MADLab: Masking and Multiple Bug Diagnosis

Allan Scott

c/o Gordon Ruckdeschel
375 Harvard St, Apt #1
Cambridge, MA 02138
(617) 492 5284
email: allans@aisb.ed.ac.uk
jas@cee.hw.ac.uk

Abstract

Multiple bug modelling involves the construction of procedural student models in which there are more than one bug. Multiple bug modelling is interesting because it poses difficulties for hand diagnosis and may provide a means of explaining the bug instability found in empirical data. Previous work on multiple bug modelling has identified multiple bug interactions as a challenge for diagnostic algorithms (Burton, 1982). The term bug masking is used here to define the situation where there is no simple relationship between the presence of an atomic bug in a multiple bug hypothesis and the performance of the hypothesis. The Masking and Diagnosis Lab (MADLab) is a domain independent toolkit designed to investigate the role of bug masking in multiple bug diagnosis. Experiments with the MADLab system in the domains of Prolog unification and Prolog control have provided evidence on the causes of bug masking and the effect of bug masking on the difficulty of multiple bug diagnosis.

1. Introduction

Multiple bug modelling involves the construction of procedural student models in which there are more than one bug. The complex interaction of bugs in multiple bug diagnosis make multiple bug modelling difficult to undertake by hand. For this reason most empirical work in the buggy modelling paradigm has focused on atomic bug modelling. Recent research has indicated that atomic bug modelling has failed to explain the instability of atomic bugs found in student behaviour (Payne et. al., 1992). Payne and Squibb found that students who displayed evidence of atomic bugs did not do so consistently. Multiple bug interaction has been put forward as a possible explanation for bug instability (Tokuda et. al, 1993). The fact that multiple bug modelling is difficult to undertake by hand and the possibility that multiple bug modelling will provide a better explanation for student behaviour provides the motivation for continuing research into multiple bug modelling.

Work on the DeBuggy system (Burton, 1982), however, identified bug interactions as a challenge for multiple bug diagnosis. Burton introduced the concept of `hiding' to cover the case where there is no evidence of an atomic bug which appears in the best compound. Analysis of examples of multiple bug diagnosis has lead to the more general concept of bug masking:

"Bug masking occurs where there is no simple relationship between the presence of an atomic bug in a multiple bug hypothesis and the performance of the multiple bug hypothesis" (Scott, 1994, p18)

The concept of bug masking is more general than the concept of hiding because it takes into account negative evidence for a hypothesis and different ways of evaluating the performance of an hypothesis.

2. Masking and Diagnosis Lab

The Masking and Diagnosis Lab (MADLab) is a domain independent toolkit designed to address three questions about the role of bug masking in multiple bug diagnosis:

2.1 How can bug masking be measured?

A measure of bug masking is important because it allows the designer of a diagnostic system to have access to information about an important property of the diagnostic problem and allows the development of experiments concerning bug masking in multiple bug diagnosis (see section 4).

2.2 What causes bug masking?

A large number of components go into the definition of a diagnostic problem: the definition of atomic bugs, the choice of questions given to students, student answers to questions, and the criteria on which an hypothesis is evaluated. Most of these are under the control of the designer of the diagnostic system. It is useful for the designer to know which factors matter, since design decisions can affect the amount of bug masking in a diagnostic problem.

2.3 Does bug masking affect the difficulty of a multiple bug diagnosis?

Burton's work suggests that bug masking leads to more difficult diagnostic problems. How can the difficulty of a diagnostic problem be assessed? If a clear relationship between bug masking and the difficulty of a multiple bug diagnosis can be found this would be useful in the choice of diagnostic algorithms.

2.4 Design Decisions

The key design decisions made in the development of the MADLab system were as follows:

Representation of diagnostic hypotheses as binary strings. In order to do this the effect of changing order in a multiple bug diagnosis was side-stepped.

Choice of a production system representation for procedural knowledge.

Providing tools for the construction of cardinal evaluation functions for diagnostic hypotheses. These evaluation functions are constructed with the modelling objectives of accuracy and parsimony in mind.

Application of the epistasis variance measure of the interaction of alleles in a binary string representation (Davidor, 1990) as a measure of bug masking. The epistasis variance measure is essentially the error variance of a simple linear regression: the more alleles interact in complex ways the higher the epistasis variance measure. The choice of epistasis variance addresses the question of how bug masking can be measured.

Application of standard optimisation algorithms for binary strings to multiple bug diagnosis (steepest ascent hillclimber, canonical genetic algorithm and enumeration). The difficulty of a multiple bug diagnosis can be assessed in a crude way by the comparison of a steepest ascent hillclimber with the canonical genetic algorithm (or enumeration where possible). The canonical genetic algorithm (or enumeration) is expected to outperform the steepest ascent hillclimber on more difficult optimisation problems.

3. Experiments

Experiments with the MADLab system were conducted in two domains: student errors with Prolog unification, and student errors with Prolog control. In both domains a production system model of the correct skill was defined and atomic bugs were defined by the mechanism of dropping and adding buggy productions to the correct skill. Atomic bugs were defined to be consistent with previous empirical work.

Data for the Prolog unification domain came from two undergraduate Prolog tests at the University of Edinburgh, whilst data for the Prolog control domain came from a survey conducted by David Duncan (Duncan et. al., 1994).

Experiments were designed to investigate the causes of bug masking and the effect of bug masking on the difficulty of multiple bug diagnosis. The first involved the application of the epistasis variance measure of bug masking and a comparison of results between different students and different data sets. The second experiment involved comparing the relative performance of the hillclimber, and the canonical genetic algorithm (in the Prolog unification domain) or enumeration (in the Prolog control domain), with results of the epistasis variance measure for each diagnostic problem.

4. Results

The results of these experiments were as follows:

4.1 Causes of bug masking?

Results indicate that the choice of questions and student answers affect the amount of bug masking measured in a diagnostic problem. Since the epistasis variance measure is not normalised it is difficult to tell how much these factors matter or their relative importance in determining bug masking.

4.2 Does bug masking affect the difficulty of multiple bug diagnosis?

A steepest ascent hillclimber was adequate for the vast majority of diagnostic problems. Without evidence from more difficult diagnostic problems it is difficult to find evidence for or against the view that bug masking affects the difficulty of multiple bug diagnosis.

5. Conclusion

The development of the MADLab system has provided partial answers concerning the role of bug masking in multiple bug diagnosis. In order to achieve the experimental goals of the MADLab system, an application of the MADLab system to more difficult diagnostic problems is essential. This work could be usefully undertaken in a domain where people find multiple bug diagnosis to be difficult. An example of such a domain is spelling where student misconceptions in a number of different types of knowledge contribute to observed student errors. Future work could also be usefully directed towards extending the MADLab approach to deal with multiple bug diagnosis where the order of bugs in an hypothesis matters, and to the development of improved measures of bug masking.

References

Burton, R. B. (1982) Debuggy: diagnosis of errors in basic mathematical skills. In Sleeman, D. H. and Brown J. S., (eds). Intelligent Tutoring Systems, pages 157-183, Academic Press, London

Davidor, Y. (1991). Epistasis Variance: a viewpoint on GA hardness. In Rawlins J. A., (ed), Foundations of Genetic Algorithms, Kaufman, San Mateo, Calif.

Duncan, D. and Brna, P. and Morss, L. (1994), A Bayesian Approach to Diagnosing Problems with Prolog Control Flow. In Proceedings of 4th Intl. Conf. User Modelling.

Payne, S. J. and Squibb, H. R.(1992). Algebra Mal-Rules and Cognitive Accounts of Error. Cognitive Science, 14:445-481.

Scott J. A. (1994). MADLab: Masking and Multiple Bug Diagnosis. Unpublished Thesis, Dept. of Artificial Intelligence, University of Edinburgh.

Tokuda, N. and Fokuda, A. (1993). A probabilistic inference scheme for hierarchical buggy models. Int. J. Man-Machine Studies, 38:857-872.