INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'09
24 - 27 AUGUST 2009, STANFORD UNIVERSITY, STANFORD, CA, USA

MECHANISMS OF MODEL CHANGE IN OPTIMIZATION AND INVENTIVE PROBLEM SOLVING METHODS

Ivana Rasovska[1], Sébastien Dubois[1] and Roland De Guio[1]
(1) INSA de Strasbourg

ABSTRACT

Two problem solving techniques can be used to solve inventive problems: optimization methods and inventive methods. Looking for a solution could be described as browsing a knowledge space. This knowledge space will be introduced in order to study the way both methods solve the problems. Both solving techniques explore this space through operators that enable shifting from one space to another and also involve model changes. Especially the mechanisms of model change issued from inventive solving methods seem to be really important to solve inventive problems. The inventive solving methods are more generic than optimization methods and cannot be formalised in the same way. Hence a general model of problem solving will be proposed based on notions from C-K theory in order to describe the operators of optimization and inventive methods at the same level. In addition to provide a formal description language for design reasoning, the C-K theory illustrates well the mechanisms to browse the problem space. Finally the process of inventive problem solving will be illustrated through an example.

Keywords: inventive design, TRIZ, optimization solving methods, CSP, C-K model

1 INTRODUCTION

What does an inventive problem exactly mean and what kind of model is associated to? The inventive problem is a design problem that has no solution in the known conditions. Inventive design, which tries to solve the inventive problems, is a specific activity which differs from the traditional design done in the research and development departments [2]. An invention supposes to invent something, i.e. to propose something new, something not known until now. Inventive problems can concern any known field, technical or not, especially some principle, product or fact relevant to this field. Designing a new technical system means making a technical system evolve [1]. Two kinds of situation can lead to this evolution: increasing the system efficiency by optimization of its parameters or re-designing the system when new parameters are introduced during the resolution, i.e. when some working principle is changed. A hypothesis is that two kinds of problem solving techniques can be used to solve inventive problems: optimization methods to optimize the system parameters and inventive ones to change the problem model. The logical succession is that optimization techniques are used firstly because of their good formalization. If no solution is found, the designer will resort to inventive solving techniques which are much more creative but not yet well formalized. In order to use these two solving approaches in a coherent manner a unified representation model should be proposed supporting both solving mechanisms.

The previous analysis of solving techniques from the two categories was proposed in [3], where CSP (Constraint Satisfaction Problem) was presented as an optimization method and TRIZ (inventive methods and models based on dialectics) as an inventive solving method. In fact, TRIZ solving mechanisms propose to define incoherent constraints in form of contradictions to make designers more creative in the problem resolution. This remains the constraints in the CSP approach and underlies the choice of the two solving techniques. The respective question “Is it possible to improve problem solving strategy for the inventive problems by matching the CSP and TRIZ solving principles?” was discussed. In order to answer this question similarities and differences between the two strategies were shown both from the problem representation point of view and from the problem solving point of

view. The comparison revealed similarities of both representation models while the solving principles were recognized as rather complementary.
The goal of this paper is to identify and to compare principles of model change in both solving strategies. Especially the mechanism of model change issued from inventive solving methods seems to be really important to solve inventive problems. Looking for a solution could be described as browsing a knowledge space. This knowledge space will be defined in order to study the way both strategies solve the problem and so browse this space will be studied. Furthermore operators corresponding to model changes and permitting to pass from one space to another will be studied. However neither CSP nor TRIZ methods provide a clear definition of the properties of the knowledge space in which they look for the problem solution. Hence a model of general problem solving will be proposed based on notions from C-K model [2]. The C-K model proposes a formal language to describe design reasoning and illustrates mechanisms to browse the problem space. Finally the proposed problem solving model will be illustrated on the application example.

2 COMPARISON OF OPTIMISATION AND INVENTIVE STRATEGIES

The comparison of CSP and TRIZ strategies done in [3] revealed similarities of both representation models. In general, the concept of parameters and contradictions in inventive approaches can be matched with the concept of variables and constraints in the CSP approach. On the contrary, the solving principles are rather complementary. The CSP solving methods propose only partial resolution, i.e. one looks for the solution of a new relaxed problem as close as possible to the initial one. On the other hand, the CSP offers quite a lot of approved solving algorithms and software tools. The TRIZ methods try to solve the initial problem, and provide operators to change the representation model if necessary, but they do not provide formalised algorithms. Therefore the conclusion is that the mutual enrichment of CSP and TRIZ solving principles is possible by the proposal of a unified representation model common for both approaches. The possible strategy for problem solving will be to search formal and computable CSP models which can use dialectical approaches, or conversely enrich computable CSP models by empirical data issued from dialectical approaches. The unified model should contribute to better understanding and thus better representation of inventive problems that are not solvable by optimisation methods. With such a problem model we should be able to introduce solving principles from inventive approach into the approved optimisation algorithms and use the optimisation software tools as well. So the operators of model change from inventive methods could be implemented into CSP solving algorithms to optimise problem solving.
At the beginning, the synthesis of complementarities and differences between CSP and TRIZ representation models are introduced in order to lay down the basic principles for our unified representation model. The definition of the knowledge spaces used to specify a design problem and to search its solution will be introduced. Then the operators used in each solving process will be presented to illustrate how both methods browse the knowledge spaces to look for a problem solution.

2.1 Synthesis of problem models

The problem representation model of CSP is based on a set of variables that can represent physical parameters of the system and on the domains of variables defining possible values of the variables. Furthermore the CSP representation model introduces a set of constraints restricting the values that variables can take simultaneously. The constraints describe relations between the variables of the system; i.e. these relations can illustrate conditions in which the system can operate, given objectives of system functions or relations between physical parameters. A solution in CSP is an assignment of a value from its domain to every variable so that all the constraints are satisfied all together. In the case of inventive problems where no solution is found and which are called over-constrained problems in CSP, solving methods try to minimize the number of not satisfied constraints. The research space of solving methods in CSP is characterized by a set of assignments of all problem variables without verification of constraints satisfaction. The solution space of CSP is then a set of assignments of all variables which satisfy all constraints or in the case of over-constrained problems which satisfy a maximum of constraints (one speaks of constraints relaxing).
In TRIZ representation model two kinds of parameters are defined (action parameters and evaluation ones) with their respective values to be satisfied. The action parameters with their required values describe different possible configurations of the system (physical parameters...) on which one can operate. The evaluation parameters with their required parameters describe solution objectives (desired

results...) and their satisfaction is fully required. TRIZ methods are looking for a contradiction inside the system model inherent to a problematic situation. A system of contradiction based on linking between a physical contradiction and two technical contradictions is proposed in [4]. The physical contradiction reflects the impossible nature of the problem by identifying one action parameter of the system that has to be in two different states. The technical contradiction expresses the opposition between two evaluation parameters of the system. Solving the inventive problem means eliminating these contradictions and in order to do this the TRIZ methodology proposes different principles.
The final comparison of CSP and TRIZ model is illustrated on table 1. The parameters in contradictions and the variables in CSP can be matched. The main difference between CSP and TRIZ is that TRIZ differentiates evaluation and action parameters and does not permit to operate on the evaluation ones. This can be translated as a required unary constraint in CSP which has to be satisfied. The notion of binary constraint as a relation between two variables in CSP is close to the notion of technical contradiction in TRIZ. On the contrary, the two strategies are different from the problem solving point of view; this will be shown in the next section.

Table 1. Comparison of representation models and solving methods

	TRIZ	CSP
Model of system	Action parameters Possible values of parameters	Variables Domains of variables (unary constraints)
	Link between physical and technical contradiction	Binary constraints
Objective	Evaluation parameters and their required values	Variables Domains of variables (unary constraints)

2.2 Synthesis of solving methods

In order to compare different solving modes and different principles of model changes in CSP and TRIZ methods, we have proposed in [5] the definition of problem space browsed by both methods. The previous analysis of the browsed space involved definition of three distinct spaces:

- Specific problem space is defined by variables (parameters) of the problem which are limited by their domains D_i. The dimension of this space is equal to the number of variables defined by the inventive problem.
- Problem space is also defined by variables (parameters) of the problem but these are not limited by their domains. The dimension of this space is equal to the number of variables too.
- Solution space is defined by all possible variables concerning the system the inventive problem concerns. So the dimension of this solution space is infinite.

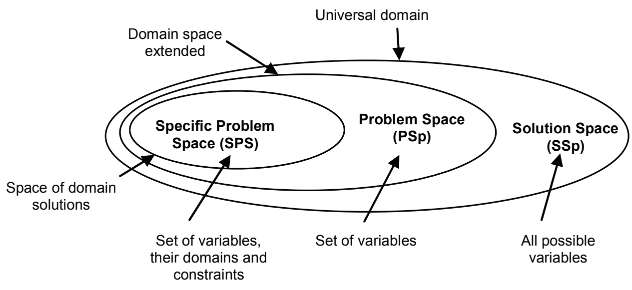

Figure 1. Definition of knowledge spaces

These spaces could be compared with the ones defined to make the difference between routine, innovative and creative design in [6]:

- Routine design proceeds within a well-defined state space, all the design variables and their possible range being known and the problem being one of instantiation.
- Innovative design refers to situations where the space of known solutions is extended by making variations or adaptations to existing designs. The range of values of existing design variables being thus extended.

- Creative design implies the formulation of the state space.

Thus the SPS is equivalent to the space of domain solutions, the PSp is equivalent to the extended domain space and the SSp is equivalent to the universal domain.

2.2.1 Operators of model change issued from CSP

CSP uses different optimization algorithms to solve a problem [7]. In the optimization strategy looking for a solution could refer to: searching the best solution according to a criterion, searching any solution the fastest way or showing that there is no possible solution. There are two kinds of optimization algorithms:

- The complete algorithms browse the whole solution research space and could guarantee the quality of the found solution measured by objective function. The most general problem solving method is "generate & test" which generates valuations and then tests their satisfaction. The backtracking algorithm incrementally extends a partial solution towards a complete solution. Both algorithms have efficiency issues and are too time-consuming till not executable. The most widely used optimisation algorithm is Branch and Bound which uses an objective function to guarantee the quality of solution.
- The incomplete algorithms search any possible solution the fastest way. They do not browse the whole research space and do not guarantee the quality of found solution. Some examples of such algorithms are consistency techniques (node, arc, path consistency) which remove inconsistent values from domains of variables or the constraint propagation which combines backtracking with consistency techniques (look back and look ahead methods). Other examples are stochastic and heuristic methods based on "generate & test" algorithms and using the smart generator of complete valuations or connectionist approaches like artificial neural networks.

Among different CSP methods solving the over-constrained problems – problem that are not solvable by simple optimization methods - two main principles appear:

- To define and evaluate preferences between constraints and to violate the constraints defined as "weak": this is used in particular in constraints hierarchy [8]. The main rule is that weaker constraints do not cause dissatisfaction of stronger constraints. Two main kinds of solving algorithms are used: the refining method (e.g. DeltaStar) solves constraints from stronger to weaker level and local propagation (DeltaBlue, SkyBlue) repeatedly selects uniquely satisfiable constraints.
- To relax the original problem by relaxing certain constraints or variables in a way that a new problem could be solved is called Partial Constraint Satisfaction Problem (PCSP) [9]. As the PCSP is a generalisation of CSOP (Constraint Satisfaction Optimisation Problem), it can use almost all optimization algorithms with a specific objective function measuring maximal satisfaction as a form of optimization.

In brief, while optimisation algorithms explore the specific problem space and do not permit to extend the solution research space, the solving methods for over-constrained problems permit to extend the solution research space within a more general problem space. This means that solving principles of constraint hierarchies and PCSP start from initial problem defined by the specific problem space 1 (SPS1) and extend this space by relaxing certain constraints and variables in order to define a new specific problem space SPS2. This space is larger than SPS1 but always covered by respective Problem Space characterized by the set of variables describing the initial problem (see figure 2).

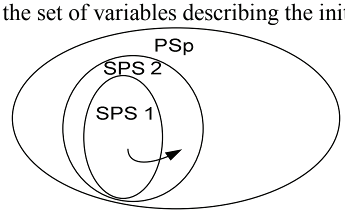

Figure 2. Model change mechanism of optimization methods

Different kinds of operators evaluating the model change are proposed in the analysed over-constrained problem solving methods.

The constraint hierarchy [8] uses a "better-comparator" evaluating and comparing different found solutions. A comparator is a relation over valuations of variables that respects the hierarchy. It defines a partial order over valuations. There are three groups of comparators:

- Locally-better comparators consider each constraint individually.
- Regionally-better comparators extend locally-better comparators to enable comparison of valuations which are incomparable by locally-better comparators.
- Globally-better comparators consider combined errors of individual constraints at each level.

To specify the problem space, the PCSP [9] uses the notion of distance between the initial problem and the changed solvable one and defines a necessary and sufficient solution distance. This metric can also compare problems, i.e. compare the values being considered at a given stage of search with the "best" solution found so far. One way is to compare the number of solutions not shared by the two problems. Another way is to compare the number of permitted value combinations not shared by the constraints of the two problems.

2.2.2 Operators of model change issued from TRIZ

Solving inventive problems in TRIZ means solving the identified contradictions. TRIZ methods propose different principles organised as knowledge bases. Three empirical knowledge bases are recognized and consist of a set of 40 principles proposed to guide the model change for problems formulated as technical contradictions, a set of 11 principles for problems formulated as physical contradictions and a set of 76 rules for problems represented through the characterization of substances and fields interactions.

Examples of some principles are: segmentation, separation of contradictory properties in time, and so on… For instance, one can add a new parameter that was not considered in the initial specific problem space but can be found in an extended solution space. This can be represented as the creation of a new specific problem space which belongs to a new problem space (cf. figure 3):

- The specific problem space SPS2 is defined by a new set of parameters, and their respective domains;
- The problem space PSp2 is defined by a new set of parameters.

These new spaces belong to the solution space characterized by a set of possible parameters describing the problem.

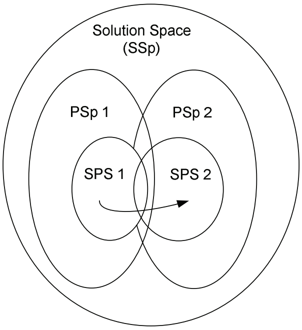

Figure 3. Model change mechanism of inventive methods

Some of the TRIZ principles are principles to guide the definition of a new Problem Space, these inventive principles could be compared with the set of four operators proposed in [6]:

- Combination: importation of parts from various designs and combination of these parts
- Mutation: modification into a structural element
- Analogy: associations to generalization outside the current domain
- First principles: investigation of the requirements without any structural representation.

All those operators are to be used when it is necessary to build a new Problem Space and when no typical adjustment of the existing problem model could lead to a solution.

2.3 Partition of knowledge space by solving methods

The previously defined knowledge spaces can be matched with each solving strategy according to their principles of transition between the spaces during the problem solving (cf. figure 4). The optimization problem solving methods cover only the specific problem space and look for the problem solution in this limited knowledge space. The adapted optimization solving methods deal with over-constrained problems which do not have any solution in the specific problem space. They are able to look for the solution in more general problem space due to the relaxing of problem constraints. The inventive methods permit to introduce new parameters in problem resolution and so cover all solution space corresponding to the problem.

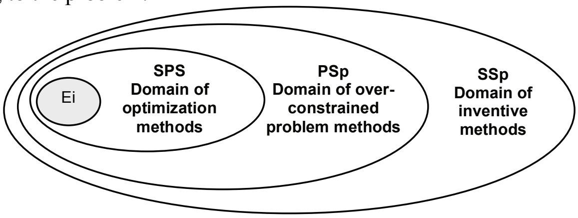

Figure 4. Partition of knowledge space by solving methods

The previous analysis of solving modes in optimization and inventive methods illustrates the advantages and disadvantages of each approach. While the inventive methods explore new problem spaces in the general solution space, the optimization methods suggest browsing a limited research space in several different systematic ways. The main principle of optimization solving methods adapted for over-constrained problems is to relax problem variables and constraints when the problem cannot be solved by classical optimization methods. The initial problem is changed in order to be less restrictive and thus to be solved by optimization methods. However, this does not guarantee solving the initial problem. In order to minimize this risk, CSP solving methods propose some operators which evaluate the distance between the initial problem and the relaxed solvable one or which define variables or constraints to relax according to their preferences or importance. TRIZ methods minimize the risk by the distinction of action and evaluation parameters defining the « role » of each parameter within the formulation and resolution of the problem. TRIZ methods create a new problem space by changing only action parameters. The space defined by a set of evaluation parameters (E_i on the fig. 4) is preserved and common for different problem spaces proposed by TRIZ methods. Contrary to TRIZ methods, CSP ones do not allow the introduction of a new parameter, and so do not really change the representation model of the problem.

3 MODEL OF GENERAL PROBLEM SOLVING

While the CSP operators are well defined by constraint programming languages, the inventive operators are more generic and cannot be formalized in the same manner or at least they are not yet formalized the same way. It is thus necessary to find a way describing both kinds of operators at the same level and so to be able to compare them. A suitable tool for formulating and defining the inventive model change operators is the C-K theory. It proposes a formal language to describe different design theories. It offers a description model for the design reasoning process. Considering the general purpose of the C-K description model, it is not as precise as constraint programming languages. Hence the C-K notions can be used as ontology to describe optimization concepts as well as inventive concepts manipulated within the problem solving process. First, the basic C-K notions will be introduced in order to be used in the general scheme of problem solving presented in the second part of this section.

3.1 Introduction of basic notions from C-K theory

C-K theory introduced in [2] proposed a formal framework based on a unified reasoning model explaining and describing existing design theories. This theory is based on the distinction of two expandable spaces: a space of concepts C and a space of knowledge K. A knowledge space is a space of propositions that have a logical status for a designer. The logical status is an attribute defining the degree of confidence assigned to a proposition. A concept is a proposition or a set of propositions that have no logical status in K. The design is defined as the process where a concept generates other concepts or is transformed into knowledge. It is characterized by the expansion of the two spaces through four interdependent operators C->C, K->K, K->C and C->K. The internal operators C->C and K->K extend every space individually. In the domain of concepts (C) it concerns classical rules in the set theory about partition or inclusion. Two kinds of partition are distinguished: restrictive and expansive. If the property added to a concept is already known in K as a property of one of the concerned entities, the partition is restrictive. If the property added is not known in K as a property of one of the entities involved in the concept definition, the partition is expansive. In the domain of knowledge K it concerns classical rules of logic and propositional calculus and new proving axioms can be proposed. The external operators K->C and C->K transform propositions of K into concepts in C and inversely using each space helping the other to expand. The K->C operator creates disjunctions and generates potential alternatives of concepts. The C->K operator creates conjunctions which could be accepted as finished designs; this corresponds to validations in classical design. The four operators form the design square shown on figure 5. Design is the process by which K->C disjunctions are generated, and then expanded by partition or inclusion, to reach C->K conjunctions.

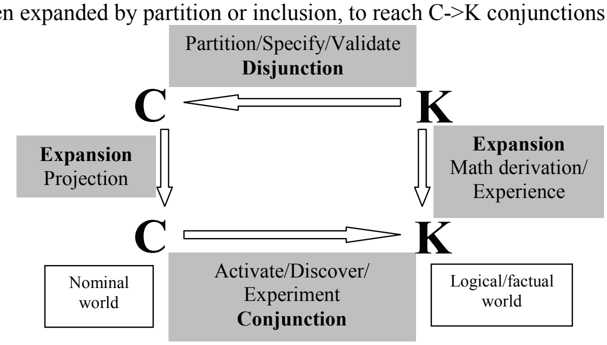

Figure 5. The design square

3.2 Model of general problem solving

In this section, a general mechanism of problem solving is described to formalize and unify both problem solving strategies – optimization and inventive. Our goal is to control and optimize the process of model change in the resolution of inventive problems. First the knowledge spaces were proposed to identify the solution research space of solving methods. Basic notions of C-K model were presented to help in formalization of inventive operators of representation model change. The general diagram of problem solving (see figure 6) illustrates the parallel between knowledge spaces defined in this paper and spaces used in C-K model.

At the beginning a problem is described within an initial Specific Problem Space (SPS1) with known parameters, their values and required system functions or required objectives described by constraints or contradictions. First the optimization methods are used to look for a problem solution in the closed Specific Problem Space. If the solution of the problem is found the solving process is finished. Otherwise a solution research space is extended to a new Specific Problem Space (SPS2) by operators of model change from over-constrained problem solving methods. Till now, the space of knowledge from C-K model was explored because all optimization methods use propositions with logical status (true or false). They look for solutions again in the space of knowledge because they are able to say if constraints are satisfied or not. The CSP operators of model change are so K->K operators in C-K model.

If no solution is found until now, the inventive methods corresponding to operators of model changes are used. These operators enable the extension of solution research space from one problem space PSp1 to another problem space PSp2, as shown in section 2.2.2. The operators and their use can be illustrated by operators from C-K model. According to the nature of used operators, spaces of concepts or knowledge will be expanded. K->C operators will create disjunctions in concept space and thus will activate new concepts not used for the moment (for example introduction of a new parameter in the problem description as a new proposed alternative). C->C operators will expand the concept space by partition or inclusion (possible values will be assign to a new parameter). C->K operators will activate knowledge which can be applied to the new concepts in the concept space (a new parameter with certain value will be evaluated according this knowledge and will obtain a logical status – if true then it will be accepted as finished design). K->K operators will create new knowledge (new proving axioms can be created concerning a new parameter). Finally to prove that the use of operators is meaningful and enables a step towards the resolution, the evaluation criterions should be established. The concrete example will be presented in the next section.

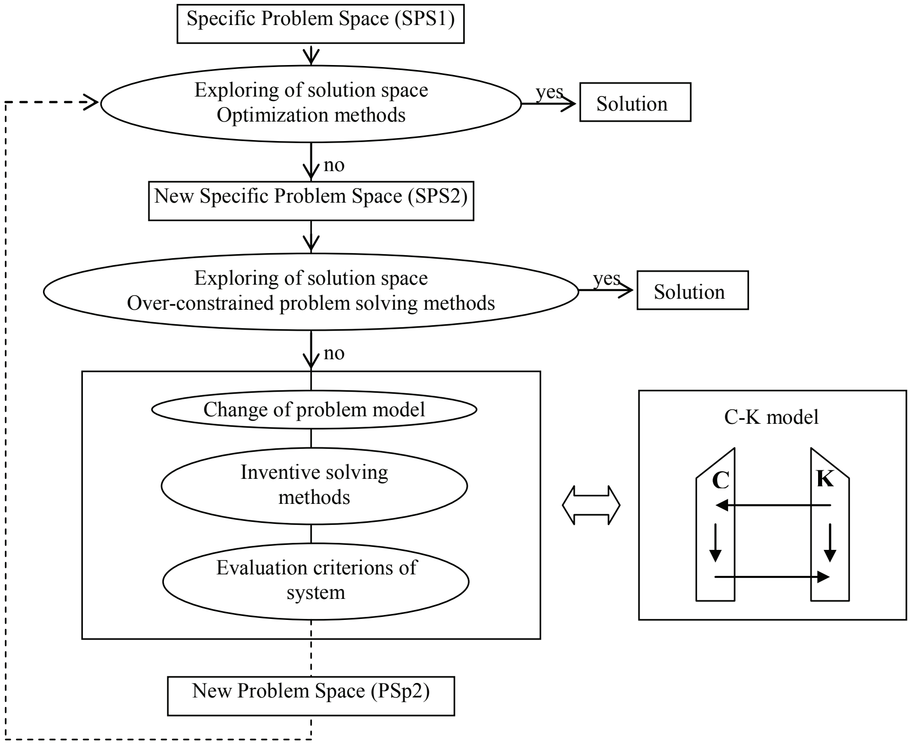

Figure 6. Model of general problem solving

4 ILLUSTRATION ON CASE EXAMPLE

In order to illustrate the different concepts inherent to the methods of problem resolution, let one consider a soldering iron (cf. figure 7) made of a resistive element, a connector, a metal tip and a handle. One can design a soldering iron which increases the precision of the welding. An initial analysis has led to a reformulation of the objective in the technical terms: decrease the distance between the hand and the tip.

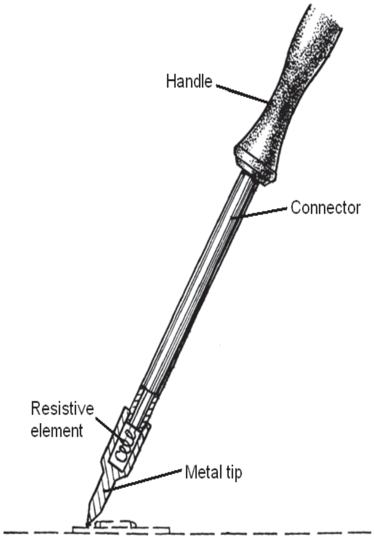

Figure 7. Soldering iron

4.1 Representation model of the soldering iron

The soldering iron has been modeled by the following set of action parameters, parameters that can be changed in order to redesign the soldering iron:
{length of the tip; temperature of the tip; impedance of the resistance; length of the connector; material of the connector; length of the handle; material of the handle}.
The evaluation parameters, which have to be met, have been identified as:
{cost of the system; temperature of the handle; precision of the system}.
The objective is to achieve a low cost of the system, a high precision of the system and a low temperature of the handle in order to prevent the user from burning.
The following set of equations model the problem:
$Temperature_{handle} = f(temperature_{tip}, material_{tip}, material_{connector}, length_{connector}, material_{handle}, length_{handle})$
$Temperature_{tip} = f(material_{tip}, impedance_{resistance})$
$Cost_{system} = f(material_{tip}, material_{connector}, length_{connector}, material_{handle}, length_{handle})$
$Precision_{system} = f(length_{connector}, length_{handle})$

4.2 Problem solving by operators of model change

The first attempt to resolve such a problem is to test if an optimization of the parameters could be done in accordance with the defined set of ranges for the action parameters. Considering that a first attempt consists in changing as few things as possible, one can imagine that the materials of the different elements do not have to be changed. Then the optimization mainly consists in looking for a length of connector enabling the shortest distance between the handle and the tip and avoiding a too high temperature of the handle.
One can consider that such a configuration cannot be found, and that it is thus necessary to extend the domain of the variables. It is then quite obvious to look for new materials to design the soldering iron, either a new material for the connector, either a new material for the handle. The relaxing of the constraint on the material of the connector could be defined as the use of two operators (cf. fig. 8):

- relaxation of a constraint
- choice of the constraint to relax

This enlargement of the domain of the variables is an extension of the initial domain, but keeping the same set of parameters, and thus, to a certain extent, the same model of problem. At least the dimension of the domain space is the same.

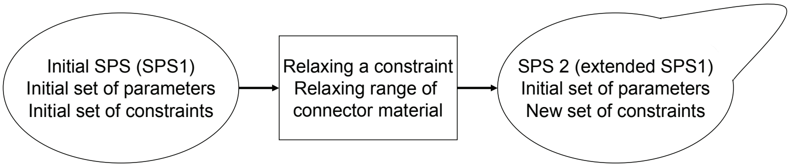

Figure 8. Optimisation operator

This new SPS could be browsed to look for a solution; it means to find a configuration of the parameters in accordance with the new ranges of the parameters satisfying in the same time the precision and the cost requirements. If no solution can be found in the extended Specific Problem Space, a new constraint can be relaxed until all the constraints are relaxed and the Problem Space is browsed. The operators to relax and to choose a constraint are common operators in the methods to solve over-constrained CSP.

If no solution exists in the Problem Space, it means that the problem is an inventive one. In [10], the problem is analysed and solved by the TRIZ methods. The analysis of the problem leads to the formulation of a contradiction which is recognized to be the core of the initial problem model: “The length of the connector has to be important in order to decrease the temperature of the handle, but the length of the connector has to be low in order to decrease the distance between the handle and the tip and thus to increase the precision of the system”.

The previous contradiction points out that whatever the materials could be, the problem model will always imply limitations due to the two relations: $\text{temperature}_{\text{handle}}=\text{f}(\text{length}_{\text{connector}})$ and $\text{precision}_{\text{system}}=\text{f}(\text{length}_{\text{connector}})$. If one tries to solve this problem a new Problem Space has to be defined in which the following relations will not be true anymore.

One can illustrate the redefinition of a new PSp by the use of one of the operators presented in [6], the operator of analogy: “Is there a suitable concept in another context?” specified to the considered problem: “Is there a suitable concept in another context to create a high difference of temperature with a little distance?” An analogy could be done with double glazing, which uses air as a resource to make thermal insulation. So the operator leads to the definition of a new value for the parameter $\text{material}_{\text{connector}}$. The connector must be made of air.

The specifications of the solution, and so the progressive definition of the Problem Space, are collected:

- the handle must be near the tip
- the handle and the tip must be separated by air
- the connector must be long in order not to transfer heat by conduction

Now the solution could be recognized and formulated through the use of a new operator: introduction of a new parameter, the shape of the connector. The connector must be long and with a loop-like shape (cf. figure 9).

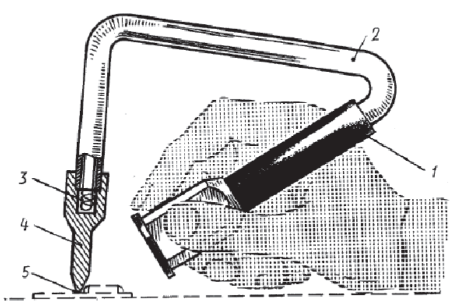

Figure 9. Solution of the soldering iron problem, from [10]

The use of the two operators implies the definition of a new Problem Space, characterized by the new set of parameters, in which Specific Problem Space could be defined, characterized by specific sets of constraints (see figure 10).

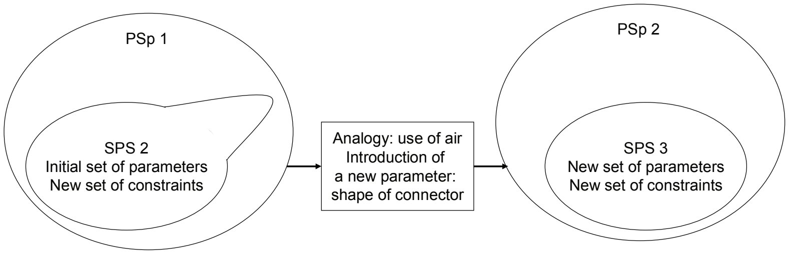

Figure 10. Inventive operators

The example is proposed just to illustrate the different kinds of operators and their inherent consequences in terms of browsed space. The solution has no specific value, as it can be discussed and compared with other solutions, but the only points to consider are the way it is built and the kind of treatment the operators imply.

5 CONCLUSION

The objective of this article was to propose a way of comparing problem solving principles from optimisation approaches and inventive ones. The C-K theory formalism has been used to describe these principles at the same level. The way the principles explore the considered knowledge spaces has been defined. The consideration of the complementary aspects of both families of solving principles is of great interest and it lays the emphasis on the necessity of defining a unified model that enables to shift easily from an optimisation approach to an inventive one.

Each inventive method involves one or more operators of model changes. First, every operator of model change and its use should be described in more details. The mutual enrichment of optimization and inventive methods will support a precise description of the inventive principles involving proposition of algorithms. Secondly, the efficiency of operators should be measured in order to prove a progress in the problem resolution. Later the whole process of inventive problem solving could be described as a succession of single model changes.

REFERENCES

[1] Cavalucci D. and Weil R.D. Integrating Altshuller's development laws for technical systems into the design process. *CIRP Annals – Manufacturing Technology*, 2001, 50(1), pp. 115-120.

[2] Hatchuel A. and Weil B. A new approach of innovative design: an introduction to C-K theory. In *International Conference on Engineering Design, ICED'03*, 2003, Stockholm, Sweden, 14.

[3] Dubois S., Rasovska I. and De Guio R. Comparison of non solvable problem solving principles issued from CSP and TRIZ. *Computer-Aided Innovation (CAI)*, 2008, 277 (Springer Boston).

[4] Khomenko N., et al., A framework for OTSM-TRIZ-based computer support to be used in complex problem management. *International Journal of Computer Applications in Technology*, 2007, 30(1), pp. 88-104.

[5] Rasovska I., Dubois S. and De Guio R. Comparaison des modes de résolution de méthodes d'optimisation et d'invention. In *8ième Congrès International de Génie Industriel, CIGI'09,* Tarbes, june 2009, submitted.

[6] Rosenman M.A. and Gero J.S. Creativity in design using a design prototype approach In *Modeling Creativity and Knowledge-Based Creative Design*, 1993 (J.S.a.M. Gero, Mary L., Editor, Lawrence Erlbaum Associates, Inc.: Mahwah, NJ, USA), pp. 111-138.

[7] Barták R. Constraint Programming: In Pursuit of the Holy Grail. In *Proceedings of the Week of Doctoral Students (WDS'99)*, Part IV, MatFyzPress, Prague, June 1999, pp. 555-564.

[8] Borning A., Freeman-Benson B. and Wilson M. Constraint hierarchies, *LISP and symbolic computation: An International Journal*, 1992, 5, pp. 223-270.

[9] Freuder E. and Wallace R. Partial Constraint Satisfaction, *Artificial Intelligence*, 58, 1992, pp. 21-70.

[10] Gasanov A.I. Birth of the Invention, *Moscow: Interpraks*, in Russian, 1995, ISBN 5-85235-226-8, pp. 432.

Contact: Roland De Guio
INSA de Strasbourg
LGECO
24 Boulevard de la Victoire
67084, Strasbourg
France
Tel: +33(0)388144749
Fax: +33(0)388144799
E-mail : roland.deguio@insa-strasbourg.fr
URL: http://www.insa-strasbourg.fr/fr/lgeco/

Roland DE GUIO is professor at the National Institute of Applied Science (INSA) of Strasbourg, France. He worked 13 years in the area of application of operational research and data analysis techniques to production flow analysis and design problems. Since 2000, his main interest is developing holistic approaches, technologies and computer based tools that support innovative design.