Using GDE in Educational Systems

Kees de Koning and **Bert Bredeweg**
Department of SWI, University of Amsterdam
Roetersstraat 15, 1018 WB Amsterdam, the Netherlands
{kees,bert}@swi.psy.uva.nl

Abstract

In intelligent educational systems, assessment of what the learner is doing is a prerequisite for proper, knowledgeable guidance of the educational process. We propose to use existing techniques from the field of model-based reasoning for this purpose. This paper describes how a modified version of GDE can be exploited in diagnosing a learner's problem solving behaviour. The problem solving task for the learner is structured prediction of behaviour. We present models of this problem solving knowledge that adhere to the representational requirements of model-based reasoning, and show how GDE-like diagnostic techniques can be employed to determine those reasoning steps that the learner cannot have applied correctly given the observations. Our approach of diagnosing the learner's problem solving *behaviour*, rather than his or her *misconceptions*, induces an educational strategy that focusses on learning from errors and stimulates the learner's 'self-repair' capabilities.

Introduction

One of the main bottlenecks in individualising education is the assessment and interpretation of the learner's problem solving behaviour, often referred to as *cognitive diagnosis*. As observed by Self, theories on model-based diagnosis aim at providing general frameworks for diagnosis, and thus "if cognitive diagnosis is indeed a type of diagnosis ..., it should be covered by these frameworks" (Self 1992). In this paper, this claim is investigated by reusing existing ideas and techniques in the context educational systems. Based on an explicit model of the subject matter, we apply the GDE paradigm (de Kleer & Williams 1987) to assess the learner's problem solving behaviour.

The problem solving task that the learner has to acquire is qualitative prediction of behaviour. Qualitative reasoning has long been recognised as an important aspect of human reasoning, and a preferable way of inducing understanding of the underlying principles in physics education (Chi, Feltovich, & Glaser 1981; Larkin *et al.* 1980; Elio & Sharf 1990). An additional advantage of the domain of qualitative reasoning is that simulators exist that can perform behaviour prediction on the basis of a description of some system (*e.g.*, QPE (Forbus 1990), GARP (Bredeweg 1992)). These simulators can be used to automate the process of creating diagnostic models.

This paper presents the STAR[1] framework, with a focus on its diagnostic component. Two key issues are addressed with respect to the application of model-based diagnosis in an educational context. Firstly, in the next section we exactly define the diagnostic problem that we want to solve: diagnoses are defined in terms of *reasoning steps* that cannot have been applied correctly by the learner given the observations. In accordance with this definition, we specify the models of the learner's reasoning behaviour that are needed to facilitate this diagnostic process, and the mapping of these models onto the component-connection paradigm used for "device models". We discuss how the right grain size of the reasoning steps is determined, and how the different knowledge types can be distinguished in the diagnostic models. The models are generated from the output of a qualitative simulator, and hierarchical structure is added automatically. Secondly, the diagnostic techniques need adaptation to work in the educational context. Because of the nature of the diagnostic task, a different probe selection algorithm is required. An example of the working of the diagnostic engine is provided in a prototype system called STAR^{light}.

One advantage of our approach lies in the fact that it puts educational diagnosis on a solid basis, and that it provides a generic approach for diagnosis of problem solving behaviour. More importantly, the STAR framework advocates a specific teaching strategy by focussing on the errors in the *behaviour* of the learner rather than on *misconceptions* in the learner's knowledge. We discuss the merits of this focus in detail.

A "Device Model" for Qualitative Prediction Of Behaviour

In a typical domain such as electronics, consistency-based diagnosis can be characterised as follows: given a model of a device in terms of components and connections between these components, plus a set of observa-

[1] System for Teaching About Reasoning.

tions, find those minimal sets of components that cannot behave according to their specified behaviour given the observations. In the context of teaching problem solving, we restate this characterisation as:

> Given a model of the problem solving task in terms of individual reasoning steps and data connections between these reasoning steps, plus a set of observations about the learner's problem solving behaviour, find those minimal sets of reasoning steps that cannot have been applied correctly by the learner given the observations.

Important to note is that this definition deviates from the commonly accepted definition of *cognitive diagnosis*, being "the process of inferring a person's cognitive state from his or her performance" (Ohlsson 1986): we do not try to determine the 'internal' cognitive state of the learner, but instead only diagnose his or her 'external' reasoning behaviour. As a result, the diagnosis consists of reasoning steps that cannot have been performed correctly, *i.e. bugs* in the learner's reasoning process, rather than *misconceptions* in the learner's knowledge. In other words, diagnoses are defined at the behavioural level, and not at the conceptual level (*cf.* (Dillenbourg & Self 1992)).

The Mapping

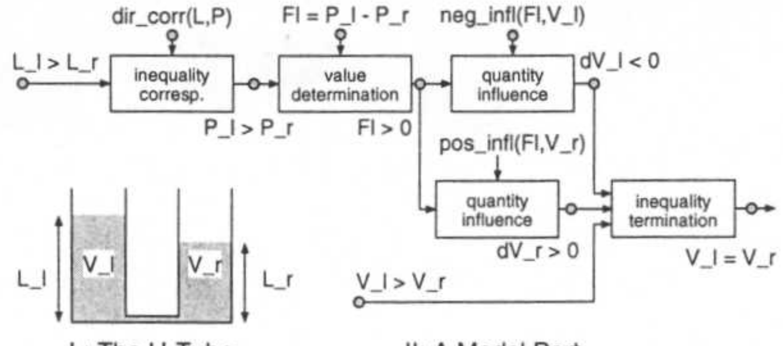
The first issue to be addressed is how to define a model of the problem solving task. This model should adhere to the representational requirements of model-based reasoning, that is, it should consist of context-independent components and connections between these. Because the execution of a problem solving task can be seen as performing a set of inference operations (*i.e.*, reasoning steps) on a data set, we model each reasoning step as a component. This means that in the model for a particular prediction, each *application* of an inference is represented as a component. As an example of such a model, consider the model fragment in Figure 1. For the U-Tube system in Figure 1-I, this model represents the derivation of the change in the inequality between the volumes: because the level is higher at the left, the pressure is higher as well, and therefore a flow exists from left to right, which means that the left volume is decreasing and the right one is increasing. As a result the levels will become equal. In the model, this reasoning trace is represented by five different components, representing four inference types. As an example, consider the leftmost component of type inequality correspondence that is used to derive that the pressure is higher at the left because the level is higher there as well. Technically speaking, the inference component has two inputs, namely the inequality between the levels $L_l > L_r$ and the directed correspondence that exists between the level and the pressure *dir_corr(L,P)*.

Grain Size of the Models In total, 16 different inference types are defined for the task of qualitative prediction of behaviour (de Koning 1997). This set is based on experimental research on how student and (human) teachers communicate about prediction problems. From protocols taken of a student and teacher discussing the behaviour of a physical device, we extracted the terminology and the individual reasoning steps that are made. The different types of reasoning steps encountered in the protocols formed the basis for the set of 16 components (de Koning & Bredeweg 1995). The experimental base of the inference types ensures that the reasoning steps are at the appropriate grain size of type to support an educational communication. For instance, the inequality correspondence component does not model a simple one-step inference from a qualitative reasoning point of view. However, the experimental research showed that this level of inference is considered primitive by both learners and teachers, and hence we do not want to model it in more detail.

Retrieval Components The model introduced so far represents the reasoning steps that make up the problem solving task. The 'contents' of these reasoning steps, *i.e.* the support knowledge that is used to make the inference, is represented as an additional input to the reasoning component. For instance, the inequality correspondence component has two inputs: the inequality $L_l > L_r$ *and* the relation *dir_corr(L,P)*. However, these inputs are of different nature: the first one is a given, something that the learner is supposed to see or read on the screen, and hence it can be assumed to be known by the learner. The second input *dir_corr(L,P)* embodies knowledge that the learner may not (yet) master. From a diagnostic point of view, this difference is important: in model-based diagnosis, all inputs are assumed to be correct, and a diagnosis can only be in terms of deficient components. Hence, no diagnosis can be found that expresses the fact that the learner does not know the relation between level and pressure, although it may express that the learner does not know how to *apply* the relation in this situation. We therefore introduce an additional component type called *retrieval.* Retrieval components have one input and one output, and the output is equal to the input if the component is functioning correctly. A "faulty" retrieval component hence represents the situation in which the learner does not know (cannot reproduce or "retrieve") an expression like *dir_corr(L,P)*.

Base Model Generation

A "device model" as introduced above, in the following referred to as the *base model*, is specific for one prediction of behaviour: although the component types are generic for the task, each prediction for a specific system such as the U-tube requires a new model to be generated. This is done on the basis of the output of a qualitative simulator called GARP (Bredeweg 1992): the simulator generates a prediction of behaviour, and a post processor transforms this output into a model representing all individual reasoning steps that a learner should master in order to solve the prediction problem.



I : The U-Tube

II: A Model Part

Figure 1: An Example Base Model Part

Adding Hierarchies

The base model contains all reasoning steps that are necessary for a correct prediction of behaviour, and hence such models tend to be rather large: a complete model for the behaviour of a system such as the balance depicted in Figure 2 consists of 665 components and 612 points. Furthermore, the nature of qualitative reasoning results in an incomplete set of behaviour rules, and hence an incomplete prediction engine, which reduces the efficiency of algorithms such as GDE (de Kleer & Williams 1987). As a result, applying GDE directly on the base model is not feasible in a run-time educational environment. In the case of electronics, such problems are usually addressed by focussing techniques such as hierarchical diagnosis (*e.g.*, (Mozetič 1991)).

The main difference for knowledge models is that the hierarchical structure is not readily available from the blue prints, but has to be generated run-time for each base model. We therefore developed algorithms that automatically add hierarchical structure to a base model. Three different types of abstraction are subsequently applied to the model: firstly, the model is simplified by *hiding* all reasoning steps (components) that are not essential (although technically speaking necessary) to the main behaviour of the system. Secondly, sequences (*chunks*) of reasoning steps are collapsed into single components. Finally, a third abstraction is made that reduces each *specification* of a behaviour state, and every *transition* between two states, into one single component. For a detailed description of the hierarchical layers and the algorithms used for producing them, see (de Koning, Bredeweg, & Breuker 1997; de Koning 1997).

The process of generating a hierarchical model of the task of qualitative prediction of behaviour is fully automated.

Diagnosing Knowledge Models

The hierarchical models adhere to the representational constraints of model-based diagnosis, and hence techniques such as GDE can in principle be applied without modification.

In GDE, the diagnostic process consists of three steps: conflict recognition, candidate generation, and candidate discrimination. Conflict recognition amounts to finding (minimal) sets of components that, if assumed to be working correctly, results in behaviour that conflicts with the observations. From each of these sets (called *conflicts*), at least one component should be faulty for the overall behaviour to be consistent with the observations. Candidate generation creates those sets of components (called *candidates*) that cover each conflict. Candidate discrimination is concerned with sequential diagnosis: given a set of observations, the set of possible diagnoses may not be satisfactory, and additional observations are then necessary to discriminate between the possible candidate diagnoses.

The first two steps are applied without significant modification. The third step however is different. The reason is that the nature of the components is significantly different in knowledge models and digital circuits. In a digital circuit, two components of the same type may behave according to the same rules, but are still physically distinct instances. In knowledge models, this is not necessarily the case. One reasoning step applied correctly in one part of the model is very likely to behave correctly as well in another part: by their nature, different components of the same type are likely to fail collectively. A learner that does not know how to apply an inequality correspondence is likely to exhibit the same error (faulty inequality correspondence) at several places in the model. The only exception is formed by components of the retrieval type: here, different instantiations are indeed independent operations, because they refer to the retrieval of different knowledge facts. The error of not correctly 'retrieving' the relation between level and pressure is usually not related to an incorrect retrieval of the negative influence of the flow rate on the volume.

This different nature of knowledge models is exploited by the diagnostic algorithm: the failure probability of a set of instances of the same component type is defined to be equal to that of a single component. For example, a candidate diagnosis [IC_1,IC_2,IC_3,IC_4] consisting of four failing inequality correspondence components has the same probability as a single component candidate [IC]. We actually interpret a candidate at the level of *generic* inferences, instead of at the level of individual, *instantiated* reasoning steps: the first candidate can be interpreted as "unable to calculate inequality correspondence", which is at this level of interpretation a single fault. This interpretation does not hold for the single component candidate: this may well be an incidental instantiation error or slip.

For retrieval components, it is possible to employ an additional heuristic in candidate discrimination: because most errors made in the experiment appeared to be caused by missing or confused domain knowledge, retrieval components can be assumed to have a higher *a priori* failure rate than inference components. Similarly, higher-level, *decomposable* components have a higher *a priori* failure rate than individual base model components, because these components incorporate a number of reasoning steps in the base model. Note that this *a priori* failure rate is inspired by a structural feature of the model, namely the number of inference components, rather than by the semantics of the inferences themselves.

The STAR Diagnostic Engine

On the basis of the above considerations, we designed a new algorithm for candidate discrimination. The algorithm is a variant on the *half split* approach that can deal with multiple faults and differing *a priori* failure rates. The half split approach aims at finding the point that optimally splits the set of components that contributes to a symptom: given the set of components CpS that contributes to a symptom, the *splitting factor* for a possible measure point p is defined as $|CpS_{bp} - CpS_{ap}|$, where CpS_{bp} is the subset of CpS contributing to the value of p ("before p") and CpS_{ap} the subset CpS not contributing to the value of p ("after p"). In our case, simply taking the difference in numbers of components does not work: components are no longer synonym with candidates. Hence, we introduce the *weighted cardinality of a candidate*, facilitating the comparison of candidates. The weighted cardinality of a candidate expresses its probability in terms of the number and type of components it consists of.

The algorithm for candidate discrimination is given below.

1. Collect the set of all possible measure points MPS by tracing backwards from the set of *symptoms* (*i.e.* observations that do not match the predicted value) through the model.

2. For each candidate Ca_i in the candidate set CaS, calculate its *weighted cardinality* WC_{Ca_i}.
   Let R be the number of retrieval components in Ca_i;
   Let H be the number of decomposable components in Ca_i;
   Let T be the number of other component *types* in Ca_i;
   The weighted cardinality of a candidate is defined as $WC_{Ca_i} = 0.7 * R + 0.5 * H + T$.

3. Let CpS be the set of components that contribute to the set of symptoms. Define the *unnormalised probability* of a component $Cp \in CpS$ to be $\sum \frac{1}{W_{Ca_i}}$ for all candidates Ca_i containing Cp.

4. For each point p in MPS, let CpS_{bp} ("before p") $\subset CpS$ be the set of components that contribute to the value of p and let CpS_{ap} ("after p") $= CpS \setminus CpS_{bp}$.

5. For each point p in MPS, calculate its splitting factor SF_p. Let UP_{bp} be the sum of the unnormalised probabilities of the components in CpS_{bp}, and UP_{ap} the sum of the unnormalised probabilities of the components in CpS_{ap}. The splitting factor SF_p of measure point p is defined as $|UP_{bp} - UP_{ap}|$.

6. Order the probe points in MPS according to their splitting factor SF_p: the best probe point is the one with the smallest value for SF.

The algorithm first determines the possible measure points (step 1). In step 2, the weighted cardinality of each candidate is calculated. The definition of a candidate's weighted cardinality embodies the different aspects discussed above: retrieval and decomposable components have a higher *a priori* failure rate and are counted individually. Components of the same type are counted only once. By its definition, the lower the weighted cardinality of a candidate, the higher its probability. Subsequently, we can map these weighted cardinalities on the individual components, yielding the *unnormalised probability* of a component (step 3). This probability expresses the different candidates that a component is part of: when a component belongs to more than one candidate, knowing its status will provide more information. Hence, the higher a component's unnormalised probability, the more important it is to focus the diagnostic process on this component. The unnormalised probabilities of the components are used in calculating the splitting factor for each measure point (step 4). As defined in step 5, the candidate discrimination algorithm does not deliver one probe point, but a list of possible probe points ordered to their discriminating power (*i.e.*, their splitting factor). Although the diagnostic machinery can reason about the expected *results* of a certain probe point, it cannot determine the *costs* of a specific probe within the current educational context. As a result, the most effective probe suggested may be very expensive, in the sense that it does not fit in with the current dialogue. In this case, the educational system may select another probe point from the list.

The discrimination algorithm should be viewed as an implementation of a number of 'rules of thumb'. Especially the definition of the weighted cardinality of a candidate includes some numerical interpretations of qualitative observations that may prove to be non-optimal. For example, the fact that a retrieval component is counted as 0.7 is a somewhat arbitrary quantification of the observation that more errors are made in the domain knowledge than in the reasoning knowledge. Due to the hierarchical structuring of the models, the number of components that is diagnosed at once is usually small. Hence, the impact of such choices will be relatively small.

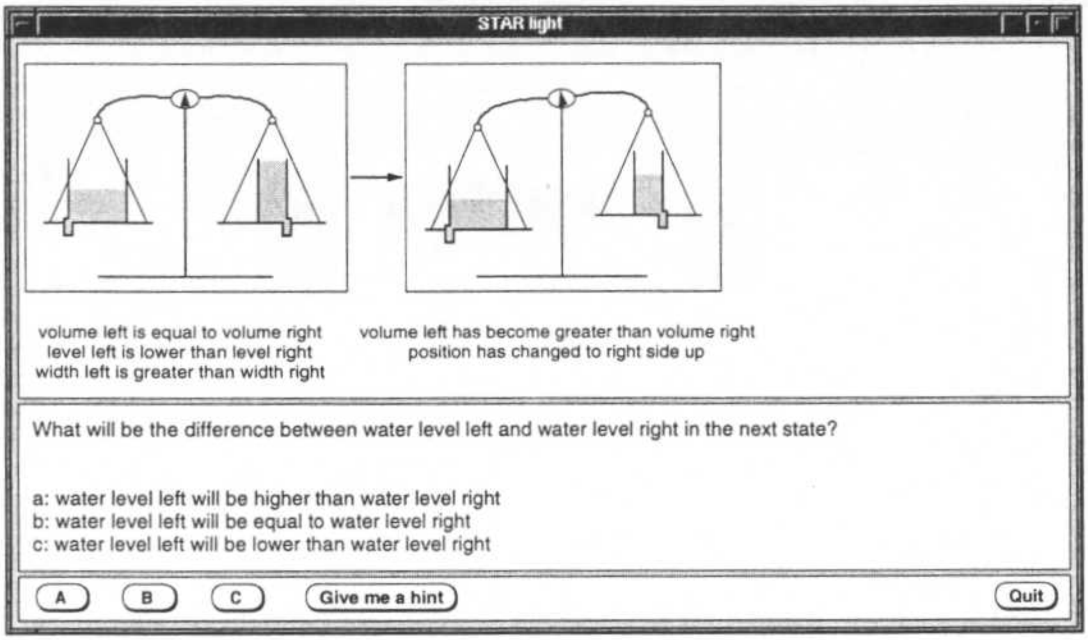
The STAR^{light} Prototype

To illustrate the ideas presented above, a diagnostic session is presented from the tests we performed with the STAR^{light} prototype. The prototype fully implements model generation, hierarchical model aggregation, and the diagnostic engine as presented in the previous sections. Furthermore, elementary implementations of a question generator and an explanation generator are added to facilitate a simple question-answer dialogue with a learner about the behaviour of some (physical) system. A screen dump of the STAR^{light} prototype is shown in Figure 2. The screen shows a typical question about the behaviour of a physical system called the *balance system*: two containers filled with water are placed on a balance, and each container has an outlet at the bottom. The weight of the empty containers is assumed equal, but their size differs. The task of the learner is to predict what happens to the water columns and to the balance once the outlets are opened.

We conducted a small set of test sessions with 9 learners. To exemplify the diagnostic machinery, we present a fragment of the interaction between the learner and the system. The indented phrases in italics are think-aloud remarks made by the learner. In this example, the initial behaviour state of the balance (the leftmost situation depicted in Figure 2) has just been presented to the learner. The dialogue starts when the learner is asked to predict the second state (the right situation in Figure 2), which is at this moment not yet visible to the student.

1. What will be the difference between volume left and volume right in the next state?
2. *Hey, look, the volumes are equal. I didn't take notice of that.*
3. *Yes, I think they stay equal?*
4. *b: volume left will be equal to volume right*
5. That's not correct.
6. Before we go on with the prediction, I would like to know what you think about the present situation.
7. Please answer the following question:
8. What do you know about the difference between the decrease in volume left and the decrease in volume right?
9. *Volume left decreasing slower than volume right?*
10. *No, at the same speed I would say.*
11. *b: volume left is decreasing at the same speed as volume right*
12. That's not true.
13. Please answer the following question:
14. What do you know about the difference between outflow left and outflow right?
15. *Well I would say that the outflow is equal, but that is now when the pressure [incomprehensible]*
16. *Ah! Is that it? The outflow left is smaller*
17. *c: outflow left is smaller than outflow right*
18. That's right.
19. Please answer the following question:
20. which quantity causes a change in volume ?
21. *The outflow, or the pressure, yes the outflow*
22. *c: outflow*
23. That's right.
24. I think you don't fully understand the following:
25. the difference between outflow left and outflow right has a direct influence on the difference between the decrease in volume left and the decrease in volume right.
26. This means that because outflow left is smaller than outflow right volume left is decreasing slower than volume right.
27. Let's continue with the exercise where we left off.
28. *All right, interesting...*

To explain the diagnostic process underlying this interaction, consider Figure 3. The answer given in line 4 supplies the observation $V_l = V_r$ in the output of the state transition component ST. ST models the transition from the first to the second state at the highest hierarchical level. In the prototype system, the subject matter sequencing is simplified to only asking the output of each subsequent transition component, and not of any specification component. Hence, a conflict at the highest level results in a decomposition of *two* components: the preceding transition component plus the previous specification component (for more details, see (de Koning 1997)). Hence, SS and ST are decomposed into the seven-component model depicted. The first call to the diagnoser delivers one conflict: $\langle$CII, IT$\rangle$, and hence two diagnoses [CII] and [IT]. The combined inequality influence CII is a higher-level component that summarises the calculation of a derivative (in this case, the ratio of the decreases in volume $\delta V_l > \delta V_r$) from an (in)equality (the ratio of the water levels $L_l < L_r$). The inequality termination IT determines the new inequality between the values in the next state. The only probe point that yields information about the candidates [CII] and [IT] is in between these components. Hence, a question is asked about the inequality between the derivatives of the volumes ($\delta V_l > \delta V_r$, line 8). The answer given in line 11 is incorrect, yielding a single-fault diagnosis [CII]. Because this is a higher-level component, it is decomposed into a lower-level model as shown in Figure 4. The next diagnostic cycle yields one conflict $\langle$TIC, R$_3$, II$\rangle$ and three candidates [TIC], [R$_3$], and [II]. For the two existing measure points, the splitting factors are determined by the discrimination algorithm. For the point *neg_infl(Fl, V)*, the splitting factor is $|\frac{1}{0.7}-(\frac{1}{0.5}+\frac{1}{1})| = 1.57$, for $F_l < F_r$ it is $|\frac{1}{0.5} - (\frac{1}{0.7} + \frac{1}{1})| = 0.43$. $F_l < F_r$ has the lowest value, and thus the highest discriminating power. Hence, this one is questioned (line 14) and answered correctly (line 17). This results in the new conflict $\langle$R$_3$, II$\rangle$ and two candidates [R$_3$] and [II]. The last probe on *neg_infl(Fl, V)* in line 20 delivers the inequality influence component II as a final single-fault diagnosis. In line 24–26, an explanation is generated for this compo-



volume left is equal to volume right
level left is lower than level right
width left is greater than width right

volume left has become greater than volume right
position has changed to right side up

What will be the difference between water level left and water level right in the next state?

a: water level left will be higher than water level right
b: water level left will be equal to water level right
c: water level left will be lower than water level right

Figure 2: The STAR^{light} Prototype

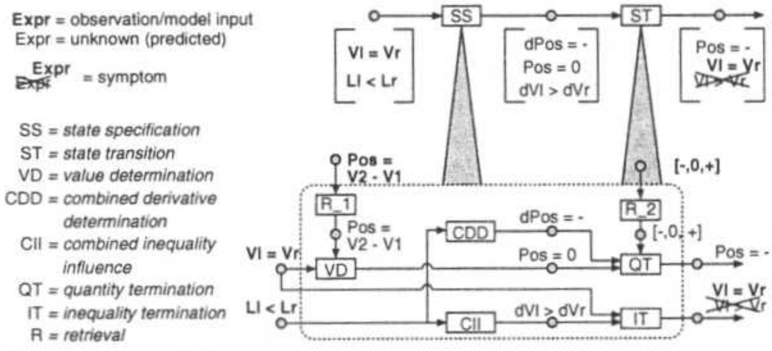


Figure 3: First Diagnostic Cycle

nent.

In the test sessions, a total of nine learners were using the prototype for about half an hour each. Of these nine learners, four had some experience in qualitative prediction of behaviour (the 'advanced learners'), whereas five had no relevant foreknowledge (the 'novices'). In total, 707 questions were answered, and there were 30 diagnostic sessions. Running on a 200 Mhz Pentium Pro platform under Linux, most diagnoses were calculated within one second, with a maximum of four seconds. The average number of probes needed to determine a satisfactory diagnosis was 2.7; the longest sequence of probes was eight, which occured twice.

Although the experimental setup and the number of subjects do not allow for drawing firm conclusions, the overall performance of the diagnoser is satisfactory: with an average of three probe questions, the diagnoser is capable of identifying one or more 'faulty components' in the model. These faulty components represent those inferences that cannot have been performed correctly by the subject.

The test sessions show a difference in competence for advanced learners and novices. The system performs very well in 'fine tuning' the learner's reasoning process. When the learner has some understanding of the prediction task, but lacks the necessary domain facts or the subtleties of the reasoning process, the diagnoses are adequate and helpful. On average, only a few probes are needed to pin down a unique diagnosis. Furthermore, the probe questions are often helpful in learning because they trigger self-repair. In such cases, the final explanation does not indicate an existing error, but serves as a confirmation of the self-repair. When the learner does not have any initial knowledge about structured

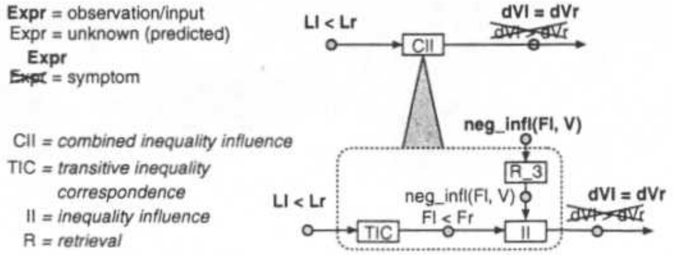


Figure 4: Second Diagnostic Cycle

behaviour prediction, the performance is less optimal: the probe questions, the explanation following the final diagnosis is not always helpful (although correct). To some extent, this result can be ascribed to the limited capabilities of the STAR*light* system with respect to phrasing questions and explanations. More importantly, novices are not always able to understand the detailed explanations of the system because they miss the necessary 'surrounding' knowledge: their problem solving behaviour may be as yet too unstructured to be discussed in terms of individual reasoning steps. In addition, novices may benefit from a more gradual introduction of the subject matter in terms of different increasingly complex models, as is for instance conjectured by the theory of *causal model progression* (White & Frederiksen 1990).

Discussion

People learn from their errors. The value of learning from errors has been recognised in influential educational philosophies such as Socratic tutoring (Collins & Stevens 1982) and LOGO (Papert 1980). The principle also plays an important role in contemporary discovery or explorative learning environments (*cf.* (van der Hulst 1996)). By means of exploration and experimentation, a learner can develop models of the subject matter knowledge involved. Errors can be a valuable aid in adjusting and refining the models developed by the learner. Knowledgeable support of the learner's trial-and-error behaviour can help the learner to learn from his or her errors in an effective and efficient way.

An emphasis on learning from errors requires a view on education that is not commonly practised by human teachers: they appear to rely mainly on pattern recognition on the basis of known misconceptions, rather than on detailed diagnostic search (*cf.* (Breuker 1990)).On the one hand, this is influenced by traditional educational views on errors: "School teaches that errors are bad; the last thing one wants to do is to pore over them, dwell on them, or think about them." (Papert 1980). And, maybe even more important, detailed structured diagnosis is often computationally infeasible for human teachers.

Compared to model-based diagnosis of reasoning behaviour, diagnosis on the basis of bug catalogues is not merely another technique to arrive at the same result. Instead, the use of pre-stored bugs involves a significantly different approach to educational guidance: in this case, the diagnostic activity is aimed at matching known misconceptions that may explain the errors made. As a result, diagnosis is often heuristic and shallow, and does therefore not play a decisive role in the teaching process. A focus on learning from errors as supported by the STAR framework yields a different teaching style: instead of directly mapping errors on misconceptions to be remediated, zooming in on the specific bug that causes the error guides the learner in discovering this error and maybe self-repairing it.

Summarising, the way in which the STAR framework provides support is one that is generally difficult or even impossible for human teachers. In the STAR approach, every error in the learner's problem solving process can be traced back to a reasoning step that cannot have been applied correctly given the observations. The probing mechanism accounts for a structured sequence of questions that, as a side effect, will stimulate the learner's self-repair behaviour. The educational methodology following from this approach is difficult or even impossible to realise with other means than knowledge-based educational systems. The diagnostic task as it is defined in the STAR framework is too complex to be feasible for human teachers. Moreover, even if they would be capable of (learning) to diagnose problem solving behaviour in a structured and detailed way, the actual application in educational practice will never be cost-effective. The STAR framework allows for close monitoring and detailed diagnosis of the learner's reasoning behaviour, showing the large potential of knowledge-based systems in education. In particular, the framework shows that the transfer of existing, solid techniques like model-based diagnosis can be successfully employed to alleviate long-standing bottlenecks in educational systems.

Conclusions

The task of cognitive diagnosis is often considered to be too complex to be cost-effective. The STAR framework counters this view by providing a generic and automated approach to diagnosing the problem solving behaviour of the learner. By exactly scoping the task of diagnosis within education, and by defining "device models" for the learner's problem solving task, it becomes possible to reuse techniques from the field of model-based diagnosis. We defined educational diagnosis as the identification of (necessary) reasoning steps that have not been performed correctly. For the task of qualitative prediction of behaviour, we designed and implemented techniques to automatically generate the necessary "device models" from the output of a qualitative simulator. Using a hierarchical variant of GDE, we showed that model-based diagnosis can be used to identify a learner's errors made in individual reasoning steps. This focus on errors in the learner's problem solving behaviour strongly influences the educational approach of the system. People learn from their errors, and the diagnostic probes can help the learner in detecting these errors. This view on education is different from the one underlying most diagnostic approaches. Instead of focussing on tracing *misconceptions* in the learner's knowledge that can be remediated, our approach can help the learner in detecting errors. These errors result in an explanation by the system, but also often stimulate the learner in self-repairing them.

Further development of the STAR framework can be pursued in various directions. The current framework is based on qualitative prediction of behaviour, which is applicable to reasoning about many types of systems such as physical devices, ecological systems, and economical systems. Within the context of qualitative reasoning, the scope of the framework can be enlarged to incorporate other problem solving tasks such as monitoring, design, or diagnosis.

A second interesting direction for extension is the development of other generic educational functions based on the model representations. Sophisticated techniques for subject matter sequencing and discourse planning are envisaged that take advantage of the hierarchical models of problem solving knowledge as defined and generated within the STAR framework.

References

Bredeweg, B. 1992. *Expertise in qualitative prediction of behaviour.* Ph.D. Dissertation, University of Amsterdam.

Breuker, J. A., ed. 1990. *EUROHELP: Developing Intelligent Help Systems.* Amsterdam: EC.

Chi, M. T. H.; Feltovich, P. J.; and Glaser, R. 1981. Categorization and representation of physics problems by experts and novices. *Cognitive Science* 5:121–152.

Collins, A., and Stevens, A. L. 1982. Goals and strategies for inquiry teachers. In Glaser, R., ed., *Advances in Instructional Psychology*, volume II. Hillsdale, NJ: Lawrence Erlbaum Associates.

de Kleer, J., and Williams, B. C. 1987. Diagnosing multiple faults. *Artificial Intelligence* 32:97–130.

de Koning, K., and Bredeweg, B. 1995. Qualitative reasoning in tutoring interactions. *Journal of Interactive Learning Environments* 5:65–81.

de Koning, K.; Bredeweg, B.; and Breuker, J. 1997. Automatic aggregation of qualitative reasoning networks. In Ironi, L., ed., *Proceedings of the Eleventh International Workshop on Qualitative Reasoning*, 77–87. Pavia, Italy: Istituto di Analisi Numerica C.N.R.

de Koning, K. 1997. *Model-Based Reasoning about Learner Behaviour.* Amsterdam: IOS Press.

Dillenbourg, P., and Self, J. A. 1992. A framework for learner modelling. *Interactive Learning Environments* 2:111–137.

Elio, R., and Sharf, P. B. 1990. Modeling novice-to-expert shifts in problem-solving strategy and knowledge organization. *Cognitive Science* 14:579–639.

Forbus, K. D. 1990. The qualitative process engine. In Weld, D. S., and de Kleer, J., eds., *Readings in Qualitative Reasoning about Physical Systems.* Morgan Kaufmann. 220–235.

Larkin, J. H.; McDermott, J.; Simon, D. P.; and Simon, H. A. 1980. Expert and novice performance in solving physics problems. *Science* 208:1335–1342.

Mozetič, I. 1991. Hierarchical model-based diagnosis. *International Journal of Man-Machine Studies* 35(3):329–362.

Ohlsson, S. 1986. Some principles of intelligent tutoring. *Instructional Science* 14:293–326.

Papert, S. 1980. *Mindstorms: Children, Computers, and Powerful Ideas.* New York: Basic Books.

Self, J. A. 1992. Cognitive diagnosis for tutoring systems. In *Proceedings of the European Conference on Artificial Intelligence*, 699–703.

van der Hulst, A. 1996. *Cognitive Tools.* Ph.D. Dissertation, University of Amsterdam.

White, B. Y., and Frederiksen, J. R. 1990. Causal model progressions as a foundation for intelligent learning environments. *Artificial Intelligence* 42:99–157.