Predicting Perceptual Chunks with a Computational Model of Problem Solving with Diagrams

PETER C. R. LANE, FERNAND GOBET AND PETER C–H. CHENG
ESRC Centre for Research in Development, Instruction and Training
School of Psychology, University of Nottingham
University Park, Nottingham NG7 2RD
United Kingdom
email: {pcl,frg,pcc}@psychology.nottingham.ac.uk

ESRC Centre for Research in Development, Instruction and Training
March 2000
Technical Report No. 67

Abstract

This paper describes CHREST+, a computational model which learns perceptual chunks to solve problems using a diagrammatic representation. Perceptual chunks are pieces of familiar information, as retrieved by a sensory device. In earlier work on chess expertise, a successful computational model, CHREST, has been developed of how such chunks can be acquired and stored in a discrimination network. CHREST+ is an extended version which learns associations between chunks for problem and solution states to create a knowledge base of information for problem solving. We compare the use of chunks by the model and human subjects in a problem-solving domain where unknown quantities are computed from electric circuits using a diagrammatic representation. We also discuss how the learning mechanisms of CHREST+ differ from those of ACT-R and Soar.

Email: {pcl,frg,pcl}@psychology.nottingham.ac.uk
Phone 0115 9515302, Fax 0115 9515324

1 INTRODUCTION

The use by humans of perceptual chunks is a central idea in the literature on skill acquisition (e.g. Simon, 1981). However, computational models of problem solving have not always made such a representation central to their learning mechanisms. Hence the role of pure chunking in problem solving has not been sufficiently investigated to determine where more complex knowledge is required. In addition, theories of efficient representations (e.g. Larkin & Simon, 1987) suggest that concrete visual information is preferentially used by humans over the abstract, making a model which learns and uses such information of interest to developers of such representations. The aim of this paper is to show that perceptual chunks can be learnt in a computational model and used to support problem-solving behaviour with a diagrammatic representation, and to compare the model with the performance of human subjects.

This paper proceeds as follows: In the first section we define CHREST+, and describe its mechanisms for retrieving and storing perceptual chunks. The second section contains a description of a problem-solving domain, where subjects and the model use a diagrammatic representation to compute unknown quantities in electric circuits. Data from an experimental study is used to support the theory that humans learn perceptual chunks. The section also compares the performance of CHREST+ with that of the subjects. The third section compares the learning mechanisms of CHREST+ with those of ACT-R and Soar.

2 CHREST+ : LEARNING PERCEPTUAL CHUNKS FOR PROBLEM SOLVING

CHREST+ is an extended version of CHREST, originally derived from the EPAM (Elementary Perceiver and Memorizer) model of Feigenbaum and Simon (1984). CHREST (Chunk Hierarchy and REtrieval STructure) contains two important extensions to the discrimination network learning mechanisms from the earlier EPAM model: first, CHREST can associate information between different nodes using lateral links (Gobet, 1996) and second, information from several nodes may be grouped into a larger structure known as a template (Gobet & Simon, in press). This model has proven highly successful in matching the recall performance of human chess players using both game and random positions (Gobet & Simon, in press). Further applications of CHREST include learning to play chess (Gobet & Jansen, 1994), the balance-beam task (Gobet, 1999) and linguistic phenomena (Croker, Pine & Gobet, 2000; Gobet & Pine, 1997; Jones, Gobet & Pine, 2000).

The CHREST+ model further adds an output device (pencil) and specific links (equivalence links) in the discrimination network to encode information about which chunks are ‘solutions’ to which other chunks. The model is illustrated in Figure 1, and described as follows: first, how patterns are acquired; second, how patterns are stored within the discrimination network and so become familiar chunks; third, how chunks are combined within STM to form larger chunks; fourth, how the model uses stored information to help it solve problems.

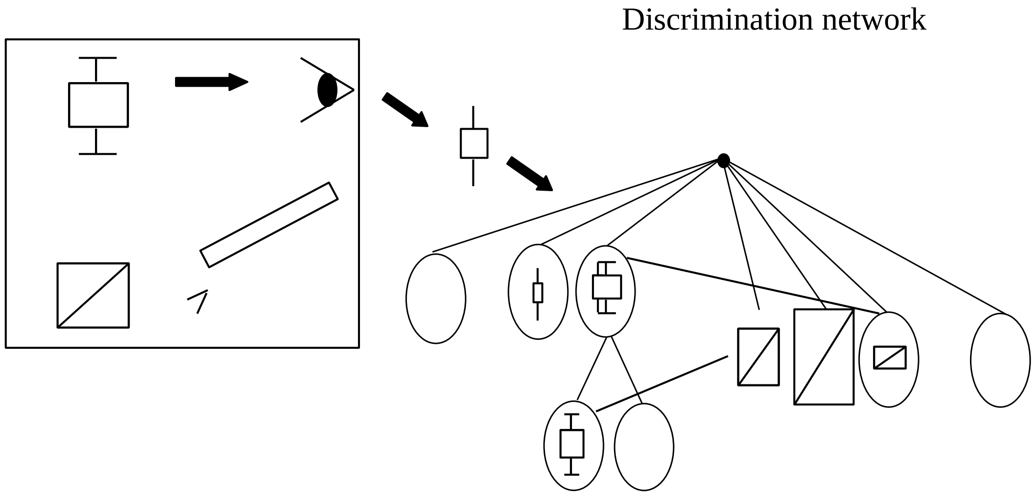


Figure 1: The CHREST+ model

2.1 Acquiring patterns from the environment

Perceptual chunks are pieces of familiar information retrieved by a sensory device; although a variety of patterns may be retrieved, the term ‘chunk’ refers to any unit of information that has been familiarised and has become meaningful (Richman, Gobet, Staszewski & Simon, 1996). Examples of chunks include: collections of letters retrieved whilst reading, collections of pieces on a chess board, and groups of components in an electric circuit. The CHREST model is based on the visual acquisition of chunks, as in chess expertise (Gobet & Simon, in press), and we employ the same mechanisms for retrieving and combining perceptual chunks in CHREST+.

Initially, patterns are input to the model via a simulated eye, which obtains information from a restricted field of view. The simulated eye has two essential procedures: first, a default strategy for eye movement, e.g. from top to bottom, or following salient features such as lines; and second, a means for representing what is being seen, i.e. the features used to describe the pattern.

In order for the model to work effectively, the eye-movement strategy and representations used must be appropriate to the domain under investigation. However, certain fundamental principles can be used to design a good representation before testing the model. These principles require the patterns retrieved from the environment to be generalisable and represent a meaningful decomposition of the domain. In the chess domain, this level is naturally that of individual pieces, in the electric circuit domain described below, this level is that of interconnected resistors. Patterns acquire the status of perceptual chunks once they have been learnt in the discrimination network, a process described next.

2.2 Storing patterns in the discrimination network

CHREST+ uses a hierarchical retrieval structure, known as a *discrimination network*, as its long-term memory. Information learnt by the system is stored in *nodes* which are interconnected with *test links*; this stored information, the node *images*, are the chunks learnt by the system. Learning a pattern begins by sorting it through the network, beginning from the *root node.* Sorting proceeds by following test links; each test link can only be traversed if the item of information on the link is a part of the current pattern. When no further test links can be followed, the image at the node reached is returned as the chunk retrieved by that pattern. The pattern is then compared with the image: If the pattern matches the image, then *familiarisation* occurs, in which the image is augmented with information from the pattern. If the pattern mismatches the image, then *discrimination* occurs, in which a new node and new test link are added to the node reached; the new node will have an empty image. The part of the pattern which mismatches the image is sorted through the discrimination network, beginning from the root node, and the image of the node reached is used as the test for the new test link; thus, the tests in the network are themselves chunks, and so reflect the amount of information in the network.

One further learning mechanism is the ability to form *equivalence links*. These links are formed based on conditions in the STM, and are described in the next section.

2.3 Combining chunks in STM

The Short-Term Memory (STM) acts as a store for perceptual chunks which the model is currently processing. Patterns perceived by the eye are initially passed to the discrimination network for sorting and learning. As each pattern is sorted through the network, the image at the node reached is retrieved and entered into STM; this is the perceptual chunk identified by the system in the current eye fixation. However, the eye has a restricted field of view from which to retrieve information, and most domains require larger pieces of information to be manipulated. The formation of the larger units is the primary function of STM, which is essentially a queue of chunks. Various lengths of STM can be used, but in the experiments in this paper the length is set to 4. One of the chunks in STM is identified as the *hypothesis* chunk: this chunk gathers together information across several eye fixations, thus representing the largest chunk currently observed.

Information is accumulated in the hypothesis as follows. Initially, STM is empty. On the first eye fixation, the chunk retrieved from the discrimination network based on the observed pattern is placed in the hypothesis slot. On the second eye fixation, the retrieved chunk is first placed into the STM queue. This chunk is then combined with the hypothesis chunk, the combined chunk is placed in the hypothesis and passed to the discrimination network for learning, as if it were a directly observed pattern. The old hypothesis will then enter the STM queue.

STM has a further role for managing how information for problem solving is learnt. The system will naturally encounter information about problem states and solution states. In the example domain below, each of these states is a diagrammatic representation, and during training the model is shown a diagram for the problem and a diagram for its solution. When STM contains separate chunks for a problem and a solution diagram, then an association between these two chunks is created in the discrimination network, known as an *equivalence link*. The next section describes how these links are used for problem solving.

2.4 Problem solving with CHREST+

The eye's limited field of view necessarily means that the chunks learnt by CHREST+ will be variously sized sub-elements of the perceived object. The learning mechanisms for forming the discrimination network tend to place these chunks into a hierarchy, with smaller chunks near the root of the network, and larger chunks further away, where more test links must be traversed to reach them. Problem solving with CHREST+ begins by decomposing the problem into chunks, and creating a solution for each chunk independently (this possibility is an important requirement for the generalisation of information learnt by CHREST+).

Recognising a chunk in the given problem may require more than one eye fixation before attempting to construct a solution for it. Hence the model, after each eye fixation, considers the new chunk formed from combining the eye fixation with the previous hypothesis (i.e. the current largest chunk so far recognised). The combined pattern is searched for in the discrimination network: if this combined pattern is not the image of some node in the network, then it is unfamiliar to the system. The model then constructs the partial solution using the previous hypothesis, which was a recognised chunk. After which the model proceeds with the chunk from the current eye fixation as the new hypothesis from which to look for a further recognisable chunk.

Chunks are used for constructing steps in the solution. This is done directly if they are linked to a second chunk by an equivalence link; this second chunk is used to provide the set of steps to include in the solution. If not, the hierarchy of the network is utilised to retrieve a solution for a subset of the chunk, by looking at its parent chunks in the network to see if they have equivalence links. In this manner, the model attempts to solve a given problem based on its decomposition into perceptual chunks. The actual steps taken by the model in solving each chunk depends on the domain; we describe one example in the next section.

3 COMPUTING UNKNOWN QUANTITIES IN ELECTRIC CIRCUITS

In order to evaluate the effectiveness of a problem-solving model which learns perceptual chunks, it must be tested in a domain where such chunks are the key to successful learning. We have chosen the task of computing unknown quantities in electric circuits using the diagrammatic representation known as AVOW (Amps Volts Ohms Watts) diagrams, which represent electric circuits and the domain laws of electricity in terms of diagrams and constraints in their composition. AVOW diagrams are described in Cheng (1999), and Figure 2 provides an overview. Essentially, each resistor in an electric circuit is represented as a separate AVOW box; the dimensions of the box are scaled to represent the indicated quantities. Composition of individual boxes is used to represent a circuit of several resistors; the rules for composition preserve the underlying physical laws of electric circuits. In working with this representation, subjects and the model must first produce a scaled AVOW diagram using the dimensions for the provided quantities; the constraints in the diagram ensure all the rules of electricity are followed, and so the subject can ‘read off’ the value of any unknown quantity simply by measuring the appropriate dimension. This section contains an overview of the results from one experiment on human subjects, investigating their use of perceptual chunks; more detailed analyses may be found in Cheng (1998; 1999a) and Lane, Cheng and Gobet (submitted). Understanding how humans learn using these kinds of representations is an important element in the design of effective educational material (Cheng, 1999b).

(c) Illustrated … e not shown).

3.1 Human subjects

Six subjects[1] were asked to construct AVOW diagrams using an electronic sketchpad; the computer retains records of each drawing and measuring action, and various timings. After an initial 15 minutes' training session in using the electronic sketchpad, subjects were presented with a graded sequence of problems, working up to relatively complex circuits containing up to twelve resistors. After each circuit was attempted, the correct AVOW diagram and solution were shown to the subjects. Our interest here is whether, for the complex circuits, subjects showed any evidence of 'chunking'. To test for this, we analysed the timing and sequences of drawing operations on the complex circuit shown in Figure 3(a), with its solution in Figure 3(b). Figure 4 contains graphs for three of the subjects. The graphs plot the reflection times for each action in the subjects' solution of the test problem: reflection time is the time between the end of one action and the start of the next. As the graphs show, the subjects produce their solutions in stages, with peaks in the reflection times followed by a succession of more rapid actions. The stages completed by the six subjects show a roughly similar pattern, with similar parts of the circuit being completed within each stage. We interpret these results as supporting the role of perceptual chunks in problem solving.

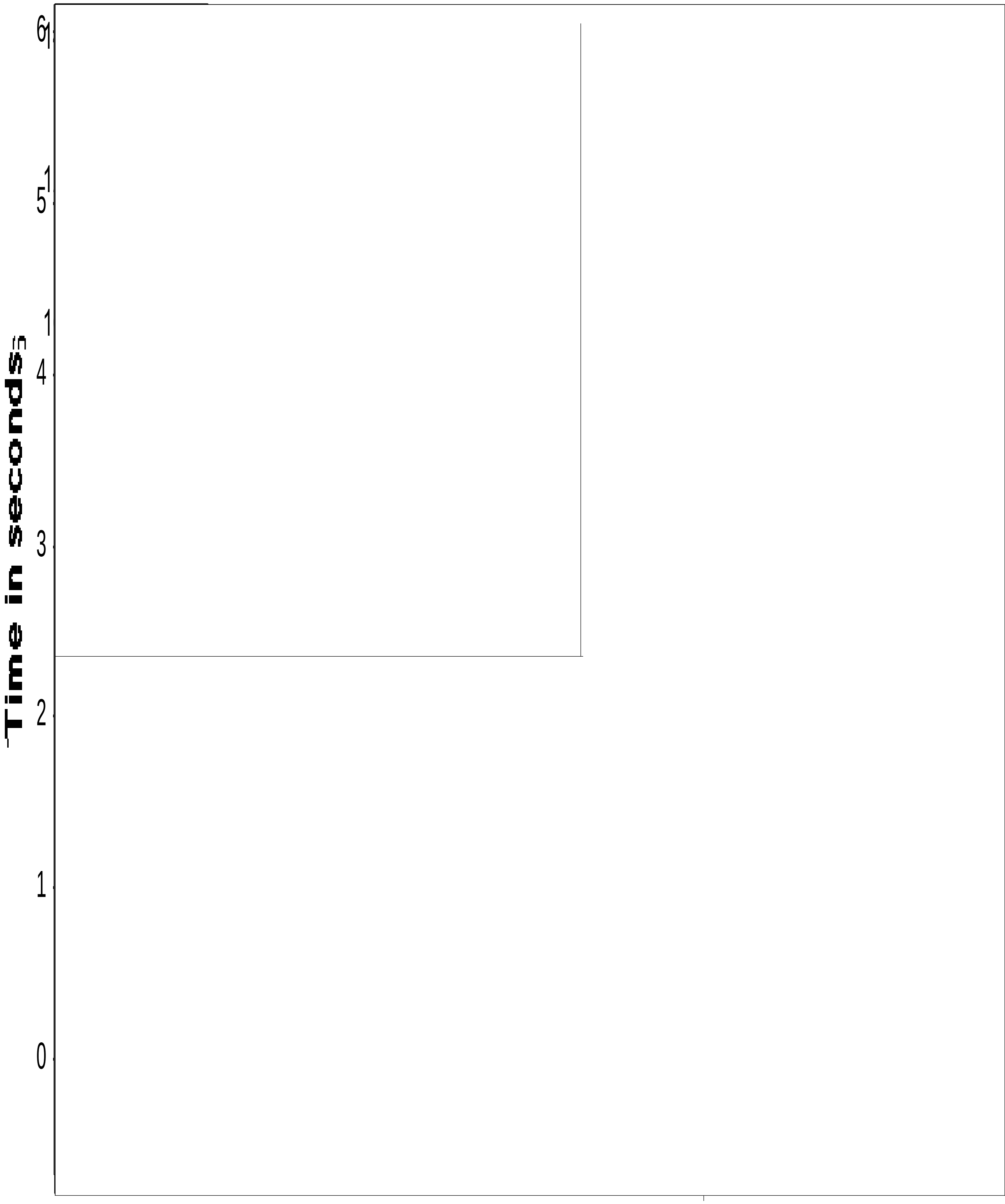
[1] The authors would like to thank Lucy Copeland for conducting this study.

3.2 CHREST+

To apply CHREST+ to this domain, two sets of representations and two sets of mechanisms are required. The two mechanisms are for: *input*, converting what is seen into an internal representation; and *output*, converting an internal representation into an external one. Two input representations are required: one for the circuit diagrams based on the input mechanism, and one for the AVOW diagrams, based on the input and output mechanisms.[2] Based on observations of human subjects, and the principles described earlier for representing perceptual chunks, the internal representations for circuit and AVOW diagrams captures their topological form only; circuits are represented as interconnected resistors and AVOW diagrams as aligned boxes. The output mechanism for drawing an AVOW diagram must incorporate the quantitative information of the appropriate line length, and thus, for each unconstrained line, the system attempts to locate the required quantity from the circuit diagram or, if this is not present, simply draws a line of arbitrary length.

We tested CHREST+ on the same sequence of problems as in the study on humans; for each problem, CHREST+ was required to produce an AVOW diagram, and was then given the correct diagram for training. As CHREST+ retrieves perceptual chunks from its network, the model reports back during solution which chunks it had identified and was using. Also, as intended drawing operations become constrained during solution, they are output by the model in rapid succession; these two forms of feedback enable us to compare the chunking performance of the model with that of the subjects. The chunks retrieved by the model are shown in Figure 3(c); the correspondence of the chunks with those of the human subjects is shown by the numbering in Figure 4. The total amount of information learnt during the above test may be estimated from the size of discrimination network: 72 chunks were learnt (42 circuit and 30 AVOW), with 11 equivalence links.

[2] The model only computes AVOW diagrams from circuits; the reverse task would require an additional output mechanism for drawing circuits.


Time in seconds
6
5
4
3
2
1
0

(a) CP

HP V

(c) E

LA

Figure 4

on the s

4 DISCUSSION OF LEARNING MECHANISMS

The previous sections have described CHREST+, a computational model which learns perceptual chunks to solve problems using a diagrammatic representation. Although CHREST+ is currently restricted in the kinds of problem it can tackle, it is worth making a brief comparison of its learning mechanisms with those in the more established models of ACT-R (Anderson & Lebiere, 1998) and Soar (Laird, Rosenbloom & Newell, 1986). Table 1 contains a summary of the types of knowledge and some aspects of the learning mechanisms in these three models.

	CHREST+	ACT-R	Soar
Declarative Knowledge	Yes	Yes	No
Procedural Knowledge	No	Yes	Yes
When Does Learning Occur?	Continuous learning	New rules through new experience; tuning statistics continuously	Chunking of solution to an impasse
Form for Learnt Knowledge	A hierarchy of perceptual chunks	Production-rules grouped by problem space.	Production-rules grouped by problem space.
Retrieval of Knowledge	Serial match and serial use of chunks	Parallel match but serial fire of rules	Parallel match and parallel fire of rules
Interaction of Stored Knowledge	Seriality imposed by hierarchy	Statistics are used for conflict resolution between matching rules	Operators suggested by rules are 'voted for' by other rules
Generalisation	Presence of similar (or ancestor) chunks	Variabilisation and use of dependencies	Variabilisation only of relevant features

Table 1 : Summary of three cognitive architectures: CHREST+, ACT-R and Soar.

CHREST+ currently only uses declarative knowledge, although production-rule-type procedural knowledge may be included in the same framework (Gobet, 1996). Of greater interest is how the hierarchical arrangement of information in CHREST+ differs from the essentially unstructured set of production rules, within each problem space, used by ACT-R and Soar. This difference means that, instead of the parallel match of all rules required by ACT-R and Soar, CHREST+ can use serial procedures for knowledge retrieval and use, although several chunks can be considered simultaneously in STM. This emphasis on seriality in CHREST+ is derived from EPAM and its presence in Simon's ideas on cognition in general (e.g. Simon, 1981). Further, because CHREST+ only retrieves a single chunk, all the interactions between chunks are taken care of by the discrimination network. In comparison, ACT-R uses a set of statistics learnt for each rule to drive a conflict-resolution strategy, whereas Soar allows all rules to 'fire' and suggest possible operators, which are then 'voted for' by further rules. This simplicity allows CHREST+ to adopt a rapid and continuous process of learning. Due to the added cost of computing interactions between rules or operators in ACT-R and Soar, these models learn new rules in response to well specified situations during the problem solving process; CHREST+ learns in response to the direct input patterns with the structure of connections within its network determining its knowledge for problem solving. Finally, the generalisation of information learnt by CHREST+ is based on the quality of the perceptual decomposition; that in ACT-R and Soar relies additionally on the power of variabilisation and specific mechanisms for selecting the relevant conditions from which to create a production.

However, the most important distinction between CHREST+ and both ACT-R and Soar is not so much the lack of production-rules, but instead that training instances are not substantially processed; the only processing that occurs is a decomposition of each instance into a number of perceptual chunks, an approach similar to lazy learning (e.g. Aha, 1997). The value of this decomposition in rapid recall and the recognition of instances from a human's domain of expertise has been demonstrated in the non-problem-solving version of CHREST (Gobet & Simon, in press). Our extension of this mechanism to actual problem solving offers an explanation of how experts can rapidly solve problems from their domain of expertise, and also offers a solution to the knowledge proliferation problem which does not involve parallelism (Lane, Cheng & Gobet, 1999).

REFERENCES

Aha, D. W. (Ed.) (1997). AI Review Issue on Lazy Learning, 11:1-5.

Anderson, J. R., & Lebiere, C. (1998). The atomic components of thought (Lawrence Erlbaum).

Cheng, P. C-H. (1998). A framework for scientific reasoning with law encoding diagrams: Analysing protocols to assess its utility. In M. A. Gernsbacher & S. J. Derry (Eds.) Proceedings of the Twentieth Annual Conference of the Cognitive Science Society (pp. 232-235). Mahwah, NJ: Erlbaum.

Cheng, P. C-H. (1999a). Electrifying representations for learning: An evaluation of AVOW diagrams for electricity. (ESRC CREDIT Technical Report No. 62, University of Nottingham).

Cheng, P. C-H. (199b). Unlocking conceptual learning in mathematics and science with effective representational systems. Computers and Education, 33, 109-130.

Croker, S., Pine, J. M., & Gobet, F. (2000). Modelling optional infinitive phenomena: A computational account of tense optionality in children's speech. In Proceedings of the Third International Conference on Cognitive Modelling.

Feigenbaum, E. A., & Simon, H. A. (1984). EPAM-like models of recognition and learning. Cognitive Science, 8, 305-336.

Gobet, F. (1996). Discrimination nets, production systems and semantic networks: Elements of a unified framework. Proceedings of the Second International Conference of the Learning Sciences (pp. 398-403). Evanston, III: Northwestern University.

Gobet, F. (1999). Simulations of stagewise development with a symbolic architecture. In J. P. Dauwalder & W. Tschacher (Eds.), Dynamics, synergetics, and autonomous agents. Singapore: World Scientific.

Gobet, F. & Jansen, P. (1994). Towards a chess program based on a model of human memory. In H. J. van den Herik, I. S. Herschberg, & J. W. Uiterwijk (Eds.) Advances in computer chess 7. Maastricht: University of Limburg Press.

Gobet, F., & Pine, J. (1997). Modeling the acquisition of syntactic categories. Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society (pp. 265-270). Hillsdale, NJ: Erlbaum.

Gobet, F. & Simon, H. A. (in press). Five seconds or sixty? Presentation time in expert memory. Cognitive Science.

Jones, G., Gobet, F., & Pine, J. M. (2000). Learning novel sound patterns. Proceedings of the Third International Conference on Cognitive Modelling, Groningen.

Laird, J. E., Rosenbloom, P. S., and Newell, A., (1986). Chunking in SOAR: The anatomy of a general learning mechanism. Machine Learning, 1, 11-46.

Lane, P. C. R., Cheng, P. C-H., and Gobet, F. (1999). Learning perceptual schemas to avoid the utility problem. In M.Bramer, A.Macintosh and F.Coenen (Eds.) Research and Development in Intelligent Systems XVI: Proceedings of ES99, the Nineteenth SGES International Conference on Knowledge Based Systems and Applied Artificial Intelligence, Cambridge, UK, pp.72-82 (Springer).

Lane, P. C. R., Cheng, P. C-H., and Gobet, F. (submitted). Predicting perceptual chunks acquired during learning: Applying CHREST+ to problem solving with diagrams.

Larkin, J. H. & Simon, H. A. (1987). Why a diagram is (sometimes) worth ten thousand words. Cognitive Science, 11, 65-99.

Richman, H. B., Gobet, F., Staszewski, J. J., & Simon, H. A. (1996). Perceptual and memory processes in the acquisition of expert performance: The EPAM model. In K. A. Ericsson (Ed.) The road to excellence. Mahwah, NJ: Erlbaum.

Simon, H. A. (1981). The sciences of the artificial. Cambridge, MA: MIT Press.