How to Become Immune to Facts

M. J. Coombs, H. D. Pfeiffer, W. B. Kilgore and R. T. Hartley

Computing Research Laboratory
New Mexico State University
Box 30001/3CRL
Las Cruces, NM 88003-0001
mcoombs@nmsu.edu

ABSTRACT

Though many of the current symbolic problem solvers have attempted to combat "brittleness" with learning procedures, here at the Computing Research Laboratory (CRL) an alternative approach has been developed based on achieving symbolic plasticity through model generation. This paper describes the evolutionary-Model Generative Reasoning (e-MGR) architecture that implements this approach, and discusses e-MGR's from a as a "connectionist" system perspective.

1. Introduction

A widely reported result concerning knowledge-based, symbolic problem solvers is that they are brittle in the sense that they only respond as intended in narrow, well-specified domains (Coombs and Alty, 1984; Fields and Dietrich, 1987; Hewitt, 1985; Holland, 1986). This is a serious weakness, since few prototype systems survive the process of being upgraded to a full scale (real-world) problem solving system.

Researchers have frequently responded by incorporating a general learning procedure into a symbolic system. The Model Generative Reasoning (MGR) project at the Computing Research Laboratory (CRL) (Coombs and Hartley, 1987; Coombs et al., 1989) has questioned this practice; noting that, while a procedure that can learn may improve performance on subsequent problems, it will not generate even partial solutions to the problem in hand. What is required is a general method to confer plasticity (the ability to alter behavior appropriately the first time a novel problem is encountered) on automated problem solvers. The MGR project is concerned with devising symbolic processing architectures to confer such plasticity.

2. Plasticity in Problem Solving

Brittleness typically occurs when task environments are sufficiently open, or complex, that all relevant relationships between potential input (facts) and knowledge (domain definitions) cannot be defined reliably in advance. Observation of human problem solving in such an 64 unstructured" task environment indicates that success more frequently results from flexible application of existing concepts and strategies (see Tolcott et al., 1989; Woods, 1986), and rarely requires the learning of new concepts over multiple trials. Human problem solvers readily create new relevance relations between existing concepts, achieving plasticity through the novel application of existing knowledge rather than through the creation of new knowledge from irregularities noted over multiple trials.

Conventional problem solvers in artificial intelligence (AI) typically pass through a 44 generate/evaluate" control cycle, where problem solving operations are determined by the evaluation stage. MGR achieves plasticity by removing "evaluate" from the control cycle and driving problem solving *within* the generation stage through feedback from characteristics of the

current set of proposed solutions to the operators that generate new solutions. These operators create new implicit relevance relations by the automated modification of knowledge structures, while the role of evaluation is reduced to a decision procedure for terminating the generation stage. This approach is related to problem solving using artificial neural networks, where novel solutions are created by the *superposition* of previous solutions, rather than the method of discrete conjunction largely used by symbolic systems.

In the search for mechanisms to achieve symbolic plasticity, we have been struck by the similarity between MGR and its treatment of unstructured task environment (Coombs and Hartley, 1987; Fields et al., 1988a; 1988b), and the immune network model of the mammalian immune system (Fanner et al., 1986). This similarity has since been reinforced by the comparisons made in Farmer (Fanner, 1990 "The Rosetta Stone for Connectionism") who provides a framework for unifying neural networks, classifier systems, and immune nets as classes of 44 connectionist" systems. These systems are defined as dynamical systems where: (i) variables are constrained to a finite list of connections, and (ii) their structures can change with time. The comparison is made using 7 categories, namely *nodes, states, connections, parameters, interaction rule, learning rule,* and *graph dynamics.* We refer to the view of MGR discussed in this paper as evolutionary-MGR (e-MGR) and sketch the architecture using these categories.

3. **An Immune Model of Reasoning**

The basic task of the immune system is to distinguish between chemical patterns that represent self and non-self, and to attempt to eliminate non-self (Fanner, 1990 p24). This task is extremely difficult, being estimated by Fanner as comparable to recognizing a million human faces (Farmer, 1990 p25). These are recognized, to mix metaphors, through the production of 10' distinct antibodies from 105 genes. Moreover, it is achieved through the use of three building blocks: (i) *antibodies,* molecules that serve to tag, and so classify, foreign material (antigens); (ii) *macrophages,* cells that remove material tagged by antibodies and influence the production of the cells that produce antibodies; (iii) *lymphocytes,* cells that both produce antibodies and discriminate different classes of antibody.

By analogy, the basic task of a problem solver may be seen as distinguishing between explicable and inexplicable facts, F challenging the system from the world, and neutralizing the latter by fusing them together into explanatory model structures, M, using stored definitions, D, as "glue". This is achieved through three operation that correspond to the three immune system operations of tagging with antibodies, removal of tagged material by macrophages, and production of antibodies by lymphocytes. These are: (i) *classification,* which tags subsets of facts from the set F with "antibody" assumptions from the set A to give the set of tagged facts T; (ii) *specialization,* in which a set of tagged facts T is neutralized by fusion with subsets of definitions from the set D to give a set of models M; *(iii) fragmentation,* which produces new assumptions A' by mapping memories from the set of tagged facts T onto a superset of the tagged facts in the models M.

These operations can be represented as a closely coupled set of functions, with coupling at F, M and A. In the worst case:

1. Classification

$$Cl: \mathrm{A} \times 2^{F} \rightarrow 2^{T}$$

2. Specialization

$$Sp: 2^{\mathrm{T}} \times 2^{\mathrm{D}} \rightarrow 2^{\mathrm{M}}$$

3. Fragmentation

$$Fr: 2^{\mathrm{M}} \times 2^{\mathrm{T}} \rightarrow 2^{\mathrm{A}}$$

4. The e-MGR Architecture

Building on the earlier MGR architecture (Coombs et al., 1990), the e-MGR is logically a multi-instruction, multi-data parallel virtual machine (MIMD) that accepts input from the databases F, D, and M. F is a fact database that receives all input from external agents; D is a definition database that contains all of the system's persistent expectations; M is a model database that contains all expectations currently under development. In contrast to the basic MGR architecture, in e-MGR we partition M into three sections: (i) A, which is an assumption database containing system constructions deemed to be sufficiently fit to regard as factual "for all practical purposes"; (ii) T, which is a database of tagged items selected from A and F to be presented to e-MGR for interpretation during the current cycle; (iii) M itself, which as in MGR is the database of current interpretations. All objects in the databases are represented as connected, multi-labeled, bipartite, oriented graphs called *conceptual structures* CS*;* each operator therefore works at a graph level. The organization of the operations into the e-MGR system architecture is represented in Fig. 1.

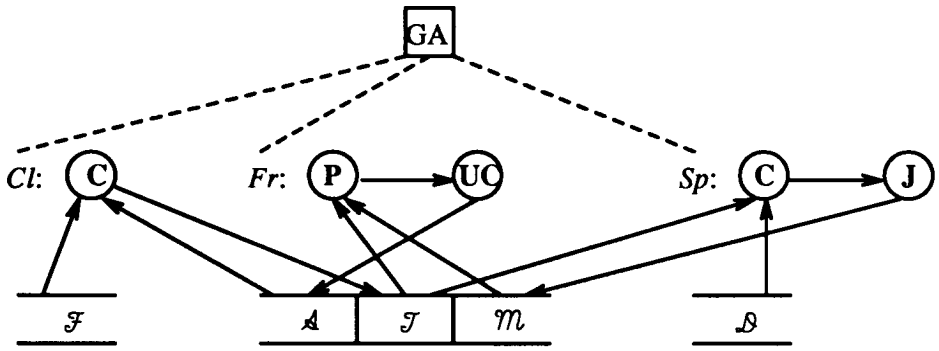

Figure 1: A data-flow diagram of the E-MGR architecture. Detailed description of the lower-level operators join, **J**, cover, **C**, project, **P** and uncover, **UC**, are given in Coombs et al., (1990). Informally, **C** identifies a subset of definition graphs that has some predefined set cover relation to all of the labeled nodes in a given subset of tagged fact graphs; **J** merges two graphs at a single point where both graphs contain related node labels; **P** is the inverse of join in that it seeks to identify related labels between graphs; **UC** is the inverse of cover in that it partitions graphs in the neighborhood of sub-graph boundaries.

Using Farmer's (Farmer, 1990) categories, assumptions are the system *nodes* (antibody types) since they represent the classes of fact that the system can recognize on each cycle and thus can be made available by tagging for ingestion. They therefore represent the current set of system relevance relations, and as such, are the objects upon which system fitness values are defined. This also makes them the principal source of halting rules (e.g., the existence of a single assumption that incorporates all the facts in F).

System states (free antibody/antigen concentrations) are defined in terms of the fitness values of the current set A, and mappings between A and F resulting in the set T.

Connections (chemical reaction of antibodies) are established between the graph objects assumptions, A, facts, F, tagged facts, T, and definitions, D - via the three operators - classify, *Cl*, specialize, *Sp*, and fragment, *Fr*. These operators are integrated into the action-part of classifier-type control strings, where the condition part consists of a set of pre-defined system state variables (e.g. measures of the *complexity*, or fact *covering* power of the graphs in A).

System *parameters* (reaction affinity, lymphocyte concentration) are strength and specificity relations defined between the Witnesses of the current assumption set A and the values of potential state variables specified in the condition part of the control strings. Strength and specificity are defined following classifier system guidelines (Holland, 1986).

Graph dynamics for e-MGR will be implemented using a genetic algorithm. Control is thus seen as a process of breeding a set of control strings through reproduction, crossover, and mutation under feedback from the fitness measures on the assumption population in order to optimize some desired relationship between A and F (Goldberg, 1983). This enables us, in our basic system, to use the bucket brigade as a *learning algorithm* with a simple linear threshold *interaction rule.*

5. Simulations of e-MGR

Reasoning in e-MGR may be characterized as an optimization process in which conceptual "glue" is added and subtracted from facts in order to equilibrate the system. A simulator has therefore been implemented to permit the exploration of different gluing strategies, and their effectiveness in different task environments defined in terms of the initial conditions of F and D. In particular, we have focused on the conditions required for convergence in e-MGR to a stable set of "attractor" models.

As an example of a problem that can be analyzed using the simulator, we have chosen Fuller's "Four Triangles Out of Two" problem (Fuller, 1975). Figure 2 gives the initial conditions given to the simulator.

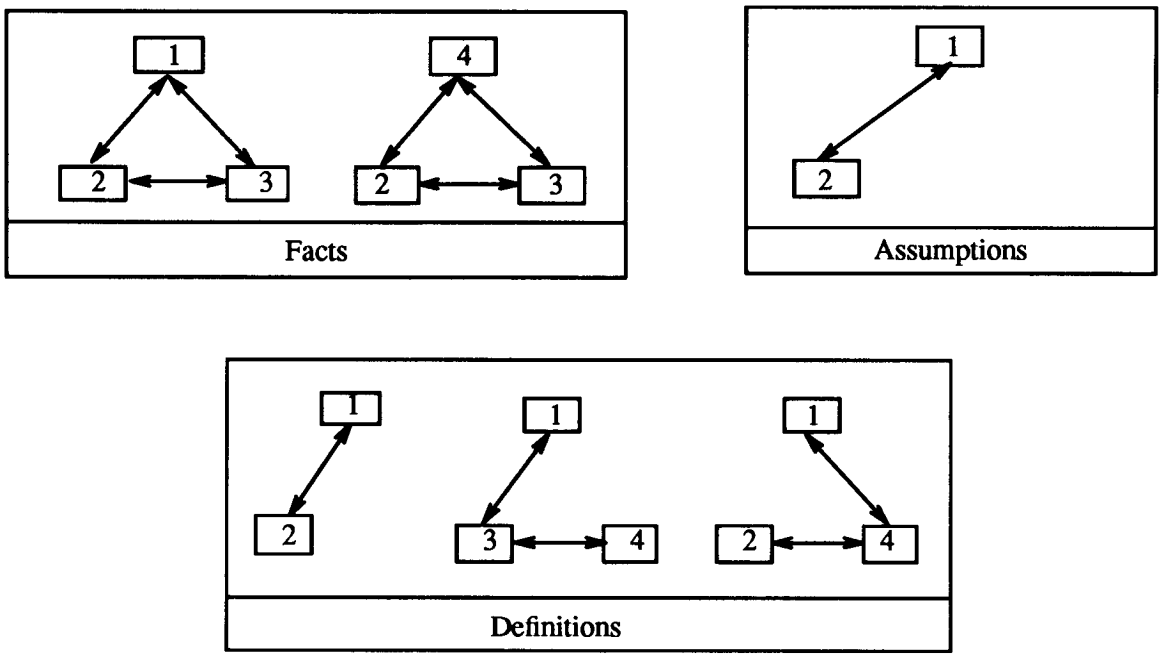

Figure 2. Given are the three sets F = $\{C_i\}$, A = $\{C_j\}$ and D = $\{C_k\}$ of bipartite graphs that form the starting conditions to the "Four Triangle" example.

Through the cycling of the operators by the simulator, a process of "adding" and "removing" glue, different gluing strategies can be analyzed. For this initial four triangle problem, cover adds the least glue to fuse the set of tagged facts, and uncover may leave a single glue node in place around fragments. Given the two triangles environment, this strategy

robustly converges to the four triangles "attractor" model depicted in Figure 3.

Further testing of different glueing strategies (covering and uncovering methodologies) are currently being evaluated.

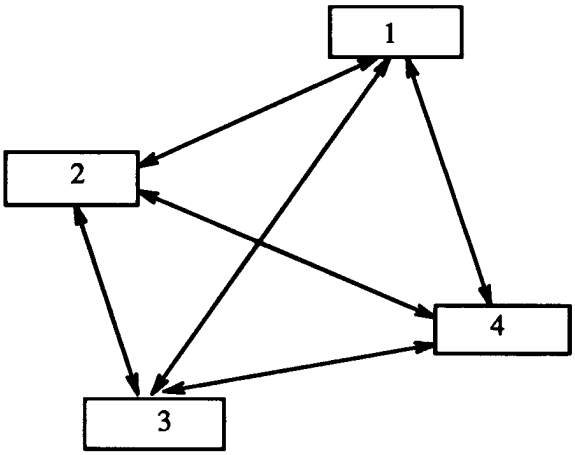

Figure 3. Given is the "attractor" model found in the "Four Triangle" example.

6. Issues and Relations

Our major interest in the e-MGR architecture is to explore the origin, and value, of widely used AI reasoning principles such as parsimony and concilience. In e-MGR approach these concepts are defined through the specification of an appropriate recognition and transformational dynamics for graph structures.

It may be noted that the three e-MGR operations can be interpreted both in terms of the evolutionary cycle → *amplification* → *mutation* → *selection* → and in terms of Peirce's (Peirce, 1934) explanation cycle → *induction* → *abduction* → *deduction* →. The *Cl* operator implements an amplification process on facts, classifying them with reference to concept clusters that have evolved to be relatively persistent within the system. This can also be seen as a form of induction, a relevance relation being induced between a fact and a persistent assumption. Similarly, Sp implements a mutation process by which tagged facts are integrated using definitional material to form alternative, specialized interpretations (models). Alternatively, the system abduces an interpretative context for the tagged facts. Finally, Fr implements a process by which a new set of assumptions are selected to tag facts in the next cycle. Fragmentation is a deductive process by which new assumptions are deduced from models. It is anticipated that these relations will help us integrate mechanisms of reasoning with the "logical" viewpoint central to symbolic AI.

References

Coombs, M. J. and J. L. Alty (1984) Expert systems: An alternative paradigm. *international Journal of Man-Machine Studies* 20: 21-44.

Coombs, M. J. and R. T. Hartley (1987) The MGR algorithm and its application to the generation of explanations for novel events. *International Journal of Man-Machine Studies* 27: 679-708.

Coombs, M. J., C. A. Fields, and R. T. Hartley (1990) A Brief Overview of the Model Generative Reasoning (MGR) Problem Solving Architecture. MCCS-89-168, CRL, Las Cruces, NM.

Farmer, J. D. (1990) A Rosetta Stone of Connectionism. LA-UR-90-228, Los Alamos National Laboratory, NM.

Farmer, J. D., S. A. Kauffman, N. H. Packard, and A. S. Perelson (1986) The immune system, adaptation, and machine learning. *Physica*, 22D: 187-204.

Fields, C. A. and E. S. Dietrich (1987). Transition virtual machines. I: The TVM problemsolving architecture. Technical Report # 3CR/Al-87-06.

Fields, C. A., M. J. Coombs, E. S. Dietrich, and R. T. Hartley (1988a) Incorporating dynamic control into the Model Generative Reasoning system. *Proceedings of ECAI-88.* London: Pitman, 439-441.

Fields, C. A., T. Eskridge, R. T. Hartley and M. J. Coombs (1988b) Experimental analysis of dynamic control strategies for the MGR architecture: Simulation environment and initial results. *Proceedings of the Seventh Conference of the Society of Artificial Intelligence and the Simulation of Behaviour.* London: Pitman.

Fuller, R. B. (1975) *Synergetics Exploration In The Geometry Of Thinking.* New York: Macmillan Publishing Co, Inc.

Goldberg, D. E. (1983) Computer-aided gas pipeline operation using genetic algorithms and rule learning. Doctoral dissertation, University of Michigan. *(Dissertation Abstracts International* 44: 3174B)

Hewitt, C (1985) The challenge of open systems. *Byte 10:* 223-242.

Holland, J. H. (1986) Escaping brittleness: The possibilities of general-purpose machine learning algorithms applied to parallel rule-based systems. In: R. Michalski, J. Carbonell, and T. Mitchell (Eds) *Machine Learning: An Artificial Intelligence Approach*, Vol. 2. Los Altos, CA: Kauftnann.

Pierce, C. S. (1934) Scientific method. In: A. W. Burks (Ed) *Collected Papers of Charles Saunders Pierce*, Harvard: Harvard University Press.

Tolcott, M. A., F. F. Marvin and T. A. Bresnick (1989) *Situation Assessment and Hypothesis Testing in an Evolving Situation.* Technical Report 89-3, Decision Science Consortium Inc, Contract # MDA903-86-C-0332, Army Research Institute.

Woods, D. D. (1986) Paradigms for intelligent decision supporL In: E. Hollnagel, G. Mancini and D. D. Woods (Eds) *Intelligent Decision Support in Process Environments.* Heidelberg: Springer-Verlag.