Benefits of knowledge-based interactive learning environments: A case in combinatorics

Hélène Giroire, Françoise Le Calvez, Gérard Tisseau
Université Pierre et Marie Curie, LIP6, Paris, France
Helene.Giroire@lip6.fr, Francoise.Le-Calvez@lip6.fr, Gerard.Tisseau@lip6.fr

Abstract

Solving combinatorics exercises is a difficult task for students because the modeling activity involved is not usual. To help students to train themselves, we have built an AI based computer assisted learning system that gives them a specific framework to solve exercises and teaches them a general problem solving method. In this article we present the benefits of using such a system in a learning process, in relation to standard teaching (as in the classroom).

1. Introduction

Solving combinatorics exercises is a difficult task for students. To help the students to train themselves, we built a knowledge-based interactive training system that relies on a method that we called the "constructive method" [7]. Using this method, students are able to model the problems and their solutions and become conscious of the underlying mathematical elements concealed behind their everyday language.

Designing such a system has forced us to define and to name mathematical concepts in domains which are generally not directly taught in the classroom. We think that making these concepts visible is a significant contribution to making learning more efficient.

This paper is organized as follows. In section 2, we show why combinatorics is a difficult domain. Modeling is the heart of combinatorics problem solving, which makes this domain interesting for scientific training. In section 3, we briefly present the software we have designed and developed to help students overcome these difficulties. In section 4, we present the specific benefits that students can draw from our system. In section 5, we explain how our system records the tracks of the students' sessions and the benefits of this recording for students as well as for teachers. In section 6, we present the first results

2. Combinatorics: a difficult domain

In the domain of mathematics, proposed problems are often solvable by a deductive method: starting from the given facts, considered as a set of axioms, the learner can apply inference rules to obtain new facts and thus find a way to the property to be proved or to the result of the requested calculation. In order to evaluate a learner's activity, a system can observe which rules are tried, whether they are correctly applied, whether the learner reaches the goal effectively and if the way to the goal is a good one. In the course of the search the system can guide the learner at each step, for example by giving hints about the relevance of (the) new obtained facts, by detecting dead ends, by trying to guess the followed plan and by proposing sub-goals.

While this kind of process is promoted in such domains as the transformation of algebraic expressions [6] and proofs of geometric properties [4], it is not suited to all domains. In combinatorics (counting the number of ways of arranging given objects in a prescribed way), for example, this unsuitability is apparent in the students' reactions to the problems: "I can't represent the problem", "I don't know where to begin from", "I understand the solution given by the teacher but I don't understand why mine is wrong", "I'm proposing a solution but I have no way of finding out if it is right or wrong". We interpret these reactions as representational difficulties: the main part of the solving process does not come from a clever chaining of inferences or calculations, but from the elaboration of a suitable representation and from the transformation of one representation into another, equivalent one. The importance of representations in problem solving is studied for example in [2], [5], [8].

We are particularly interested in these domains because they offer a good opportunity to acquire some important abilities, such as the ability to elaborate models.

3. A knowledge – based interactive training system for combinatorics exercises

From our experience of teaching combinatorics in the classroom, we have defined the mathematical bases of a solving method that we called “the constructive method”. It is adapted to the usual students' conceptions and gives access to the mathematical theory of the domain. This method consists in making students build a generic element (called *configuration*) of the Solution set. This element is built as a union of elements, each one verifying a constraint. Then, the students reason about this construction to obtain the numerical solution [7]. The combinatorics problems considered (from undergraduate university mathematics courses in France), were classified according to the type of their *configuration* solutions. To each type of configuration we associate an interface which enables students to describe their construction as a set of constraints. Figure 1 shows the ongoing solution of exercise 1. This solution is composed of a universe and a construction in two stages. The screenshot is taken during the definition of the second stage.

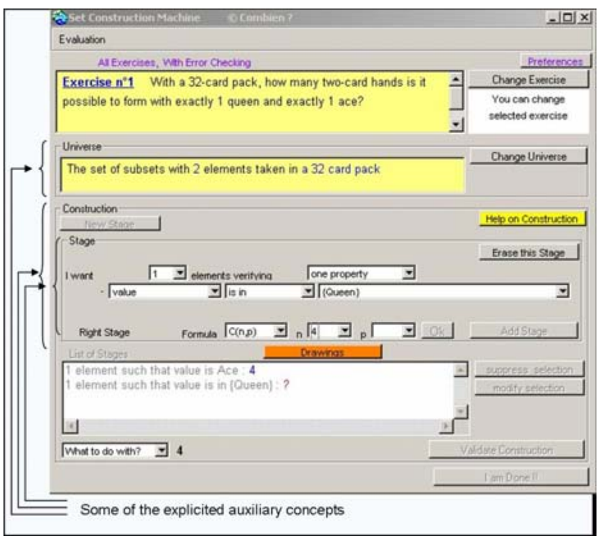

Figure 1. An interface during the construction of a solution

These interfaces are AI systems which are built in such a way that the students' errors can only be "relevant" errors, i.e. errors with a mathematical meaning. These errors are detected at each step of the construction. Then, they are pointed out to the students by giving them hints. These hints do not correct the errors but help the students to discover by themselves why the ongoing construction is incorrect. The aim of this help formulation is to train students to overcome their difficulties. As printed out in different studies overcoming one's errors allows one to learn. [3]

The AI techniques implemented in the system enables it to guide the students in their construction and to detect the errors incrementally. Thus it induces students to be able to justify their solution, to become conscious of the underlying mathematical elements, and more generally to acquire some abilities to elaborate models.

4. The benefits of using technology

Numerous articles have shown that computer assisted learning makes it easier for many students to overcome their learning difficulties in different areas such as, for example, working at their own pace, working in conditions others than those of the classroom… In this section we develop other types of benefits brought about by technology-facilitated learning.

4.1. Explicit concepts

To design a learning system, the taught domain must be formalized. In mathematics, the domain is generally formalized but this formalization does not supply the ways of expressing and representing it in an interactive educational system. The enlargements needed are mainly at a meta level. A problem, a solution, an error, or a step of reasoning must be explicitly represented. The general theory is therefore specialized by introducing auxiliary concepts specific to the domain. The institutional theory generally does not refer to these concepts. It is minimal and general and presents the bases that are logically sufficient. Furthermore, this theory is built to teach the right reasoning but it has no general criteria to analyze wrong reasoning, which requires a deeper knowledge of the domain.

In combinatorics, we had to introduce concepts to represent the problems and their solutions (constructions, steps, filter, property, multiplicity), and some knowledge associated with these concepts (constraints associated to a correct construction, comparison of two constructions, constructive method…).

The use of the system has shown that introducing explicit concepts enables to structure the solving process. Explicitly using a specific vocabulary ("construction", "step", "property"…) allows students to become conscious of a framework in which they can make the most of the general theoretical knowledge.

4.2. A learning framework

Usually, students have no general method to solve combinatorics problems. One aim of our software is to give them a systematic way to solve them. To do so, our software forces them to give a structured solution by using the constructive method. This method is not totally new for students. It only formalizes an intuitive approach as can be seen from the solutions they give (sometimes, wrong ones) before using the system.

At the beginning, students feel uncomfortable with this constraining framework. First, it forces them to face several obstacles without bypassing possibilities. Second, sometimes it limits their means of expression. Third, they feel it prevents them from reaching their main objective directly: namely giving the numerical solution.

But, quickly (after solving a few exercises), the constraining framework is felt as helpful, safeguarding and guiding.

Answers to a questionnaire pointed out the interest of such a framework: "The compulsory process helps us to solve, it's a clear framework", "we learn to well split exercise solving in steps", "the system forces me to clearly explicit my construction and to break it down into several simple steps".

4.3. Detection of errors on the spot

Generally in a learning process, the errors made by students are of importance because they constitute particularly favorable times for learning [3]. When students solve exercises in classrooms, they have to wait for model answers. When the solution is given to them, they have forgotten the context of research so that the best time to benefit from the correction has passed. Moreover, several isomorphic solutions exist and the model answer may not correspond to the solution planned by the student.

In our constructive method, we have defined how to build some right *constructions* for an exercise and at the same time, we have defined the theoretical elements to detect a wrong construction. The construction is elaborated step by step. Each step corresponds to a concept (auxiliary concepts of the domain see §4.1.). For each step of the solution, the system, based on this theory, can detect that an ongoing solution will never lead to a right solution. Thus, the system points out the error very early in the building of the solution. The system can signal the error immediately or record it to delay the message. We chose to signal it immediately as this strongly increases the ability for students to understand their mistakes because those are linked with the last concepts used.

4.4. Appropriate teaching times

When a student adds a new step to his/her solution, the system detects the possible errors according to the theoretical elements defined to detect a wrong construction. The system displays an error message for each misused concept. These messages are especially relevant because they are given in the context of the specific ongoing solution of this student; and when the solution is being build . This message is composed of hints about the misused concept, to help the student to find a correct answer and continue the process, see Figure 2.

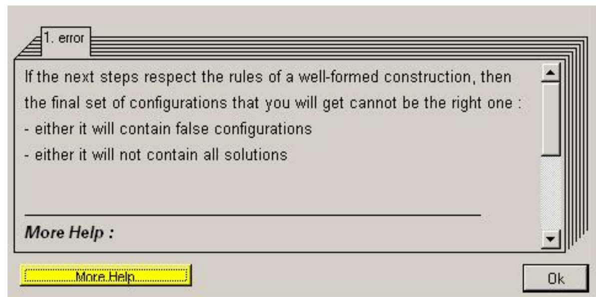

Figure 2: An error message

The pedagogical approach chosen for our system is that students have a stronger motivation to understand and learn when they have made mistakes than when they have to listen beforehand to a course. This is the reason why the course is available in different pieces in the system. Some pieces are displayed to students when they have made an error (using the “More Help” button as shown in Figures 2 and 3). Other pieces are displayed if students use the “help” buttons. All these pieces of courses are linked to the context of the ongoing solution. Thus, the course becomes an answer to questions that students really ask themselves

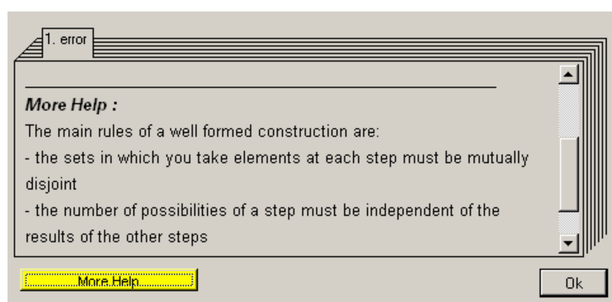

Figure 3. A piece of course in the error context of figure 2

These pieces of course and especially the definitions of the auxiliary concepts as universe, step, referential… and instructions on how to use the system are also available in the user's manual.

Visualizing an instance of part of the ongoing solution helps students to understand why the solution becomes wrong. Using the "drawings" button, students have the opportunity to see a randomly created instance of the ongoing configuration. This button is useful all the time, but if the student uses it when an error is detected, the system displays an example of a wrong configuration to confront him/her to his/her error. For example, if the generic configuration to be built is a card hand, and the ongoing solution is wrong the system displays a configuration with a card that does not belong to any final configuration.

4.5. Usage analysis

Students learn by doing but also by analyzing their works a posteriori. One advantage of computer assisted learning is the possibility of automatically recording and analyzing students' works. Our system records all the actions of the students during the solving process and some information about their possible errors. Some functionalities of our system enable students and teachers to reconsider the sessions:

- Recorded files can be replayed so that students or teachers can examine the different actions made during the building of the solutions. This functionality is also used to enable a student to visualize the building of a solution from a pre-recorded right solution;
- Various analyses of these recordings are made a posteriori, to present to students a summary of their works, to suggest a course revision, to make statistics for teachers... This is developed in the following section.

5. Analyzing the tracks

A track recording contains the image of a student's session (i.e. all the solving process of the exercises completed or attempted during a session). It contains all the student's actions and answers, possible types of errors, indications of time, and information about the reactions of the system.

5.1. Track recording structure

Through the interface students edit their solution as a tree-like structure. An object of the conceptual model of the combinatorics domain that we have defined values each node of the tree. Each input (answer) by the learner, its corresponding moment (in seconds), and the reactions of the system are structured according to the model of the solution and recorded. The file contains this information using the Descript language [10][1] (see figure 4), which is the language used to express all the resources of the system. We have built a translator from Descript to XML and track analysis is performed from the XML files.

In figure 4, we can see concepts such as universe, construction, property, the corresponding moments ([time]), the reactions of the system as choose ("choisir"), verify stage, and a type of error detected ("erreur").

```
Machine_CE ,
  ( filsChoixExercice :
     ( choisir :
        [ time : 10 ] , 'Ex n°6 : 12  cards out of
                32 with 7 Hearts, 4 Aces, 2 Spades'
      valider :
        [ time : 11 ] , 'Validate' )
    filsSolution :
     ( filsUnivers :
●●●
         , 'Ok' )
      filsConstruction :
         ( creerInstance :
            [ time : 24 ] , 'New Stage'
          filsEtape :
             ( saisirCardinal :
                [ time : 35 ] , '7'
              fixerNbProprietes :
                [ time : 38 ] , 'one property'
              filsDeclaProp1 :
●●●
                 [ 'Verify Stage'
                  message :
                     ( erreur : 'IncorrectlyIntersectedFilter' ) ]
```

Figure 4. Part of a recorded track

5.2. Aims of data analysis

Both teachers and learners are involved in the learning process: therefore we analyzed tracks from the two points of view.

5.2.1. For teachers

Using the analyzing system, teachers choose to look at information about one student or a set of students.

For one student the notion of session enables the teacher to look at his/her progression. The time spent

[1] Student can choose the language (French, English…) for the interface but since the designers are French native speakers, the technical language used in the descript language is French.

for solving an exercise is calculated in connection with the rank of the exercise in the list of exercises. The errors made in each exercise are collected, and their continuity (i.e. when the same errors are made in several exercises following one another) can be examined.

For sets of students, two kinds of reports are useful to the teacher. One concerns a categorization of exercises according to their level of difficulty; the other concerns the categorization of students according to the type of errors they made, and their ability to complete the exercises.

From these two categorizations, teachers have elements to adapt their teaching to students. They can build personalized learning scenarios consisting of pieces of course to revise and exercises to solve.

5.2.2. For students

The analyzing system presents each student with a report of his/her session. For each exercise, it reports the total solving duration, the numbers of errors made, the types of errors, the number of hesitations, whether the exercise was completed or not, and the number of times the pieces of course have been consulted. Information about the whole session is added such as the duration of the session, the number of exercises made, and comments according to the students' categorization. It is also possible to indicate to students their groups, and thus the next corresponding learning scenarios.

6. Some results

In In 2002 and 2003, our system was experimented in various contexts. In 2004 and 2005 it was actually used in Paris 5 university, for 2nd year students studying for a science degree. The corresponding tracks were recorded and are being analyzed.

In class tests, most of the students structured their solutions as they did when using the system and used the terms introduced in the system such as universe, constraint, and property. (This seems to show that they have acquired the constructive method and that they are able to justify their numerical solution.

From the analysis of the tracks, we made a first classification of the sets of exercises and we searched for students who had difficulties on a precise point. For these, we assigned remedial tasks consisting in some pieces of course to revise and some new exercises to solve using the system.

7. Conclusion

Solving combinatorics problems is a difficult process. It requires several steps of non-intuitive modeling and reasoning.

Our system provides a learning framework which ensures that the user actually learns. It makes the steps of modeling and reasoning concrete and forces the student to name the underlying concepts used. It provides explicit resources as error messages and pieces of course and it offers visualization means (dynamically- and contextually-created examples and counter-examples) at the adequate learning time. From the recorded tracks, classifications (of students and of exercises) were made to enhance the learning process.

All these points highlight the benefits that the use of such a system adds to standard teaching.

8. References

[1] Astolfi J.-P., *L'erreur, un outil pour enseigner,* ESF, Paris, 1997.

[2] Abidin B., Hartley J.R., *Developing mathematical problems solving skills*, Journal of Computer Learning (1998) 14, pp. 278-291.

[3] R. Borasi, *Reconceiving Mathematics instruction: a focus on errors*, Norwood, NJ: Ablex Publishing, 1996, 321p.

[4] Chou S.-C., Gao X.-S., *Ritt-Wu's Decomposition Algorithm and Geometry Theorem Proving,* Proceedings of the 10th international Conference of automata deduction, Springer verlag, London, pp 207-220

[5] Cox R., Brna P., *Supporting the use of external representations in problem solving: the need for flexible Learning Environments*. Journal of Artificial Intelligence in Education, 6, 2/3, 1995, pp..239-302.

[6] Nicaud J.-F., *Building ITSs to be used: Lessons Learned from the APLUSIX Project*, Lessons From Learning, Lewis, R., Mendelsohn, P., (eds), IFIP, North Holland, 1994, pp. 181-198

[7] Tisseau G., Giroire H., Le Calvez F., Urtasun M., Duma J., *Design principles for a system to teach problem solving by modelling.* Lecture Notes in Computer Science N° 1839, ITS'2000, Springer-Verlag, Montréal, 2000, pp. 393-402.

[8] White B.Y., Frederiksen J.R., *Causal model progressions for intelligent learning environments.* Artificial Intelligence, 42, pp..99-157.

[9] http://www.lip6.fr/combien/

[10] http://www.lip6.fr/combien/descript.pdf