL CONFERENCE ON

Genetic ALgorithms in Engineering Systems: Innovations and Applications

GALESIA

'95

CHARLEY: A GENETIC ALGORITHM FOR THE DESIGN OF MESH NETWORKS

J. Hewitt*, A. Soper and S. McKenzie

University of Greenwich UK
*Charing Cross Hospital, UK.

Email: a.soper@greenwich.ac.uk

ABSTRACT

This paper presents a genetic algorithm for the design of an optimal mesh network. The problem is of relevance in the design of communication networks where the backbone switching network takes the form of a highly connected mesh in order to provide reliability in the event of switch/link failure. The proposed algorithm addresses two important aspects of the problem - topology design and capacity allocation. The optimisation is done with respect to connection costs subject to performance (delay), connectivity and capacity constraints. Connection costs are assumed to depend on distance and link capacity. Though the algorithm was designed with mesh networks in mind, it can be applied to the simpler problem of the constrained minimum spanning tree.

The algorithm has been tested on a tree network and two mesh networks. The results compare very favourably with those obtained from existing design techniques.

INTRODUCTION

Genetic Algorithms (GAs) have been successfully applied to a wide range of combinatorial problems (Goldberg 1989, Davis 1991). They are particularly useful in applications involving design and optimisation (Whitley 1989), where there are a large number of variables and where procedural algorithms are either non-existent or extremely complicated. Communication network design is one such example and is addressed in this paper.

Modern communication networks are hierarchical in structure, consisting of a high speed backbone of switches with clusters of user sites connected to the switching nodes in multipoint local access configurations. The backbone is generally a highly connected mesh with redundant links to ensure reliability in the event of node or link failure. Local access networks are generally configured as trees. The object of design is to produce a minimum cost network subject to performance and reliability constraints.

Traditionally, the network design process has always been carried out iteratively, with continual refinement (see for example Tannenbaum 1986, Ahuja 1985, Kershenbaum 1993, Schwartz 1977). For tractability, local access and backbone design are tackled separately. The problems of topological design, link capacity allocation and routing are also treated independently of each other, though decisions made in one area will affect other aspects of design. While graph theoretical algorithms, heuristics and other mathematical techniques are available for different aspects of design, there is no one procedural algorithm for solving the problem as a whole. This is an area where the GA approach could prove useful.

GAs have already been applied to several aspects of network design - the unconstrained Minimum Spanning Tree (Michalewicz 1991), concentrator assignment (Routen 1994) and routing (Sinclair 1993). In this paper we present a GA for mesh backbone design, incorporating topological design and capacity allocation. Routing is still done using Dijkstra's algorithm (Tannenbaum 1988), using available link capacity as the metric. The algorithm is intended for producing optimal minimum cost networks, while satisfying connectivity and performance (delay) constraints. The connectivity constraint is incorporated by requiring that the network be k-connected, i.e. there must exist at least k independent paths between any given pair of nodes. With k set to 1, the same algorithm could be used for deriving local access configurations, with the added flexibility of variable link speeds.

The GA has been implemented on a Sparc 10 platform using C. Preliminary results are encouraging and show that the algorithm does converge to an optimal solution reasonably quickly.

A more precise definition of the network design problem, the genetic algorithm and preliminary test results are given in the following sections.

THE NETWORK DESIGN PROBLEM

The inputs to the design process are:

Genetic Algorithms in Engineering Systems: Innovations and Applications
12-14 September 1995, Conference Publication No. 414,

given generic shape, rather than as a source of novel designs.

8. Conclusions

8.1 Lower number of invalid shapes.

The parametric description used produced a lower proportion of invalid shapes than previous representation methods of lines or approximated splines. It is clear that given reasonable ranges for each parameter, the number of invalid shapes is a small fraction of the total number of possible shapes, this allows the program to dispense with any specific checking of the shapes for validity other than that conducted implicitly by the FE package. In this respect parametric descriptions can be seen as more in tune with GA processes than previous representations.

8.2 Limits on design innovation.

The suggestion has been made that the use of generic shapes would restrict the innovation of new designs. The early work conducted on this design shows that this is not entirely correct. The minimisation of some features in the generic shape can be seen as highlighting the redundancy in the shapes description. Removal of such extraneous features may lead to novel designs.

It must still be noted that whereas redundant features may be highlighted due to their minimisation, important features that do not exist in the current generic shape cannot be simply inferred from evolutionary results.

8.3 Results, target values exceeded.

The results show that target figures for a good design can be exceeded using this FE-GA package, and the results can be found in reasonable times on an average sized workstation. The designs produced appear to be entirely practical in terms of the stress and mass constraints specified by Rolls-Royce, and the results found so far exceed those produced using the previous representations.

8.4 Proof of concept.

The work covered in this paper provides the first evidence that the Edinburgh FE-GA package can be used to generate practical efficient designs. Further to this it demonstrates that the parameterisation of the shape is no barrier to successful evolution.

9. Future Work

Future work will be directed to more detailed testing of the annulus (to include cycling information), a comparison with other optimisation techniques, and research into alternative shape representation methods.

The time taken to write the program which generates the shape from the parametric description prevents this system from being a simple and general designers tool. This problem will largely be alleviated with the integration with the Computervision CADDS5 system, where the generation of generic shapes should be relatively fast and simple.

10. Acknowledgements

The authors gratefully acknowledge the support of the SERC's EPSRC Directorate (grant number GR/J40812) and of Rolls-Royce and British Aerospace for supplying data and software.

11. References

[1] David Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addision-Wesley, 1987

[2].Richard Smith (23/8/94) *The Use of Genetic Algorithms in Shape Design Optimisation*. Internal Report No.19 Manufacturing Planning Group, University of Edinburgh, 1994

[3] Rolls-Royce Annulus Specification Document REF. JDR/RGP

[4] Emile Aart and Jan Korst (1989) *Simulated Annealing and Boltmann Machines*. Wiley

N, network size (number of nodes).

C, Connection costs, where C_{ij} represents the cost of connecting nodes i and j. It is assumed that a set of discrete link speeds are available, with costs depending on both link speed and distance between nodes.

T, traffic matrix. T_{ij} denotes the traffic (number of messages/sec) from node-i to node-j. The mean message size is also needed to calculate link flows in bits per second(bps). Exponential message interarrival times and exponential message length distributions have been assumed for purposes of delay calculation, which is done using the M/G/1 delay formula (Schwartz 1977).

k, the connectivity requirement. It is required that there be at least k node-disjoint paths between any given pair of nodes.

t_{max}, the maximum acceptable value (seconds) of mean message end-to-end delay.

The output from the design process is an optimal minimum cost network that satisfies the delay and connectivity constraints.

GENETIC ALGORITHM

Representation

Since a traditional genetic crossover operator is used a chromosomal structure needs to be defined. This consists of a list of established links in a candidate network or individual in the population. Each link is represented as a record with fields defining its end point nodes, capacity, cost, messages/sec (flow), mean message length, and link delay.

The list of links is ordered as follows:

The nodes in the network are numbered 1 to N and the end point nodes (i, j) of each link are ordered such that i<j. The list of links is sorted in ascending order by the first end point. The ordering may be destroyed during mutation and the list is reordered before the next crossover and mutation are attempted.

Initial Population

The minimum number of links in an acceptable network is first calculated from the connectivity requirement. Each individual in the population is then constructed by selecting the required number of links at random from the pool of available connections in the connection matrix C.

Crossover

A node (i) is selected at random, and segments before the first link with i as the first end point in the first parent are combined with links including and after the first link with i as the first end point in the second to get the first offspring. The remaining links in the two parents go to form the second offspring.

If the first parent does not possess any links with i as the first end point, then the first offspring is formed from segments taken solely from the second parent as above, the remainder of the parents' links forming the second. If neither network includes a link with i as the first end point, the parents are carried over unchanged into the next generation.

The linkage structure imposed by the representation is not ideal in that it does not preserve local groups of links with uniform probability along the chromosome. Links incident at the first node will have that node as their first end point and will occur in a block at the beginning of the chromosome. Links associated with the last node will have that node as their second end point and will in general be dispersed along the length of the choromosome. The representation combined with the crossover operator is thus too conservative with respect to links at the head of the chromosome and too destructive with respect to those at the tail.

Despite this drawback, the representation used provides a starting point for solving a fairly difficult optimisation problem. Coupled with the mutation operators and rates used, it is reasonably successful in producing good solutions. Improvements to both the representation and the crossover operator will be explored in future work.

Selecting candidates for mating

The Reeves (Reeves 1993) rank based scheme is used. The individuals are assigned a rank R (fittest = M = population size), and individuals are selected at random using the probability function:

$$p(R) = 2R / \{M(M+1)\} \quad (1)$$

Once two networks have been selected, two possibilities arise: 40% of the time they do not participate in mating, but are carried over into the next generation. The rest of the time, they are subject to crossover.

Mutation

Two mutation operators are defined. The first (add/delete) allows the number of links in a network to vary by plus or minus one, and the second (exchange) allows a network to be altered keeping its number of links constant.

(a) Delete a link at random in 75% of mutations of this type and add a link in the remaining 25% selected at random from the pool of available connections in the cost matrix. The bias towards removing links was found to result in better networks.

(b) A link chosen at random is replaced by another one selected at random from the pool of available connections. The exchange is unconditional in 25% of mutations, but only if there is a cost saving in the remaining 75%.

Networks which do not undergo crossover are also candidates for mutation.

Fitness

The measures of fitness are all quantities to be minimised. They are grouped together into two penalty terms. The sum of components (1) and (2) below, gives a performance measure of the network while the sum of components (3) and (4), relating to the connectivity and cost of the network will be referred to as the adjusted cost. Networks with the smaller penalties are deemed to be fitter individuals for the purpose of selection.

(1) Delay penalty - if the average delay is less than the acceptable delay threshold the penalty is zero. For average delays greater than the acceptable delay, the difference is divided by the average delay so as to lie between 0 and 1. If the flow along a link exceeds its capacity, a utilisation of .999 is used in the M/G/1 delay formula to give virtually infinite delay, so that the penalty term is 1.

(2) Message blocking penalty - if messages are blocked due to lack of a connected path or insufficient capacity, a penalty term equal to the fraction of messages blocked is calculated.

(3) Connectivity penalty - the total deficit in the number of disjoint paths between all possible pairs of nodes (as compared to the required k) is calculated, and normalised with respect to the total required number of disjoint paths in the k-connected network.

e.g. for N nodes, with N(N-1)/2 node pairs, the required number of disjoint paths (p_req) for k-connectivity is

$$p\_req = k * N(N-1)/2. \qquad (2)$$

If there are only k-2 disjoint paths between two particular nodes the deficit is 2, and the penalty term is 2/p_req.

(4) Connection cost - the sum of link costs normalised with respect to the cost of a network having the minimum number of links for the specified connectivity requirement k, but using the cost of the most expensive link. This term could range from 0 which is unlikely to infinity in theory, but once the GA starts to produce 'sensible' networks without too many redundant links, the penalty term will generally be less than 1.

Networks in the population are sorted in descending order of performance (primary key) and adjusted cost (secondary key). The fittest individuals are those at the bottom of the list, i.e. they have the highest rank.

Simulation Parameters

The same GA parameters were used for the three test cases considered: a population size of 400, a maximum number of generations of 400 and a mutation rate of 10% applied to each of the two mutations defined.

Mutation rates of 1%, 10%, 20% and 40% were tried for the multidrop problem. The rate of 10% was found to be best both in terms of speed of convergence and the fitness of the best individual.

High mutation rates have been used in other work (Freisleben and Hartfelder 1993). In the present work, the high mutation rate helps to overcome the adverse effects due to the inherent bias in the representation.

RESULTS AND DISCUSSION

The results for the best of three runs of the GA for each network design problem is given and compared with an existing design technique.

Constrained Minimum Spanning Tree (k=1)

All the network traffic is directed to node-1. The total number of nodes N was 13, and a very large acceptable delay of t_{max}=200 sec was used. Links of speed 1200 bps were available between all nodes and also of 9600 bps between node 1 and nodes 2-7.

Two types of message were allowed, with mean sizes of 960 and 19200 bits.

The Esau-Williams heuristic (Ahuja 1985), adapted to take account of varying link speeds was used to derive a network of cost 4143 and a mean message delay of 39.8 sec. The GA produced the same network in two of the three runs, found in generations 192 and 261 respectively.

Mesh Networks (k=3,4)

Six node networks were considered, with data message traffic between all pairs of nodes. The maximum acceptable delay (t_{max}) was set to 1 sec. Link speeds of 2.4, 4.8, 9.6 and 19.2 kbps were available for all connections and costs were proportional to the speeds of the links. The mean message size was set to 512 bits.

The GA produced networks satisfying the connectivity, capacity and delay requirements. These were compared with networks derived using the link deficit (heuristic) algorithm (Tannenbaum 1986), with minimum hop, minimum cost routing and using the M/M/1 delay formula (Schwartz 1977). The results are given below:

Connectivity(k), Algorithm	Cost	Delay (sec)	Generation (GA only)
3, Link deficit	825	.180	
3, GA	781	.773	107
4, Link deficit	797	.270	
4, GA	1094	.560	345

For k=3, the GA produces a lower cost network than the traditional approach, though the average delay is somewhat higher.

For k=4, the GA produces a feasible network, only somewhat more expensive. The GA converged to a population with 75% of the individuals identical to the fittest after approximately 100 generations for the three problems. Significant improvements were still obtained with further generations due to the introduction of new networks from mutations, though at a much slower rate.

We feel that the less impressive results for the k=4 case demonstrate the deficiencies of the crossover operator, a point that will be addressed in future work.

It may appear anomalous that the link deficit algorithm yields a cheaper network for k=4, despite the higher number of links. This is due to the fact that the routing procedure used in the k=3 case results in two of the link flows just exceeding 9600 bits/sec, necessitating the use of more expensive 19200 bits/sec links. It would be possible to refine the topology and the routing assignment by trial and error.

In conclusion, the results from the GA approach are very encouraging. They demonstrate the applicability of this approach to the problem of network design.

Future work will be aimed at testing the GA on larger networks. A crossover operator which is better at preserving the measures of fitness in this problem will be investigated.

REFERENCES

Ahuja V, 1985, Design and Analysis of Computer Communication Networks, McGraw-Hill.

Davis L. (editor), 1991, Handbook of Genetic Algorithms, Van Nostrand Rheinhold, New York.

Freisleben B and Hartfelder M, 1993, Optimization of Genetic Algorithms by Genetic Algorithms in the Proceedings of the International Conference on Artificial Neural Nets and Genetic Algorithms, pp 392-99, Springer-Verlag.

Goldberg D., 1989, Genetic Algorithms in Search Optimization and Machine Learning, Addison-Wesley, Reading, MA.

Kershenbaum A, 1993, Telecommunications Network Design Algorithms, Mcgraw-Hill.

Michalewicz Z., 1991, A Step Towards Optimal Topology of Communications Networks, Data Structures and Target Classification, SPIE Vol. 1470.

Reeves C., 1993, Genetic Algorithms in Modern Heuristic Techniques for Combinatorial Optimisation (Ed. Reeves, C) pp. 151-196, Blackwell Scientific Publications, London.

Routen T., 1994, Genetic Algorithm and Neural Network Approaches to Local Access Network Design, Proceedings of the 2nd IEEE International Workshop on Modelling, Analysis and Simulation.

Schwartz M, 1977, Computer Communication Network Design and Analysis, Prentice-Hall.

Sinclair M., 1993, The Application of a Genetic Algorithm to Trunk Network Routing Table Optimisation, Tenth UK Teletraffic Symposium, Performance Engineering in Telecommunications

Networks, p. 2/1-6, IEE, London.

Tannenbaum A S, 1986, 1988, Computer Networks, Prentice-Hall (editions 1 and 2).

Whitley D, Starkweather T and Fuquay D, 1989, Scheduling problems and Travelling Salesmen: The Genetic Edge Recombination Operator. In Proceedings of the Third International Conference on Genetic Algorithms, ed. J D Shaeffer, Washington D C, Morgan Kaufmann.