A Distributed Framework for Semi-Automatically Developing Architectures of Brain and Mind

Fernand Gobet[1], Peter C. R. Lane[2]

[1]School of Social Sciences and Law, Brunel University, Uxbridge, Middlesex, UB8 3PH, UK

[2]School of Computer Science, University of Hertfordshire, College Lane, Hatfield, Hertfordshire, AL10 9AB, UK

Email address of corresponding author: fernand.gobet@brunel.ac.uk

Abstract. Developing comprehensive theories of low-level neuronal brain processes and high-level cognitive behaviours, as well as integrating them, is an ambitious challenge that requires new conceptual, computational, and empirical tools. Given the complexities of these theories, they will almost certainly be expressed as computational systems. Here, we propose to use recent developments in grid technology to develop a system of *evolutionary scientific discovery*, which will (a) enable empirical researchers to make their data widely available for use in developing and testing theories, and (b) enable theorists to semi-automatically develop computational theories. We illustrate these ideas with a case study taken from the domain of categorisation.

Background

A primary aim in science is to develop theories that summarise and unify a large body of empirical data. However, there is no overarching theory in psychology[1] (or even in sub-fields of psychology, such as study of memory, emotions, or perception), which, in the way quantum mechanics organises empirical data in chemistry, imposes order on the mass of data and makes it possible to derive quantitative predictions. To compound the difficulty, there currently exist about 1,500 journals devoted to scientific psychology, and new journals are created monthly. A substantial proportion of these journals publish mainly experimental or quasi-experimental results. How can scientists keep track of this exponentially increasing amount of information without succumbing to an information overload? While progress in database management of scientific results is notable, in part due to new advances in grid-supported databases, there remains the question of how this new information can foster scientific *understanding*, as opposed to simple *accumulation* of knowledge. As proposed by Newell (1990) and others, the best approach is to develop theories implemented as computer programs, which can summarise and account for empirical data.

Computational modelling has played an increasingly important role in psychology in recent years, with models spanning a wide range of complexity: from simple models, which account

[1] While we focus on high-level cognition, the ideas we develop also apply to the study of low-level brain processes, as well as to the study of how these levels, among others, interact.

for the results of a single experimental paradigm, to ‘unified theories of cognition’ such as ACT-R (Anderson & Lebière, 1998) and Soar (Newell, 1990), which aim to cover the whole gamut of cognitive phenomena. Computational modelling involves expressing psychological theories as computer programs, which specify in detail the processes carried out Models can be separated into two approaches: symbolic (where programs manipulate symbols) and connectionist (where programs roughly simulate the way neurones function in the brain). One important advantage of theories expressed as computer models, as opposed to informal theories, is that they make it possible to derive both *qualitative* and *quantitative* predictions, even for complex psychological phenomena. However, the fragmented nature of collaboration between cognitive scientists critically hinders the development of overarching theories in psychology; as Richman and Simon (1989) put it: ‘We need comprehensive theories to permit detailed modelling of phenomena and comparison over wide ranges of different phenomena.’ Developing such comprehensive theories requires global cooperation among empirical researchers, computational modellers and theorists: the goal of our system is to develop an infrastructure supporting such cooperation.

Current difficulties facing cognitive science

We highlight three difficulties in the current approach to cognitive science, which hinder the development of truly robust and comprehensive theories.

1. Empirical research and computational modelling are each demanding subjects. An expert in one area is not guaranteed to be an expert in the other. A typical case is that a researcher involved mainly in gathering empirical evidence will limit their modelling efforts to a subset of the full range of approaches on offer, an enforced limitation deleterious to theory development.

2. Related to point 1 is the lamentable lack of an agreed-upon set of techniques for directly comparing models and evaluating them against empirical data (Newell, 1990; Roberts & Pashler, 2000). In particular, few modellers routinely use formal, automated techniques to optimise the parameters and/or the structural components of their model (Ritter, Shadbolt, Elliman, Young, Gobet, & Baxter, 2003). Clearly, using a theory to explain an empirical phenomenon is inappropriate if the model implementing the theory has not been optimised (Ritter, 1991).

3. The development of comprehensive theories requires a pooling together of information from all empirical research and modelling efforts relevant to that theory. For instance, if researcher X develops a Soar[2] model of problem solving with blocks, and researcher Y develops a similar Soar model of problem solving with sticks, a theory of problem solving based on Soar should refer to both models. However, unless researchers X and Y are in the same institution, this reference will only be performed in a qualitative manner, based on published literature. Preferable is for the empirical evidence to be compared directly and quantitatively across the two models.

Evolutionary scientific discovery

The overall aim of our research is to address these three difficulties by separating out the roles of the empirical researcher, the computational modeller, and the theorist. We then support each role with a data format for storing relevant data in an on-line database. Search

[2] Soar is a production-rule model of human cognition, developed by Newell (1990).

methods and grid technology (for storing large amounts of data and for carrying out massively parallel computation) may then be implemented to (semi-)automatically develop comprehensive, computational models of psychological phenomena; it is this final part where evolutionary techniques become key to the process of scientific discovery. A key requirement is the presence of *templates* for representing the models and empirical data in a standardised format, with the format tailored to assist the use of evolutionary, optimisation techniques.

The long-term view of our research is that different empirical researchers will conduct experiments and record their results in a unified manner. These results are made accessible on-line. The computational modeller can then access the results *from all registered researchers* simultaneously, in order to develop optimal models covering a wide range of phenomena. Each model, with its matching or refuting evidence, is then placed in an on-line database. Finally, the cognitive theorist can inspect the database of models and empirical evidence, and so come to informed decisions about the explanatory power and comprehensiveness of all competing models.

We propose that automatic and semi-automatic techniques should be developed for supporting many of these operations; for example, evolutionary algorithms (Goldberg, 1989; Holland, 1992; Koza, 1994) can optimise the fit of a class of computational models to a spectrum of empirical data. Such automated approaches are often prohibitively expensive, and hence require the power of grid computing to be feasible. Finally, although we have separated out three roles, there is nothing in our methodology prohibiting a single researcher from adopting different roles at different times; all we require is that the distinct nature of each role be separated out and handled independently. For instance, the automated optimisation techniques covering a wide spectrum of computational models will prevent an individual researcher from being locked in, through technology or other reasons, to a restricted theoretical framework.

Development of explanations

Science progresses through the development of *explanations*. We are not interested only in the fact that one model provides a good fit to a given piece of empirical data, but also in *why* the chosen model is a better candidate than other models. This ‘why’ question is often ignored, although it is a key element in the development of broad theories of cognition. By automating the development of psychological theories, we aim to support the development of meaningful answers to why particular models are better than others, in particular by supporting direct comparison between many models through the use of grid-supported parallel computation. The result of this comparison will be a collection of optimised models, each model coupled with the empirical data which it explains (the framework for this was proposed in Lane and Gobet (2003)). A second supporting technique will be tools for probing the details of the optimisation process, querying which models have been rejected and why. This information provides the data required by a theorist developing explanations for which cognitive processes generate which empirical phenomena.

Optimising models with evolutionary computation

Ideally, one should be able to use a number of computational architectures to ensure the generalisability of the techniques used. A computational model typically has a number of ‘free parameters’ – values of parameters set in accordance with the current experimental findings. A key task for a modeller is to explore the space of these parameter settings until one is found that is suitable to the given data. Usually, this is done in a sub-optimal manner (Ritter, 1991).

Our intention behind formalising the specification of a broad class of computational models is to support the application of optimisation techniques for aiding the development of theories. We will use evolutionary approaches for this purpose, including genetic algorithms (Goldberg, 1989; Holland, 1992) and genetic programming (Koza, 1994). Inspired by natural selection, evolutionary computation enacts a search for solutions to the problem of survival. It evolves large populations of genotypes (possible solutions) with the constraint that the fittest (best) tend to survive and reproduce. Artificial genotypes encode sets of parameters with genetic algorithms, or entire programs with genetic programming. What constitutes a 'solution' is governed by a fitness function determined by the specific problem; hence, if the problem is to simulate an empirical dataset, the amount of variance accounted for may be used as the fitness measure. The mathematical foundation of evolutionary computation is well established (Holland, 1992).

Although computationally simple, these algorithms are robust and powerful, exploring huge search spaces efficiently and in parallel, even when the information is noisy and subject to uncertainty. These algorithms have been used extensively in science and engineering, for example in function optimisation, pattern recognition, functional genomics, diagnostic discovery, and the analysis of noisy data (Goldberg, 1989; Kanehisa, 2000; Koza, 1994).

As our models and domains are taken from the psychological literature, a number of results already exist for particular models in particular domains. Where possible, we will check on the validity (and value) of our approach by comparing the results obtained through optimisation with those proposed by other researchers (Ritter, 1991).

Case study: Developing a cognitive model

We can more carefully illustrate the potential and implications of our approach by discussing a case study in developing a model using evolutionary techniques. The intention is not to give a complete description of the study, which would require details of categorisation and modelling not pertinent here, but instead to illustrate the complex task confronting a social scientist who wishes to construct a computational model applicable to multiple data sets. The description emphasises the form of the problem, and its method of solution using evolutionary scientific discovery.

Categorisation

The problem of categorisation requires an experimental subject to assign items into preset categories, or even to create new categories. Categorisation has seen intense study by psychologists and computer scientists for several decades (for reviews, see Fisher, Pazzani, & Langley, 1991; Gobet, Richman, Staszewski, & Simon, 1997; Murphy, 2002). Empirical psychologists gather many kinds of data, including the expectation of assigning a given item to a category, and the time to make a response. The aim of the modeller is to produce some computational description which will generate data of a similar kind.

Categorization is a good choice for illustrating our approach, as there exists a large amount of empirical data; computational models have already been developed for accounting for some of the data; this domain is characterised by clean and straightforward experimental designs, which can be easily simulated on the computer; it is easy to measure the (simulated) subject performance (typically, percentage correct or time to reach a preset criterion); and, finally, categorization is representative of the kind of experiments typically carried out in psychological research. Figure 1 illustrates the kind of problem confronting the experimental subject.

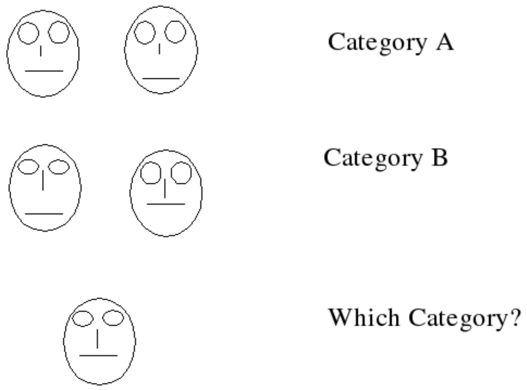

Figure 1. An experimental subject will typically be confronted with a task such as that illustrated, with some examples of two categories, and then a set of new, transfer items, to which the subject will have to provide a category.

We use data from an experiment called the 5-4 structure (Table I summarises the typical form of this experimental task), which was introduced by Medin and Smith (1981). Different forms of experimental data can be created by varying the interpretation of the four attributes. For example, by making A0 eye height, A1 eye separation, A2 nose length, and A3 mouth height, we obtain the face experiment performed by Medin and Smith (1981) and Gobet et al. (1997). The 5-4 structure spawned a myriad of follow-up studies, and thirty of these have been summarised by Smith and Minda (2000). The studies provide empirical data for the proportion of correct responses and the time to make them, achieved in different settings.

We use four classes of computational model, each of which is capable of performing the categorisation experiment. Two classes are mathematical models (prototype models and exemplar models, respectively), as discussed by Smith and Minda (2000), and use specific formulae to produce a response. The third class is a complex process model of human learning, known as CHREST (Gobet et al., 2001). The fourth is a form of connectionist network (McLeod, Plunkett, & Rolls, 1998). In a nutshell, prototype models propose that each category is represented by a best example, exemplar models that categories consist of memories of all stimuli previously perceived, CHREST that categories incrementally grow as chunks of information, and connectionist networks that categories result from the tuning of weights between nodes.

Evolving a model

We have glossed over the description of the models; however, one aspect of the models must be present. This is the *parameterisation* of the model, and its application to multiple tasks. Each class of model has its own set of parameters, and varying the values for the parameters allows every instance of the class to be obtained. Each specific instantiation of the parameter values is called an individual *model,* and each model must be capable of being applied to any of the tasks.

Training										Transfer				
	Attribute (A)					Attribute (A)					Attribute (A)			
	A0	A1	A2	A3		A0	A1	A2	A3		A0	A1	A2	A3
A Examples					B Examples					Transfer Items				
E1	1	1	1	0	E6	1	1	0	0	E10	1	0	0	1
E2	1	0	1	0	E7	0	1	1	0	E11	1	0	0	0
E3	1	0	1	1	E8	0	0	0	1	E12	1	1	1	1
E4	1	1	0	1	E9	0	0	0	0	E13	0	0	1	0
E5	0	1	1	1						E14	0	1	0	1
										E15	0	0	1	1
										E16	0	1	0	0

Table I: The 5-4 structure used in categorisation experiments (after Medin & Smith, 1981).

In summary, our approach has the following structure:

1. From a class of model, return specialised models by selecting parameters
2. For each specialised model and each experiment in the set of data to simulate
    a. apply the model in the experiment
    b. return a behaviour
3. For each model, compute a fitness function combining measures of behaviour in each experiment

The empirical data, which each model must match, provides a set of constraints. For example, one constraint, ‘SSE Avg’, is the sum-squared error of the model’s predicted performance against the average of previous experimental results. A second constraint, ‘AAD Time’, measures the average absolute difference between the model’s predicted time to make a response, and the recorded time of response in one experiment. Taken individually, it is possible to find a set of parameter values so that one instance of each class of model does well on each of these constraints. Two questions then present themselves: is there a single model instance in each of the classes which does ‘well enough’ on all of the constraints? And, how do the best instances of each class fare in comparison with those from other classes?

We use a genetic algorithm (Goldberg, 1989) to optimise the fit of models on the constraints. A complication arising because of the multiple constraints is that it can be hard to say which of two models is ‘better’ when one is better on one constraint, but worse on a second. Hence, we instead look for those models which are not *dominated* by any other. One model dominates a second if it is at least as good as the second in all constraints, but better in at least one constraint. The genetic algorithm is adapted to seek out the collection of non-dominated models.

The genetic algorithm works in the following manner. First, it creates a population of models, using *random* assignments of values to each of the models’ parameters. Second, the optimal (non-dominated) models in the current population are extracted. Third, new models are created based on the optimal models in the current population. New models are created using analogues of the evolutionary processes of cross-over (mating) and mutation. Finally, the system repeats from the second step, repeating typically for a few thousands of cycles.

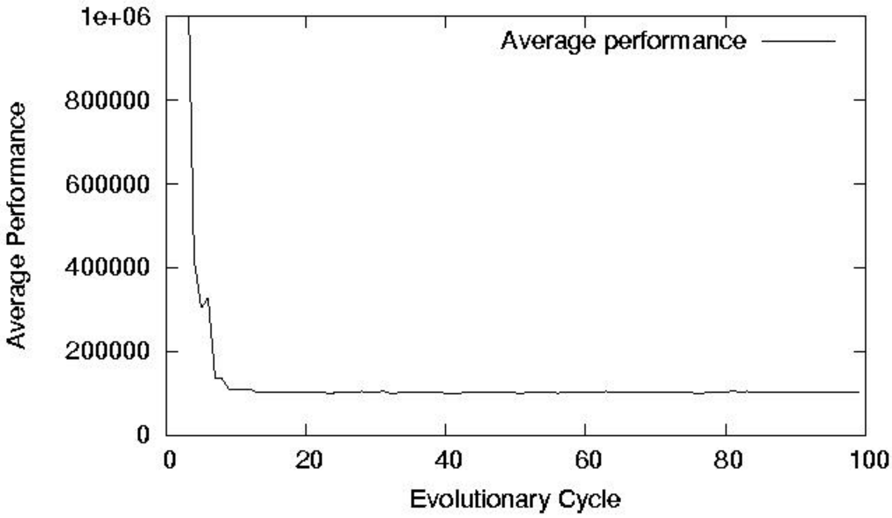

Figure 2. Average total performance of an evolving population of models.

Initially, the population of models is not particularly successful, but will, overtime, locate models which perform well on the task. Figure 2 illustrates how the *total* performance of the system on multiple constraints settles down to a figure over the first 100 cycles of a typical run. Once the evolutionary process has completed, we can analyse the performance of the evolved models on selected criteria. Based on the optimal set of models, we can determine ranges of admissable parameter values. Also, we can find the optimal model for specific criteria. For example, we find that for one particular measure, SSE of the average performance, a context model produced a fit of 0.03; this is an improvement on a published fit of 0.065 by Smith and Minda (2000).

Discussion of case study

The most important conclusion from the preceding study is that optimisation techniques, such as genetic algorithms, can generate models as good as, if not better, than those produced ‘by hand’. For example, on the sum-squared error constraint, we obtained superior levels of fit with the mathematical model to those presented by Smith and Minda (2000). Similarly, Ritter (1991) and Tor and Ritter (2004) report improvements on hand-optimised values for developing different kinds of cognitive models. This point should not be taken as a criticism of the original work, but rather a recognition that optimising the fit of a model is in itself a complex task, which must be considered as carefully as data collection or theory creation.

A second consequence of our approach is its implicit enforcement of *standardisation*. The testing of models drawn from multiple theories, and the use of differing criteria for evaluation, requires the implementor to put their data or model into a generic format. For example, the raw data obtained from each categorisation experiment must be separated into the response probabilities and the timing figures. Each experiment’s data must be represented in the same form, as a list of numbers, and the order of these numbers must match a standard order of the experimental stimuli. Similarly, each model which the genetic algorithm creates must be applicable to all of the experimental conditions. This requires the implementor of the model to conform to a standard *interface*, so that every model will be applied correctly to the given experimental constraint, and report results of a similar nature.

Standardising models and datasets

Constructing a distributed framework for developing architectures of brain and mind provides the considerable advantage of allowing different specialists to work at different parts of the problem. Empirical scientists can collect data, which are then placed in a repository for access by those developing concrete models. Modellers can provide implementations, which may then be run against multiple datasets. Those interested in optimisation may develop efficient optimisation algorithms, which apply the models to the data. A specialist in one of these three areas can contribute effectively without knowing the details of the other two. As highlighted above, current practise in modelling means that a single scientist must perform all three tasks, and the results are frequently suboptimal.

However, the three areas, although distributed, must interact in some manner. The interaction implicitly requires contributors to work towards standards, if their contributions are to be useful within the total framework. We can consider the requirements for standardisation separately for the models and for the data.

Models

There are two sets of constraints for the models: applicability to multiple experiments, and uniformity for treatment in optimisation. Being applicable to multiple experiments requires the implementor of the model to ensure that the model carries out each of the standard operations. For instance, in the case study, the implementor must ensure the model both provides a probability of responding with a given category, and also provides timing data as to how long the response takes. In extending the case study, we may find the implementor is required to ensure that the model carries out certain problem-solving tasks in a specified manner. The importance of this conformity should not be underestimated. One factor leading to fragmentation within the modelling community is the lack of opportunity for quantitative comparisons between models. By ensuring that models are applicable to at least a standard set of empirical settings, we can increase the opportunities for such comparisons.

The second constraint is that the model be suitable for treatment in optimisation. Within the context of our evolutionary approach, all this requires of the model is that its free parameters be ‘exposed’ to the optimisation algorithm, so that the range of possible values may be explored automatically.

Datasets

Empirical data must be provided to the system in a format so that a model can be applied to it. For example, if we consider the categorisation experiments, the experimenter will construct a set of stimuli, train the experimental subjects on part of the data, and then test the subjects on some new, transfer data. Each subject’s response will be recorded, as will the time to make that response. Currently, we design the datasets in the form of *standard templates*. The dataset first specifies the kind of experimental setting, categorisation, and then provides the training stimuli. Next, the dataset contains instructions for training and testing the model, providing a formalised description of how the experiment is conducted. This description is then used when obtaining quantitative results from the computational models, and ensures that the model data is comparable with the subject data.

General discussion

Resources

One of the problems with finding a cognitive model is the vast number of potential models which may be defined. Each one of the potential models must be evaluated against each of the constraints. The small problem considered above, the categorisation problem, explores around 50,000 models, but each of the constraints is reasonably quick to run. Hence, the whole system runs in about an hour, on a single computer. However, a more complex task applied to a large pool of experimental data can require 24 hours or longer to run. These run-times make the problem resource hungry. Three possible solutions present themselves.

First, by analysing the behaviour of the models, we can fix the value of many parameters within the model. For example, the time for certain processes to run within CHREST can be determined in one set of experiments, and then this value used in all future experiments. These additional constraints considerably reduce the number of models which must be searched, decreasing the number of cycles required in the genetic algorithm. Second, because the system is running many times, employing the same algorithm and the same constraints, it is frequently the case that the same computation will be performed multiple times. Low-level programming techniques, such as memoization, can be employed to obviate the need to recompute complex results many times. Third, by taking advantage of grid technology, we can run the application in parallel on faster processors, and considerably improve the time required to process a single task. Even with the previous two solutions included, we still anticipate the need for grid technology to make the search for complex cognitive models tractable.

Generality for the social sciences

In this paper, we have focused attention on *quantitative data* provided by *controlled experiments* in *psychology*. It is useful to consider whether our approach can be generalised along the three axes implied by these terms. Specifically, can it be used with *qualitative* data, *observational* data, and *other fields* in social sciences?

Qualitative data

Qualitative data are common in social sciences. Typical data-collection methods include direct observation (e.g., through videotaping), interviews, and use of various types of texts (e.g., books, newspapers, websites). In psychology, the recording of think-aloud verbal protocols is common, in particular in the study of problem solving (Newell & Simon, 1972). Several of these data can be quantified, for example by computing the frequency of contents words or the relations between these words. Then, standard statistical and mathematical techniques can be used to analyse categorical data (frequency of contents words) and graphs (relations between contents words). Thus, our approach should be able to handle qualitative data, although with varying levels of difficulty that will depend on the current state of formal theorizing. For example, while modelling think-aloud protocols in chess problem solving is within reach of current technology, modelling somebody carrying out an informal conversation is certainly many decades away.

Observational data

While experimental data are not rare in the social sciences, for example in education, psychology, and the relatively new field of experimental economics, it is probably the case that most of the data collected—certainly those accessible in databases over the internet—are of an observational nature. Typical examples include data on the national and international economy, socio-economic data from national censuses, and data on world records in sports. A number of techniques have been developed in statistics (e.g., multiple regression/correlation analysis; multivariate analysis) and in data mining (e.g., CART, or decision trees) to deal with this class of data. Some of these techniques have in-built optimisation devices (e.g., minimizing the sum of squared errors in multiple regression analysis). A limit of these techniques is that, while excellent at picking up patterns in these datasets, they provide limited scope for generating general theories. In principle, the techniques we have described in this paper could be used with observational data as well, in particular if they provide information about time and thus capture dynamical aspects of the variables under scrutiny.

Social sciences beyond psychology

As should be apparent from our discussion, our approach should be of interest to social sciences in general, both for those using experimentation and those relying more on observational data. However, an inescapable requirement is that theories within these fields are expressed formally, or at least with sufficient detail and clarity. Without this, there is obviously no way that they could be translated into computer programs—a condition of our method. Incidentally, our approach highlights one of the disadvantages of informal theories: they cannot be optimised or otherwise manipulated formally, which deprive them of most of the benefits we have highlighted in this article. For an elaboration of this question, see Gobet (2000a,b), as well as Gobet and Waters (2003).

Further work

The alert reader would have noticed strong links between our approach and research topics in the philosophy of science. Indeed, one of the motivations behind our work was to develop techniques that enable meaningful comparison between different theories making predictions on the same tasks. This constitutes one of the thorniest difficulties in theory development. While the methods we have described were based primarily on measures of goodness of fit with empirical data, we also plan to incorporate information about the number of degrees of freedom and measures of parsimony of the theories. Expanding the current set of templates into more complex *specification languages* will make it easier to measure this information, which has been notoriously difficult to quantify (Pitt, Myung, & Zhang, 2002; Simon, 1977).

This research constitutes the first attempt to use large-scale optimisation techniques to tackle the problem of optimising computational models within cognitive science, and, more generally, in psychology. To the authors' knowledge, this is also true in the wider social sciences. The increasing importance of quantitatively verified, predictive theories suggests that optimising models across multiple datasets will assume a greater importance in future, and so generate a wider application for evolutionary approaches to scientific discovery.

In addition, we intend to take advantage of grid technology in managing distributed databases to develop an infrastructure, supporting the roles of the empirical researcher and theoretician in providing data to and summarising the results from the optimisation techniques. Grid technology is critical to the success of this project as, apart from the need to handle large amounts of data, the key computational processes of optimisation require large levels of computing power.

Although only a pilot, this project has made good progress in setting up an infrastructure for distributed collaboration and management of a complex scientific endeavour: developing theories of human cognition. Achieving this goal required a close collaboration between the authors—a social and a computer scientist. Complementary expertise enabled us to develop powerful computational tools supporting the development of meaningful theories in the social sciences. The significance of this project outside of psychology is also high as, with suitable modification of the data structures, the infrastructure we develop can be easily generalised to other scientific fields.

This paper has deliberately limited itself in presenting a standard template and an optimisation method for optimising extant theories. This limitation has enabled us to illustrate our evolutionary approach in a concrete case study, which already produces surprising results. Can we go one step further and propose methods enabling us to semi-automatically, or even automatically, develop theories from scratch? In a related project (Gobet & Parker, in press), we have proposed means to automatically 'evolve' theories, in that case theories mapping brain structures to cognitive functions. Although there are differences with the methods proposed here, the need for encouraging researchers to build up databases of results and the use of evolutionary techniques is apparent in that project as well.

Conclusion

Modern science is confronted with an impressive and overwhelming amount of empirical data. An important technique for attempting to understand these data is to develop simulation models which cover as many of the datasets as possible. In this paper, we have proposed a distributed framework to which empirical researchers and modellers can contribute information from their specialism. Using a case study, we have demonstrated the value of evolutionary scientific discovery in developing optimal models. Through the use of grid technology, we suggest that our framework provides a robust way of developing theories, both within cognitive science, and the social sciences more generally.

References

Anderson, J. R., & Lebière, C. (Eds.). (1998). *The atomic components of thought*. Mahwah, NJ: Erlbaum.

Fisher, D. H., Pazzani, M. J., & Langley, P. (1991). *Concept formation: Knowledge and experience in unsupervised learning*. San Mateo, CA: Morgan Kaufmann Publishers.

Gobet, F. (2000a). Long-term working memory: A computational implementation for chess expertise. In N. Taatgen & J. Aasman (Eds.), *Proceedings of the Third International Conference on Cognitive Modelling* (pp. 150-157). Veenendaal, The Netherlands: Universal Press.

Gobet, F. (2000b). Some shortcomings of long-term working memory. *British Journal of Psychology*, *91*, 551-570.

Gobet, F., Lane, P. C. R., Croker, S., Cheng, P. C. H., Jones, G., Oliver, I., & Pine, J. M. (2001). Chunking mechanisms in human learning. *Trends in Cognitive Sciences*, *5*, 236-243.

Gobet, F., & Parker, A. (in press). Evolving structure-function mappings in cognitive neuroscience using genetic programming. *Swiss Journal of Psychology*.

Gobet, F., Richman, H. B., Staszewski, J. J., & Simon, H. A. (1997). Goals, representations, and strategies in a concept attainment task: The EPAM model. *The Psychology of Learning and Motivation, 37*, 265-290.

Gobet, F., & Waters, A. J. (2003). The role of constraints in expert memory. *Journal of Experimental Psychology: Learning, Memory & Cognition*, *29*, 1082-1094.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison-Wesley.

Holland, J. H. (1992). *Adaptation in natural and artificial systems*. Cambridge, MA: MIT Press.

Kanehisa, M. (2000). *Post-genome informatics*. Oxford: Oxford University Press.

Koza, J. R. (1994). *Genetic programming II. Automatic discovery of reusable programs*. Cambridge, MA: MIT Press.

Lane, P. C. R., & Gobet, F. (2003). Developing reproducible and comprehensible computational models. *Artificial Intelligence, 144*, 251-263.

McLeod, P., Plunkett, K., & Rolls, E. T. (1998). *Introduction to connectionist modeling of cognitive processes*. Oxford: Oxford University Press.

Medin, D. L., & Smith, E. E. (1981). Strategies and classification learning. *Journal of Experimental Psychology: Human Learning and Memory*, *7*, 241-253.

Murphy, G. L. (2002). *The big book of concepts*. Cambridge, MA: The MIT Press.

Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.

Newell, A., & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice-Hall.

Pitt, M. A., Myung, I. J., & Zhang, S. (2002). Toward a method of selecting among computational models of cognition. *Psychological Review, 109*, 472-491.

Richman, H. B., & Simon, H. A. (1989). Context effects in letter perception: Comparison of two theories. *Psychological Review, 3*, 417-432.

Ritter, F. E. (1991). Towards fair comparisons of connectionist algorithms through automatically optimized parameter sets. In *Proceedings of the Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Erlbaum.

Ritter, F. E., Shadbolt, N. R., Elliman, D., Young, R. M., Gobet, F., & Baxter, G. D. (2003). *Techniques for modelling human performance in synthetic environments: A supplementary review*. Wright-Patterson Air Force Base, OH: Human Systems Information Analysis Center.

Roberts, S., & Pashler, H. (2000). How persuasive is a good fit? A comment on theory testing. *Psychological Review, 107*, 358-367.

Simon, H. A. (1977). *Models of discovery and other topics in the methods of science*. Dordrecht: Reidel.

Smith, J. D., & Minda, J. P. (2000). Thirty categorization results in search of a model. *Journal of Experimental Psychology: Learning Memory and Cognition, 26*, 3-27.

Tor, K. and Ritter, F. E. (2004). Using a genetic algorithm to optimize the fit of cognitive models. In *Proceedings of the Sixth International Conference on Cognitive Modeling*. Mahway, NJ: Erlbaum. (pp. 308-313).