Basic Properties of Attribute Grammar Encoding

Talib S. Hussain
Computing and Information Science Dept.
Queen's University
Kingston, Ontario
hussain@qucis.queensu.ca

Roger A. Browse
Computing and Information Science Dept.
Psychology Dept.
Queen's University, Kingston, Ontario
browse@qucis.queensu.ca

Appeared in Late Breaking Papers at the Genetic Programming 1998 Conference (July 22-25, Madison, Wisconsin), p. 256.

An important problem in evolutionary computing is the design of genetic representations of neural networks that permit optimization of topology and learning characteristics (Yao, 1993). Neural networks that are modular in topology and function are ideal targets for the development of genetic representations. Modular architectures may be more suitable in addressing large scale problems, and it has been suggested that they are capable of decomposing problems into subtasks that facilitate solutions to complex problems (Jacobs et al., 1991). In this paper, the usefulness of the Network Generating Attribute Grammar Encoding (NGAGE) (Hussain and Browse, 1998) as a genetic representation for modular neural networks is discussed.

A promising approach to genetic representation of neural networks is the use of grammars to depict a process in which neural networks may be generated. An evolutionary algorithm may manipulate the instructions that generate a network, rather than manipulating the networks that are generated by the grammar. Cellular encoding is one such grammar-based system (Gruau, 1995). However, within cellular encoding, identical subtrees do not always expand in identical ways as they may be dependent on the context of the rest of the tree for the development of network connections. This characteristic emerges primarily from the application of cellular encoding's REC and WAIT productions, and affects the behavior of the genetic operators used to optimize genetic codes. In particular, a subtree crossover operator may not transfer fixed meaning.

NGAGE follows basic principles similar to those of cellular encoding, but endeavors to introduce a stricter notion of semantic identity to the subtrees in the genetic representation and a more explicit notion of a constituent module. It uses an attribute grammar to specify a neural network. An attribute grammar (Knuth, 1968) is a context-free grammar in which symbols have semantic attributes and production rules specify not only the replacement of symbols, but also the evaluation of the attributes of those symbols. In NGAGE, a sequence of production rules, represented in a tree, is used as the genetic representation.

An NGAGE grammar analogous to the basic cellular encoding is presented in Figure 1. The grammar defines a module as the result of applying a subtree in the gene to a non-terminal symbol M. A module is characterized by a set of nodes, which includes externally accessible input and output subsets, and a set of connections among all the nodes. Productions outside of a subtree may not effect changes to the internal connectivity of the module that subtree represents; they may only add new external connections to the input and output nodes of the module. These connections may, without loss of power, be considered not part of the semantic identity of the module.

Start: $S \rightarrow M$
In_Nodes **of** S = In_Nodes **of** M;
Out_Nodes **of** S = Out_Nodes **of** M;
All_Nodes **of** S = All_Nodes **of** M;
Connections **of** S = Connections **of** M;

SEQ: $M_1 \rightarrow M_2\ M_3$
In_Nodes **of** M_1 = In_Nodes **of** M_2;
Out_Nodes **of** M_1 = Out_Nodes **of** M_3;
All_Nodes **of** M_1 = [All_Nodes **of** M_2 $\cup$ All_Nodes **of** M_3];
Connections **of** M_1 = [Connections **of** M_2 $\cup$ Connections **of** M_3 $\cup$
full_connect (Out_Nodes **of** M_2, In_Nodes **of** M_3)];

PAR: $M \rightarrow L_1\ L_2$
In_Nodes **of** M = [In_Nodes **of** L_1 $\cup$ In_Nodes **of** L_2];
Out_Nodes **of** M = [Out_Nodes **of** L_1 $\cup$ Out_Nodes **of** L_2];
All_Nodes **of** M = [All_Nodes **of** L_1 $\cup$ All_Nodes **of** L_2];
Connections **of** M = [Connections **of** L_1 $\cup$ Connections **of** L_2];

MAP: $L \rightarrow M$
In_Nodes **of** L := In_Nodes **of** M;
Out_Nodes **of** L := Out_Nodes **of** M;
All_Nodes **of** L := All_Nodes **of** M;
Connections **of** L := Connections **of** M;

END: $M \rightarrow n$
In_Nodes **of** M := [node];
Out_Nodes **of** M := [node];
All_Nodes **of** M := [node];
Connections **of** M := [];

Figure 1: Modular NGAGE grammar

Future research on NGAGE shall address the inclusion of localized learning as properties of the terminals, the formal specification of existing neural network models as grammars, the modular combination of grammars from multiple models, and the selection and design of genetic operators which exploit the properties of NGAGE.

Gruau, F. (1995) "Automatic definition of modular neural networks," *Adaptive Behavior*, *3*, p. 151-183.

Hussain, T. and Browse, R.A. (1998) "Network generating attribute grammar encoding," *1998 IEEE International Joint Conference on Neural Networks*, p. 431-436.

Jacobs, R.A., Jordan, M.I., Nowlan, S.J. and Hinton, G.E. (1991) "Adaptive mixtures of local experts," *Neural Computation*, *3*, p. 79-87.

Knuth, D. E. (1968) "The semantics of context-free languages," *Mathematical Systems Theory*, *2*, p.127-145.

Yao, X. (1993) "Evolutionary artificial neural networks," *International Journal of Neural Systems*, *4*, p. 203-222.