Towards an Epistemology of Intelligent Design and Modelling Environments: The Hypothesis Testing Approach

CLAUS MÖBUS
Department of Informatics, Learning and Teaching Systems
C.v.O. University, D - 26111 Oldenburg
and OFFIS Institute, D - 26121 Oldenburg, Germany
E-Mail: Claus.Moebus@informatik.uni-oldenburg.de

Abstract: The main purpose of Intelligent Design and Modelling Environments is to offer students the opportunity to acquire knowledge while working on a sequence of given or self-selected problems chosen from the application domain. Earlier, we developed intelligent problem solving environments (IPSEs, Möbus, 1995) for various curricula and applications. Now we extend this approach to intelligent design and modelling environments in the areas of pneumatic circuits, chemical polymerisation reactions, economic simulation games, causal modeling, and diagnosis. On the surface being very different our systems follow a common design theory: the student acquires knowledge by testing his own hypotheses.

First we want to show that hypothesis testing plays a fundamental role in a *cognitive science* orientated theory of knowledge acquisition (ISP-DL-Theory). This theory is the basis of our systems. In a case study three of our most recent systems (PULSE, WULPUS, MEDICUS) and their relationship to hypothesis testing are discussed.

Then we define the concept of hypothesis testing in a *logic* framework. We describe knowledge acquisition events and learning effects. It is argued that knowledge acquisition stimulated by Intelligent Design and Modelling Environments is based fundamentally on self explaining the responses of the system to the student's hypotheses. Furthermore, we will show how the concepts of hypotheses testing and self explanation of the system‘s feedback apply to the four case systems presented.

Thus the topic of this paper is the epistemological implication:

If *there is empirical evidence for the ISP-DL-Knowledge Acquisition Theory which relies on self explanation*
then *Intelligent Design and Modelling Environments should be designed like our systems which stimulate self explanation.*

Introduction

It has been well recognized that the development of intelligent help systems raises difficult questions, like: How is help and instructional material to be designed? When should remedial information be supplied? Why is the same information useless to one person and helpful to another? Existing intelligent tutorial and help systems have not always provided satisfactory answers to such questions. For example, the information delivered to the learner may assume too little or too much knowledge, the user interaction is too restrictive, or tutoring and help strategies are unprincipled and ad hoc. These shortcomings are basically still true (Self, 1990). To make some progress a theoretical framework seems to be necessary. It should be sufficiently detailed to enable specific design decisions and predictions. At the same time it should be so general that it is applicable to different domains. This paper is a further step in that direction: we try to describe the epistemology of systems assuming the correctness of ISP-DL-Theory.

From our point of view Intelligent Design and Modelling Environments seem to be the most cost effective intelligent systems for the communication of problem solving knowledge. Though they contain an expert system or an oracle that can check the correctness of students' solution proposals, they lack other expensive components like a teaching or a student model. The curricular component in form of a teaching model is abandoned in favor of a simple sequence of task-relevant problems. The student model which should be responsible for the individualization of system responses is missing, too. Instead of that individualization is achieved by the ability of the system to respond intelligently to student hypotheses. An expert system (or an oracle) and the current student hypothesis are sufficient to generate adaptive help.

To avoid design errors the design of Intelligent Design and Modelling Environments should be guided by a *psychological* theory of knowledge acquisition. Our work is based on ISP-DL-Theory, an acronym for "Impasse-Success-Problem-Solving-Driven-Learning" (Möbus, 1995; Möbus, Schröder & Thole, 1994, 1995; Möbus, Thole & Schröder, 1993a, 1993b). ISP-DL is influenced by the cognitive theories of Anderson (Anderson, 1986, 1989), Newell

(1990) and Van Lehn (1988) as well as by the motivational "Rubikon" theory of Heckhausen (1987, 1989) and Gollwitzer (1990). The empirical evidence of the if-part of our epistemological implication is a question of cognitive psychology. Because there is much work published by others and by us we only give some pointers to the relevant literature. The theory and design principles for Intelligent Design and Modelling Environments which can be (informally) derived from it are sketched and briefly discussed in part 2.

To demonstrate the feasability of these ideas we presented three case studies earlier (Möbus, 1995) in the domains of the derivation of functional programs, modeling time-discrete distributed systems, and room configuration tasks. In part 3 of this paper, we present three new case studies from even more different domains (pneumatic circuits, business simulation, and modelling and diagnosing in medical domains) with very different (monotonic and nonmonotonic) problem spaces. Compared to the systems presented earlier (Möbus, 1995), there is a shift from closed problem solving for a fixed set of given tasks towards design, simulation, modelling, and diagnosis. This shift also includes the integration of conceptual domain knowledge. In these case studies, it is shown how close one can stick to a special design philosophy despite differences in knowledge domains and despite the use of very different AI-techniques (model checking, quantitative constraints, and Bayesian networks).

In part 4 we summarize and abstract the results of the case studies. We define formally the concept of a hypothesis in a knowledge revision framework. We show that *hypothesis testing* can be integrated into *theory revision* and *knowledge acquisition* processes of an abstract problem solver. We discuss the question *when* knowledge acquisition events will happen and *what* kind of knowledge is acquired when working with these Intelligent Design and Modelling Environments. We present our main hypothesis that knowledge acquisition stimulated by Intelligent Design and Modelling Environments is based fundamentally on self explanation: the student should try to explain the responses of the system to his hypotheses by himself.

ISP-DL: A Theory of Knowledge Acquisition and Design Principles

From our own empirical investigations (Möbus & Schröder, 1993) we concluded that it is fruitful to describe learning as an interplay of impasse- and success-driven learning (Möbus, 1995). Learning has two aspects: the process of knowledge optimzation occurs after a solution has been found. This process is *deductive* in the sense that the new optimized knowledge is a logical consequence of old knowledge:

$$\textbf{background knowledge} \cup \textbf{evidence} \models \textbf{optimized knowledge}$$

The more interesting knowledge acquisition process occurs after solutions have been found with the help of heuristics. This process is *inductive*:

$$\textbf{background knowledge} \cup \textbf{new knowledge} \models \textbf{evidence}$$

so that heuristics can be seen as inductive inference rules.

When do we expect hypothesis testing activities? We assume that the problem solver has a solution proposal for the given task. This is evaluated by mental or real time simulation or asking an oracle (eg. the Intelligent Design and Modelling Environment). When there is negative feedback the student realizes an impasse. The reaction to that is planning and use of weak heuristics. One of them is testing a hypothesis: that means asking the system questions concerning the solution status of parts of the original defective proposal: "Is this part of my solution proposal embeddable in a correct solution?".

The ISP-DL Theory motivates the following principles:

(1) The Intelligent Design and Modelling Environment should *not constrain and interrupt* the problem solver but offer information only on demand. According to the theory, information is only helpful at impasse time. We think that it is important first to let the learner develop her/his own solution ideas and then later optimize his solutions. As novices are rather "creative" in generating unusual solutions the systems should be sufficiently powerful.

(2) The student should have the opportunity to obtain detailed feedback and information *any time*. Since impasses are possible at *different phases* of problem solving, the system must offer support in the problem solving phases *planning*, *implementation*, and *evaluation*.

(3) The learner should be enabled to *make use of her/his pre-knowledge* as much as possible when asking for help. Thus the information provided should be conditional to his hypotheses and preknowledge to avoid follow-up impasses.

(4) The information provided should in *grainsize and amount* be *tailored to the knowledge state* of the problem solver. If the grainsize of the information is too fine or too coarse and the amount not synchronized to the knowledge deficit, then the problem solver has to filter or generate new information which can have undesirable emotional effects preventing progress. An (expensive) *student model* is needed only if there is a set of help alternatives to *choose* from.

(5) It is necessary that the learner is free in the choice of his problem solving operators and her/his interaction modality. We should offer an Intelligent Design and Modelling Environment for *free and unconstrained problem solving and modelling.*

Case Studies

Now in addition to the three systems described earlier (Möbus, 1995), we want to describe three more systems (PULSE, WULPUS, MEDICUS) which are designed according to ISP-DL theory and which enable hypothesis testing for the student.

The knowledge domains of the three systems differ eg. with respect to the availability of expert knowledge. In PULSE *expert knowledge* is not readily available for the correct deduction of pneumatic circuits from function diagrams, though it is possible to check the correctness of a students' solution proposal: model checking. In this respect, PULSE is an application of PETRI-HELP, a system described in Möbus (1995). Students solve the problem only with rules of thumb or with heuristics. In WULPUS *expert knowledge* represented as quantitative constraints is also in principle available in order to derive good decisions, though the learner may not use it. In MEDICUS, the expert knowledge is probabilistic. Much of this knowledge is hidden in linguistic terms like fuzzy concepts and relations, and even experts hesitate to state numerical relations e.g. between expositions to poisonous substances and certain diseases. The knowledge is more complex and interrelated than in the other domains.

PULSE (Pneumatic Learning and Simulation Environment) supports the construction of pneumatic circuits. PULSE is designed to encourage explorative learning. It is developed in cooperation with the DIHT Society of Professional Training and the local chambers of commerce. PULSE consists of the following parts:

- A set of construction tasks of increasing difficulty, for example, constructing the circuit for a pneumatic ramp. If a task is selected, a verbal description and a function diagram describing the desired behaviour of the to-be-designed circuit are presented to the user.
- An editor for constructing pneumatic circuits
- A hypotheses test environment for stating hypotheses about the correctness of the actual proposal with respect to the function diagram. PULSE then analyses the hypothesis and delivers error feedback and, on further request, explanations.

Figure 1 shows the pneumatic circuit for a lifter. The learner creates a solution proposal for a task specification, expressing her or his knowledge, intentions, and heuristics. The learner‘s solution proposal is analysed by model checking (e.g., Clarke et al., 1986), that is, it is verified whether the solution proposal behaves as specified in the function diagram. Model checking can be viewed as an oracle, that is, it can analyse any solution proposal created by the learner, and comment on it. The system is also able to explain why a solution proposal is buggy. This explanation uses conceptual knowledge about the parts used in the construction (cylinders, valves, etc.), and their connections.

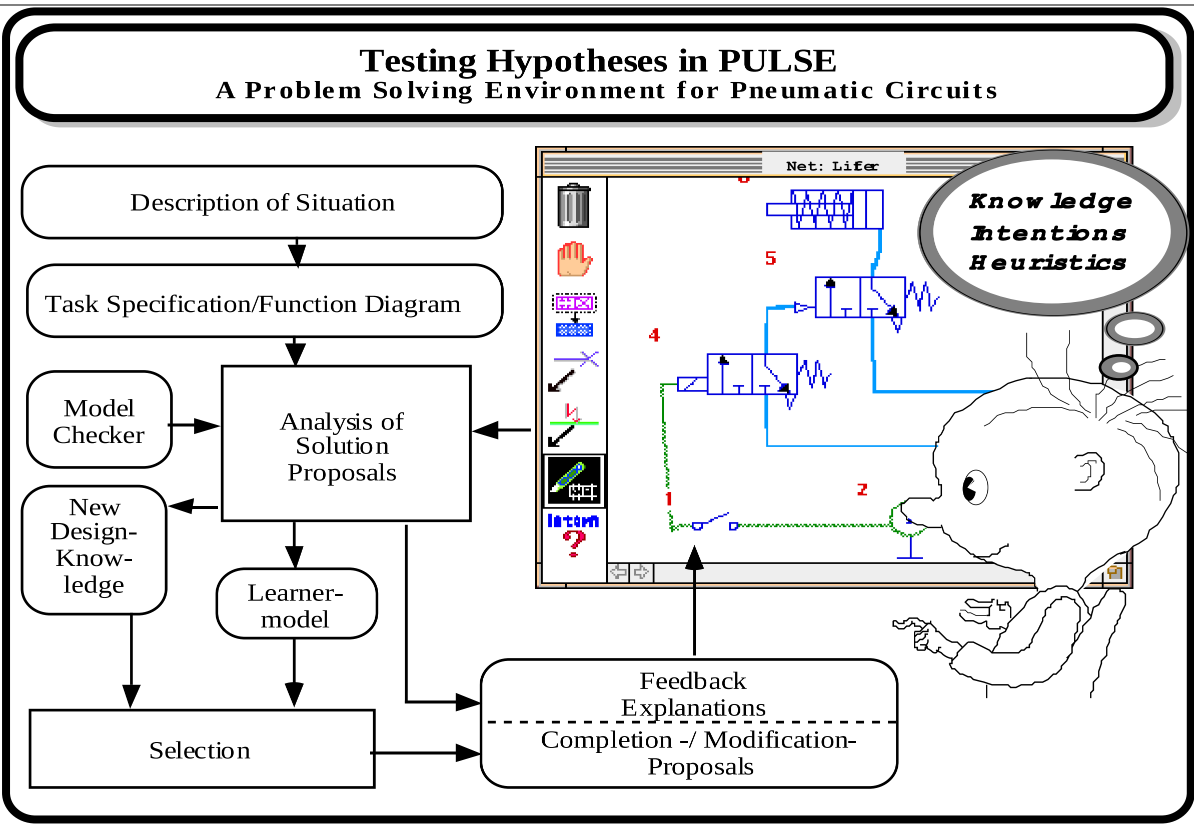

Figure 1: The Intelligent Design Environment PULSE

WULPUS (German acronym for Knowledge-based Support for LUDUS, a Simulation Game for Business Strategies; Figure 2) is designed to support users in acquiring basic knowledge in economics and business management, and to help learners recognise the relationships between business goals, decisions, and outcomes in an oligopoly situation. There are three important differences of WULPUS to the „classical“ simulation game approach. Firstly, in „classical“ simulation games the learner states decisions and inspects their outcomes. But in WULPUS the learner states business goals and hypotheses about their reachability. For example, the user states sales goals, specifies marketing decisions, and revises the goals if needed. Secondly, in the classical approach decisions cannot be undone. In WULPUS, the learner may try different decisions, evaluate their consequences, and test various hypotheses before making a „real“ decision („simulation in the simulation“). Thirdly and most importantly, in WULPUS the learner gets qualitative and quantitative feedback and explanations in response to hypotheses testing. This feedback carries the knowledge necessary to gain an understanding about the economical relationships. So new knowledge is presented right when it is needed. This differs sharply from the classical situation, where the learner usually has to acquire a lot of pre-knowledge *before* he is able to play a simulation game successfully.

WULPUS consists of two main parts. In the first part, the learner tries to find a marketing mix by testing hypotheses that fulfills his goals most closely. After settling on a specific marketing mix, the learner specifies the decisions necessary for the marketing mix. These decisions concern the fields of ordering and storing material for production, personnel, financial situation, and so on. Again, in case of impasses the learner may ask for additional information clarifying the relationships between the actual concepts at hand.

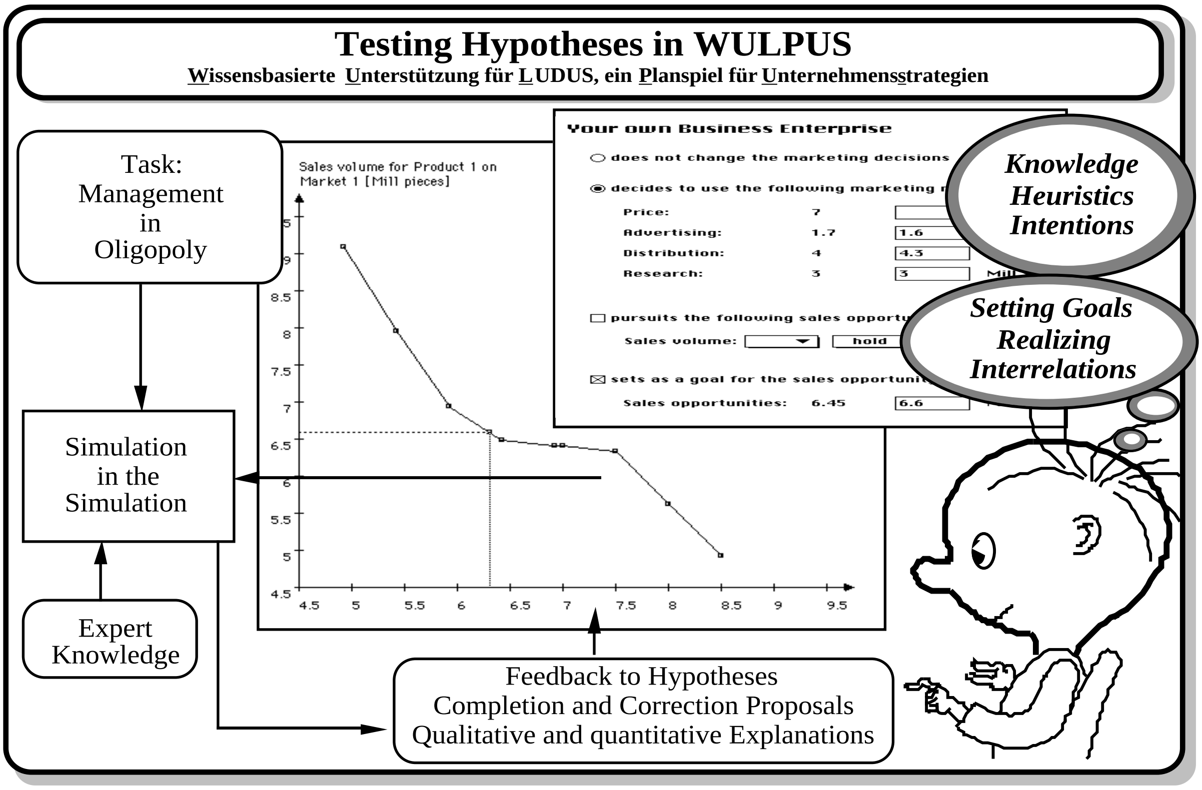

Figure 2: The Intelligent Decision Support and Simulation Environment WULPUS

MEDICUS (Modelling, explanation, and diagnosis support for complex, uncertain subject matters) is an intelligent environment currently developed in cooperation with several medical institutions. MEDICUS supports modelling and diagnostic reasoning in the fields of environmental medicine and human genetics. These domains are two yet new subdomains of medicine receiving increasing research efforts, but still consisting of largely fragile and uncertain knowledge. In MEDICUS, uncertainty is handled by the Bayesian network approach.

Thus the *modelling task* for the user consists of creating a Bayesian network for the problem at hand. But since we want mathematically untrained persons to work with MEDICUS, the user may alternatively state propositions verbally and let the system generate a Bayesian network proposal. This differs from existing reasoning systems based on Bayesian networks, i.e. in medical domains, which contain a built-in knowledge base that may be used but not created or modified by the user. When the system generates a graph proposal, the user can refine it qualitatively and quantitatively. On the qualitative level, is is checked whether the independence and dependence assumptions implied by the graph are in accordance with the modelling intentions of the user. On the quantitative level, the user can specify apriori and conditional distributions, and the system computes marginal distributions and, after specifying evidence, aposteriori distributions.

The *diagnostic reasoning task* for the learner consists of using the network for stating diagnostic goals, and for proposing diagnostic hypotheses and examinations. The system gives propagation-based recommendations about what diagnostic hypotheses to pursue next, and what diagnostic information to look for.

With respect to environmental medicine, the area of environmental monitoring is a central application field. This concerns planning, executing, and evaluating in-room investigations for poisonous substances contained in the air, in drinking water, and in food. From the point of view of our cooperation partners, MEDICUS will serve a *quality ensuring function* in this respect. Another application field is history taking, here we plan to integrate MEDICUS within diagnostic training sessions within postqualification courses for physicians. With respect to human genetics, we will provide applications of MEDICUS that can be used in consulting via communication networks.

In MEDICUS, *hypotheses testing* comes into play in the modelling component and the diagnosis support component. In modelling, the user states the hypothesis that his proposal is compatible with dependence and independence assumptions elicited in a dialog. In diagnosis, the learner may state diagnostic hypotheses and hypotheses about what information should reasonably be acquired next.

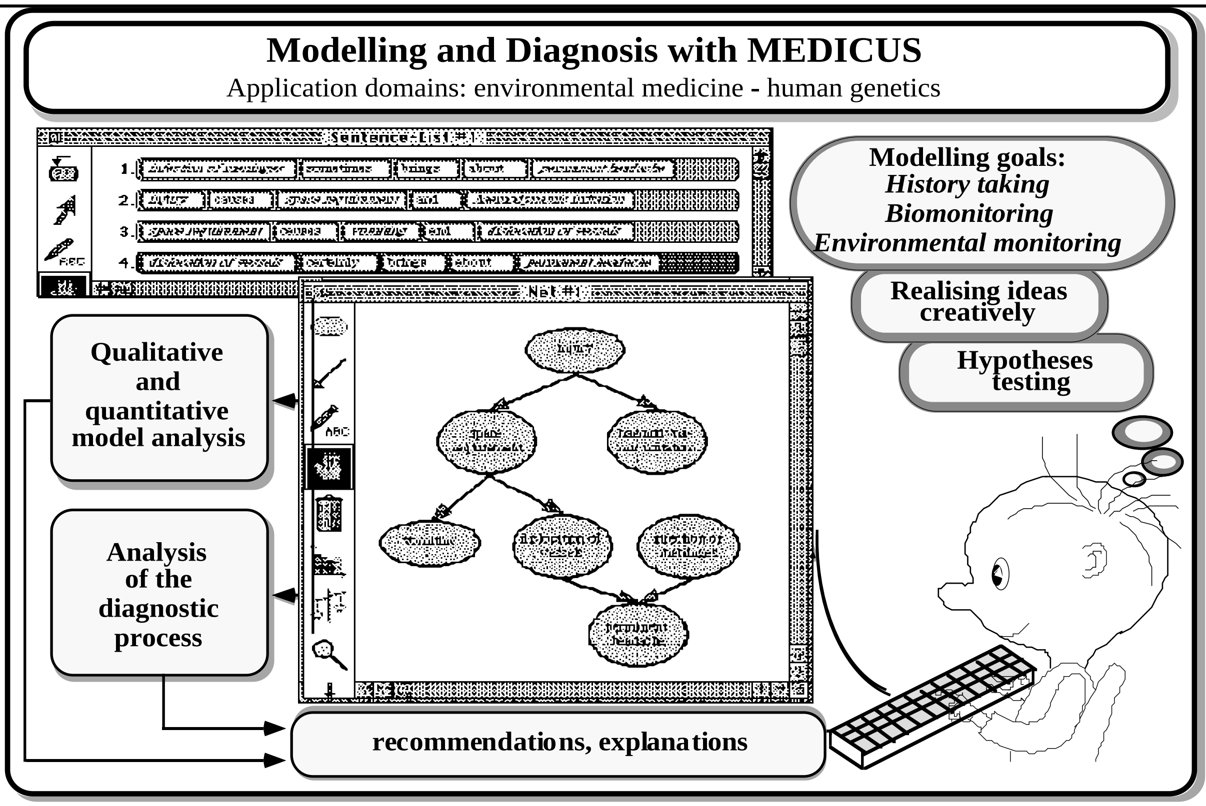

Figure 3: The Intelligent Modelling and Diagnosis Environment MEDICUS

Theory Revision, Hypotheses, Knowledge Acquisition and Self Explanations

As we stated before the formulation and testing of hypotheses is an important concept in the development of Intelligent Design and Modelling Environments. Though we may have an intuitive idea what a hypothesis is we try to give a formal definition. The definition is embedded in the concept of theory revision (De Raedt, 1992). We try to be as abstract as possible so that hypothesis testing in various Intelligent Design and Modelling Environments can be subsumed as special cases. The main points are summarized in Figure 4.

According to ISP-DL theory there are several steps when acquiring knowledge with Intelligent Design and Modelling Environments. (1) The problem solver generates with his subjective theory S evidence E, which may be a solution proposal. From the viewpoint of an ideal expert this proposal may be wrong. (2) This proposal E is submitted to the system. If the proposal is in error it cannot be explained by the system‘s domain theory T. (3) So the problem solver can generate a hypothesis and partition his proposal E into two parts Efix and Emod. The student has the hypothesis that Efix can be embedded into a correct solution. (4) Now, the system generates with its theory T a system response to the hypothesis. E' is a system generated solution proposal, which contains Efix. E'mod is help information for the student which in our Intelligent Design and Modelling Environments is shown to the student on demand. (5) After these events (hopefully) we have some knowledge acquisition events. The student tries to explain E' with its parts Efix and E'mod to himself. As a result of this, the learner generates new knowledge S'mod. As indicated in (6), with this new knowledge he is able to understand E'. According to (6) this is an inductive inference, because S'mod can be inferred inductively from $S \setminus S_{mod} \cup \{E'\}$. The comparison of (1) with (5) results in a revised theory S'.

(1) Problem Solving: $S \models E$

(2) Incorrect Proposal: $T \not\models E$

(3) Stating Hypotheses: $E = E_{fix} \cup E_{mod}$ **and:** $T \models E_{fix}$

(4) Completion Proposal: $T \models E'$

where: $E' = E_{fix} \cup E'_{mod}$ **and:** $T \models E'_{mod}$

(5) Self Explanation: $S' \models E'$

where: $S = S_{fix} \cup S_{mod}$

and: $S' = S_{fix} \cup S'_{mod}$

(6) (Inductive)
Knowledge Modification: $S \setminus S_{mod} \cup S'_{mod} \models E'$

respectively: $S \setminus S_{mod} \cup \{E'\} \mid< S'_{mod}$

Figure 4 : Problem Solving, Hypotheses Testing, Selfexplanation and (inductive) Knowledge Modification

The following table relates S, E, T, Efix, and E'mod to the case studies described above.

Theory	PULSE	WULPUS	MEDICUS
S	knowledge about circuits construction heuristics	knowledge about marketing heuristics	medical knowledge diagnostic strategies
E	pneumatic circuits	marketing mix configuration of decisions corporate objectives	verbal or formal model, set of diagnostic decisions
T	model checker conceptual knowledge	business expert knowledge quantitative constraints	medical expert knowledge qualitative and quantitative probabilistic knowledge diagnostic strategies
Efix	part of circuit	part of decision configuration	part of model, or subset of decisions
E'mod	system-generated completion proposal: part of circuit	system-generated completion proposal: decisions	system-generated completion proposal: part of model, decisions

Summary

We tried to show how a cognitive theory of knowledge acquisition (ISP-DL) motivates the concept of Intelligent Design and Modelling Environments and how the hypothesis testing capability can be described on a logical metalevel and implemented in various domains. The implication was an epistemological research question, the if-part a topic of cognitive psychology, the then-part a matter of AI and the complete modus ponens a project in cognitive science.

References

Anderson, J.R. (1986). Knowledge Compilation: The General Learning Mechanism. In: R.S. Michalski, J.G. Carbonell, T.M. Mitchell, Machine Learning II. Kaufman, 289-310

Anderson, J.R. (1989). A Theory of the Origins of Human Knowledge, Artificial Intelligence, 40, 313-351

Chi, M.T., De Leeuw, N., Chiu, M.-H. & LaVancher, Chr. (1994). Eliciting Self-Explanations Improves Understanding, Cognitive Science, 18, 439-477

Clarke, E.M., Emerson, F.A. & Sistla, A.P. (1986). Automatic Verification of Finite-State Concurrent Systems Using Temporal Logic Specifications. ACM Transactions on Programming Languages and Systems, Vol. 8, No. 2, 244-263

De Raedt, L. (1992). Interactive Theory Revision, San Diego: Academic Press

Gollwitzer, P.M. (1990). Action Phases and Mind-Sets, in: E.T. Higgins & R.M. Sorrentino (eds), Handbook of Motivation and Cognition: Foundations of Social Behavior, Vol. 2, 53-92

Heckhausen, H. (1987). Perspektiven einer Psychologie des Wollens, 121 - 142, in: H. Heckhausen, P.M. Gollwitzer & F.E.Weinert (eds), Jenseits des Rubikon: Der Wille in den Humanwissenschaften, Berlin: Springer Verlag

Heckhausen, H. (1989). Motivation und Handeln, Heidelberg: Springer, (second ed.)

Möbus, C. (1995). Towards an Epistemology of Intelligent Problem Solving Environments: The Hypothesis Testing Approach, in J. Greer (ed), Artificial Intelligence in Education, Proceedings of AI-ED 95, Washington, D.C., August 16-19, 1995, Charlottesville: AACE, 138-145

Möbus, C., Schröder, O. (1993). The Acquisition of Functional Planning- and Programming Knowledge: Diagnosis, Modelling, and User-Adapted Help, in G. Strube, K.F. Wender (eds), The Cognitive Psychology of Knowledge. The German Wissenspsychologie Project (Advances in Psychology Series), Amsterdam: Elsevier (North-Holland), 233-261

Möbus, C., Schröder, O. & Thole, H.J. (1994). Diagnosing and Evaluating the Acquisition Process of Programming Schemata, in J.E. Greer, G. McCalla (eds), Student Modelling: The Key to Individualized Knowledge-Based Instruction (Proceedings of the NATO Advanced Research Workshop on Student Modelling, in St.Adele, Quebec, Canada), Berlin: Springer (NATO ASI Series F: Computer and Systems Sciences, Vol. 125), 211-264

Möbus, C., Schröder, O. & Thole, H.J. (1995). Online Modeling the Novice-Expert Shift in programming Skills on a Rule-Schema-Case Partial Order, in: K.F. Wender, F. Schmalhofer & H.D. Böcker (eds), Cognition and Computer Programming, Norwood N.J.: Ablex publishing Corporation, 63-105

Möbus, C., Thole, H.-J. & Schröder, O. (1993a). Interactive Support of Planning in a Functional, Visual Programming Language, in P. Brna, S. Ohlsson, H. Pain (eds), Proceedings AI-ED 93, World Conference on Artificial Intelligence and Education, Edinburgh, 362 - 369

Möbus, C., Thole, H.-J. & Schröder, O. (1993b). Diagnosis of Intentions and Interactive Support of Planning in a Functional, Visual Programming Language, in D.M. Towne, T. de Jong, H. Spada (eds), Simulation-Based Experiential Learning, Berlin: Springer (NATO ASI Series F: Computer and Systems Sciences, Vol. 122), 61 - 76

Newell, A. (1990). Unified Theories of Cognition, Cambridge, Mass.: Harvard University Press

Self, J.A. (1990). Bypassing the Intractable Problem of Student Modelling, in C. Frasson, G. Gauthier (eds), Intelligent Tutoring Systems, Norwood: Ablex, 107-123

VanLehn, K. (1988). Toward a Theory of Impasse-Driven Learning, in H. Mandl, A. Lesgold (eds), Learning Issues for Intelligent Tutoring Systems, Berlin: Springer, 19-41

Acknowledgements: I want to thank the following researchers from our unit on "Learning and Teaching Systems" without their contribution this paper would have been only a dream: "Luftschloß" ("castle made of air"): Jörg Folckers, Hermann Göhler, Karsten Rommerskirchen, Olaf Schröder, Heinz-Jürgen Thole, Janine Willms.