A Heuristic TRIZ Problem Solving Approach based on Semantic Relatedness and Ontology Reasoning

Wei Yan¹,², Cecilia Zanni-Merk², François Rousselot², Denis Cavallucci¹, and Pierre Collet²

¹LGECO/INSA Strasbourg, 24 Boulevard de la Victoire, 67084 Strasbourg Cedex, France
²LSIIT/BFO Team (UMR CNRS 7005) - Pôle API BP 10413, 67412 Illkirch Cedex, France
{wei.yan,cecilia.zanni-merk,francois.rousselot,denis.cavallucci}@insa-strasbourg.fr,pierre.collet@unistra.fr

Abstract. The theory of inventive problem solving (TRIZ) was developed to solve inventive problems in different industrial fields. In recent decades, modern innovation theories and methods proposed several different knowledge sources, whose use requires extensive knowledge about different engineering domains. In order to facilitate the use of the TRIZ knowledge sources, this paper explores a heuristic TRIZ problem solving approach. Firstly, TRIZ users start solving inventive problem with the TRIZ knowledge source of their choice. Then other similar knowledge sources are used according to a calculation of semantic relatedness. Finally, heuristic solutions are returned by ontology reasoning on the knowledge sources. The case of a "Diving Fin" is used to show the heuristic TRIZ problem solving process in detail.

Keywords: TRIZ, Semantic Relatedness, Ontology Reasoning.

1 Introduction

TRIZ, the theory of inventive problem solving, was developed in the middle of the 20th century by G. S. Althshuller. The goal of this methodology was, initially, to improve and facilitate the resolution of technological problems [1] [2].

During the development of TRIZ, different knowledge sources were established in order to solve different types of inventive problems. These knowledge sources are all built independently of the application field, but their levels of abstraction are very different, making their use quite complicated. The main reasons for this difficulty are:

1. The wealth of knowledge available in TRIZ is available for solving a large variety of inventive problems but access to the needed specific knowledge might be troublesome.
2. Using a recommendation proposed by TRIZ for solving a specific problem requires extensive knowledge of different engineering domains and is not currently supported by the methodology. As a consequence, the user is supposed to possess a high degree of expertise in engineering design.

In order to facilitate the TRIZ problem solving process, the development of an intelligent knowledge manager is explored in this paper. On the one hand, according to the TRIZ knowledge sources ontologies, the knowledge manager offers to the users the relevant knowledge sources of the model they are building. On the other hand, the manager has the ability to fill "automatically" the models of the other knowledge sources. Firstly, TRIZ users start solving an inventive problem with the chosen knowledge source, and then other similar knowledge sources are obtained based on the links among the TRIZ knowledge sources. Finally, heuristic solutions are returned by ontology reasoning on the knowledge sources.

The remainder of the paper is organized as follows. Section 2 introduces the main tools and techniques of classic TRIZ. Section 3 discusses the heuristic TRIZ problem solving in detail, including the framework, the semantic relatedness method and also the ontology reasoning mechanism. Section 4 elaborates the whole process in terms of the case of a "Diving Fin". Section 5 concludes with a summary and outlines some directions for future research.

2 The TRIZ Problem Solving Approach

This section presents the TRIZ problem solving approach and the TRIZ knowledge sources.

The TRIZ problem solving approach is made up of three phases, as shown in Fig.1:

1. The "formulation" phase, where the expert uses different tools to express the problem in the form of a contradiction [1] network or another model.
2. The "abstract solution finding" phase, where access to different knowledge bases is made to get one or more solution models. Generally, in this step, TRIZ users are required to have wide experience on the TRIZ knowledge sources. They need to be capable of choosing the accurate abstract solution according to the current abstract problem.
3. The "interpretation" phase, where these solution models are instantiated with the help of the scientific-engineering effects knowledge base, to get one or more solutions to be implemented in the real world.

2.1 The TRIZ knowledge sources

The knowledge sources for solving inventive design problems, including 40 inventive principles, 11 separation methods and 76 inventive standards, are used to eliminate technical contradictions, eliminate physical contradictions and provide common problem-solving methods respectively [2].
Inventive principles A new problem can be solved by the use of a proper inventive principle, after the problem has been formulated as a technical contradiction in terms of predefined generalized parameters: "a generalized parameter to be improved versus a generalized parameter which deteriorates". Inventive principles only recommend a method for eliminating the contradiction. For example,

**Inventive Principle 35**: Change of physical and chemical parameters

1. Change the object’s aggregate state.
2. Change concentration or consistency of the object.
3. Change the degree of flexibility of the object.
4. Change the temperature of the object or environment.

Separation methods An advanced form of contradictions is physical contradictions. To model an inventive problem as a physical contradiction, a physical object that must have two conflicting properties has to be identified. To solve problems containing physical contradictions, separation methods for physical contradiction elimination are used. Among them, there are:

1. Separation of conflicting properties in time.
2. Separation of conflicting properties in space.
3. Separation of conflicting properties at micro level.

Inventive standards They are drawn from the fact that most inventions refer to a conceptual modification of physical systems. If problems from different domains result in identical physical models, this means that the problems are similar. Therefore, they can be solved by applying the same method.
Standards are built in the form of recommendations, and generally, formulated as rules like \( \text{If } \langle \text{Condition1} \rangle \text{ and } \langle \text{Condition2} \rangle \text{ then } \langle \text{Recommendation} \rangle \). Both conditions permit recognizing the typology of the problem associated to the standard. This way, for a built problem model, there exist a certain number of recommendations allowing the construction of the corresponding solution model. An example:

*Inventive Standard 2.4.12*: Applying electrorheologic fluid.

If a magnetic fluid cannot be used, the electrorheologic fluid may be useful.

The missing links among the knowledge sources In order to present objects, ideas, situations and relationships in the inventive problem solving process, as stated in [3], TRIZ experts use the knowledge sources for their own specific applications from their point of view. The knowledge sources for resolution are situated at different levels of abstraction and at different levels of “closeness to reality”. In fact, inventive principles, for example, even if they seem to refer to concrete reality (for instance, ”inert atmosphere”) are conceptually more abstract than inventive standards, which refer to concrete substances or fields. The imagination effort to be done by the user to apply an inventive principle is much more important than that the one to be done if he wants to use an inventive standard.

3 The Heuristic TRIZ Problem Solving Approach

In this section, firstly the framework of the heuristic TRIZ problem solving approach is proposed. Then we introduce the retained semantic relatedness methods to find the missing links among the TRIZ knowledge sources. Finally, the TRIZ knowledge sources ontologies are presented and the mechanism of ontology reasoning is elaborated in detail.

3.1 Framework

The framework of the heuristic TRIZ problem solving approach is given in Fig.2, which is made up of a preprocess and a main process. The preprocess includes two steps:

1. Use semantic relatedness methods to find the missing links among the TRIZ knowledge sources, and store the obtained similarities in a database;
2. Establish the TRIZ knowledge sources ontologies and set up ontology reasoning rules. Ontologies are used to formalize the main concepts in the TRIZ knowledge sources, and the Semantic Web Rule Language (SWRL)[5] is used to describe the reasoning rules.

The main process with the results of the pretreatment consists of four phases:
1. Search an abstract solution with one of the TRIZ knowledge sources. According to conventional TRIZ problem solving methods, TRIZ users choose a TRIZ knowledge source, for example using inventive principles, to eliminate a technical contradiction;

2. For the abstract solutions obtained in phase 1, select the similar knowledge sources on the basis of the similarities stored in the database;

3. According to the TRIZ knowledge sources ontologies, choose the appropriate reasoning rules from the rule set;

4. Implement ontology reasoning to provide heuristic solutions.

3.2 Semantic Relatedness Methods for Defining the Missing Links among the TRIZ Knowledge Sources

Considering the way the TRIZ knowledge sources are represented as short texts, we explore two methods to calculate the semantic similarity among short texts: basic matching and improved matching with word weight. Regarding the basic matching, it has been discussed in previous publications [11].

Word weight (word frequency) is proposed to measure the different ways words act on the short text similarity. We use here $tf \times idf$, in which two parts of the weighting were proposed by Gerard Salton [7] and Karen Sparck Jones [8] respectively, to calculate the word weight.

Given two short texts $A_1$ including words sequence $A_{11}, A_{12} \cdots A_{1n}$ and $A_2$, including $A_{21}, A_{22} \cdots A_{2m}$, the most similar words in $A_2$ are selected for each word in $A_1$. However, this is not enough to estimate the similarity between $A_1$
and \( A_2 \), because the inverse situation is not considered, that is, the most similar words in \( A_1 \) for each word in \( A_2 \) also need to be considered. Taking this into account, an improved method of matching with word weight is proposed:

\[
s(A_1, A_2) = \sum_{i=1}^{n} w_{w1i} \times \max_{j=1}^{m}(s(A_{1i}, A_{2j})) + \sum_{i=1}^{m} w_{w2i} \times \max_{j=1}^{n}(s(A_{2i}, A_{1j}))
\]

where \( s(A_{1i}, A_{2j}) \) and \( s(A_{2j}, A_{1i}) \) represent the word similarity of \( A_{1i} \) and \( A_{2j} \), \( 1 \leq i \leq n, 1 \leq j \leq m \) (the way to calculate them was introduced in [11]). \( w_{w1i} \) is the word weight of the \( i \)th word in \( A_1 \) and \( w_{w2i} \) is the word weight of the \( i \)th word in \( A_2 \).

### 3.3 The TRIZ Knowledge Sources Ontologies

In general, an ontology is composed of concepts and relationships that are used to express knowledge about the modeled field. Therefore, the constitution of a framework for problem-solving based on different knowledge sources as the aim of ontologies will be considered.

Here we use the process of constructing the inventive standards ontology as an example, which are the foundations for the so-called Su-Field analysis. Generally, Su-Field modeling includes the construction of a generic problem model and a generic solution model, both in the form of Su-fields. The transformation from the generic problem model to the generic solution model is implemented through the use of the appropriate inventive standards. Finally, the solution model is interpreted to the specific solution for the specific problem. The minimum technical system consists of two substances - an object, a tool to act on the object, and a field.

The inventive standards ontology is established in the following hierarchy:

1. There are two different classes of Su-Field: Problem_Model and Solution_Model. Each model consists of three elements - a Field, a Tool and an Object;
2. There are five kinds of Fields: Magnetic, Mechanic, Chemical, Thermal, and Electric and five sorts of Substance: Material, Instrument, Human-being, Component, and Environment;
3. Inventive_Standards are divided into five Classes, and each Class has several Groups;
4. The properties \( PM \_hasField \), \( PM \_hasTool \) and \( PM \_hasObject \) are defined to describe the relationships respectively between the Problem_Model and Field, Tool and Object;
5. The properties \( SM \_hasField \), \( SM \_hasTool \) and \( SM \_hasObject \) are defined to describe the relationships respectively between the Solution_Model and Field and Substance with different roles.

Compared with Description Logic (DL), Web Ontology Language (OWL) [4], constructed based on SHIQ and described in the form of Extensible Markup
Language (XML), makes it easier and much flexible to reuse and share the knowledge base. According to the ability of description and reasoning, OWL-DL is used to describe the concepts and their relationships in the TRIZ knowledge sources in Protégé [9]. A fragment of the inventive standards ontology is shown in Fig.3.

![Fig. 3. The fragment of the inventive standards ontology](image_url)

### 3.4 Ontology Reasoning for Providing Heuristic Solutions

**Ontology Reasoning Rules** Compared with RuleML and RIF aiming to tackle broad problem of rule interchange, Semantic Web Rule Language (SWRL) [5], is an OWL-specific rule language, which allows users to write rules that can be expressed in terms of OWL concepts to provide more powerful deductive reasoning capabilities than OWL alone.

The rules for each knowledge source can be established using SWRL. Taking the inventive standards ontology as an example, we build the rules for the *Inventive Standard 1.1.1* - If there is an object which is not easy to change, add the missing necessary elements. We set the SWRL rules as follows:

\[
\begin{align*}
\text{Problem Model}(?x) \land \text{PM isSolvedBy}(?x, ?y) \land \text{IS1.1.1}(?y) \land \\
\text{PM hasFieldNum}(?x, 0) & \rightarrow \text{PM hasField}(?x, \text{AdditiveField1}) \\
\text{Problem Model}(?x) \land \text{PM isSolvedBy}(?x, ?y) \land \text{IS1.1.1}(?y) \land \\
\text{PM hasToolNum}(?x, 0) & \rightarrow \text{PM hasTool}(?x, \text{AdditiveSubstance1}) \\
\text{Problem Model}(?x) \land \text{PM isSolvedBy}(?x, ?y) \land \text{IS1.1.1}(?y) \land \\
\text{PM hasObjectNum}(?x, 0) & \rightarrow \text{PM hasObject}(?x, \text{AdditiveSubstance2})
\end{align*}
\]

where \text{PM hasFieldNum}, \text{PM hasToolNum} and \text{PM hasObjectNum} are defined to evaluate the missing necessary elements. According to the reasoning rules, the instances \text{AdditiveField1}, \text{AdditiveSubstance1} and \text{AdditiveSubstance2} are connected automatically to the instances of the \text{Problem Model}, which is qualified...
for the inventive standard 1.1.1, and are returned to TRIZ users together with the instances of the Solution Model.

**Ontology Reasoning Engine** Jess [6] is a high-performance rule-based engine and scripting environment written entirely in Sun’s Java language. SWRLJessBridge [10] is usually used to translated ontologies and rules from the OWL and SWRL formats to the Jess format.

The process to reason with the TRIZ knowledge sources ontologies includes four steps:

1. Instantiate the TRIZ knowledge sources ontologies, and convert them into the Jess format using SWRLJessBridge;
2. Infer with the Jess engine;
3. Generate the asserted ontologies in Jess format;
4. Transfer the asserted ontologies from the Jess format to the OWL format.

### 4 The Case of the "Diving Fin"

A software prototype was developed in a Java 1.7.02 platform, WordNet 2.0, Protége 3.4.3 and Jess 7.1p2 on a Windows environment, to test this approach taking the case of a "Diving Fin" as an example, which has been also solved by TRIZ experts.

Even if the use of diving fins becomes popular, there are still some problems with them. The divers need to make a great effort to push the water, which often makes them tired. Generally, the diving fin should be small for offering minimal resistance to water in order to minimize the effort of the diver and, at the same time, it should also be big in order to push water more efficiently. As a result, there is a contradiction between "Ease of manipulation" and "Kicking efficiency".

As shown in Fig.4(a), there are two interactions between the Tool "diving fin" and the Object "water", that is, a useful one is "push" and a harmful one is "resist". The harmful interaction should be minimized.

![Fig. 4. The problem model of the case of a "Diving Fin"](image-url)
TRIZ experts usually start to solve this problem using the knowledge source of inventive principles, by setting a technical contradiction and using the appropriate tool to use them. The proposed inventive principle to use is **IP36: Phase transitions**: Use phenomena occurring during phase transitions. The abstract description of this principle makes difficult to experts to find a detailed solution for this case.

Our approach proposes, then, to find heuristic solutions using other semantically similar knowledge sources. For example, the most similar inventive standards for **Inventive Principle 36** are selected (Table 1).

<table>
<thead>
<tr>
<th>Inventive Principle</th>
<th>Inventive Standard</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Inventive Principle 36</strong></td>
<td><strong>Inventive Standard 1.1.2</strong> Introduce additive in the S1 or S2 enhancing controllability or imparting the required properties to the SFM; <strong>Inventive Standard 1.2.4</strong> Add a new field (F2) to neutralize the harmful effect (or transform the harmful effect into another useful effect);</td>
</tr>
</tbody>
</table>

Then, ontology reasoning is launched to find concrete solutions to the case. Rules associated to **Inventive Standard 1.1.2** include:

\[
\begin{align*}
\text{Problem Model}(x) & \land PM\text{ isSolvedBy}(x, y) \land IS1.1.2(y) \\
& \land PM\text{ hasFieldNum}(x, 1) \land PM\text{ hasToolNum}(x, 1) \\
& \land PM\text{ hasObjectNum}(x, 1) \rightarrow PM\text{ hasTool}(x, \text{AdditiveSubstance1})
\end{align*}
\] (5)

\[
\begin{align*}
\text{Problem Model}(x) & \land PM\text{ isSolvedBy}(x, y) \land IS1.1.2(y) \\
& \land PM\text{ hasFieldNum}(x, 1) \land PM\text{ hasToolNum}(x, 1) \\
& \land PM\text{ hasObjectNum}(x, 1) \rightarrow PM\text{ hasObject}(x, \text{AdditiveSubstance2})
\end{align*}
\] (6)

After reasoning, as described in Fig.4(b), an "AdditiveTool" or an "AdditiveObject" are added to the problem model.

![Fig. 5. The concept of tubular shear thickening fins](image)

For example, using a shear thickening liquid as an "AdditiveTool", the concept of tubular shear thickening fins is proposed as shown in Fig.5. This concept
is a truly inventive concept, since there are no existing widely distributed and produced products in industry based on this principle (only prototypes, patented fabrics, dampers in the truck industry, etc.).

5 Conclusions

In order to facilitate the use of the TRIZ knowledge sources, this paper explores a heuristic TRIZ problem solving approach. Firstly, TRIZ users start solving inventive problems with the TRIZ knowledge source of their choice. Then other similar knowledge sources are used according to a calculation of semantic relatedness. Finally, heuristic solutions are returned using ontology reasoning.

Perspectives of future work include improving the semantic relatedness methods to make the similarities among the TRIZ knowledge sources more accurate and exploring a more comprehensive rule base for the TRIZ knowledge sources ontologies. On the short term, our research will focus on facilitating the transformation of the obtained generic heuristic solutions into specific solutions based on use of the physical effects knowledge source.

References