Mathematics in Virtual Knowledge Spaces: User Adaptation by Intelligent Assistants

Sabina Jeschke[1], Thomas Richter[1], Ruedi Seiler[1]

[1]Berlin University of Technology, Faculty II – Mathematics & Natural Sciences, 10623 Berlin, Germany

Scientist's and engineer's workplaces are about to change: Numerical software and computer algebra systems remove the burden of routine calculation, but demand the ability to familiarise with new concepts and methods quickly. Given the rapid growth of knowledge in today's sciences, traditional "learning on supply" is no longer applicable; instead, learning and teaching methods have to be established that drive learners towards efficient self-controlled learning.

The value added by the New Technologies is the ability to enrich the areas of classical educational fields by some limited form of "intelligence" and by suitable interfaces to allow combinations of these areas. Thus, we propose tools that are not only able to adapt to the learning process of the student, but that are also smart enough to point towards additional background information, and smart enough to find the required information in the given context.

We present the application of New Media in four areas: Knowledge management, i.e. the presentation of mathematical content, intelligent lexicon toolkits, homework training courses and Virtual Laboratories.

Keywords: Intelligent Assistant; User Adaptation; Semantic Retrieval; Virtual Laboratory

1. Introduction: Why Intelligent Assistants?

Mathematics is the key technology of the 21st century: Besides being a research field of its own, it is the key ingredient for studies in engineering sciences, physics, computer science and many other fields. Teaching mathematics therefore means teaching a very broad, heterogeneous audience with varying fields of interest; teaching at the Berlin University of Technology means especially handling increasing student numbers with decreasing funding. Luckily, mathematics by itself is a highly structured field using a very precise language. Its internal structure is built on well-developed entities, e.g. fields, vector spaces, linear mappings, all integrated into a well accepted ontology. Given that more and more of the computational tasks are solved by the computer today, the demand for *understanding the concepts* and *interpreting the results* of the electronically performed operations becomes a major task of the mathematical education. Therefore, the structure of mathematics has to be understood - and can be exploited as well to aid the design of electronic tools supporting the learning process.

Presumably this technical assistance for teaching mathematics is suitable and fruitful as more structure is found in mathematics than in any other field taught at universities. Given the heterogeneous background of students and the versatility of fields that *require* mathematics as a precondition for advanced studies, we must, however, *go beyond* the first generation of eLearning [4] which was often not much more than computer assisted document management. We therefore need technology that provides enough flexibility to adapt to the requirements of the field and the learning process of the student. The goal of a rich variety of learning scenarios and the integration of multiple applications, as well as the desired scientific broadness of the contents increases the complexity of the user interfaces furthermore. eLearning environments therefore have to evolve from complex toolkits to systems that contain a certain amount of autonomy, enough to support human operation processes by a degree of artificial intelligence.

In short: We need *intelligent assistants*. Therefore the concepts which have been recently developed in the field of Artificial Intelligence have to be adapted to Virtual Knowledge Spaces and their components. The impact of intelligent assistants reaches from adaptivity to specific usage patterns up to active support of the learning or research process itself.

Corresponding author: Thomas Richter, e-mail: thor@math.tu-berlin.de, Phone: +4930 3142 7684, Fax +4930 3142 4413

In the following, a classification of eLearning support into four categories is presented, always keeping in mind that these have to be understood as intertwined and entangled with each other: The content area, presenting matter in a structured way as it is typically developed in lectures and courses, the training area providing a framework for homework assignments and hands-on training, the semantic retrieval area that answers individual requests by organising its contents in knowledge nets and visualising its results, and last but not least the virtual lab area that allows self-controlled learning by providing the infrastructure for experiments and hands-on training.

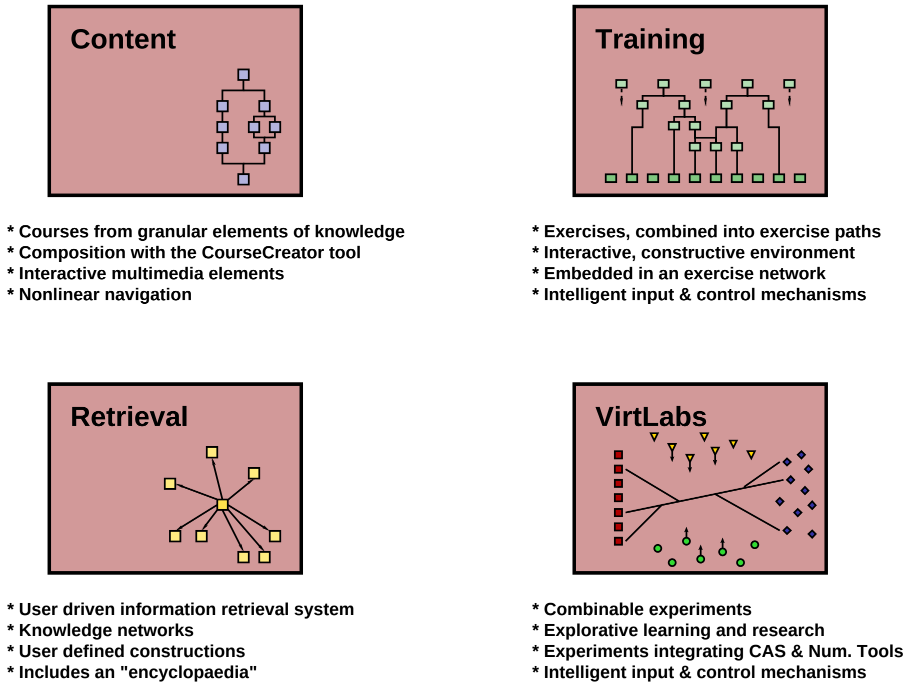

Fig. 1: Categories of eLearning.

2. Content Area

The content area is the electronic representation of the content of a specific course or lecture; to aid the lecturer and the student, the topic of a course is separated into minimal knowledge atoms following a specific matter ontology, e.g. theorems, conclusions or examples are enriched by interactive applets and composed to courses. This concept allows on the on hand efficient reuse of existing atoms and their re-composition to courses to simplify the task of the lecturer, and on the other hand it allows non-linar navigation for students to either follow courses or refresh their knowledge in exam preparation.

In its current form, created to a major extent by S. Jeschke and R. Seiler within the Mumie-project [8], it consists of a data base providing the knowledge atoms, an application server delivering the contents as HTML data and a course creator tool that defines the linkage of the atoms - as vertices in a graph - to a complete topic. A course runs like a red thread through this graph.

Clearly, this design allows reusing existing material easily to adapt courses to changing requirements. The prospects for intelligent assistants in the content area are manifold: They would observe the usage patterns of both the lecturer creating and the student using a course and thus would allow adapting the presented material to the corresponding user. Looking at the vertices of the course-graph, an assistant would be able to select the proper presentation style of the content that fits the learner's style best, e.g. mathematical exact representation vs. visualisation. But assistants could also help lecturers to combine the atoms to courses given the demands of the audience and the author at hand, and thus act on the edges of the course graph as well. Ideally, an assistant would be able to make a proposition of a full course, given the dependencies between the atoms and the preferences and demands of the audience.

3. Semantic Retrieval Area

While the Content Area requires that the content it provides to students and teachers is already integrated into a network of knowledge atoms, the main focus of the Semantic Retrieval Area is to construct these networks from mathematical texts formulated in natural language. The resulting knowledge nets represent connections between terms as well as dependencies between mathematical statements. A retrieval system will then extract the required connections between the terms and statements for the user, for example in the form of a graph.

In its current form, mainly developed by N. Natho as an additional tool within the Mumie-environment [8], the existing software implements a semi-automatic natural language parser that analyses mathematical texts for their internal structure. Concerning the Content Area, due to the high technical formality of the language used in mathematical texts, extracting the semantics should be easier than in any other field. Though it is likely that the system will remain semi-automatic, requiring help by a human mathematician.

Even though this system already qualifies as an intelligent assistant by itself, especially its retrieval component could be enhanced by making use of assistants that try to find a representation of the contents suitable for a given user profile, thus to adapt the answers of the system to its user. Clearly, another application of the semantic retrieval system would be the initial step of reverse-fitting an existing lecture in paper form into the Content Area.

4. Training Area

The Training Area provides students with highly-structured exercises to delve into the matter of the lectures to a higher degree than by just following the lecture. The keywords here as in all other fields discussed so far are *granularity* and *structure* of the exercises. Thus, a given assignment is structured into subproblems to be solved by the student such that a corresponding easier exercise could be given to focus on trouble points in case the student gets stuck. In other words, the Training Area is designed around a hierarchically structured graph of exercises providing individual learning units. This type of adjustment enables learners to gradually enhance their competencies in self-directed problem solving.

As shown in the graph, training networks are typically no trees. They are true graphs containing the same nodes, even on different levels in the hierarchical network, as the position of a node depends on the context where the exercise represented by it is needed.

The Training Area concept in its current state, developed mainly by S. Jeschke and R. Seiler [8], is mainly based on Java applets that state classical exercises in the field of undergraduate linear algebra, allowing the student to gain the required score to be admitted to the final exams.

One of the major challenges in developing multimedia-based learning and teaching platforms is to drive dynamical validation forwards, i.e. using internal or external tools that are able to validate *dynamically* and *on the fly* the correctness of a statement. As far as the field of mathematics is concerned, computer algebra systems, automatic proving — if available — as well as specific numerical software have been used. The possibilities made available by linking external tools to a training environment are currently not well exploited. A central reason for this unnecessary restriction is the lack of proper interface definitions — or even the total lack of any interface at all — at both the training tools as well as the external software. Intelligent validation tools are characterised by their tolerance towards various notations and formulations of the same subject. To achieve this tolerance, it is not only necessary to encode complex answers semantically, but also to use and interpret the same semantic correctly within the validation toolkit. The semantic coding of scientific content is therefore an important research challenge (cf. [7], [9], [12]).

The prospects for intelligent assistants in this field are manifold, and the functionality provided by them is in most cases imperative to make learning within the Training Area fruitful. As discussed above, user input must be validated and qualified, and if weaknesses show up, intelligent feedback in the form of hints or reinforcing exercises must be generated. Additionally, intelligent feedback must be given to the learner as to why his exercise might have failed, and which matter to study or repeat. Intelligent tutoring will help the Training Area tools to provide exercises that both adapt to the learning process of the student as well as the preferred presentation form of the contents — for example students might either prefer a mathematical exact definition or a handwaving demonstration that appeals to intuition. By observing the user behaviour an intelligent tutoring system would then be able to offer a suitable exercise to train the deficiencies of the student.

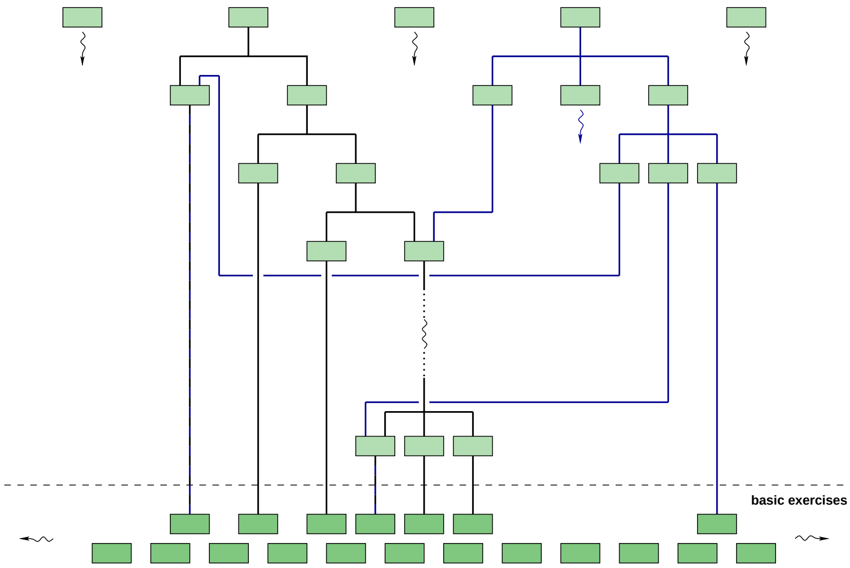

Fig. 2: Training Network of Exercises showing graph like dependencies with basic exercises in the bottom row.

5. VirtLab Area

Virtual Laboratories use the metaphor of a scientific laboratory as a guiding line for the design of the software. Applications of laboratories range from practical support for traditional lectures, over homework assignments and practical training for students up to aiding researchers in experimentation and visualisation. Thus, virtual labs have to deal with the problem of a very broad audience even more than any other of the discussed topics:

To address lecturers, it must be easy to setup and perform experiments, optimally by graphically combining the required components. An important aspect for practical training courses is that a given problem has to be solved within the interaction of a group of learners through cooperation. It is reasonable to demand the support for cooperative learning scenarios as an important feature of this concept of Virtual Laboratories. Last but not least, to make a Virtual Laboratory applicable to real-life research problems, its flexibility and proper integration of well-accepted computer algebra systems is imperative.

The given demands have consequences on the architecture of the software at hand: To offer interfaces that are adapted to the corresponding group of users, Virtual Laboratories are separated into kernels carrying out the mathematical and numerical modelling of the physics, and user interfaces that allow the users to set up and control the experiments as well as to inspect the results in a suitable visualisation. A third layer, the so-called “Connectors” interact between kernel and user interface, allowing to link virtual labs of varying origin to each other by providing both a software-technical as well as a semantical transformation between distinct implementations.

In its current form, mainly developed by Th. Richter, the existing Virtual Lab [11] focuses on the field of statistical physics and statistical mechanics. This field ideally combines mathematical research and its application to important problems of natural science, engineering and economy. Probability, analysis, dynamical systems, cellular automata are the important mathematical disciplines which can be brought to action in an environment of interesting applications like image data compression, denoising, phase transitions, transmission from microscopic reversibility to macroscopic irreversibility for large numbers of constituents.

The kernel consists of a C++ server that implements a configurable cellular automaton implementing the microscopic laws of the corresponding experiment, as well as add-on tools to measure observables like entropy or particle number on these setups. Both, microscopic laws and measurement tools are formulated in a problem-specific language and compiled and linked into the core at run-time. A wide range of experiments formulated in this language is already available, ranging from the Ising model, lattice gases, image denoising, measuring intrinsic energy, entropy, magnetisation and much more [6].

To address the various user groups the offered user interfaces range from simple Java applets to be used as browser plug-ins to an Oorange interface allowing free graphical composition of the toolkits to an interface towards the computer algebra software “Maple” to run and evaluate even complex setups.

In order to support users, the laboratory also provides assistants which user interfaces might download from the server in the form of compiled Java code. These assistants may either provide simplified or streamlined user interfaces for a specific problem at hand, or might guide the user towards exploring a certain experiment using so-called storyboards. For that, the assistants observe user behaviour, try to provide intelligent feedback and redirect the user to additional experiments to gain further insight.

Acknowledgements: The support by the DFG (Deutsche Forschungsgemeinschaft) in the framework of the DFG Research Centre Matheon (Mathematics for key technologies) and by the BMBF (Bundesministerium für Bildung und Forschung) in the framework of the Mumie-Project is greatly acknowledged.

References

[1] JavaBeans. http://java.sun.com/products/ejb/ .
[2] N. Dahlmann, S. Jeschke, R. Seiler, and U. Sinha. MOSES meets MUMIE: Multimedia-based Education in Mathematics. In Freddy Malpica, Andrés Tremante, and Nicoletta Sala, editors, International Conference on Education and Information Systems: Technologies and Applications, pages 370–375, Orlando, Fla., 2003. International Institute of Informatics and Systemics.
[3] S. Jeschke. Mathematik in Virtuellen Wissensräumen – IuK-Strukturen und IT-Technologien in Lehre und Forschung. PhD thesis, Technische Universität Berlin, Berlin, April 2004.
[4] S. Jeschke and R. Keil-Slawik. Next Generation in eLearning Technology – Die Elektrifizierung des Nürnberger Trichters und die Alternativen. Informationsgesellschaft. Alcatel SEL Stiftung, 2004.
[5] S. Jeschke, M. Kohlhase, and R. Seiler. eLearning-, eTeaching- & eResearch-Technologien - Chancen und Potentiale für die Mathematik. DMV-Nachrichten, Juli 2004.
[6] S. Jeschke, T. Richter, R. Seiler. VideoEasel: Architecture of Virtual Laboratories on Mathematics and Natural Sciences. Proceedings of the 3rd International Conference on Multimedia and ICTs in Education, June 7-10, 2005, Careres/Spain, to appear June 2005.
[7] Math Working Group, W3C. MathML. http://www.w3.org/Math/
[8] Mumie community. Mumie. http://www.mumie.net .
[9] O. Caprotti and D. Carlisle. OpenMath and MathML: Semantic Mark Up for MathematicsOpenMath. http://www.openmath.org.
[10] Oorange: The Oorange development environment. http://www.oorange.de.
[11] Thomas Richter. VideoEasel. http://www.math.tu-berlin.de/~thor/videoeasel .
[12] Saarland Universität, TU Eindhoven, RISC Linz. OMDoc. http://www.mathweb.org/omdoc/
[13] Todd Scallan. a corba primer. http://www.omg.org .