INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 05 MELBOURNE, AUGUST 15 – 18, 2005

A FORMAL MODEL FOR THE REPRESENTATION OF PROBLEMS BASED ON TRIZ

Dubois Sébastien, Lutz Philippe, Rousselot François, Caillaud Emmanuel

Keywords: Inventive design, UML, ontology building

1 Introduction

Designing, and especially in inventive design, is mainly understanding and solving problems. Simon [1] describes the designer activity as a problem forming, finding and solving activity. Nevertheless, if the importance of the problem solving process is admitted, a lot of methods exist to guide the inventive process without integrating this fact. TRIZ [2] proposes to answer this lack by focusing the design on a problem framing and solving process. On a generic point of view, TRIZ can be described as a rule-based problem solving method. In fact it is a whole of methods and tools which aim is to identify problems and formulate them through generic frames enabling the use of patterns of solutions.

One of the disadvantages of TRIZ is its lack of formalization which implies a lot of difficulties to implement it and even to understand it. To fulfill this lack a formalization, a process is described in this article and a resulted partial model is presented. This model is partial as it is only focused on the TRIZ problem formulation frames.

The implementation of the model is presented to show the interest of the model and to validate its efficiency. This implementation is argued by the description of a problem formulated by the use of a prototype of software.

2 TRIZ as a rule-based model transformation system

TRIZ, Russian acronym of Theory for Inventive Problem Solving, is a theory made of several tools and methods to guide the inventive design process by a problem-focused approach. Several methods exist to formulate and solve the problems and these methods have interactions that make the theory coherent. But the models and the methods are only semi-formal ones, disabling a good implementation.

2.1 A structured approach

The TRIZ approach is based on the building of a convergent process. As was said before, the aim of TRIZ is to focus the design process on a problem framing and problem solving process. At least three goals co-exist, understanding the foundations why a new technical system is required, understanding the difficulties of this new system realization, and, at last, overcoming these difficulties by the conceptualization of a solution. Simon, in [1], call these three goals problem forming, problem finding and problem solving. The satisfaction of this three goals process is an iterative process, as problem framing, understanding the difficulties of the new system realization, and problem solving, the conceptualization of a solution, are not one-step processes. These processes are iterative ones, the proposal of a potential concept of solution leading, most of the time, at a better understanding of the problem to be solved, thus at the taking into account of a new parameter of the problematic situation.

To represent this three-goal process and to make it efficient, TRIZ proposes to build a convergent process, convergent in the meaning of "going towards the solution". This convergent process is represented on the figure 1. The first goal of the process is the understanding of the reason why a new technical system is required. This goal is assumed, partially, by the formulation of the Ideal Final Result. This Ideal Final result is an utopian concept solution based, on the one hand on the maximization of the benefits of a system, the efficiency of the satisfied functions, and on the other hand on the minimization of the drawbacks of the system, no harmful function, no resource consumption, and so on. The aim of this ideal solution is double. First it enables the guiding of the convergent process into a direction that assume to be beneficial. Secondary, the Ideal Final Result provides a criteria to judge the potential concept solution in regard of this ideality, and thus to validate or to sort them.

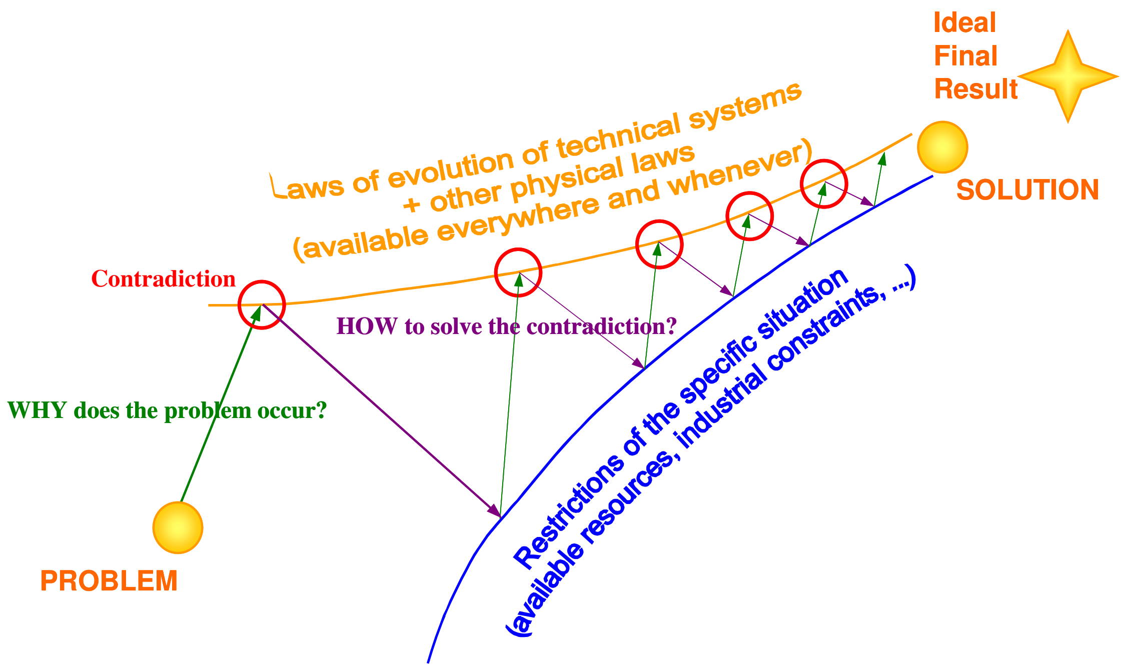

Figure 1. The TRIZ convergent process

The second goal of the process is the understanding of the difficulties linked with the new system realization. In TRIZ these difficulties are formulated through a particular shape: the contradiction model. The interest of this model is that it is the result of an abstraction phase only focusing on the elements of the problematic situation directly implied in the problem. The abstraction phase aim at identifying the root cause of the problem. This root cause is an objective law, in TRIZ semantics. An objective law is a particular rule that is available everywhere and whenever. The existence of this rule does not depend of the person solving the problem nor of the period when the problem is to be solved. Two major objective laws categories exist: the laws of technical systems evolution and the laws of physics. One of the main axiom of TRIZ is to consider that each system evolve in regard of a set of laws. It means that whatever the considered technical system is, its evolution is predictable as the evolution will respect at least one of the laws. The second category of laws is the whole of laws of physics, as for example the law of gravity. Of course the law of gravity, as all the laws of physics, is not really available everywhere, but it has a domain of validity large enough to consider it as an objective law in a considered domain.

The third goal of the process is overcoming the difficulties by the conceptualization of a solution. In TRIZ, this goal is assumed by a two-step phase. The first phase is the application of problem solving tools. We talked previously of the contradiction model as the model to represent the problems in TRIZ. In fact it exist three kind of models, each of this model corresponding to different levels of abstraction, from very technical level to very generic level. For each model, TRIZ proposes a specific tool to transform the model of problem into a model of solution. This model of solution indicates the mechanism to solve the problem, in a conceptual way. The second step of the resolution phase is then the implementation of this conceptual mechanism of resolution. The implementation of the conceptual solution aims at identifying the resource that could be used. The identification of this resource has to be done in regard of the availability of the resource and of industrial constrains.

2.2 But only semi-formal models

The convergent process of TRIZ is a complete process to guide the problem resolution in design. It is based on a set of axioms. Without the respect of these axioms, the process is not coherent. It is really important, in order to use TRIZ as a theory enabling a systematic problem resolution in design to clearly understand and use its different models and tools into a coherent whole based on the systemic relations of these different components. On a first approximation, it seems that TRIZ propose formal models and a formal methodology. But a deeper analysis shows that these formal elements are not complete as they do not enable the clarification of TRIZ as a system. Two examples to illustrate the lacks. First example, the links between the different models of problem are not explicit. Second example, the formal methodology to implement the convergent process, ARIZ [3] (Russian acronym for Algorithm of Inventive Problem Solving), is dedicated for people having at least eighty hours of formation. This requirement is made to assume people using ARIZ are able to build, by themselves, the links between the different formal steps and to understand the underlying axioms.

Based on this remarks, we consider that TRIZ proposes semi-formal models as some bricks of the theory are formalized but not enough to understand the theory as a coherent whole and not enough to make it usable for non experts. Some deviance are already observed due to theses links. Lot of people use these semi-formal elements during their creativity phases to increase their efficiency. Even if the use of TRIZ solving tools could be helpful to increase creativity, doing this without considering the whole theory and without including TRIZ axioms, does not provide the building of a convergent process and so is not having a problem solving activity, in the strict sense of TRIZ.

The proposal to fill these lacks is to formalize all the elements and axioms of TRIZ. This analysis is really important for two reasons. First it will increase the transferability of the theory, facilitating its understanding and so its acquisition. Second, building of formal models and methods will facilitate the computerization of these models and methods. The aim of this paper is to present a first brick in the formalization process, by the description of the formal models to represent the problems, in accordance with the TRIZ frames.

3 The formalization process

The formalization process completed to achieve a formal model of the problem formulation frames of TRIZ is based on three steps: semantic analysis of the corpus, object-oriented representation of the frames and validation of the representation by an analysis in description logics.

3.1 The corpus analysis

The main difficulty linked with TRIZ study is that the amount of reference texts is relatively poor. So the first task is the definition of the corpus of knowledge that will be analyzed to identify and represent the concepts. The objective of the study is to formalize the TRIZ problem formulation frames. It is thus required to understand which kind of information is included into a complete problem formulation. As the advantage of the TRIZ frames is to enable the use of resolution tools, and as the facilitation of this use is an objective of this study, the information required to implement the rules have also to be included into the analyzed corpus.

The main component of the corpus, in which the problem formulation concepts have been identified, is ARIZ, the algorithm for problem solving. The analysis of this algorithm enables the identification of numerous concepts and of links between these concepts, but not the clarification of their meaning. The clear definition of these concepts is acquired through the analysis of the TRIZ literature or through the observation of TRIZ experts during their problem resolution activities.

A second component of the corpus is the text of inventive standard solutions [4], a particular TRIZ resolution tool. This text is a whole of seventy-six heuristic rules of problem transformation and it is a rich, and the most formal, representation of the problematic situations solvable with TRIZ. The analysis of the discriminating parameter of each rule enables the identification of all the distinguishing parameters of the problematic situations, and then the construction of a complete problem formulation model, directly orienting to the adequate model transformation rule.

The analysis of the texts have been supported by the use of a linguistic engineering station [5] aimed at building interpretable data from multi-texts corpus. These data are a list of word sequences candidate to a particular interpretation, it could be terms, relations or a whole of relations to be interpreted as a generic class. This tool enables the identification of all the notions available in the corpus and the validation of the ones that have to be included into the problem formulation model.

3.2 Object-oriented representation of the frames

The built model [6] is the object-oriented representation of the TRIZ problem formulation frames. It does not include all the concepts of the theory, especially the laws of technical systems evolution are not available. As a first step of the process of TRIZ knowledge formalization, the objective is to focus on TRIZ problem formulation frames, as they are one of the main components of the static components of TRIZ knowledge, the laws of system evolution are more a dynamic aspect of the knowledge.

The model is represented on figure 2, in UML, and is described below. The next description of the model is the description of the TRIZ frames to represent the real elements and the problems.

A resource is the basic element representing a real object characterized by its localization and described by a whole of parameters. A resource could be localized inside the system implied in the study, in an adjacent system or, more globally, in the environment. A resource could be a field, a geographical zone (a space), a temporal period (a time) or a material resource. The nature of the field could be magnetic, mechanic, electric, chemical or thermic. A material resource could be a system or a substance in regard of its decomposability. A system is composed of, at least, four material resources, enabling its functionality. This four main elements assume the roles of motor, transmission, tool and control to provide the function.

A function is the modification of the value of a parameter of a resource. The realization of the function modifies the parameter from an initial value into a final value. The function operates during an operational time and within an operational zone. The importance of the function could be principal, if it is the function for which the system has been designed, or technical, if it is a sub-function enabling the realization of the principal one. The type of the function could be useful or harmful. It is harmful if it is a non desired function to be eliminated, resulting of the realization of other useful functions. A function could participate to the realization of a super-function and could be decomposed into sub-functions.

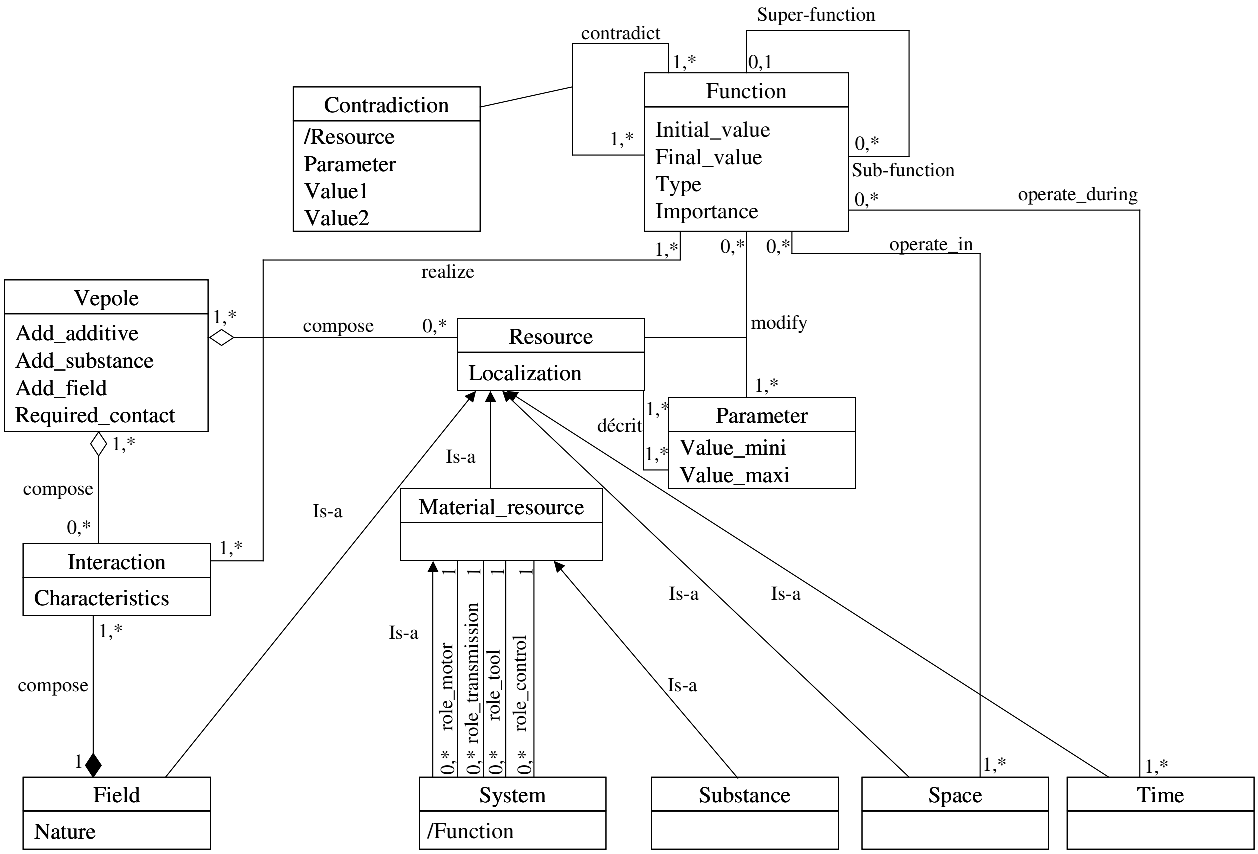

Figure 2. UML model of TRIZ frames

A contradiction arises if one or several functions require that the parameter of a resource has one value, when an other, or several others, functions require this parameter to have an other value.

An interaction is the representation of the action of a material resource on another material resource. This action is produced by a field. The interaction could be satisfying, excessive, insufficient or harmful. It is harmful if it is a non desired interaction, which is to eliminate (for example the Joule effect of a component inside an electric circuit). If the interaction is useful, it could be excessive, if it is realized more than necessary and then occurs disturbances (the sun radiations on the skin in summer, for example). A useful interaction could also be insufficient if its action has to be increased (the oxygen inflow in high altitude, for example). At last, if the interaction is to be kept as is, it is a satisfying one.

A Vepole model is made of interactions and of resources. This model is characterized by the additivity constrains and by the contact necessity. The possibilities to add a field, an additive or a substance are thus defined, as the possibility to break the contact between two substances of the Vepole model.

3.3 Validation in description logics

Once the model is built, its validation has to be achieved. The aim of this validation is to assume that the model is coherent both on ontological level and on hierarchical level.

- The coherence of the model on ontological level could be assumed by the creation and test of non valid instances or by the belonging research. For example, in an ontology describing a family, an instance defining a human which is both a sister and a father is a non valid instance. In the same manner, the result of the belonging research of a feminine individual has only to deliver the classes Women, Mother or Sister. These validations enable the test of the completeness of the definition of the concepts, to check that each concept has enough discriminating parameters to describe it without any ambiguity.

- The definition of the whole of concepts generates a hierarchy of concepts and then enables the building of a graph to represent this hierarchy. The aim of the graph is to have a vision of the descendants of the concepts. This representation is a way to validate the coherence of the model.

To perform the validation, the built model has been defined and tested by the resort to description logics. The main interest of description logics, to validate an ontology, is the calculation of the hierarchical links by subsumption: the concept A subsumes the concept B if the concept A is more generic than the concept B, i.e. if the whole of individuals represented by the concept A includes the whole of individuals represented by the concept B. The used tool [7] to validate our model enables an automatic construction of the hierarchy graph and a fast detection of incoherencies through the use of commands as:

- (concept-consistent? conceptA conceptB) to check if two concepts are consistent

- (instance individual concept) to check if an individual is a member of a concept

- (most-specific-concepts individual) to list the less generic concepts to which an individual belongs

The formalization and the validation of the model has been performed by to use of tools from Artificial Intelligence. Once validated it is possible to implement it in order to make it operational.

4 Implementation of the model

The objective of the implementation of the model is to propose an informatics tool to guide the instantiation of the model. As the model represents the TRIZ problem formulation frames, the aim of the tool is to facilitate the problem formulation, in regards of the TRIZ frames, mostly to facilitate the formulation of the contradiction inherent to a problem and the formulation of the inherent Vepole model.

The implementation of the model is a two-step process, the first step is the definition of the strategy of instantiation and the second step is the construction of the interface.

4.1 Strategy of instantiation

The instantiation of the model is assumed by a whole of questions. The role of each question is to enable the user to understand the way the problem will be formulated and to make him give an information. Each piece of information is in fact the value of a parameter of a concept.

But it is evident that the well understanding of the model, and the good instantiation could only be assumed if the questions are organized. This organization is the order in which the questions will be asked, that means, the order in which the model will be instantiated. This organization is called the strategy of instantiation.

The strategy of instantiation is defined in accordance with one of the main concepts of TRIZ, the concept of ideality. This concept precises that each evolution of a system has to increase its level of ideality. A criteria to evaluate the level of ideality of a system is given in equation 1.

$$I = \frac{\sum Useful\_Functions}{\sum Harmful\_Functions + \sum Costs} \quad (1)$$

This criteria indicates that the evolution of a system could be reached through the increasing of the functionalities of the system, through the decreasing of the annoyances of the system, or through the decreasing of the expenses linked to the system (money or energy costs). Based on this criteria, it is possible to propose four kind of problems linked to the system development:

- the creation of a new system
- the addition of a new functionality to an existing system
- the improvement of the efficiency of an existing functionality
- the suppression of an harmful effect.

The strategy of instantiation is then directly oriented by the nature of the problem to be solved. For example, if the problem to be solved is due to the improvement of the efficiency of an existing functionality, it means that the function to be improved is already known. The strategy of instantiation can then begin by collecting: the function to improve, the product on which the function acts, the system performing the function, etc… However, the creation of a new system requires to first identify the product on which the future system will have to act, the parameter of the product that will be modified by the system, so to define precisely the function and the nature of the tool that will act on the product.

4.2 System prototype

The prototype of software to implement the model has been developed in JAVA. It is the construction of JAVA windows to enable the display of the questions, and the capture of the answers. Several kind of windows exist, depending of the nature of the question.

- Some questions are open-ended, for example the capture of the name of the study is totally free.

- Some questions are closed-ended, for example the choice of the kind of problem is limited to those described in section 4.1.

- Some questions are open-ended but a list of possible answers is proposed, for example the choice of the system to be studied is assisted by the proposal of all the previously defined systems, but a new one could be included in the base.

- Some steps in the strategy of instantiations are only validation steps, to confirm the previously captured information by the coherence of the global concepts formulation. For example, some steps provide the formulation of the problem through the contradiction frame to assume all the information have been captured in an adequate form.

An example of JAVA window is given in the figure 3.

Figure 3. A JAVA window with closed-ended list

Each answer to a question enables the instantiation of a parameter of an instance of the previously presented model. This standard way to proceed is not sufficient to assume the coherence of the instantiation, the implementation of the model is done in an expert system shell, JESS [8]. The use of this shell enables the definition of rules, and their automatic execution. For example, when a new function is defined a specific rule launches the questioning of the product on which the function is performed. These rules are additional questions that have the priority on the standard questions from the strategy of instantiation.

The use of JESS enables the definition of a second kind of rule which automates the instantiation of some specific parameters of instances. For example, the Vepole model is characterised by its composition, if the Vepole model is made of at least two substances and a field, it is defined as complete. This specification of the completeness of the Vepole model is automatically assumed by specific rules in the shell.

5 Example

In this section an example of problem formulation, resulting of the use of the prototype, is presented. It is a result extracted from experimentations conducted with eight students having a short introduction to TRIZ. The aim of these experimentations was to enable the students to work on a free subject. After four hours of analysis of their problem with the classical methods of TRIZ, they used the prototype to measure the benefits in terms of understanding of their problem.

One of these experiences is presented below. The underlying rules of the system, to activate the questions and to automate the instantiation, are not presented, but only the questions they generate.

It handles on the building of a contradiction model about a problem concerning tongue piercing, where jewels fastened on the piercing on the tongue are often lost and consequently ingested.

Question: Give a name to the project. {STRING: Study}

Answer: Tongue piercing

Q: What is the objective of the study? {Improvement of an existing system}

A: Improvement of an existing system

Q: What is the name of the system to improve? {STRING: System}

A: Fixing system of the piercing

Q: What is the main function realized by “fixing system of the piercing”? {STRING: Function}

A: Maintain the piercing

Q: How do you wish to improve the system “fixing system of the piercing”? {Improvement of a useful function; Addition of a new useful function; Elimination of a harmful effect; Reduction of costs}

A: Elimination of a harmful effect

Q: What is the harmful effect, the disturbing function, to suppress? {STRING: Function}

A: Impromptu unscrewing

Q: The realization of a function modifies an object. Which object is modified by the function “Impromptu unscrewing”? {STRING: Ressource}

A: The jewel

Q: “The jewel" is : {Substance; System}

A: Substance

Q: “Impromptu unscrewing” is the modification of a specific parameter of “the jewel”. Which parameter is it? {STRING: Parameter}

A: The fixing

Q: What is the value of “The fixing” before the realization of “Impromptu unscrewing”? {STRING: Value}

A: Good

Q: What is the value of “The fixing” after the realization of “Impromptu unscrewing”? {STRING: Value}

A: None

Q: Why does “The fixing” “None” appear? {This is the RESULT of the realization of a useful function; It PARTICIPATES to the realization of a useful function}

A: This is the RESULT of the realization of a useful function

Q: Which useful function is not perform if “The fixing” is not “None”? {STRING: Function}

A: The jewel interchangeability

Q: The following contradiction has to be solved: “The fixing” has to be “None” to assume the realization of “The jewel interchangeability”, but “The fixing” has not to be “None” to avoid “Impromptu unscrewing”.

Q: The system responsible of “Impromptu unscrewing” will now be analyzed. Give a name to this system. {STRING: System}

A: Fixing system

Q: During the realization of “Impromptu unscrewing”, an element of “Fixing system” is directly in contact with “The jewel”, directly acts on “The jewel”. Which element is it?

A: The thread

Q: Which parameter of “The thread” could be modified to avoid “Impromptu unscrewing”?

A: Unscrewability

Q: Which value of “Unscrewability” of “The thread” avoids the realization of “Impromptu unscrewing”?

A: Null

Q: If "Unscrewability" of “The thread” is “null”, is the problem solved?

A: No

Q: What is the actual value of “Unscrewability” of “The thread”?

A: Existing

Q: If it cannot be changed then “Unscrewability” of “The thread” being “Existing” enables the realization of a useful action. Which action is it?

A: Jewel interchangeability

Q: The following contradiction has to be solved: “Unscrewability” of “The thread” has to be “Existing” to perform “Jewel interchangeability”, but “Unscrewability” of “The thread” has to be “Null” to avoid “Impromptu unscrewing”.

Based on this formulation, several concepts of solution have been found, by analogy to the closing systems of toxic liquids (bleach cap) or to bulbs with bayonet base.

The global experimentations have been analysed through the following criteria:

- The identification of the core of problem
- The formulation of a contradiction
- The required time
- The number of iterations required to achieved a satisfying formulation
- The interest of the formulated problem
- The identification of concepts of solution.

The results of the experimentations for the eight students is given in table 1.

Table 1. Results of the experimentations

	Built Model		Modelling approach			
	Core	**Contradiction**	**Time**	**Iterations**	**Interest**	**Solutions**
Student 1	No	Yes	40'	1	No	Yes
Student 2	Yes	Yes	45'	1	Yes	Yes
Student 3	No	Yes	1 h	2	Yes	No
Student 4	Yes	Yes	55'	2	Yes	Yes
Student 5	Yes	Yes	1h	1	Yes	Yes
Student 6	No	Yes	50'	2	Yes	No
Student 7	Yes	Yes	40'	1	Yes	Yes
Student 8	No	No	45'	2	Yes	No
Total	**50%**	**88%**	**49'**	**1,5**	**88%**	**63%**

The analysis of the experimentations shows satisfying results:

- the use of the prototype is not a long-time process, the average time of 49 minutes has to be compared with the formulation of the contradiction with the TRIZ methods which required two days of analysis;

- the interest is mainly recognized by all the students;

- more than half of the students have formalized concepts of solutions after the use of the prototype.

A remark can be made for the identification of the core of problem. For the students who do not succeed in identifying the core of their problem, it comes from a lack of knowledge about the given problem. The use of the prototype, then, does not lead to the identification of the problem to be solved, but it enables the identification of the knowledge that have to be collected.

These experimentation have to be pursued to be more consistent, but at least, it enables to validate the interest of the formalization of TRIZ problem formulation frames.

6 Conclusion

In this paper was presented a general approach to formalize some elements of a global theory of inventive problem solving, TRIZ. The main benefits of this approach is to enable the understanding of the concepts inherent to the theory and of their organization. The formal model is limited to the problem formulation frames and now has to extended to all the remaining concepts of the theory. Especially, the laws of system evolution have to be formalized in order to build a global view of the dynamics of the model. For the moment only the static aspects of the concepts have been represented, as the concepts are a part of a theory and the dynamics aspects required to build the links with the non-included concepts.

An other benefit of the model is its implementation. This implementation enables the use of the model and then the validation of its usefulness. Some improvements are now in development for the interface, especially to enable the display of the evolution of the contradiction formulation. The aim of this display is to provide a better understanding of the role each element of the model is playing in the problem formulation process.

But even with an incomplete model, and with a primitive prototype, the obtained results, through the experimentations, are really satisfying and encouraging.

References

[1] Simon, H.A. *Problem Forming, Problem Finding, and Problem Solving*. in *1st International Congress on Planning and Design Theory*. 1987. Boston, USA.

[2] Altshuller, G.S., *Creativity as an Exact Science*. 1988, New York: Gordon and Breach.

[3] Altshuller, G.S., *The Innovation Algorithm: TRIZ, systematic innovation and technical creativity*. Technical Innovation Center, Inc. ed. 1999, Worcester, MA.

[4] Altshuller, G.S., *System of Standard Solutions for Inventive Problems*. 1987, OTSM-TRIZ Technologies Center: Minsk, Belarus.

[5] Rousselot, F., P. Frath, and R. Oueslati. *Extracting concepts and relations from corpora*. in *12th European Conference on Artificial Intelligence, ECAI'96 workshop on Corpus-Oriented Semantic Analysis Workshop*. 1996. Budapest: John Wiley & Sons, Ltd.

[6] Dubois, S., *Contribution à la formulation des problèmes en conception de systèmes techniques. Etude basée sur le TRIZ*, in *Sciences Pour l'Ingénieur*. 2004, Université Louis Pasteur: Strasbourg, France.

[7] de Bertrand de Beuvron, F., et al. *The Description Logic Reasoner CICLOP (Version 2.0)*. in *7th Workshop on Automated Reasoning*. 2000. London, Great Britain.

[8] Friedman-Hill, E., *Jess in Action: Rule-Based Systems in Java*. 2003: Manning Publications Co. 443 pages.

Corresponding author:
Dubois Sebastien
INSA de Strasbourg
LICIA
24 boulevard de la Victoire
67084 Strasbourg Cedex
France
Phone: (00) 33.388.144.700
Fax: (00) 33.388.144.700
e-mail: dubois@mail.insa-strasbourg.fr