Comparison of non solvable problem solving principles issued from CSP and TRIZ

Sebastien Dubois[1], **Ivana Rasovska**[2] **and Roland De Guio**[3]

INSA Graduate School of Science and Technology, 24 Boulevard de la Victoire, 67084 Strasbourg Cedex FRANCE, [1] sebastien.dubois@insa-strasbourg.fr, [2] ivana.rasovska@insa-strasbourg.fr, [3] roland.deguio@insa-strasbourg.fr

Abstract: Inventive problems from many domains are usually problems we are not able to solve. This problem insolvability is often due to the incomplete or unmatched representation model of the problem that does not correspond to the given problem. In this paper, we introduce two problem solving theories for the solutionless problems: Constraint Satisfaction Problem (CSP) and dialectical based methods and models (TRIZ). It is an exploratory analysis of both theories in order to compare grounding approach and tools of both theories. Their potential complementarities will be defined in further objective to improve problem solving strategy for the inventive problems by matching the CSP and TRIZ solving principles. We consider that it will contribute to better understanding of non-solvable problems, i.e. to improve representation models of the problems and to make the problem solving more accurately.

Keywords: Constraint Satisfaction Problem, Over-constrained systems, TRIZ

1. Introduction

Problem solving methods could be categorized in accordance with their resolution. One can recognize two kinds of problems: optimization ones, for which a solution can be found, at least theoretically, by adjustment of the value of problem parameters within the framework of a given model; and inventive problems, which requires some changes of the model of the problem in order to be solved. Among others, two different problem solving theories propose solving principles for such type of problems: constraint satisfaction problem (CSP) and dialectical based methods and models.

There are several reasons for choosing a CSP to represent and solve a problem. Firstly, set of constraints is a natural medium for people to express problems in many fields and is easily understood by users. Secondly, CSP algorithms are essentially very simple but can still find solution quickly. The constraint satisfaction involves finding values for problem variables subject to constraints on acceptable combinations of values [1]. Problems, where it is not possible to find valuation satisfying all the constraints, are called over-constrained. These over-constrained problems correspond to the solutionless optimization problems the solution of which requires changing the initial model of the problem.. Typical CSP solving methods are designed for solving the optimisation problems; nevertheless several strategies for dealing with overconstrained problems are proposed.

TRIZ [2] is a theory designed for inventive problem resolution in technical domain, but several proposals have emerged to apply its axioms in other fields [3].One among its main approaches for problem resolution is to state the problem in the shape of contradictions and use them for finding a contradiction free model within the framework of given objectives. An interesting point is that TRIZ propose principles for separating the contradictory properties of a situation, which leads to get satisfactory contradiction free model of problem.

We propose an exploratory analysis in order to compare grounding approach and tools of both theories and explore their mutual complementarities. In order to do so, we shall introduce concepts of representation model and solving principles will be presented; for CSP and dialectical approaches successively. Then similar points and differences of their model changing approaches will be defined and the building stones for a comparison of their solving principles will be established. A concrete example will be shown to illustrate this analysis. The evaluation of approaches for our purpose as well as ideas of their possible match will be discussed in the conclusion.

2. Constraint Satisfaction Problem

In this section the basic notions of CSP concerning the representation model and problem solving principles are introduced. Constraint satisfaction problems along with constraint networks have been studied in Artificial Intelligence starting from seventies. Constraint satisfaction has wide fields of applications, in areas ranging from temporal reasoning, scheduling problems, expert systems and robotics to machine vision.

2.1. Representation model

The basic notion of CSP theory is the constraint. It is a relation among several variables, each taking a value in a given domain. A constraint restricts the possible values that variables can take; it represents some partial information about the variables of interest. A **constraint satisfaction problem** model consists of:

- a set of variables $X=\{x_1,\ldots,x_n\}$,
- for each variable x_i, a finite set D_i of possible values (its domain),
- a set of constraints $C=\{c_1,\ldots,c_k\}$ restricting the values that the variables can simultaneously take.

A **solution to a CSP** is an assignment of a value from its domain to each variable, in such a way that the constraints are satisfied all together. In our applications, variables can describe the physical system parameters while constraints may describe both relations between these parameters and given objectives. Over-constraints problems are problems the constraints of which cannot be satisfied all together.

2.2. CSP problem solving principles

The traditional algorithms for constraint satisfaction are not able to solve over-constrained systems although the stochastic algorithms can maximize the number of satisfied constraints. Therefore, some alternative approaches have been proposed to solve over-constrained problems or generalize the notion of constraint respectively. We give a simple outline of these approaches:

- **Extending Constraint Satisfaction Problem** associates some valuation (usually a number) to each constraint and enables relaxing of constraints according to their preference level expressed by valuation. As the constraints in classical CSP are crisp these alternative approaches propose to enable non-crisp constraints. Examples of such method are fuzzy CSP proposing a preference level with each tuple of values between 0 and 1 [4], probabilistic CSP dealing with uncertainty in CSP [5] or weighted CSP taking into account for example the costs [6].
- **Partial Constraint Satisfaction Problem** is based on scheme of Freuder and Wallace [1] that allows the relaxation and optimization of problems via weakening the original CSP. Partial constraint satisfaction involves finding values for a subset of the variables that satisfy a subset of the constraints. The method "weakens" some of constraints by enlarging their domain in order to permit additional acceptable value combinations on a "similar" but different problem than initial given problem.
- **Constraint hierarchies [7]** describe the over-constrained systems of constraints by specifying constraints **with hierarchical preferences**. In many situations we can state required (hard) and preferential (soft) constraints. The required constraints must be hold but other constraints are merely preferences, it is tried to satisfy them as far as possible, but solutions that do no satisfy soft constraints may be generated. A constraint hierarchy consists of a set of constraints, each labelled as either required or preferred at some strength. An arbitrary number of different strength is allowed.

- **Alternative and generalized approaches** propose general frameworks to model features of various CSP problems. Among these approaches two approaches are the most popular. A compositional theory for reasoning about over-constrained systems is an extension to constraint hierarchies permitting to consider compositionality and incrementality in constraint logic programming [8]. This theory defines a scheme for composing together solutions to individual hierarchies and shows that hierarchy composition can be expressed very simply using multisets. The semiring-based constrained satisfaction is based on the observation that a semiring (that is, a domain plus two operations satisfying certain properties) is all what is needed to describe many constraint satisfaction schemes [9]. In fact, the domain of the semiring provides the levels of consistency (which can be interpreted as cost, or degrees of preference, or probabilities, or others), and two operations define a way to combine constraints together. Other way, the semiring specifies the values to be associated with each tuple of values of the constraint domain.

All these methods can be classified into two general principles:

- to state or evaluate preferences of the constraints or the combinations of constraints and to relax the "soft" ones (extending CSP, constraint hierarchies, alternative approaches)
- to relax the original CSP by modifying some constraints in such a way that the modified CSP has solutions.

For our purpose we will study these two general principles in their basic form: constraint hierarchies and partial constraint satisfaction problem.

Constraint hierarchies specify hierarchical preferences of the constraints, hard constraints are required and soft (preferential) constraints are satisfied as much as possible. The solution of the problem is found by relaxing the soft constraints. In constraint hierarchies [10], [7], each constraint is labelled by a preference expressing the strength of constraint – called **labelled constraint**. The labels can be expressed by names like required, strong, medium, weak and weakest. An arbitrary number of different strengths is allowed. A **constraint hierarchy** H is a finite set of labelled constraints. The set of constraints with the same label composes a **hierarchy level** H_j. A valuation for the set of constraints is a function that maps variables in the constraints to elements in the domain of variables over which the constraints are defined. A **solution to the constraint hierarchy** is a set of valuations for the variables in the hierarchy such that any valuation in the solution set satisfies at least the required constraints.

There is a number of reasonable candidates for the predicate better, which is called a **comparator**. The comparator formally describes the idea that satisfaction of a stronger constraint is strictly preferred to satisfaction of an arbitrary number of weaker constraints. A detailed summary of constrained hierarchies and the solving algorithms can be found in [7]. In general, there are two types of special algorithms for solving constraint hierarchies: refining algorithms and local propagations.

For our purpose, we retain that this approach can only relax the soft constraints but change neither problem variables nor their domains. It is supposed that the problem is not well stated and so one can modify or in this case relax the constraints. This may be relevant in some domain applications such as planning or design where the constraints do not need to be a part of the described physical system. In this case the constraints make part of system objectives and do not describe relations in the physical system.

Contrary to the constraint hierarchies, **Partial constraint satisfaction problem** (PCSP) weakens both the constraints and the variables (their domains are enlarged) to permit additional acceptable value combinations. It involves finding values for a subset of the variables that satisfy a subset of the constraints [1]. The goal is to search a simpler problem (the representation model of the problem) we can solve. A problem space is partially ordered by the distance between the original problem and the new simpler one. A problem space is a partially ordered set of CSPs where order ≤ is defined the following way. Let (*sols(P)* denotes the set of solutions to a CSP called *P*: P1≤ P2 if *sols(P1)* is a superset of *sols(P2)*.A solution to a PSCP is a problem P' from the problem space and its solution, where the distance between P and P' is less than N. If the distance between P and P' is minimal, then this solution is optimal.

Four ways to weaken a CSP [10] are possible: 1. enlarging the domain of a variable, 2. enlarging the domain of a constraint, 3. removing a variable and 4. removing a constraint. All previous cases can be considered in terms of enlarging the domain of a constraint only [1].

To solve the problem, partial constraint satisfaction problem weaken variables and constraint domains thus the representation model is enlarged. Nevertheless, PCSP does not permit to introduce a variable offering a new point of view (new dimension) to the problem and thus permitting to solve it more accurately.

3. Dialectical approaches

This section focus our attention on methods and models of TRIZ, a theory for inventive problem solving, based on dialectical thinking. One of the main characteristics of dialectical thinking is that it places all the emphasis on change [11]. Dialectics is looking for contradictions inside phenomenon as the main guide to what is going on and what is likely to happen. Basing evolution of systems on the elicitation, understanding and resolution of contradictions is also one of the main characteristics of TRIZ. TRIZ theory aims at understanding the way technical systems evolve and developing methods and tools for inventive technical problems solving. The principles of TRIZ have been widely applied in many domains. One of the benefits, which will be considered here, is the existence of models to represent problems and of principles to guide the change of model from a non-solvable one to a solvable one.

3.1. Representation models

One of the main ideas of TRIZ based theories is to identify inside systems the contradictions inherent to a problematic situation. In its original representation, contradictions, in TRIZ, are defined at three different levels:

- **Administrative contradiction**, which is the definition of a situation where an objective is given, but not satisfied.
- **Technical contradiction**, which is the expression of the opposition between two parameters of a system, when the improvement of one factor implies the deterioration of another factor.
- **Physical contradiction**, which objective is to reflect the impossible nature of the problem by identifying one parameter of the system that has to be in two different states.

Studies have been proposed to enlarge the scope of applicability of TRIZ principles and methods to problems not relative to technical systems. Among these studies OTSM-TRIZ proposed a description of the system of contradictions, which proposes a link between a physical contradiction and two technical contradictions. The system of contradictions is represented in bold in figure 1.

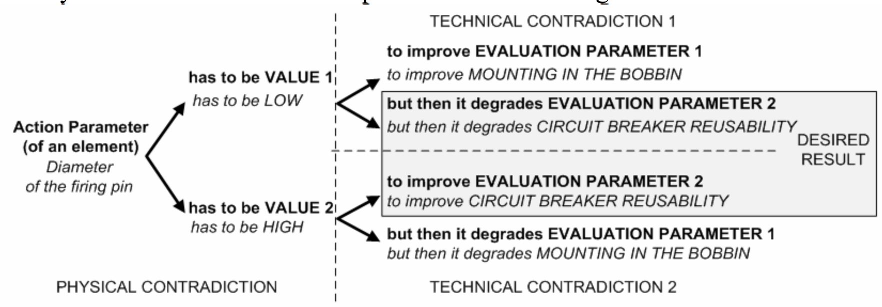

Figure 1 OTSM-TRIZ system of contradiction

This system of contradictions is based on the existence of a physical contradiction and of two technical contradictions that justify the need of the two different states of the physical contradiction. The two technical contradictions are complementary as they correspond to the increasing of the first parameter that implies the decreasing of the second; and of the increasing of the second parameter that implies the decreasing of the first. The two parameters of the technical contradictions are defined in [12] as taking part in describing the objective, they are called Evaluation Parameters, whereas the parameter of the physical contradiction is a mean to make the situation change, defined as Action Parameter.

3.2. TRIZ problem solving principles

As there exist different levels of problem formulation, there exist, in TRIZ different levels for problem resolution. In TRIZ one can recognize three empirical knowledge bases used to guide the model change.

- A set of 40 principles combined in a matrix is proposed to guide the model change for problems formulated as technical contradictions. For example, a principle is: segmentation (divide an object into independent parts; or divide an object into parts so that some of its part can be easily taken away; or increase the degree of object segmentation).
- A set of 11 principles is proposed to guide the model change for problems formulated as physical contradictions. For example, a principle is: separation of contradictory properties in time.
- A set of 76 rules is proposed to guide the model change for problems represented through the characterization of substances and fields interactions ("SFM" model in TRIZ terminology). For example, a rule is: If there is an object which is not easy to change as required, and the conditions do not contain any restrictions on the introduction of substances and fields, the problem is to be solved by synthesizing a SFM: the object is subjected to the action of a physical field which produces the necessary change in the object. The missing elements being introduced accordingly.

4. A comparison of CSP and TRIZ

This chapter will at first establish complementarities and differences between the previously defined models of problems' representation coming from CSP and from TRIZ based approaches. In a second part, the differences between the principles to solve problems and their possibilities to change the representation model of the problem will be discussed in regard of their potential complementarities to improve problem solving strategy for the inventive problems.

4.1. Comparison of representation model

If trying to build analogies between the two models of problem representation, of the CSP and of the system of contradiction, one can notice that problems in CSP are described by a set of variables and constraints on these variables. These constraints are of three kinds: required values for variables to satisfy the problem, domain of possible values for variables, and set of relations between the variables.

In TRIZ-based approaches, problems are modelled by two types of parameters (evaluation and action) and set of values. The evaluation parameters and their required values define the objective of resolution, whereas action parameters and their values define means to act on the problem.

Parameters in contradictions and variables in CSP models can be matched. The main difference between CSP and contradiction models is that, contrary to CSP, contradiction model differentiates evaluation parameters and action parameters. Evaluation parameters represent the desired domain for solutions and action parameters impact system and so represent the possible domain of variables. In CSP the methods to solve problems could operate both on evaluation and action parameters.

Let us consider an electrical circuit breaker. When an overload occurs, the overload creates a force (due to magnets and electrical field) which operates a piece called firing pin. The firing pin opens the circuit by pressing the switch, located in the circuit breaker. In case of high overload, the firing pin, this is a plastic stem, breaks without opening the switch. Components are presented on figure 2.

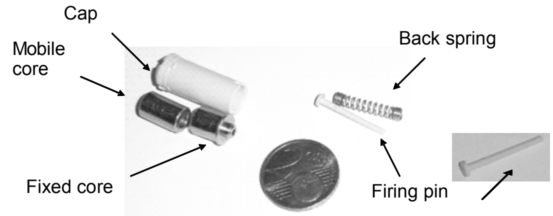

Figure 2 Components of electrical circuit breaker.

The problem has been studied and the main system parameters and their domain have been defined as: A1: firing pin material (plastic – 1, metal – 0) ; A2: core internal diameter (high – 1, low – 0) ; A3: core external diameter (high – 1, low – 0) ; A4: firing pin diameter (high – 1, low – 0) ; A5: spring straightness (high – 2, medium – 1, low – 0) ; E1: circuit breaker disrepair (satisfied – 1, unsatisfied – 0) ; E2: circuit breaker reusability (satisfied – 1, unsatisfied – 0) ; E3: spring core mounting (satisfied – 1, unsatisfied – 0) ; E4: firing pin bobbin mounting (satisfied – 1, unsatisfied – 0) ; E5: normal mode release (satisfied – 1, unsatisfied – 0) ; E6: firing pin initial position return (satisfied – 1, unsatisfied – 0). The system behaviour was modelled by Design of Experiments and it is shown in table 1b.

The relations between system's parameters are described in the form of equations representing constraints in the table 1a. As example the following constraint: "If the firing pin material is plastic then there is an irreversible degradation of the circuit breaker" is defined as "(A1=1) => (E1=1)" in the table 1a. The objective is to satisfy all the constraints, i.e. all evaluation parameters are equal to 1. In the table 1b we note that there is no such solution, so the problem is over-constrained. The possible problem solving by constraint hierarchies and partial constraint satisfaction problem is shown in 4.2.

The analysis of the data by TRIZ approach leads to the identification of a set of contradictions among which the most important has been identified by experts as

being the contradiction on the firing pin diameter, represented in italic in figure 1. This corresponds to the set of constraints in the CSP approach that could not be satisfied at the same time. So in general we are not able to solve the problem.
Since the comparison between the models is done, let tackle the comparison between the solving principles, this will be the object of the next part.

Table 1 a) Constraints for CSP model.

Constraints
(A1=1) ➔ (E1=1) (A1=0) ➔ (E1=0)
(A2=1) ✧ (A3=0) ✧ (A4=1) ➔ (E2=1) (A2=0) ✧ (A4=0) ➔ (E2=0)
(A2>A4) ⇔ (E3=1)
(A3=1) ⇔ (E4=1)
(A5=0) ⇔ (E5=0)
(A5≠0)⇔ (E6=1)

b) DoE for the circuit breaker example.

A1	A2	A3	A4	A5	E1	E2	E3	E4	E5	E6
1	1	0	0	1	1	0	1	1	1	1
0	1	1	1	1	0	1	0	0	1	1
1	0	1	0	0	1	0	1	0	0	0
1	1	0	0	0	1	1	1	1	0	0
1	0	1	0	1	1	0	1	0	1	1
0	1	0	1	2	0	1	0	1	1	1
1	0	1	1	0	1	0	1	0	0	0
1	0	0	0	1	1	0	0	1	1	1
0	1	0	0	2	0	1	0	1	1	1

4.2. Comparison of solving principles

The first element of comparison between CSP and TRIZ is the aim of each category of principles. A second element is the mechanism these principles use to transform the problem model into a solution model.

Two over-constrained solving methods issued from CSP (constraint hierarchies and PCSP) use relaxing of constraints while aiming and solving the problem. Constraint hierarchies will specify constraints with hierarchical preference and will relax soft ones. This could be done on constraints concerning the domains of both action and evaluation parameters and on constraints concerning the relations between variables. In our example, the evaluation parameters E2 and E4 described by related constraints are considered hard and E1, E3, E5 and E6 are considered soft without preferences between them. This statement of required and preferential constraints is done by experts. In this case, the equivalent solutions are coloured in grey in the table 4. The comparison of TRIZ solving principles with the constraint hierarchies leads to the conclusion that such a type of hierarchy is implicitly proposed in TRIZ. As the parameters in TRIZ are categorized into two kinds: evaluation and action ones, and as the evaluation parameters are parameters that have to be fitted to solve the problem, analogy presented in table 2 can be defined. To solve the problem in constraint hierarchies it is possible to relax action parameters as well as evaluation parameters and their constraints.

Table 2 Parallel in modelling between TRIZ and Constraint hierarchies.

TRIZ	CSP
Domains of Action Parameters	Soft constraints
Domains of Evaluation Parameters	Soft and hard constraints

Generic principle of PCSP is the enlarging of the domain of a constraint; this principle could lead to two totally different actions. Either the enlarging of the domain will concern an action parameter; either it will concern an evaluation one. In our case, we can enlarge the domain of the evaluation parameters E5 and E6 and so the fourth line of the table 4 becomes a solution of the partial problem.

Relaxing a constraint when it concerns an evaluation parameter is something that is not admitted in TRIZ-based approaches, as it is considered changing the problem and not solving it. This is one of the main principles in CSP tools, to change the problem into a less constrained one, but then it cannot always be considered as solving the initial problem. If the problem "how to live ten days without water" is considered an over-constrained one, trying to solve the problem "how to live two days without water" is not solving the initial problem.

Relaxing a constraint when it concerns an action parameter is changing the representation of the system. This is something that can be considered in TRIZ-based approaches. Resolving the previously described example of breaking circuit with TRIZ methods leads to change the problem model. Bellow are given two methods for guiding the change of model and their possible interpretation.

1. *Separation in space: try to separate the opposite requirements in space.* The firing pin diameter is low in accordance with the bobbin diameter but high to avoid breaking. This can be done by enlarging the bobbin diameter, this means by locating the spring outside of the core.
2. *Elimination of harmful interaction by modification of existing substances. If there are a useful and harmful effects between two substances and it is not required that these substances be closely adjacent to one another, but it is forbidden or inconvenient to use foreign substance, the problem is solved by introducing a third substances (modification of the existing substances) between these two substances.* A part of the fixed core becomes movable and acts as the firing pin, thus the magnetic surface and the pin rigidity are increased. The pin has a high diameter from the fixed core to the mobile core and a low diameter but in a more resistant material from the mobile core.

The two presented rules to guide the change of model leads to the introduction in the initial model of problem representation of a new action parameter: spring location in the first case and fixed core mobility in the second one.

The table 3 summarizes the general comparison of two studied problem solving principles – TRIZ and CSP.

Table 3. Comparison of TRIZ and CSP models and methods for resolution.

	TRIZ	CSP
Model of system	Action parameters Link between physical and technical contradiction	Variables Domains of variables Constraints
Objective	Evaluation parameters + required values	Constraints
Methods to change model	Enlarge domain of action parameter Introduce new action parameter	Enlarge domain of variable
Solved problem	Initial problem	New problem

5. Conclusion and perspectives

An exploratory analysis of two different solving theories was proposed in order to compare grounding approach and tools of both theories. Now we will discuss some advantages and disadvantages of each theory according to their capacity to change the representation model of the problem for successful problem resolution.

The domain of CSP is quite well formalised by a fixed representation model. There exists a number of proved solving algorithms and quite a lot of automated CSP systems and informatics tools. On the contrary, CSP proposes only partial resolution of the problem by constraint relaxing. This means that CSP does not solve the initial problem but a new one which is sufficiently closed to the initial one. This approach does not permit to introduce a new variable. In consequence we cannot use any operator helping to pass from an actual representation model to a new one.

The TRIZ approach aims at solving the initial problem that means it allows the real change of the representation model. This approach distinguishes between action parameters and evaluation parameters and thus specifies the unchangeable objectives. Its solving principles are independent from the application domain. The big disadvantage is that there are neither formalised algorithms nor developed software tools to extract and analyse contradictions for the moment.

We consider that the match of CSP and TRIZ solving principles will contribute to better understanding of non-solvable problems. A new operator could be introduced in order to improve a CSP representation model, i.e. to pass from an old model that does not fit to a new one. Therefore the changed representation model of the problem will make the problem easier to solve by actual solving strategies. The possible strategy 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 repetitive using of CSP solving strategies can help to characterize partial solutions or optimums according some criterion which is not possible in TRIZ approaches. On the contrary, CSP was not founded for inventive problems and so the model changing strategies are very basic and could be improved by TRIZ methods.

References

1. Freuder E., Wallace R.: Partial Constraint Satisfaction, Artificial Intelligence, 58, pp. 21-70 (1992).
2. Altshuller G. S.: Creativity as an Exact Science, Gordon and Breach, New York (1988).
3. Khomenko N., De Guio R.: OTSM Network of Problems for representing and analysing problem situations with computer support, 2nd IFIP Working Conference on Computer Aided Innovation, Technical Center Brighton, Springer, USA (2007).
4. Ruttkay Zs.: Fuzzy Constraint Satisfaction in Proceedings of the 1st IEEE Conference on Evolutionary Computing, Orlando, pp. 542-547 (1994).
5. Fargier H., Lang J: Uncertainty in constraint satisfaction problems: a probabilistic approach. In Proceedings of ECSQARU-93, pp. 97-104 (1993).
6. Schiex T., Fargier H., Verfaillie G.: Valued Constraint Satisfaction Problems: Hard and Easy Problems. In Proceedings of International Joint Conference on Artificial Intelligence , Montreal (1995).
7. Borning A., Freeman-Benson B., Wilson M.: Constraint hierarchies, LISP and symbolic computation. An International Journal, 5, pp. 223-270 (1992).
8. Jampel M.: A Compositional Theory of Constraint Hierarchies, pp. 189-206. Jampel, Freuder, Maher (Eds.), Over-Constrained Systems, Lecture Notes in Computer Science (1996) .
9. Bistarelli S., Fargier H., Montanari U., Rossi F., Schiex T., Verfaillie G: Semiring-Based CSPs and Valued CSPs: Basic Properties and Comparison. Jampel, Freuder, Maher (Eds.), Over-Constrained Systems, Lecture Notes in Computer Science pp. 111-150 (1996).
10. Bartak R.: Modelling Soft Constraints: A Survey, Neural Network World, vol. 12(5), pp. 421-431 (2002).
11. Rowan J., Ordinary Ecstasy. The Dialectics of Humanistic Psychology, Brunner-Routledge (2001).
12. Eltzer T., De Guio, R.: Constraint based modelling as a mean to link dialectical thinking and corporate data. Application to the Design of Experiments, 2nd IFIP Working Conference on Computer Aided Innovation, Springer, Brighton, USA (2007).