NOMaD: Applying a genetic-algorithm/heuristic hybrid approach to optical network topology design

M. C. Sinclair

Dept. of Electronic Systems Engineering, University of Essex,
Wivenhoe Park, Colchester, Essex CO4 3SQ.
Tel: 01206-872477; Fax: 01206-872900; Email: mcs@essex.ac.uk

1 Introduction

This paper describes the use of a genetic-algorithm (GA)/heuristic hybrid approach in a tool for optical network modelling, optimisation and design (NOMaD) being developed by the author at the University of Essex [1] and, in particular, early results from its application to virtual-topology design.

NOMaD is used as part of the author's own research into the application of GA/heuristic hybrid optimisation techniques to network design, as well as in several research projects, including two Advanced Communications Technologies and Services (ACTS) projects [2]: WOTAN (Wavelength-agile Optical Transport and Access Network) and OPEN (Optical Pan-European Network).

2 Application Areas

The design problems that NOMaD seeks to address will, at least initially, be those presented by multi-wavelength all-optical networks, including combined access/core networks at national level and transport networks at the international, as well as consideration of both 'flat' and hierarchical approaches. The two key application areas are:

- Topological design, starting by making the choice of links in both the virtual and physical topologies (the virtual topology ignores the underlying physical reality of which ducts and fibres will actually be used), and then building on that to network extensions (additional node placement), and perhaps eventually node placement for an entire network.

- Routing and wavelength allocation, initially taking account of only the virtual topology, but then recognising the need to include duct and fibre choice.

To begin with, NOMaD will model networks with static traffic demands, but subsequently it is hoped to incorporate dynamic traffic into the network model. Further, although at the start the focus will be on cost optimisation (whilst meeting certain performance and reliability constraints), additional performance metrics will be added later. It is also envisaged that NOMaD will allow the interactive study of network failure scenarios as part of the overall assessment of a design.

3 Overall Architecture

NOMaD is being developed as an overall architecture that will be extended by several people in a variety of directions to serve the needs of their individual projects. Consequently, its users are not just given a single unchangeable piece of software, but rather a framework which they can actively add to for themselves. As a result, NOMaD has to be easy to understand, flexible and extensible. The overall layered architecture of NOMaD is shown in Fig. 1.

To aid both understanding and extensibility, the network optimisation, modelling and design toolset is being developed using object technology. The analysis and design is being documented using Booch Notation [3], and the implementation done in the C++ programming language. This process is aided by the use of a CASE tool, Rational Rose/C++ [4], which generates skeletal C++ code directly from the diagrams and specifications created by the author.

To provide flexibility, rather than only providing a C++ interface for users of the NOMaD network toolset, a NOMaD-specific tcl extension is also being developed. Tcl [5] is a scripting language designed to be extended by the incorporation of application-specific commands, coded in C or C++. Consequently, `NOMaDsh` can be used either interactively to access the underlying network toolset—creating networks, modifying them, assessing their cost/performance,

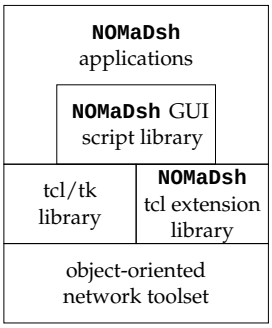

Figure 1: Overall NOMaD architecture

etc.—or via scripts written using a combination of tcl and the NOMaD-specific commands. In addition, a library of graphical user interface (GUI) scripts is being developed, using the tk toolkit (tcl/tk), that allow `NOMaDsh` scripts to edit and display networks.

Further details on the initial architecture of NOMaD can be found in [1], including the object-oriented network toolset, the tcl scripting layer and the GUI.

4 GA/Heuristic Hybrids

Within the field of GA research, there are two 'schools of thought' with respect to GAs. One approach [6] is to try and develop an algorithm that is robust and general in its application *i.e.* it makes use of no problem-specific information—the focus of the research is on the development of a better GA for all problems. The other [7] is to blend the traditional GA with problem-specific operators, often based on existing heuristic algorithms for a particular problem (or problem domain), such that the combined algorithm performs better than either a 'pure' GA or the heuristic alone. This approach is more pragmatic in nature, with the better solution of particular engineering problems being the primary focus of research.

The GA/heuristic hybrid approach thus adopts a problem-specific encoding, and then develops and employs problem-specific operators that together combine the best existing heuristics for the problem with an overall GA framework. The motivation for this approach is simple, according to Davis: "Although genetic algorithms using binary representation and single-point crossover and binary mutation are robust algorithms, they are almost never the **best** algorithms to use for any [specific] problem" [7]. It is almost always possible to develop a problem-specific GA/heuristic hybrid that will outperform either the heuristic or a traditional GA alone, and this is one of the approaches that is used in NOMaD.

For a brief survey of work on the application of GAs to network design, see [8], which indicates that even without the incorporation of heuristic-based operators, GAs are often able to match heuristics alone in terms of results, if not in speed. Building on this promise, GA/heuristic hybrids should prove even better for network design.

5 O-O Network Toolset

Clearly, NOMaD's object-oriented network toolset is too large to present in detail here. Consequently, we will focus on those elements that, on the one hand, represent the structures undergoing adaptation, and on the other, implement the overall GA/heuristic hybrid framework itself.

The main `NetworkModel` class diagram is given in Fig. 2, which shows the `Network` class, composed not only of `Node`s, `Link`s, and `PathLossSeq`s (ordered sequences of `Path`s), but also several implementation classes used to represent the `Network`'s adjacency matrix, connection matrix (which records `Link` levels in hierarchical networks) and the traffic requirements. To date, the focus has been on the virtual-topology design of both flat and hierarchical networks, but in future, the network model will also incorporate classes to represent a network's physical topology (*e.g.* `Duct`s and `Fibre`s). Within NOMaD, `Network` objects are themselves the structures undergoing adaptation—a highly-structured and very problem-specific representation that requires no decoding or interpretation.

Fig. 3 is the main `GA` class diagram, illustrating the `GeneticAlgorithm` class, with its `Population` of `Individual`s and its `OperatorPool` of `Operator`s. An `Individual` simply consists of a single `Network` object and some additional accounting information. As in Davis [7], the probabilities of individual `Operators` being applied to an `Individual` adapt through the course of a run using a credit-assignment algorithm, rather than being chosen with a fixed probability throughout. Whenever an `Individual` is created whose fitness exceeds that of the best `Individual` so far (`bestIdv`), the improvement in fitness is credited to the `Operator` responsible. In addition, a decreasing fraction of the credit (say, 0.5 and 0.25) is awarded to the parent `Operator` (*i.e.* the `Operator` that created this `Individual`'s parent, `parentOp`) and grandparent `Operator` (*i.e.* `grandParOp`), to ensure that two- and three-`Operator` sequences are rewarded appropriately. After a few generations (say, 4), a small proportion (say, 0.15) of the `Operator` probabilities

GA	ML	SPCO	LGCO	DEG2	DEGD
1	0.35	0.45			
2	0.35		0.45		
3	0.25		0.35	0.10	0.10

Table 1: Genetic algorithms

is reassigned according to the average credit earned per child each `Operator` has generated. In addition, `Operator`s are given a minimum probability (say, 0.05) to ensure they do not lose the opportunity to gain some credit if they, having decayed to the minimum level, are later found to be useful to further evolve the `Population`.

A variety of operators have been incorporated within NOMaD (by creating derived classes from either `UnaryOp` or `BinaryOp`) including those used in this paper: `MutateLink` (single-link mutation, ML), `SPLinkCO` (standard single-point crossover, SPCO), `LinkGroupCO` (link-group crossover, LGCO), `DegreeTwo` (degree-two operator, DEG2) and `DegreeDec` (degree-decrement operator, DEGD). Link-group crossover selects a group of nodes in one network (those whose distance from a randomly-selected node is less than or equal to the distance to a second randomly-selected node) and then exchanges the links used within that group with those used within the same group of nodes in the other network. It was believed that this would lead to better building blocks than a standard single-point crossover with a linear encoding of the network topology, as employed by some earlier authors [9, 10]. The degree-two operator alters a network by adding links to nearest nodes, where necessary, making all nodes at least of degree two (*i.e.* connected to two other nodes), to ensure adequate reliability and avoid a cost penalty. The degree-decrement operator simply deletes a random link from the highest-degree node (if there is one of degree greater than two), hopefully thereby reducing the network cost. In general, operators derived from `UnaryOp` are analogous to mutation in a traditional GA, but can be heuristics that incorporate arbitrary amounts of problem-specific information. Likewise, `BinaryOp` subclasses are analogous to traditional crossover, although once again, there is considerable potential for problem-specific heuristics.

6 Experimental Results

To assess the relative performance of the above operators, five test networks were generated using the approach described in [11], although further modified to ensure reasonable node separations. Each network had fifteen nodes, used a 1,000km × 1,000km

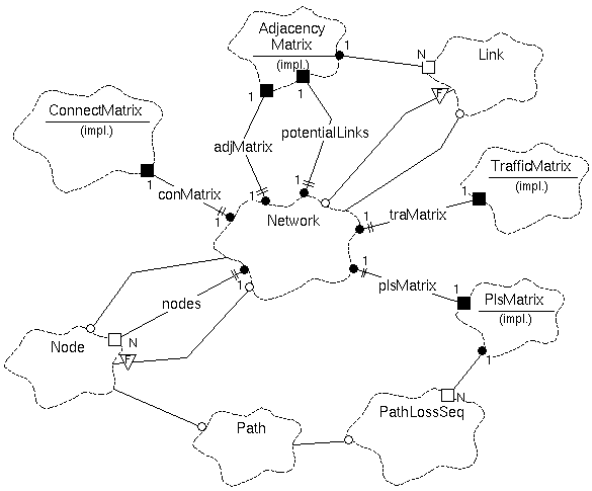

Figure 2: `NetworkModel` class diagram

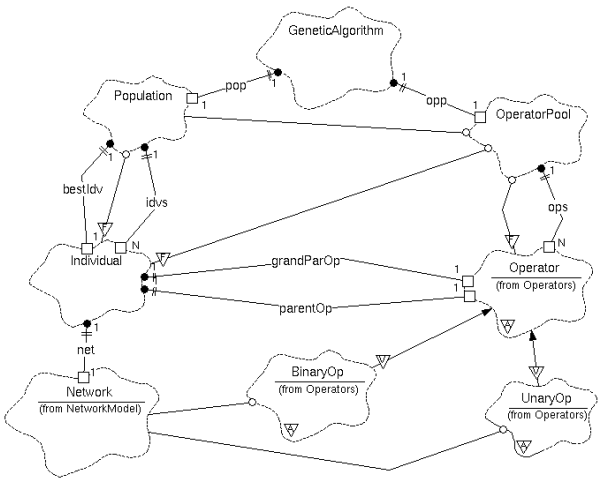

Figure 3: `GA` class diagram

area and carried an overall traffic of 1,500Gbit/s. The network costs were assessed using the model in [9].

Three genetic algorithms were employed. Each had a population size of 200, used tournament selection (of size 4), and was limited to a maximum of 5,000 trials (*i.e.* somewhat more than 25 generations). The operators used, and their initial probabilities (chosen with a few trial runs), are given in Table 1.

The three GAs were each applied five times with different random seeds to each of the five networks, and the lowest cost obtained for each group of independent runs. In addition, the lowest overall result for each network from the three different GAs was found, and used to calculate the difference in cost (above the best) of the results found for that network by the other two GAs (see Table 2).

Clearly, for these five test networks and the cost

Network	GA1		GA2		GA3	
	Cost	Diff.	Cost	Diff.	Cost	Diff.
1	4,974,888	20,906	4,953,982	0	4,965,731	11,749
2	4,785,018	34,002	4,751,016	0	4,756,635	5,619
3	4,449,713	34,311	4,415,402	0	4,426,182	10,780
4	4,619,478	20,579	4,604,944	6,045	4,598,899	0
5	4,461,352	34,021	4,427,331	0	4,430,814	3,483

Table 2: Results

model employed, link-group crossover (GA2) is uniformly better than single-point crossover (GA1). Surprisingly, however, combining the degree-two and degree-decrement operators with link-block crossover only produced an improvement in one case; in all the others, it actually seemed to reduce performance slightly.

7 Conclusions & Further Work

The use of a GA/heuristic hybrid approach in a tool for optical network modelling, optimisation and design (NOMaD) has been described and, in particular, its application to virtual-topology design. The experimental results show that the adoption of a problem-specific crossover can enhance GA performance in this context. However, incorporation of simple heuristic operators has so far provided more mixed results. It is anticipated that, as well as further exploring those operators already developed and the operator-probability adaptation algorithm, future work will investigate other, more powerful heuristic operators to incorporate into NOMaD, as well as addressing wavelength allocation and routing, and physical-topology design. For these latter problems, more difficult than virtual-topology design, the GA/heuristic hybrid approach may better fulfill its promise. Overall, although NOMaD is still being developed, these early results are encouraging, and it is envisaged that NOMaD will continue to play a central role in both network design and GA/heuristic hybrid research at the University of Essex.

References

[1] Sinclair, M.C.: NOMaD: Initial architecture of an optical network optimisation, modelling and design tool *Proc. 12th UK Performance Engineering Workshop* University of Edinburgh, 1996, pp.157–167

[2] Hill, A.M. & Houghton, A.J.N.: Optical networking in the European ACTS programme *OFC'96* San Jose, USA, 1996

[3] Booch, G.: *Object-Oriented Analysis and Design with Applications (2nd Ed.)* Benjamin-Cummings, 1994

[4] Rational Software Corporation: *Using Rational Rose/C++ (Revision 2.7)* 1995

[5] Ousterhout, J,K.: *Tcl and the Tk Toolkit* Addison-Wesley, 1994

[6] Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning* Addison-Wesley, 1989

[7] Davis, L.: *Handbook of Genetic Algorithms* Van Nostrand Reinhold, 1991

[8] Proestaki, A. & Sinclair, M.C.: Initial survey of heuristics for optical core network design in ACTS WOTAN *Proc. 13th UK Teletraffic Symposium* University of Strathclyde, Glasgow, 1996, pp.20/1–20/8

[9] Sinclair, M.C.: Minimum cost topology optimisation of the COST 239 European optical network *Proc. Intl. Conf. on Artificial Neural Networks and Genetic Algorithms* Alès, France, 1995, pp.26–29

[10] Hewitt, J., Soper, A. & McKenzie, S.: CHARLEY: A genetic algorithm for the design of mesh networks *GALESIA'95* University of Sheffield, 1995, pp.118–122

[11] Griffith, P.S., Proestaki, A. & Sinclair, M.C.: Heuristic topological design of low-cost optical telecommunication networks *Proc. 12th UK Performance Engineering Workshop* University of Edinburgh, 1996, pp.129–140

Acknowledgements

The author is grateful to Katerina Proestaki and Noel Parnis (also of E.S.E., University of Essex) for the many interesting and stimulating discussions during the continuing development of NOMaD. They are both supported by the European Commission under the ACTS programme: Ms. Proestaki on project AC029 (WOTAN) and Mr. Parnis on project AC066 (OPEN).