UNIVERSITÀ DEGLI STUDI DI BERGAMO

Corso di Dottorato in Meccatronica, Informazione,

Tecnologie Innovative e Metodi Matematici- XXVIII CICLO

Early design Decision-making and Problem solving

Relatore:

Chiar.mo Prof. Righettini Paolo

Correlatore:

Chiar.mo Prof. Russo Davide

Tesi di Dottorato

Stefano DUCI

Matricola n. 1002540

ANNO ACCADEMICO 2015/2016

INDEX

INTRODUCTION

A problem solving activity changes with the type of problem, with the ability of individuals as well as with external conditions. Accordingly, problem solving strategies should be adapted for specific types of problem. Instead, in the field of creative problem solving, tools and methodologies are kept as general as possible and often do not even clarify the boundaries of their applicability. Thus, these methodologies often need a strong customization before using them in an industrial context.

For instance, TRIZ (theory of inventive problem solving) is organized as a toolbox. The user has to select and adapt the tools to the specific problems that he is trying to solve. A proper selection is, in fact, an adaptation of the methodology to the problem and cannot be easily performed by beginners. More in general, an effective customization of problem solving methodologies is not a trivial activity and often needs a long time to be mastered.

An ideal problem solving method should be capable of addressing most problems with minor customizations and without losing its specificity. At the same time, it should effectively integrate known problem solving strategies, such as heuristics, analogies, root cause analysis, etc. Such a method does not yet exist in literature. Thus, the basic idea is to select a method among the available ones, clearly determine its boundaries and extend them with proper customizations.

With these premises, among the overwhelming number of methods that exist in literature, the author selected Spark after a qualitative and partially subjective evaluation. Spark is a problem solving methodology conceived by adapting TRIZ for industrial product development, and it is the result of ten years of experience with Italian companies. Unlike TRIZ, Spark contains a set of tools which are ordered in a step-by-step methodology and includes marketing aspects and patents analysis.

In order to set the boundaries of a problem solving methodology it is necessary to identify what types of problems exist. In literature problems classification is marginally treated by authors in different fields. In this dissertation, I propose a unified classification for technical problems, by summarizing and extending the present state of the art in the fields of education and creative problem solving.

This classification is propaedeutic for a critical analysis of Spark. In fact, Spark is especially suited for problems that involve the generation of conceptual alternatives for the improvement of existing products, while it lacks specificity for decision-making and other problem types. Furthermore, since Spark inherits TRIZ tools for idea generation, such as the 76 standard solutions, the 40 inventive principles and the separation principles, it shares some of their shortcomings. Specifically, given a type of problem, it is not clear how to select one of these tools instead of another and they are sometimes overlapping and inconsistent.

Firstly, this dissertation deals with the adaptation of the second step of Spark for two decision-making problems: the definition of an innovation strategy and the selection of a technology among a series of alternatives. For both types of problems, the management of requirements has a central role. Specifically, I treat the aspects related to the alignment of R&D and problem solving with marketing requirements. Existing methodologies were developed to organize and rank product requirements in order to plan a reliable innovation strategy. The main difficulties of these methodologies are the transformation of customers' needs in technical requirements, the subjectivity of the evaluation, and the strong relation between requirements and the adopted technological solutions. The most structured methodologies use QFD (quality function deployment), sometimes in combination with other design methods. An easier and faster way is based on the evaluation of each product requirement by importance and satisfaction values. A procedure is proposed to make the evaluation of importance and satisfaction a more robust and consistent process. The generation of the innovation strategy with the proposed procedure has been tested for three products in a multinational company. A similar procedure is proposed for the evaluation and selection of the most promising system or technology.

Secondly, I propose an adaptation of the idea generation step of Spark for problems of unsatisfactory actions (e.g. those problems that involve insufficient or harmful actions). I analyze guidelines for the generation of conceptual solutions. Then, I define a practical ontology that allows the organization of the information contained in existing guidelines in accordance with a type of problem and make them suitable for a software implementation. This ontology is used to reorganize the 76 standard solutions for problems of unsatisfactory actions. A particular use of the new system of standards is presented to systematize the fourth step of problem formulation.

The present document has been structured as follows:

Chapter 1 describes the state of the art and identifies the main shortcomings of existing problem solving methodologies. This chapter is structured by associating problem solving tools and methods with a finite number of problem solving strategies. Following this, there is a focus on TRIZ and on the identification of research opportunities in the current scenario.

Chapter 2 proposes a new classification for technical problems that is functional to the identification of the boundaries of applicability for existing problem solving methodologies.

Chapter 3 presents a problem solving methodology called Spark. Spark is presented by highlighting its steps and expected outputs using an explanatory case study. The new classification of problems is used to clearly state the boundaries of Spark and identify future developments.

Chapter 4 proposes a new methodology for the second step of Spark, for addressing decision-making problems such as the definition of an innovation strategy to improve an existing product and the selection of a technology among a series of alternatives. Both uses of the methodology are presented with an explanatory case study.

Chapter 5 proposes a new set of guidelines for the fifth step of Spark. The new set of guidelines is especially adapted for problems of unsatisfactory actions. It is also presented a special use of these guidelines for the step of problem formulation.

Chapter 6 draws the conclusions and main outcomes of this dissertation.

Appendix contains a more detailed state of the art about problems, dealing with the meaning of knowledge, what is a problem, and presenting a collection of problem classifications partially elaborated by the author. This chapter contains useful but not fundamental discussions for the reading of this dissertation.

1 PROBLEM SOLVING STRATEGIES AND METHODS

This chapter serves the purpose of exploring the state of the art and identifying the main shortcomings of existing problem solving methodologies. This chapter is structured by associating problem solving tools and methods with a finite number of problem solving strategies. After that, there is a focus on TRIZ and on the identification of the research opportunities in the current scenario.

There seems to be a finite number of ways with which a solver can creatively or not creatively solve a problem. The collection of problem solving strategies of the following list was developed by elaborating and extending the list of Wang and Chiew [1]:

- Direct facts
- Algorithmic deduction
- Heuristics
- Analysis, synthesis and reformulation
- Sub-problem identification
- Abstraction
- Analogy
- Exhaustive search
- Root cause analysis
- Hypothesis testing
- Proof
- Lateral thinking
- Hill climbing and compromises
- Trial-and-error
- Stimuli and triggers
- Serendipity and motivation

1.1 DIRECT FACTS AND ALGORITHMIC DEDUCTION

Direct facts and algorithmic deduction are well-defined problem solving strategies for well-defined problems. Direct facts are about finding a direct solution path based on known solutions [1]. Direct facts are generally retrieved from memory as facts.

Algorithmic deduction is a known and well defined solution [1], generally obtained from the execution of a known series of operations. Humans perform very little step-by-step algorithmic deduction. In problem-solving, application of an algorithm always guarantees the correct answer, but it requires a well-defined problem.

In order to better explain the differences between these two strategies, we can use an example. When trying to solve an addition problem, such as 6+3, the result can be easily retrieved from memory as a fact [2], that means that we remember that 6+3 equals 9. An alternative is to calculate by applying an algorithm by starting from 6 and recursively adding 1 (three times).

1.2 HEURISTICS AND TRIAL AND ERROR

Although there is not a commonly accepted definition of heuristics [3], we can say that problem solving strategies based on heuristics adopt rule of thumb or the most possible solutions [1]. Heuristics are strategies derived from experience with similar problems, using readily accessible, though loosely applicable, information to control problem solving in human beings, machines, and abstract issues [4]. They are approaches to problem solving, learning, or discovery that employs a practical method not guaranteed to be optimal or perfect, but often sufficient for the immediate goals. An example of heuristic can be "If you are having difficulty understanding a problem, try drawing a picture" [5].

Trial and error could be generally considered a heuristic. However, in this dissertation, they are separated. Trial and error is the process of testing possible solutions until the right one is found. One of the most famous inventor supporting "trial and error" was Thomas Edison [6]. Although its strategy could not be defined as just "trial and error", he developed an incredible number of almost 1100 patents.

1.3 ANALYSIS, SYNTHESIS AND REFORMULATION

Analysis and synthesis are problem solving strategies that reduce a given problem to a known category and then find particular solutions. While analysis and synthesis are more appropriate terms for mathematics, the same concept can be called "reformulation" of the problem when speaking about ill-defined design problems.

According to Einstein "The formulation of a problem, is often more essential than its solution, which may be merely a matter of mathematical or experimental skill. As Bransford [7] argues, Problem formulation (or problem definition) is of unquestionable importance inside a problem solving activity. Real world problems are not pre-defined and the solver must examine the context from which the problem emerged and determine what the nature of the problem is [8]. The problem formulation phase influences the outcomes of a problem solving activity, i.e. once a specific representation of the problem is developed, a particular solution will follow from that representation [9]. Contextually, the importance of problem formulation on achieving original and creative solutions has been stated by several other authors [10–14]. Overall, problem formulation involves the understanding of the problem, the identification of a problem representation, and a reduction of complex ill-defined problems into clearer well-defined problems [8].

In the diverse word of TRIZ, the formulation of the problem is a topic of fundamental importance. The quotation by John Dewey "a problem properly defined is virtually solved" [15] is often cited and universally shared. However, it is still an open issue, and ARIZ is the demonstration of that. In fact, in ARIZ, the most representative and acknowledged tool of the TRIZ theory, the step 0 dedicated to the problem reformulation has been modified many times until its final elimination.

The first version of ARIZ dates back to 1956, but only in the 1964 version a section devoted to "Clarifying and verifying the problem statement" appeared. It remained unchanged until 1968, when the section related to problem analysis was expanded and supported by techniques for overcoming psychological barriers (Size Time Cost

- STC tool, etc.). In this version the correct problem identification was almost half of the entire algorithm. The versions belonging to the 1970s (ARIZ 71, ARIZ 75, ARIZ 77) had the problem formulation and analysis phases as large and distinct, until obtaining the 1977 version, by successive and gradual changes. ARIZ 77 was based on a single step composed of nine sub-sections, including techniques for reducing psychological inertia, comparison techniques based on existing systems on the market and patents knowledge. In the following versions (82-A, B, C, D and 85-A), the problem formulation stage remained unchanged until version 85-B, where it suddenly disappeared. The section on analysis and reformulation of the problem was eliminated, even though it was considered necessary and useful, because it was probably considered non rigorous compared to the other steps. Also G. Altshuller, the founder and creator of the TRIZ theory, was not able to find a structured procedure for the formulation of the problem.

This lack, in a context of a well-structured and guided theory, could not pass unnoticed and without any consequences. In fact, in following years, many TRIZ specialists have tried to bridge this gap. Immediately after 1985, the suspension of ARIZ developments by Altshuller and the need for a structured step guiding the formulation of the problem was perceived and thus proposed by many of his collaborators [16]. So further versions of ARIZ, containing a section on the analysis of the problem, were developed (such as ARIZ-KE-89/90, ARIZ-SMVA 91, ARIZ92, Ariz.-96SS), up to the first computer programs used to support this phase, such as those made by Ideation, Invention Machine and Iwint [17–19], which help and try to guide the user to the first phase of problem approaching, consisting of information collection and problem formulation.

1.4 SUB-PROBLEMS IDENTIFICATION

Divide and conquer is a problem solving strategy based on the decomposition of the a large, complex problem into smaller, solvable problems. "The ability to identify the general problem and generate the sub problems to be solved is crucial for real-world problem solving" [7]. It is a common strategy used in engineering and management.

Divide and conquer includes working in steps, iterative working and sub-problems design. The positive effects of this strategy are also due to the overcoming human's limitations about memory and information processing. This problem solving strategy is especially evident in design strategies based on functional decomposition [20–23].

1.5 ABSTRACTION

Abstraction is a problem solving strategy that is based on solving the problem in a model of the system before applying it to the real system. It is heavily used in engineering and physics, where models are used to simplify the real word. With the use of mathematical symbolism, for example, mathematics facts and heuristics can be used to solve a problem, such as dimensioning a shaft or calculating the time needed to stop a moving car. Abstraction techniques are used when we perform dynamic simulations, finite element analysis simulations and so on.

The strategy of abstraction is very well described with the schema of figure 1 [24–27]. In the field of creative problem solving (for design problems), abstraction is supported by schemas or models for representing the problem. There are several reasoning schema for design, such as function-behaviour-state [22], TOP model [28], Energy-Material-Signal model [29], Su-field model [24], etc. These reasoning schemas are sometimes used alongside heuristics to trigger solutions in the abstracted model. For instance, the 76 standard solutions [30] work together with the Su-field model; the Energy-Material-Signal (EMS) work with a set of compacted standard [31–33]. Synectics [34] suggest to analyse the problem with analogies and metaphors.

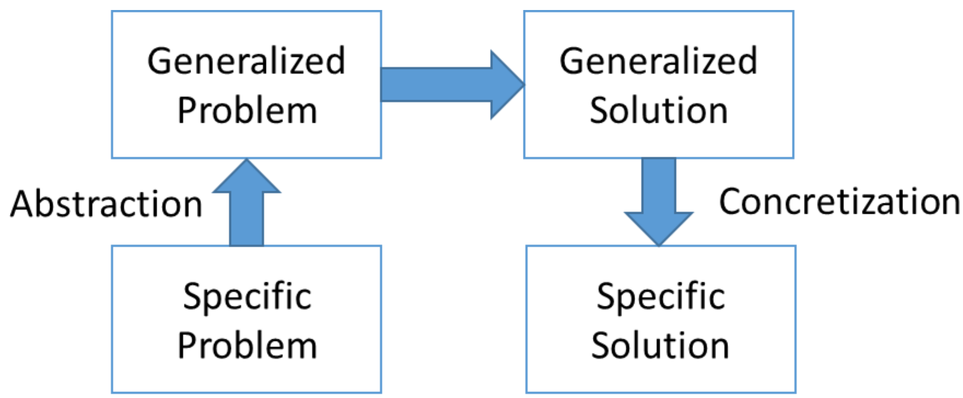


Figure 1. Abstraction for problem solving.

1.6 Analogy

Analogy is a problem solving strategy based on using or adapting an existing solution that solves an analogous or similar problems [35].

In design problems, this strategy is called design-by-analogy, an ideation or problem solving method based on analogies between products [36]. The growing interest in this topic is well demonstrated by the increasing number of publications on the subject and its importance in both problem solving and creative idea generation has been stated by many [37,38].

The concept of analogy, as well as the concept of similarity, are broad and must be intended in a design context [39]. We can interpret a similarity as the link between two products based on certain shared attributes of the products, shared relations between individual parts of the products, or shared functionality. In accordance with the latter, two products can be considered similar if they share one or more functions. For example, an escalator and a lift share the same function "lifting a person". This kind of similarity is not related to the particular structure that performs the required function. Extending the concept of similar products to the natural world, a product can be intended in a very broad sense, so that a volcano that "lifts the ashes" will be somewhat similar to an escalator that "lifts a person", having the same abstracted function "lifting a solid".

Design-by-analogy has been classified in two classes [40]: within domain and cross-domain design-by-analogy. The first one if the inspiration is taken from similar products on the same domain; the second if the products are taken from other domains.

Kolodner [41] highlighted the most representative problems of design-by-analogy, which are relevant for expert and accentuated for novices. These problems are mainly related with encoding previous solutions, retrieving these previous solutions and transferring the knowledge by mapping the relevant pieces of information. Another research was conducted by Vattam and Goel [42], finding three main challenges: findability, recognisability and understandability. In other words, from a designer point of view, the major difficulties are the identification of the best candidate for the knowledge transfer and the knowledge transfer itself.

In order to support the knowledge transfer, Nagel et al. [43] used the energy, material signal model [20] in a bio-inspired design context. Specifically, they obtained a conceptual design of a solar panel inspired by lichen. Before that, Nagel et al. [44] already explored the use of functional modelling for other three case studies: Armadillo, Puffer Fish and Housefly.

In order to support the identification of the best candidate for knowledge transfer, several information retrieval strategies have been used. Among them, WordTree [45] is a design-by-analogy method that uses WordNet [46] vocabulary to manage language relationships between words and facilitate the identification of different domains for idea generation.

AskNature is a database of elaborated information about biological systems that have been manually classified in accordance with their functionality [47]. In this way, AskNature can be queried with functional terms in order to find an appropriate candidate for design-by-analogy.

Nagel et al. [43] presented an engineering-to-biology thesaurus to support the search of functionalities into a biological field, with the aim of using biologically-related verbs to search textual sources of information. Verhaegen et al. [36] described an algorithm and methodology that, through analysis of term occurrences in patents, extracts information concerning the characteristics of products, and applies this in order to identify possible candidates for design-by-analogy exercises.

1.7 Exhaustive search

The exhaustive search is a problem solving strategy that involves the systematic search for all possible solutions. An exploration of all the possible solutions to a multi-dimensional, non-quantified complex problem can be done with morphological analysis [48]. The generation of the maximum number of solutions is also supported with brainstorming [49], where a group of people strive to find a conclusion for a specific problem by gathering a list of ideas spontaneously contributed by its members.

Exhaustive searches are also supported by recent information retrieval techniques, such as FOS (function oriented search) [50] or KOM (Knowledge Organizing Module) [51–53].

1.8 Root-cause analysis

The concept of cause and effect has been deeply described by Sloman [54]. Although causal and effect assume various roles in everyday language. Some definitions are useful to understand their meaning:

- Causal relations relate entities that exist in and therefore are bounded in time. Such entities can be called events or classes of events, because the word “event” suggest the transient character of causes and effects.
- To say that A caused B means: A and B both occurred, but if event A had not occurred (and B had no other sufficient cause), B would not have occurred either.

This awareness on cause and effect bring us to the consideration that, causal relation doesn’t merely imply that events happened together, but that there’s some generating mechanism that produces an event of one type when engaged by an event of another type. So, in some other worlds where the event has not been engaged by the event, the effect may not have occurred. These considerations are very important to distinguish mere correlation from causal relation. A causal relation has the further requirement of counterfactual dependence [55].

Cause-effect analysis is one of the most powerful and widespread ways of describing how a system works. This is confirmed by the fact that it is harnessed by several methods including Failure Modes and Effects Analysis - FMEA [56,57], the Theory of Constraints [58] and many others, such as Ishikawa diagrams [59], Kepner-Tregoe method [60], Fault Tree Analysis [61], Why-Why [62], etc. [57,63,64]. It is interesting to note that after the death of Altshuller - the father of TRIZ, almost all recent efforts to prepare a problem statement module for ARIZ have been based on Root Cause Analysis as in RCA+ , in OTSM [65,66], or on the general theory of innovation - GTI [67], and on combination with TRIZ and other methodologies, such as Axiomatic Design [68] or Failure mode analysis [56,57], etc.

1.9 Hypothesis testing and Proof

A strategy for problem solving which is widely used in research activity is hypothesis testing, which consists on assuming a possible explanation to the problem and trying to prove (or, in some contexts, disprove) the assumption [1].

On the opposite, proof is a problem solving strategy which try to prove that the problem cannot be solved (or a hypothesis is not true). The point where the proof fails will be the starting point for solving it [1].

1.10 Lateral thinking

Lateral thinking is a problem solving strategy based on creating new points of views on problems. It is to raise new questions, new possibilities, to regard old questions from a new angle, and this requires creative imagination [69]. Lateral thinking in about the creation of different social perspectives [70], artificially generating new points of views and approaching solutions indirectly and creatively [71–73]. A tool for lateral thinking is, for instance, the STC (size-time-cost) [74], which is a TRIZ tool that help you in lateral thinking by suggesting to exaggerate or minimize resources (such as "how would you realize this system if you had immense space or minimal space?).

1.11 Hill climbing and compromise

The problem solving strategy called hill climbing in based on choosing an action at each step to move closer to the goal. A simple example of the application of this strategy is the Means-ends analysis [75]: the problem solver begins by envisioning the end, or ultimate goal, and then determines the best strategy for attaining the goal in his current situation. Inside TRIZ, this strategy is used when trying to define the ideal final result of our current situation [24–27].

An opposite problem solving strategy consists on "compromises". Starting from an ideal but not feasible goal, the solver proceeds step-by-step toward a feasible solution. The solver is so accepting a compromise. The differences between "hill climbing" and "compromise" is graphically described in figure 2.

1.12 Stimuli and triggers, serendipity and motivation

Stimuli and triggers are problem solving strategies that work by proposing stimuli (textual, audio, visual) to the problem solver. This kind of strategy differs from heuristics since the trigger is based upon random associations or stimuli that are weakly associated with the problem.

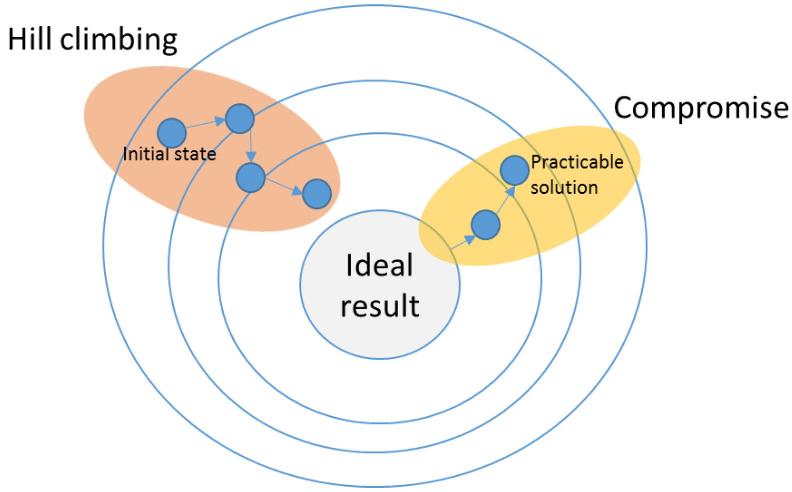


Figure 2. Hill climbing and compromise.

A tool for problem solving can serve the user as source of inspiration. The influence on the user may occur in a systematic way, when designers actively search for inspiration, or even unconsciously or by chance [76]. Gonçalves et al. [77] tried to understand which kind of idea-generation method is preferred by designers, while Chulvi et al. [78] studied the differences on design outcomes. An important group of triggers for idea generation are represented by visual stimuli, and their impact on design was explored by several authors [79,80], finding a general positive effect on creativity. The synergic effect of a combination of textual and visual stimuli on learning has been confirmed by Schnotz [81], while the effect of the representation of stimuli in problem solving has been addressed by Sakar and Chakrabarti [82]. The use of text as stimuli for idea generation was promoted by several researchers. Goldschmidt and Server [76] explained a positive effect of textual stimuli on originality, but they mentioned no effect in "practicality". Chiu and Shu [83] studied the effect of verbs as stimuli for idea generation, finding a general increment in creativity. With a different prospective, Fantoni et al. [84] started from a functional description of products to obtain a new design method, based on the analysis of functional synonyms and antonyms. The method of focal object, instead, is based on the synthesis of seemingly non-matching characteristics of different objects into something new [33].

Serendipity and motivation can be categorized as another strategy for problem solving, where the solution comes from a special predisposition of the solver in a certain moment. The solver is motivated or simply "in the best mood and best context" to innovate. In this direction, Synectics tried to identify factors in teams and individuals that favor creativity in practical contexts [85].

1.13 PROBLEM SOLVING METHODS

An open debate in the design society is about the overwhelming number of methods that are constantly produced all over the world. There are so many theories, methodologies and tools that is really difficult to select the proper one when needed. There are many methodologies that can support problem solving. In literature, several books contain collection of methods [86–88] that can be probably counted in the order of thousands. Just to give an example, in 2004, the Technology Futures Analysis Methods Working Group (TFAMWG) proposed a TFA collection consisting of 51 methods arranged in 9 "families,": (A) Creativity, (B) Expert Opinion, (C) Trend Analyses, (D) Monitoring & Intelligence, (E) Statistical, (F) Scenarios, (G) Modeling & Simulation, (H) Descriptive & Matrices and (I) Valuing/Decision/Economic. This collection has been modified slightly by Porter [89,90] in 2005 into 48 methods in 13 families (see table 1), adding (J) Roadmapping and (K) Logical/Causal Analyses.

Table 1. Future-oriented Technology Analysis Methods Families.

FTA Families	Sample Methods
Creativity Approaches	TRIZ, Future workshops, Visioning
Monitoring & Intelligence	Technology Watch, Tech Mining
Descriptive	Bibliometrics, Impact checklists, State of the Future Index, Multiple Perspectives Assessment
Matrices	Analogies, Morphological analysis, Cross-Impact analyses,
Statistical Analyses	Risk Analysis, Correlations
Trend Analyses	Growth curve modeling, Leading Indicators, Envelope Curves, Long wave models
Expert Opinion	Survey, Delphi, Focus groups, Participatory approaches
Modeling & Simulation	Innovation Systems descriptions, Complex Adaptive Systems modeling, Chaotic regimes modeling, Technology Diffusion or Substitution analyses, Input-Output modeling, Agent-based modeling
Logical/ Causal Analyses	Requirements analysis, Institutional analyses, Stakeholder analyses, Social Impact Assessment, Mitigation strategizing, Sustainability Analyses, Action analyses (Policy assessment), Relevance Trees, Futures Wheel
Roadmapping	Backcasting, Technology/Product Roadmapping, Science Mapping
Scenarios	Scenario Management, Quantitatively based scenarios
Valuing/Decision-aiding/Economic Analyses	Cost-Benefit Analysis (CBA), Analytical Hierarchy Process (AHP), Data Envelopment Analysis (DEA), Multicriteria Decision Analyses
Combinations	Scenario-Simulation (gaming), Trend Impact Analysis

A so big number of methods is easily identifiable for many other purposes than just TFA. Researchers are wondering how to deal with this quantity of methodologies, where just a few of them are specifically developed for specific types of problems. The result is confusion among users on which method should they select for their current situation. Furthermore, the introduction of new methodologies introduces new terms, new concepts which are very often overlapping with existing ones. After all, the mechanisms (or strategies) with which we perform problem solving seems finite in number (see section 7.2.1). In the author opinions, there is the need to understand the mechanisms underneath the existing methodologies, so as to organize them in an ordered framework. Specifically, a valuable advance on creative problem solving can be identified with two drives: a) the unification of existing problem classifications and eventually associating existing methodologies to specific classes, so as to know where and when a certain tool or method should be used; b) the unification of different tools and methods that share similar problem solving strategies.

1.13.1 TRIZ

TRIZ is probably the most used theory for creative problem solving in industry [91–93]. It is the acronym of the Russian "Theory of Inventive Problem Solving (TIPS)" [24] Russo and Spreafico [91] examined a wide number of publications where TRIZ were used in industrial case studies, highlighting a wide range of fields (see figure 3). Similar analysis were performed in 2009 [92] and 2013 [93] with similar results.

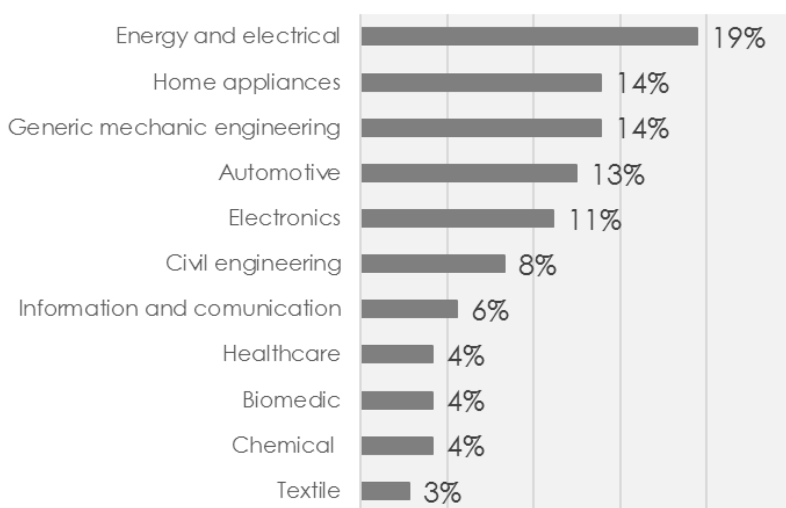


Figure 3. Survey on the uses of TRIZ in different fields of industry (from [91]*).*

TRIZ is a complex methodology that includes a wide range of tools. The most used are [91–93]:

- Contradictions [24–26]
- 40 Inventive principles [94,95]

- Evolution laws [96]
- Ideal final result [25,26]
- Su-field modeling [30,97,98]
- Standard solutions [30,97–99]
- Separation principles [25,26]
- TRIZ functional analysis [27]
- Multiscreen (or nine windows) [24]
- Effects and resources [24–26]
- ARIZ [16,100,101]
- Smart little people [102]
- Size-time cost (STC) [74]
- Tool-main useful function-object (or tool-object product) [28]

The frequency of use of these tools is shown in figure 4. TRIZ has been customized and adapted by many authors, some examples are the Unified Structured Inventive Thinking [103] (USIT) and Advanced Systematic Inventive Thinking [104]. Usually, derivate methodologies claim similar effectiveness and less learning time when compared to the original theory. TRIZ is also often integrated with other methods [91], creating variants of the methodology or totally new approaches (see figure 5).

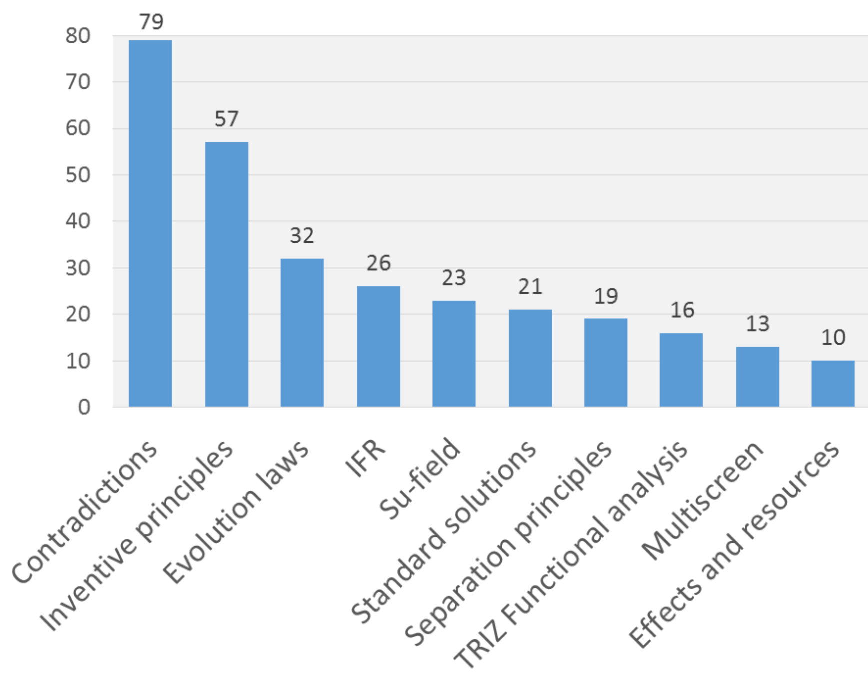


Figure 4. Survey on the frequency on the use of TRIZ tools in industry (from [91]).

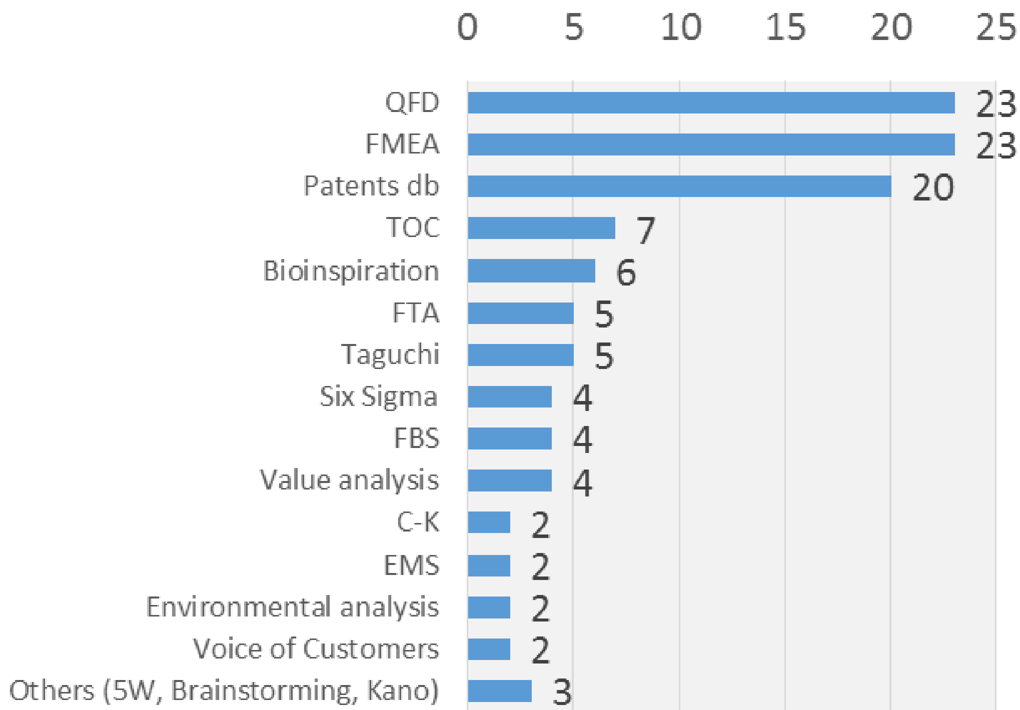


Figure 5. Integration of TRIZ with other methodologies (from a proceeding presentation [91]*).*

Despite the implementation of TRIZ in big multinational companies, such as Samsung and Intel, its use is yet limited. The first shortcoming is probably due to the long time needed for its proper learning, which can take longer than a year. Of course, this kind of timespan is not compatible with the expectation of companies. Savransky states "It is impossible to solve high degree of difficulty problems with only the knowledge of TRIZ heuristics and instruments without learning the whole TRIZ methodology and sharpening solver skills with real and educational problems" [26]. The causes of the difficulties in learning TRIZ are here summarized by the author:

- The methodology is presented in form of toolbox (except for ARIZ). There is not a specific way to choose when and where a certain tool should be used.
- ARIZ, the most structured tool, starts from an already formulated contradiction. There is not an accepted way to effectively define that contradiction.
- Some tools are overlapping. 40 Inventive principles and standard solutions for example share the same purpose of idea generation, but they are presented and used in totally different ways.
- There is, as in every field, a big amount of tacit knowledge.
- Some people are skeptical about the possibility of systematizing creative processes.

1.14 LIMITATIONS AND CRITICALITIES OF THE STATE OF ART

The previous paragraphs served the purpose of highlighting the current state of methodologies and tools for problem solving. As a result of this analysis, the author identified some limitations and criticalities in the current state of the art:

- Methodologies and tools for problem solving do not clearly states where and when they should be used (with just few exceptions).
- There are a huge number of methodologies and tools that can be used. They are kept general to be applied in many situations, but the result is a huge number of customizations to make them usable in specific contexts. This lead to confusion among potential users.
- Problem solving is rarely integrated with marketing, commercial and operational aspects of companies.
- When proving the effectiveness of solving methods, it is common to compare them with no method (something is often better than nothing).
- When comparing two methodologies, instead, since they usually do not clearly state the boundaries of their applicability, the effectiveness will depend on the type of problem. Methods are so compared without consistency. This lead to the assumption that methods will be comparable only if a consistent problem solving classification is developed and accepted. A series of random problems should be classified accordingly for testing the methodologies.

The analysis of the state of the art allowed the identification of valid research opportunities that would bring added value in the research scenario:

- Developing a unified classification of design problems.
- Sorting existing tools in accordance with specific problem types.
- Unifying tools in accordance with their problem solving strategy.

In this dissertation, the previous points have been addressed as follows:

- Definition of a new classification of technical problems.
- Identifying problems that can be addressed with Spark, a methodology used and developed at the University of Bergamo.
- Customization of the use of Spark for the definition of an innovation strategy and the selection of a technological alternative.
- Customization of Spark for problem of unsatisfactory actions in the idea generation step.

2 A NEW UNIFIED CLASSIFICATION FOR TECHNICAL PROBLEMS

This chapter proposes a new classification for technical problems that is functional to the identification of the boundaries of applicability for existing problem solving methodologies. This is considered as a necessary step before the development, improvement or comparison of any problem solving method.

Problem solving varies with the type of problem, in the way they are presented or represented, and in their elements and interactions among them (see appendix in section 7.2.1). Since problems strongly differ from one another [105], the identification of the type of problem can help in the customization of tools and procedures. Literature on design theories and methods is very concentrated on how a method can be evaluated but less concentrated on when a method should be used. As a result, a unified classification of technical problems does not yet exist and the identification of the boundaries of a certain methodology becomes difficult if not impossible.

In this dissertation, I propose a unified classification for technical problems by summarizing and extending the current state of the art in the fields of education and creative problem solving. Some authors from the world of education and artificial intelligence, such as Jonassen, Newell and Simon (section 7.3.1) divides problems on the basis of structuredness, complexity and abstractedness. J.W. Getzel [106] used characteristics that are almost entirely related with individual differences (sections 7.2.1 and 7.2.2) while other classifications are entirely based on complexity (section 7.3.4). However, these classifications did not find any application in industry. Problem classification in industry, instead (section 7.3.5), is more related with the elements of the problems and with goals, distinguishing between product and processes (such as [107]) and between small improvements and big improvements (such as [108]).

Other classifications have been developed from various experts of inventive problem solving, often without explicitly mentioning the classification itself but by using it as filter for the selection of problem solving tools. Among them, some classification are based on distinguishing the technical parameter to be improved (section 7.3.7), the amount of changes that are acceptable for the system (section 7.3.8) and the Su-field conditions (section 7.3.9). Eventually, classifications of problems can descend from functional descriptions based on language (section 7.3.10), TRIZ functional analysis (section 7.3.11) and also from the concept of contradiction (section 7.3.12).

The unified classification for technical problems is reported in table 8. The boundary line between one problem and another is not clearly defined and problems are not independent. The distinctions between them are sufficient to justify a customized methodology for each of them. The classification is not comprehensive, but it includes most of the classes mentioned in the literature review of section 7.3. Referring to Jonassen's [109] proposal, this dissertation is limited on Ill-Structured technical problems, i.e. troubleshooting, diagnosis-solution problems, situated-case problems, decision-making problems and design problems. The problem types are named in accordance with the main goal of the problem solving activity. It is important to notice that the resolution of a certain type of problem may include the resolution of other type of problems. For example, the "improving an existing product" problem may include "improving requirement "X" of a product" as one of the sub-problems.

Design problems ends with the creation of a system (artifact, product or process) in the real world. They were described also by Jonassen [109] and include design problems as intended by Koller [110]. Examples of this type of problems are: creating a new engine, creating a new boat, creating a new personal computer and so on.

Conceptual design problems identify a group of problems that ends with one or more conceptual alternative solutions. Conceptual design problems are divided in accordance with different goals:

- *Improving an existing product*: when we want to improve an existing system, i.e. obtaining a series of alternatives solutions that should be better than a reference product. Implicitly, in this type of problems we do not know the requirements that have to be improved. Examples of this type of problem are: improving a pen, improving a hairdryer, improving a boat, improving a lighter and so on.
- *Improving requirement "X" of a product*: when there is a system and we know which requirement or requirements have to be improved. This type of problems is more structured than "improving an existing product" and some examples are here listed: increasing the brightness of a screen, reducing the time needed for maintenance of a switchgear, reducing the energy consumption of a dishwasher and so on.
- *Improving an existing process*: when there is a process and we know which requirement or requirements have to be improved. Implicitly, in this type of problems we do not know the requirements that have to be improved. Examples of this type of problem are: improving a manufacturing process for the production of water bottles, improving a manufacturing process for the production of steel tubes, improving a process for wastewater treatment and so on.
- *Improving parameter/requirement "X" of a process*: when there is a process and we know which requirements have to be improved. Examples of this type of problems are: reducing glitches in a manufacturing process, reducing energy consumption in a manufacturing process, reducing waste in a manufacturing process, reducing the number of rejects and so on.
- *Solving technical contradictions*: when we want to improve both a part (or parameter) and another part (or parameter) of a technical system, but If by

certain methods one improves one of the parts (or parameter), the other one (or other parameter) deteriorates in the process [24]. Examples: I want to reduce weight without losing the mechanical strength of a device, I want to increase illumination intensity without increasing the temperature of a light.

- *Solving physical contradictions*: when the key subsystem (name) should be or has ("positive" parameter), in order to (the first requirement for the tool), the key subsystem (name) should not be or not have ("negative" parameter), in order to (the second requirement for the tool) [24]. Examples:
- *How to perform an action or function*: when we want to know how to realize a certain action or function. Examples are: how to rotate an object, how to increase the temperature of a stone, how to reduce the diameter of a tube and so on.
- *Unsatisfactory actions or interactions among elements*: when we have two elements with an interaction or action which is not satisfactory since it leads to an unsatisfactory transformation. Examples of this type of problems are here listed: a hummer hit a finger, obtaining a sore finger that I did not want; a magnet is pulling an iron core, but the core do not accelerate sufficiently; etc.

Finding the unknown causes and effects identifies a group of problems where the outputs are one or more cause and effects relationships which were not known before. This group can be considered as an extension of Jonassen's [109] troubleshooting problems and diagnosis-solution problems. It includes the following type of problems:

- *Finding the unknown reasons/causes of something that happens (emerges) in our system (that is unknown)*: when the goal is to understand the causes of an effect in a physical system [111]. In this class we include emergency problems [107], and also "*Finding the unknown reasons/causes of a disparity between expected and real results*" [107]. Examples are: finding the cause of a failure for a boiler tank that is broken, finding the cause of a missing color in a printed photo, finding the cause of corrosion in apparently safe conditions and so on.
- *Finding an unknown function of an element in a system* (from [112]): when there is a system or part of a system which is done in a certain way and we want to understand why. Examples: finding why competitors changed the form of a part of the system, finding why plastic water bottles have certain grooves and so on.
- *Failure prevention*: when the goal is to find possible undesired unknown effects and causes in our system. Failure prevention is mentioned in [108] and is one of the main purposes for the development of FMEA [56]. Some examples are: failure prevention of a car, failure prevention of a boiler and so on.
- *Finding unknown information about physical and chemical processes* [107]: when the objective is to find information which is not available in literature, such as the melting temperature of a new material, the correlation between

temperature and Young module of a material, finding if light is capable to provide force etc.

Forecasting (predictions) is to make predictions of the future based on past and present data and analysis of trends. They are identified as problems in industrial contexts [113]. Predictions is a group of problems that includes a series of problems, the following list is not comprehensive:

- *Predicting the evolution of a system:* when the objective is to understand with a certain degree of confidence, what will be the future characteristics of a system [113]. For examples: predicting which technology will be used in 50 years to cut tubes, predicting which technology will be used to store energy in 10 years, etc.
- *Finding possible future applications for a system*: when we have a system that works for a certain application and we want to know if there will be other applications of this system or technology in the future. Or we have discovered a new technology and we want to understand possible applications. Examples: where laser cutting will be used, in which applications that use alternative motors will have direct current motors, etc.

Decision-making problems (ranking or selecting) has the objective of selecting or ranking, among a series of alternatives, one or more of these alternatives, or we sort them in accordance with some criteria to identify the best one. It resembles the homonym class in [109] and it contains a various set of problems. Here follows a non-comprehensive list:

- *Selecting between a series of alternative systems*: when there are a series of conceptual alternatives that should be evaluated to select the best. For example, given a series of alternative technologies for measuring air pressure, I may want to select the most promising one.
- *Selecting or ranking the requirements of a product or process that should be changed:* when you have a list of requirements of a product and you need to select one or more of them to concentrate the following efforts for improvements. This type of problems is also called "definition of an innovation strategy". Examples are: should I improve weight or battery life of a cellphone? Should I reduce maintenance time or reliability of my product?
- *Deciding if a product should continue to stay on the market or it should be abandoned*
- *Selecting the number of versions/sizes that should be part of a product family*
- *Selecting on which product in a company to invest and innovate*

For conceptual design problems, it can be useful to further divide problems in accordance with a functional decomposition. A function can be easily described with a proper level of detail with a verb, an object, one or more parameters/adjectives of the object and one or more parameters/adjectives of the verb (see an example in figure 6).

Table 2. A unified classification for technical problems.

Groups	Problem types	
Design problems	Creating a new system, artifact, process or product	
Conceptual Design problems	Finding how to perform an action or function	
	Improving an existing product	
	Improving requirement "X" of a product	
	Improving an existing process	
	Improving requirement "X" of a process	
	Solving technical contradictions	
	Solving physical contradictions	
	Improving unsatisfactory actions or interactions	Insufficient actions or interactions
		Excessive actions or interactions
		Harmful actions or interactions
		Optimal actions or interactions
		Missing actions or interactions
Finding the unknown causes and effects	Finding the unknown causes of something that happens (emerges) in a system	
	Finding an unknown function of an element in a system	
	Finding unknown information about physical and chemical processes	
	Finding possible undesired unknown effects and causes in a system (failure prevention)	
Forecasting (Predicting)	Predicting the future characteristics of a system	
	Finding possible new applications for an existing system	
Decision-making Problems (ranking or selecting)	Selecting or ranking a series of alternative systems	
	Selecting or ranking the requirements of a product or process that should be changed	
	Deciding if a product should continue to stay on the market or it should be abandoned	
	Selecting the number of versions/sizes that should be part of a product family	
	Selecting on which product in a company to invest and innovate	

Examples of subclasses that can take advantage of a functional description are:

- How to perform an action or function: how to move an object, how to melt a solid, hot to rotate an object…

- Improving parameter/requirement "X" of a product: decrementing weight of an object, incrementing temperature of a device, incrementing velocity of a solid…
- Improving parameter/requirement "X" of a process: decrementing glitches of a process, decrementing environmental impact of a process, decrementing energy absorption of a process…
- This functional description is inspired from the functional basis [46,114]. According to Stone and Wood [114], its adoption would allow different designers to share information at the same level of detail, to generate repeatable function structures, and to compare functionality of different products for idea generation purposes. Furthermore, an increasing number of databases are organized in accordance with a functional description similar to the one in figure 6. Among them, databases of effects use functional filters called pointers to effect while databases for design by analogy, especially those for biomimetic, developed a complex taxonomy that can be elaborated to fit in a functional decomposition (section 7.3.10).

Nowadays, the choice of the verb and object have some disadvantages. Different databases have differences in language and they have customized set of verbs and objects. This forces the user to adapt the formulation of a problem to the type of database that he wants to use. For this reason, in order to automatically or manually relating verbs, flows and parameters among different database, I suggested a table of correspondences (section 7.3.10.1) in an attempt to unify different databases and making them less-dependent to differences of formulation. The results of this work are shown in the appendix in section 7.3.10.1.

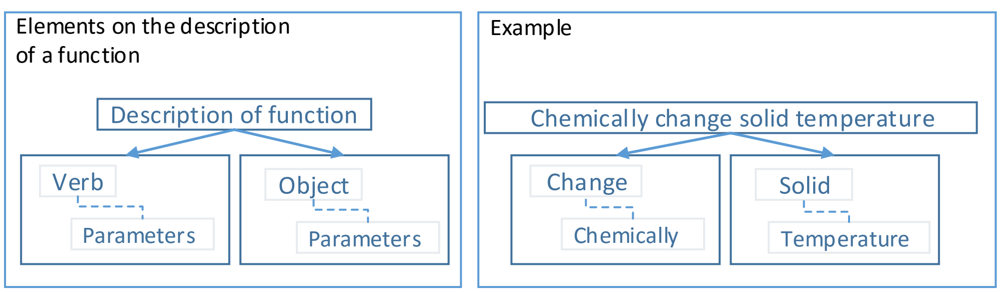


Figure 6. Decomposition of the description of functions (e.g. chemically change the temperature of a solid).

A special type of problem contained in conceptual design problems is the contradictions. In the following paragraph, I explain how contradictions can be further classified if contradictions are formalized as follows:

- Technical contradictions (TC): if one part (or one parameter) of a technical system is improved by any known method, some other part (or some other parameter) will be inadmissibly impaired [25,26].

- Physical contradiction (PC): The key subsystem (name) should be or has ("positive" parameter), in order to (the first requirement for the tool), the key subsystem (name) should not be or not have ("negative" parameter), in order to (the second requirement for the tool) [25,26].

Both definitions contain two requirements or parameters that we want to improve. These parameters are often called evaluation parameters (EP1 and EP2). In a technical contradiction there is not a precise parameter that relates EP1 with EP2, while a physical contradiction contains a third parameter, usually called control parameter (CP), that relates EP1 with EP2.

Technical contradictions were classified by Altshuller [24] in a matrix of contradictions, where columns and rows contained general description of parameters to be improved. In fact, technical contradictions can be classified in couples of parameters to be improved. Each of these technical parameters EP1 and EP2 may be described with a functional description as described in figure 6.

Physical contradictions contain also a third parameter (CP). The control parameter has been already used in [26,115,116] to create sub-classes of physical contradictions. Furthermore, since physical contradictions contain EP1 and EP2 in their formulation, they may inherit the same sub-classification of a technical contradiction. Eventually, physical contradictions may be sub-classified in accordance with their control parameters and they may be sub-sub-classified in accordance with the functional description of EP1 and EP2.

The classification of technical contradictions already found an application in the matrix of contradiction [24] but with a limited number of technical parameters (functional descriptions). Instead, there is not a practical application for the classification of physical contradictions, i.e. there are not specific tools in literature that provide different suggestions in accordance with the type of physical contradiction.

3 PRODUCT IMPROVEMENTS WITH SPARK

An ideal method for problem solving should address the most types of problems (chapter 2) without losing its specificity. Of course, there is not such a method in literature and it is not trivial to select, among existing methodologies, the most valuable starting point. With these premises the author selected Spark since it is a problem solving methodology that is born by adapting TRIZ for industrial product development, exploiting numerous feedbacks coming from ten years of experience with Italian companies. Unlike TRIZ, Spark contains a set of tools which are ordered in a step-by-step methodology and includes marketing aspects and patents analysis. In this chapter, I explain the main characteristics of Spark, identifying the boundaries of its applicability in accordance with the problem classification of chapter 2.

Spark is a methodology developed at the University of Bergamo (Italy) after ten years of experience on industrial and academic case studies. Although the development of the methodology has taken ten years of testing and slight improvements, it is only in the last three years that it has been formalized in an ordered set of tools. Spark is developed on the basis of TRIZ. Engineering students follow 40 hours of course on TRIZ and Spark at the University of Bergamo. In this context, while working in real case studies, researchers add tools and explicit knowledge to the methodology, facilitating learning and improving effectiveness. New parts and features of the method are so firstly tested by researchers, they are explained to engineering students and they are qualitatively evaluated on students' final exams.

Now, Spark is a mature methodology for solving design problems which has been tested in small and medium enterprises as well as in big multinational companies. The scalability of the methodology allows to address problems in accordance with the amount of time that can be dedicated to a certain project. Two of the main weaknesses of TRIZ are addressed: the lack of marketing considerations on technical problem solving and the lack of an ordered framework (toolbox). Marketing aspects are so introduced with a tool for the management of technical and non-technical requirements (see figure 7). Tools are positioned in an almost sequential order and classical TRIZ tools are adapted and optimized. Being this the first attempt to explain the principles of Spark, and being it a comprehensive methodology, in this dissertation, not all the tools will be treated in detail.

Thanks to the classification of problems presented in chapter 2, it will be possible to clearly state the present boundaries of Spark (3.8). Spark is composed of five main steps plus an introductive one and a conclusive one (see figure 8). Tools inside a certain step share the same purpose. Although steps are sequential, tools inside each step can be used in parallel and their order is just suggested. Tools in different steps which share the same name are considered as different tools since they are used differently.

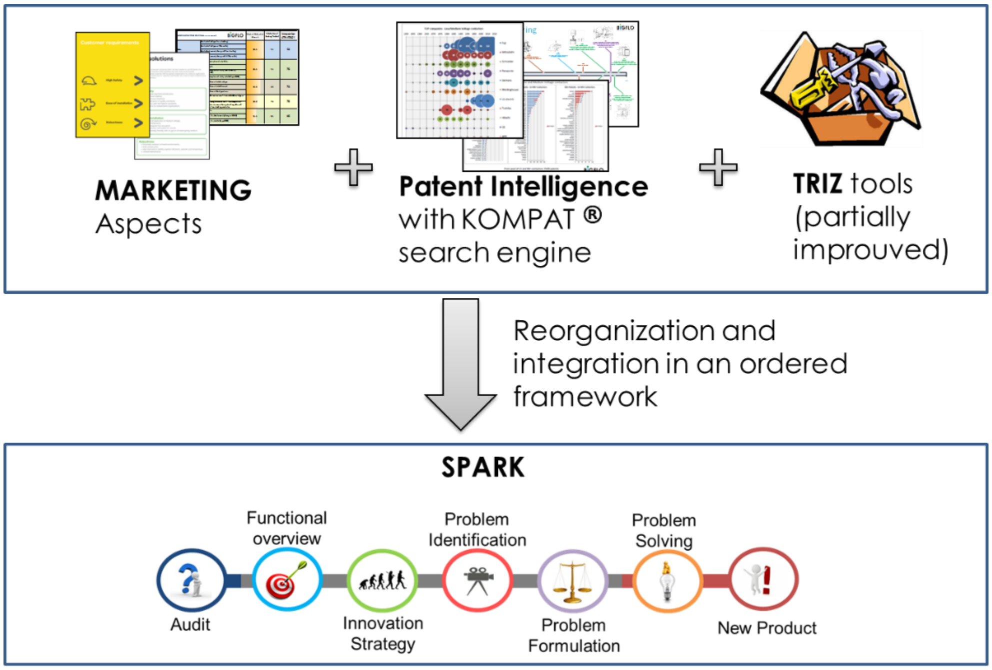


Figure 7. Spark.

The objective of each step will be well described in the next paragraph, but they can be summarized as follows:

1. Functional overview: the goal of this module is to collect and reorganize the information about the function that we want to perform with our system/product.
2. Definition of the innovation strategy: The goal is to define the innovation strategy for our product. On the one hand, tools contained in this step support the decision-making confirming the initial direction of development or changing it towards more radical innovative solutions. On the other hand, they help to quantify the effectiveness of the initial system by comparing it with competitors (ideal or real). Finally, they help the definition of the most important requirement of the new product.
3. Problem identification: the goal is to define in a very precise way all the potential solving directions for the problem.
4. Problem formulation: the goal is to formulate problems in a known form to be solved with one of the idea generation techniques of the next step.
5. Problem solving: the goal of this step is to generate solutions or partial solutions for the problem formulated in the previous step.

The first step called "audit" is a brief discussion with the committee of the problem solving activity. The last step "new product" is simply the presentation of the results to the committee, possibly selecting the most valuable solutions.

The methodology is described step-by-step in the following paragraph. A real industrial case study is used to explain implications on engineering students.

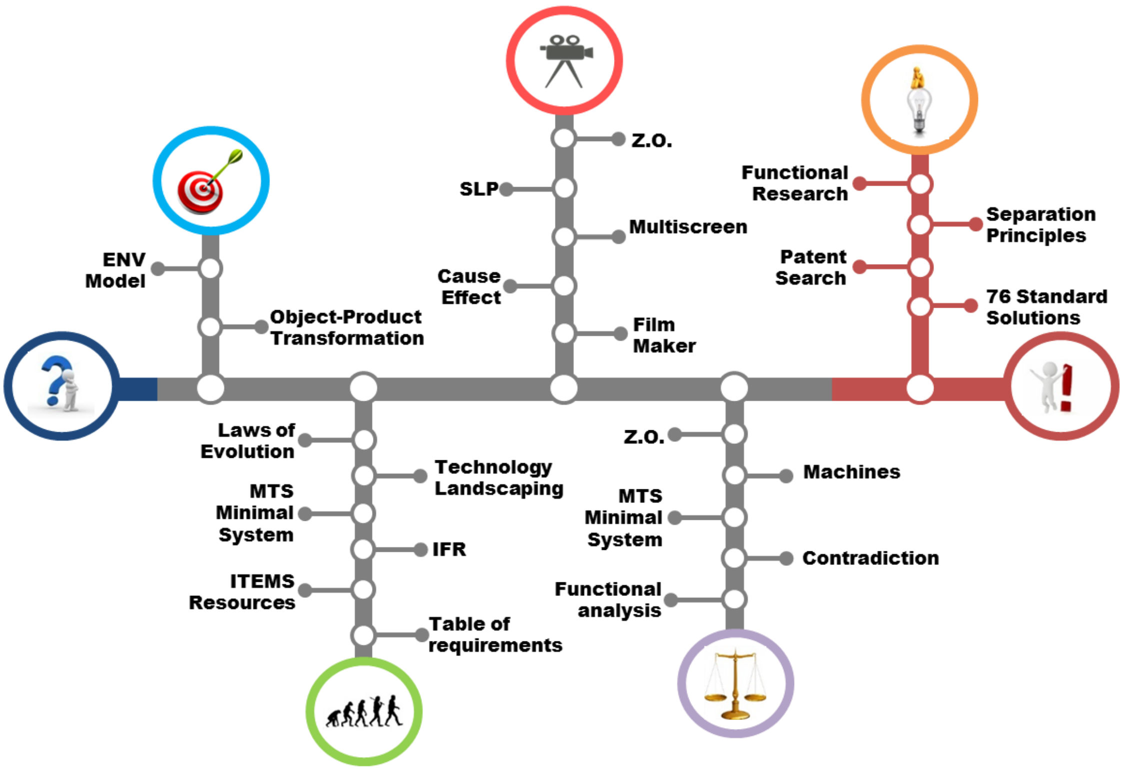


Figure 8. Tools of Spark.

3.1 THE ASEPTIC VALVE PROBLEM

A case study will be used to effectively describe the implications of Spark on engineering students. Students had to prepare a final project for the course "Innovazione di prodotto e processo" (Innovation for products and processes). Students had to use Spark to solve the proposed problem. No specific time constraints were given for the resolution of the problem, but only projects completed in less than two weeks had been analyzed.

First, students received a presentation about the problem (35 slides). Then, the problem was proposed and exposed by a CIO (Chief innovation officer) of a multinational company and students had the opportunity to ask some questions. Given the real interest of the company in the problem, it seems reasonable to assume that students were properly motivated. Here follows a summary of the problem description.

Characteristics of an aseptic valve:

- Aseptic single seat valves are used for the monitored control of fluids in aseptic processing plants. A welded stainless steel bellow is used to hermetically seal the product area from outside contamination.
- The valve is designed for the use in the food, dairy, beverage, pharmaceutical, chemical, and cosmetics industries.

Figure 9. An aseptic valve with bellow (GEA image courtesy).

Some problems:

- Fracture of hermetic sealing due to unfavorable process conditions (pressure hammers, vibrations).
- Inefficient cleaning cycles due to unfavorable housing design.
- Limited theoretical service life due to stainless steel material for hermetic sealing.
- Hermetic sealing method not suitable because of particulates or fibers in the product.

The main questions asked to students were "how an aseptic valve will look like in the future". No particular advice or clues were given to the students on preferable solutions.

3.2 Step 1: Functional Overview

The input of this step is a general description of the problem without any formalization.

The goal of this module is to collect and reorganize the information about the function that we want to perform with our system/product.

This step is assisted by the use of tools such as the object-product transformation and the ENV model that are usually used sequentially (see figure 10).

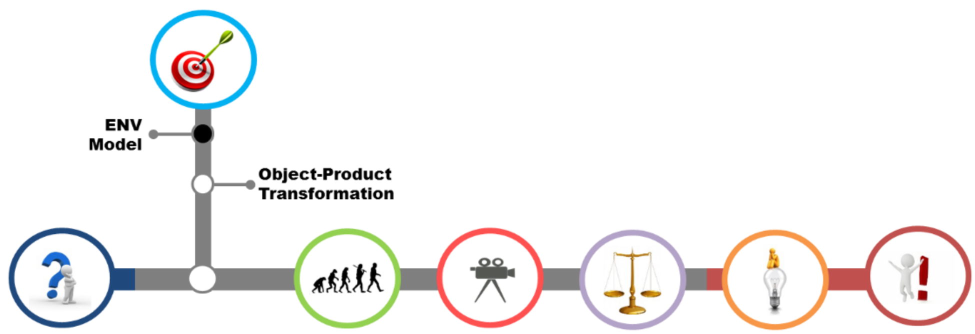


Figure 10. Tools for Functional overview.

The **object-product transformation** is an extremely simple tool that serves the purpose of identifying the most important elements of the problem. In accordance with the TRIZ theory, we here use the concept of "main useful function", i.e. the reason of the system existence.

The most important parts of the problem are so identified with the elements involved in the main useful function.

The object-product transformation is needed to overcome psychological inertias embedded in the system. Specifically, psychological inertia can be the habit to associate a function with the object that performs it, the tendency to extend partial restrictions to the whole object, the tendency to rely on past ways of doing something without actually knowing the reason, the tendency to associate common properties to words [117]. Forcing the elimination of the system when describing the function has been identified as the first ally to leave psychological inertia behind.

The tool uses a simple graphical representation as the one in figure 11.

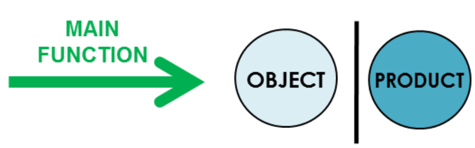


Figure 11. Object product representation.

The description of the main useful function is performed through three elements:

- The object: the object of the transformation/function.
- The product: the transformed object after the function is performed.
- The function: usually defined as a verb that describes the transformation of the object in product.

It is not necessary to be detailed about the description since the next tool "ENV model" serve that purpose. Examples on the elimination of object-product transformations are reported in figure 12.

Figure 12. Examples on the elimination of the system.

The use of a simple model such as the Object-product transformation stimulates the simplification of the starting points of the process in few sentences. The elimination of the system from the description of the "main useful function" is very useful to break psychological inertia and to define the boundaries of the problem solving activity.

In a practical context, the elimination of the system facilitates the identification an enlarged set of competitors. For instance, a company that produces pens surely consider all producers of pens as direct competitors. It is less luckily that the same company considers producer of pencils as competitors, and yet less luckily to consider producer of paper sheets as competitors.

However, when thinking at the main useful function of a pen: a sheet that is becoming written, we can easily imagine at least two other competitors: a producer pencils that cannot be erased, a sheet that writes itself.

An example on the real case study of the "aseptic valve" performed by a student is reported in table 3. The description of the function is very general and simplified.

Table 3. Object-Product transformation on the "Aseptic valve" problem (by a student).

Function	Object	Product
Isolate two parts of a duct so that the fluid is closed off	Two or more parts of a duct are linked and the fluid flows undisturbed	Isolated rooms and isolated fluid or fluids

In the case of a system that is associated with more than one function, they are treated separately one by one. For example, a hairdryer can serve the purpose of drying air but also to style your hair. In these cases, more than one object-product transformations can be highlighted.

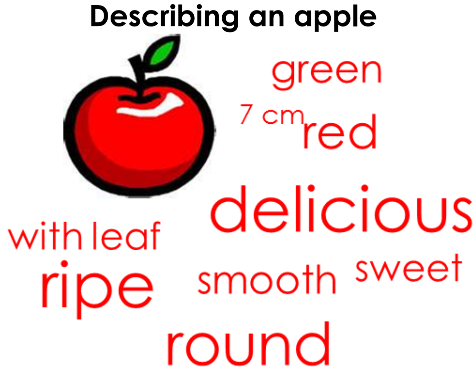
The **ENV model** is presented in the OTSM theory [65,118] as a tool to systematically describe an object. By itself, this tool helps in reducing heterogeneity of interpretation since it forces the user in describing a system or object with a set of predefined entities:

- E-The element: the name of the element of part of the element we are referring to.
- N-The parameter: the name of the parameter that we want to use for the description.
- V-The value: the correspondent value of the parameter.

A simple example of ENV model is shown in table 4.

Table 4. Partial example of ENV model: describing an apple.

Describing an apple



Element	Parameter	Value
Apple outside peel	Color	Red
Apple	Shape	Round
Apple	Diameter	7 cm
Apple edible parts	Taste	Delicious
Apple	Leaf presence	Yes
…	…	…

During the last years, a general descriptive tool such as the ENV model has been implemented in Spark for a specific purpose: the description of the main useful function. From here, I'll use an "*" to differentiate the classical ENV model from the new ENV model*.

The ENV model* is adapted by including two other concepts: time and ideality. The function "drying hair" is performed on the object "wet hair" to obtain the product "dry hair". The transformation implies a time relation between object and product. Therefore, in order to describe a transformation, we need two moments in time.

With this tool we want to describe both the current situation and the ideal situation (or desired situation). The ENV model* is so composed of 6 columns:

- E-The element: the name of the element of part of the element we are referring to.
- N-The parameter: the name of the parameter that we want to use for the description.
- VO-The value of the object: the correspondent value of the object before the transformation.
- VP-The value of the product: the correspondent value of the object after the transformation.
- VOI- The value of the ideal object: the correspondent desired value of the object before the transformation.
- VOP- The value of the ideal product: the correspondent desired value of the object after the transformation.

Actually, it is also important to understand how a certain parameter changes during the transformation. Therefore, it is suggested the use of a graph to describe how a parameter changes in time.

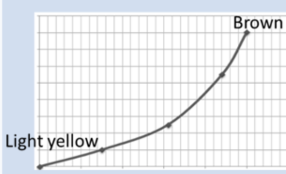
Some rule on thumb (heuristics) to obtain a better result are here summarized:

- Prefer quantitative parameters then qualitative ones, e.g. prefer length instead of dimensions, "number of leafs" instead of "leaf presence".
- Parameters can be of various nature:
    - Boolean: when the property exists or not (true, false); anyway, ask yourself if you can express the same concept with a quantitative parameter.
    - Constant: a quantitative value; always verify it is really constant, that is not a range and that it does not change in time.
    - Range: included between < >.
    - Graph: draw a qualitative graph to understand how parameters changes in time.
    - ??: lack of knowledge (such as unknown Boolean, constant, range or graph).

  - Does not matter: values that are not perceived as important for the ideal result; always verify that it is not actually a range of admissible values.
- Force yourself on seeing contradictory objectives on the ideality column.
- Consider ideal values as the ones that would help in realizing the function.
- Remember to include
  - Performances (precision, speed, efficiency)
  - Technical standards and norms
  - Non-ideal operations (when the function is performed on a special environment or situation)
  - Harmful effects to be avoided

Of course, ENV model* can be used to describe any function involved in our problem. However, its position in Spark is only in the first step. A simple function, such as cutting an apple, can take many lines of ENV model. You can easily imagine what would happen when the system is much more complicated. Therefore, ENV model is a deep analysis that is justifiable only when goals are vague (to increase structuredness) and the elements considered are few. That is why the ENV model* is used after the identification of the main useful functions in the object-product transformation, and all the elements that are not included in the function are not considered here.

Table 5. Partial example of ENV model in spark: cutting an apple.*

Element	Parameter	Value of the object	Value of the product	Ideal value of the object	Ideal value of the product
Apple inner part	Color			Always light yellow	
Apple	Shape	Imperfect Round	Imperfect Round''	Perfectly round	
Peeling	Time	15s		1s	
Apple	Diameter	<7cm-8cm>	<6,7cm-7,6cm>	??	
Apple	Taste	??	Quite good	Does not matter	Delicious
Apple	Leaf presence	Yes	No	??	
Apple seeds	presence	Yes	Yes	No	

Lack of knowledge

Elimination of constraint

Description of the present transformation

Description of the ideal transformation

Some implications of the use of the ENV-model are here summarized:

1. Facilitating the identification of lack of knowledge on the main useful function or on the desired function.
2. Facilitating the elimination of constraints (identification of parameters that can be modified without consequences).
3. Structuration of goals by reducing heterogeneity of interpretation.
4. Facilitating the identification of out-of-the-box solution paths based on the ideality approach.
5. Facilitating the identification of contradictions.
6. Facilitating the identification of innovation opportunities (when there is appreciable difference between real and ideal values.

These implications are qualitatively deduced from students' final exams and practical consultancy activity in companies. There is not yet a quantitative measure or test to demonstrate the aforementioned statements. Some heuristics, such as the use of qualitative graphs to describe parameters changes are already present in other methodologies [103].

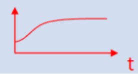
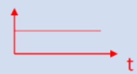
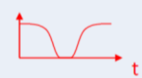
Engineering students applied ENV model* to describe the main useful function of the "aseptic valve". Student "A" produced table 6. We can see the identification of many contradictions on the ideal columns. Notice for example the contradiction of student "B" about fluid viscosity: the fluid should be with high viscosity when it has to stop, the fluid should be with low viscosity when it has to move". This is already a path for an innovative solution.

Outputs: The functional overview is not finished until all lack of knowledge are removed or considered not relevant for the following analysis. The outputs of this step are here summarized:

- A description of the main useful function performed by the present system;
- A description of the ideal/desired main useful function;

The main useful function is depurated of any parameter that involves the system. For example, if I want to describe the function of a hairdryer, I will not mention hairdryer parts or parameters, but I will describe all the parameters of the hair that are dried.

Table 6. Application of ENV model by student "A" on the "valve problem".*

ELEMENT	FEATURE	VALUE-OBJECT	VALUE-PRODUCT	VALUE-IDEAL
FLUID	TEMPERATURE			$T_{min} < T < T_{max}$ not changing fluid properties
FLUID	PRESSURE			No water hammer $p > p_{atm}$ (bacteria can't enter) $p < p_{atm}$ (no fluid waste)
FLUID	VISCOSITY			Infinite in stop 0 - finite in motion
TUBE	ROUGHNESS			Infinite in stop 0 not to have stagnant zones
FLUID	STERILITY	Commercial	Commercial	Commercial
FLUID	ORGANOLEPTIC PROPERTIES			
EXTERNAL SPACE	STERILITY	NO	NO	Commercial when there is a possible connection between internal and external spaces NO in the other cases
INTERFACE A-B	SECTION (without valve)			
BACTERIA	SIZE	µm	µm	mm (bacteria can't pass through splits and cracks)
BACTERIA	DEFORMATION	YES	YES	NO
FLUID	TYPE	Whatever type (food, beverage, dairy, pharmaceutical, chemical, and cosmetics)		

3.3 STEP 2: INNOVATION STRATEGY

The input of step 2 is the output of step 1, i.e. the description of the real and ideal useful functions.

"Innovation strategy" is designed to explore alternative systems to perform the "main useful function" and to select the requirements that deserve further analysis. It contains a variety of approaches and methods based on an evolutionary perspective for the strategic analysis of technical systems (see figure 13). On the one hand these approaches support decision-making confirming the initial direction of development or changing it towards more radical innovative solutions, on the other hand they help to quantify the effectiveness of the initial system by comparing it with the ideal system. This is to define the innovation strategy and prioritize the design requirements.

The output of the "innovation strategy" is a system with one or more requirements to be improved.

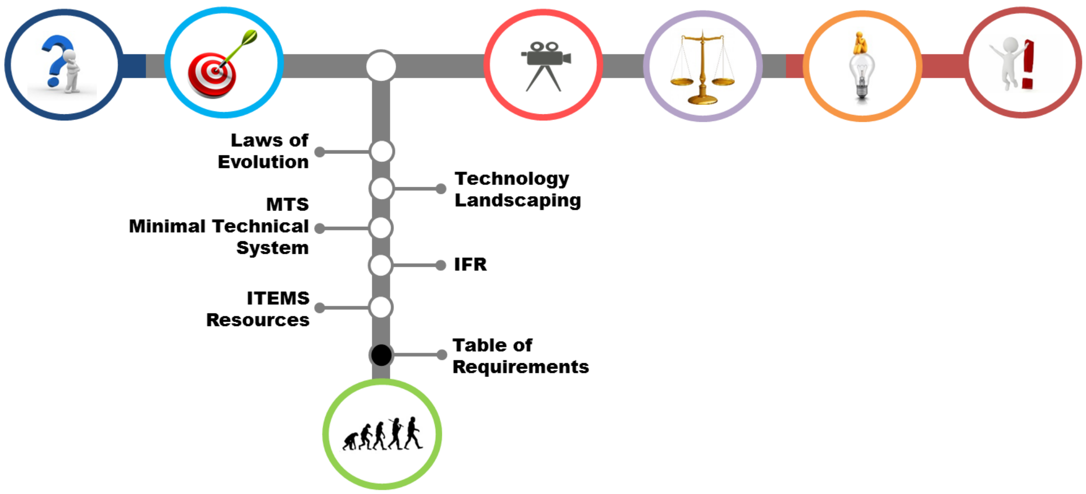


Figure 13. Tools for Innovation strategy.

The **IFR (ideal final result)** is used in step 2 to define a short description of the ideal product. Although there are different heuristics to define an ideal final result [24–27], the most used in Spark, is: "Imagine the object that transforms itself in the product, use only available resources".

Similarly, **Laws of evolution and MTS** (minimal technical system) are here used to identify possible evolutions of the system in terms of system completeness and control. There are not big differences in the use these tools if compared with the classical TRIZ theory (see [24–27]). An example of MTS used by a student for the "Aseptic valve" problem is reported in figure 14. In this version of Spark, Laws of evolution, MTS and IFR are used to trigger solutions or identify paths for improvement.

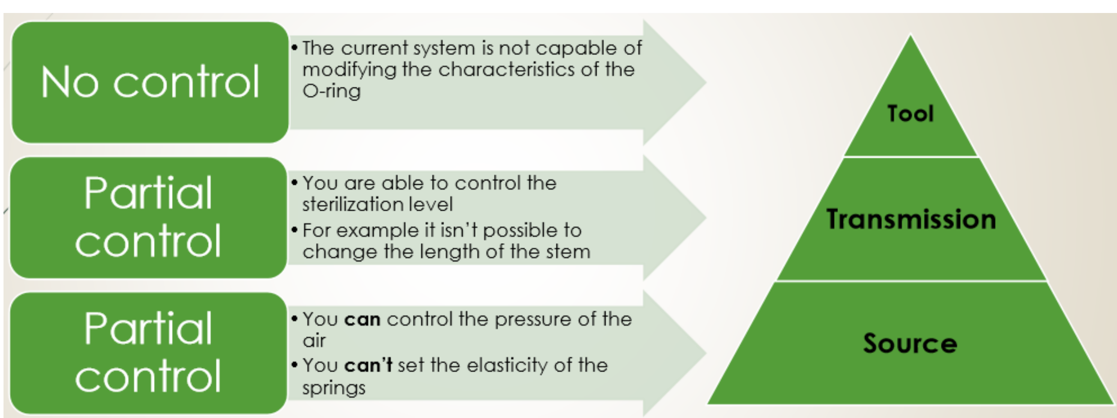


Figure 14. Example of MTS used by a student for the "Aseptic valve" problem.

Technology landscaping is a systematic search of all the possible technologies that are alternatives for a specific function. Specifically, the tools and methods used in Spark are explained in [51,52,119]. An example is shown in figure 15.

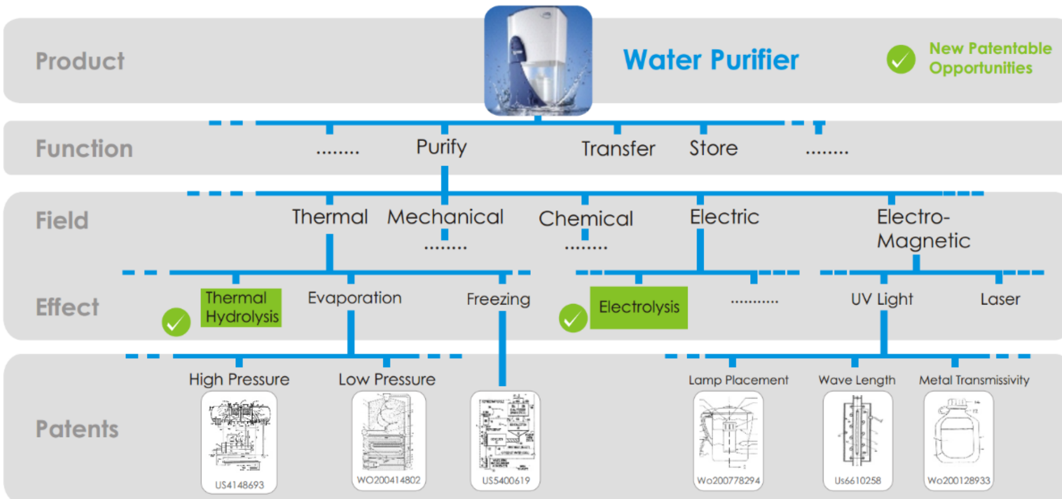


Figure 15. Technology landscaping of a water purifier.

ITEMS is an acronym od Information-Time-Energy-Material-Space. It is simple method to remember the main resources that are usually involved in a technical system.

An ideal system is the one consuming the less resources possible. ITEMS is used in Spark to compare systems in terms of their consumption of resources. An example of ITEMS used by a student in the "aseptic valve" problem is reported in table 7.

Table 7. Example of ITEMS used by a student in the "aseptic valve" problem.

	Current system	Ideal system
Information	Assembly instruction, steam parameters, pressure rate, O-ring and bellow condition…	Less than possible
Time	Long assembly time	No assembly time
Energy	Energy for the "intelligence" (sensor) and pressurized air (compressor)	No energy required
Material	A lot of components	Less than possible (None)
Space	Variable from the model. The valve occupies space inside the duct and also outside	Less than possible (No space required)

The **table of requirements** is used to sort requirements and define an innovation strategy. This tool will be better explained in chapter 4. Shortly, it is designed to collect requirements in accordance with importance and satisfaction values. Higher rankings are given to those requirements that have high importance and low satisfaction for the customers. The table of requirement is used to select the most important requirements on which focus on the next steps of the methodology. Students also used this tool (see table 8) to solve the "aseptic valve" problem. However, in students' project, this tool does not actually deserve particular attention since they do not have customers or companies' feedbacks.

Table 8. Table of requirements used by a student for the "Aseptic valve" problem.

Requirements	Importance % (X)	Satisfaction % (Y)	Market Potential (Y)
Reduced number of pieces	100	20	28,00
Small dimension	90	30	25,00
Energy saver	100	70	23,00
Duration	100	85	21,50
Adjustable (work in different position)	90	70	21,00
Sterility	100	95	20,50
Security	100	95	20,50
Avoid contamination	100	95	20,00
Reliability	90	80	20,00
Flow control (flow section control)	90	99	18,10
Constant monitoring	80	100	16,00
Easy cleaning	70	90	15,00
High quality material	70	95	14,50

Outputs: The "innovation strategy" is considered finished only when enough information of the selected system are collected. The first output is the system selected for further analysis. The second output is the requirement that has to be improved or that system. Examples of output can be: "reducing energy consumption of the hairdryer", "reducing maintenance time of the system".

3.4 Step 3: Problem Identification

The input of the third step is a system that has to be improved on a specific requirement, such as "reducing energy consumption of the hairdryer". The problem is decomposed in space and time, with tools that stimulates the identification of effects and causes (see figure 16). The goal is to define in a very precise way all the potential paths to improve the product.

At the end of this analysis, a hierarchy of the potential problems will be created.

The **Film Maker** tool is used to describe the dynamics of the current situation, representing the complexity of the problem as a sequence of events. This tool is studied to highlight the cause-effect relationships that involve time. It is composed of a sequence of states that are represented in frames on a time axis. In this sense, each state represents a picture of what is happening in a specific instant of time (see figure 17). In each frame it is asked to identify sub-problems related to the main one, possible paths for solution and resources.

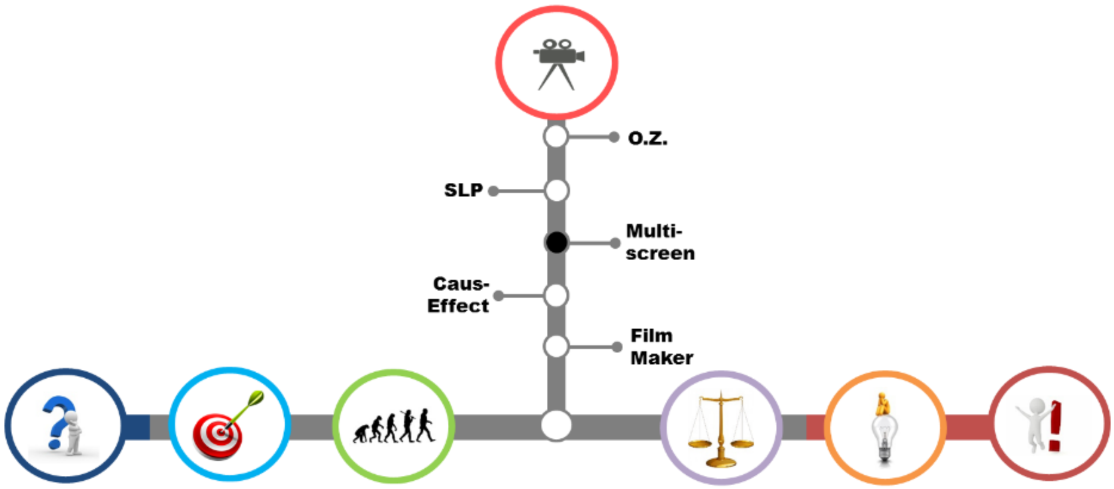


Figure 16. Tools for Problem identification.

Usually, a FilmMaker is completed starting from the instants where the problem solver has his own perception of the problem. Afterwards, new frames are added in the past and in the future in order to create a film. Each frame should contain an image or a drawing and/or a textual description (see figure 17 and 18).

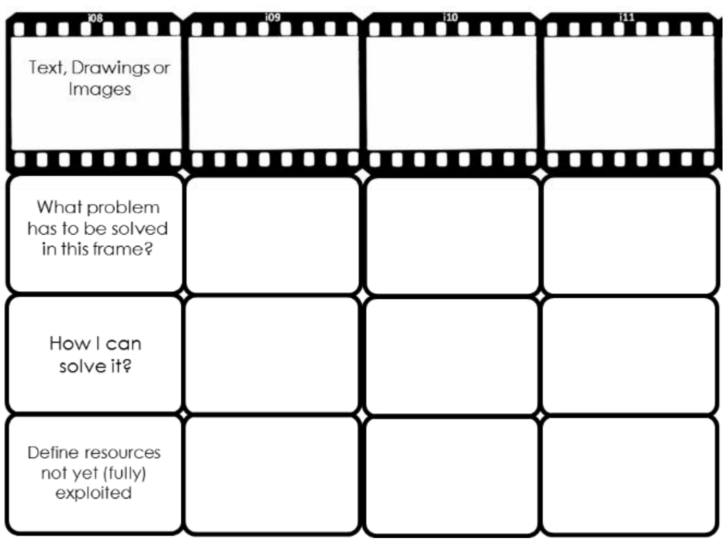


Figure 17. Schema of a FilmMaker.

The output of this tool is a sequence of frames with the aforementioned structure. From practical experience, a completed FilmMaker should contain at least seven frames, highlighting every state in which a condition is changed.

The Film Maker has some similarities with the known TRIZ tool called Multiscreen [24], but it has just one row and its purpose is quite different. Multiscreen (or 9 windows) itself has been used in some different ways from other authors [120], but in this sense, a Film Maker is more similar to the Domino Theory [121] where a long series of events can result in an unexpected situation.

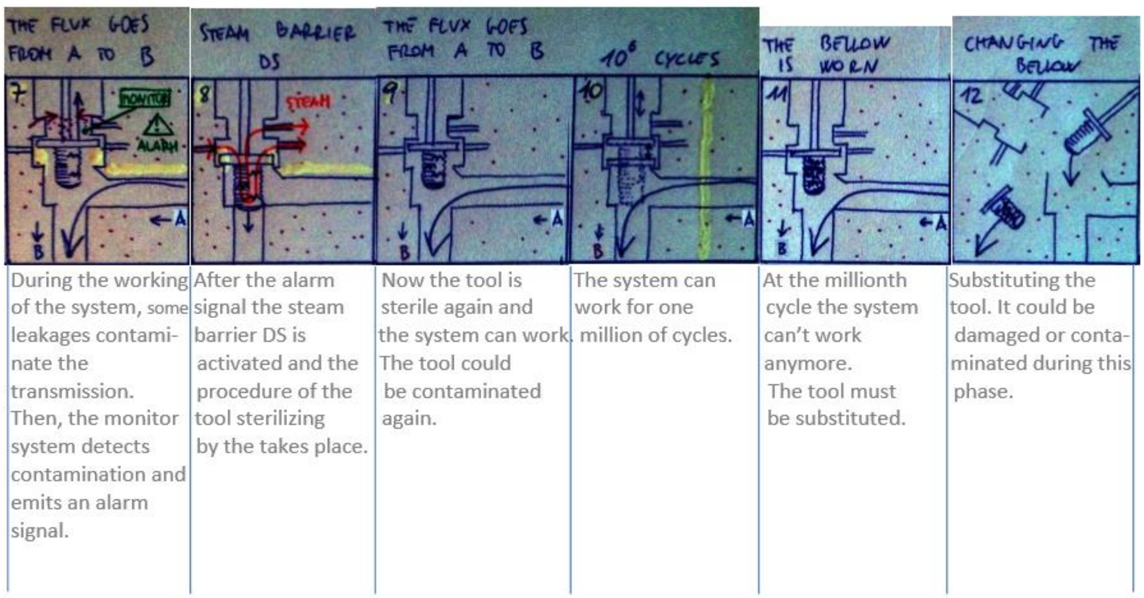


Figure 18. Extract of a FilmMaker for the "Aseptic valve Problem" done by a student.

The **O.Z. (operative zones tool)** is used as a magnifier on a selected frame of the FilmMaker. The same instant is described with different levels of zoom, facilitating the user in the identification of out-of-the-box points of view (see figure 19).

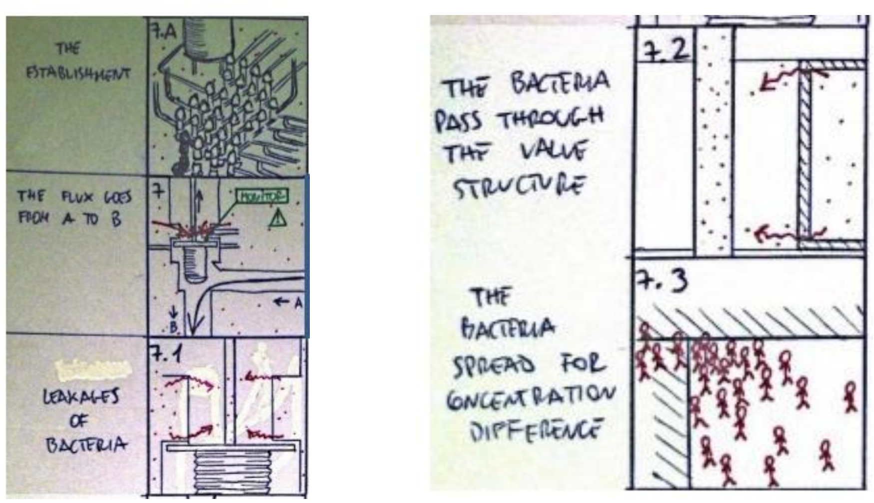


Figure 19. Extract of a O.Z. for the "Aseptic valve Problem" done by a student.

SLP smart little people, **cause and effect analysis** and **multiscreen** are not different from the classical tools used in classical TRIZ (see [24–26,61,122]).

Outputs: The outputs of "problem identification" are a series of paths that can solve the problem which was the input of the third step. These paths are not necessarily formulated in a specific form. Such as "Reducing the velocity of the stem", "Removing the stem", "Reducing the space between the stem and the seal", etc...

3.5 Step 4: Problem Formulation

The input of the "problem formulation" phase is a path for solution, indicating a parameter, object or action that should be changed or performed to improve the requirement of the system which was defined in the second step. For example, "in order to reduce the probability of contamination of the system (aseptic valve), reduce the space between the stem and the seal".

"Problem formulation" contains tool for the formalization of the problem in accordance to known form, such as contradictions or interactions among elements (see figure 20).

The output of this step is one or more problems formulated in terms of contradictions or TRIZ functional analysis.

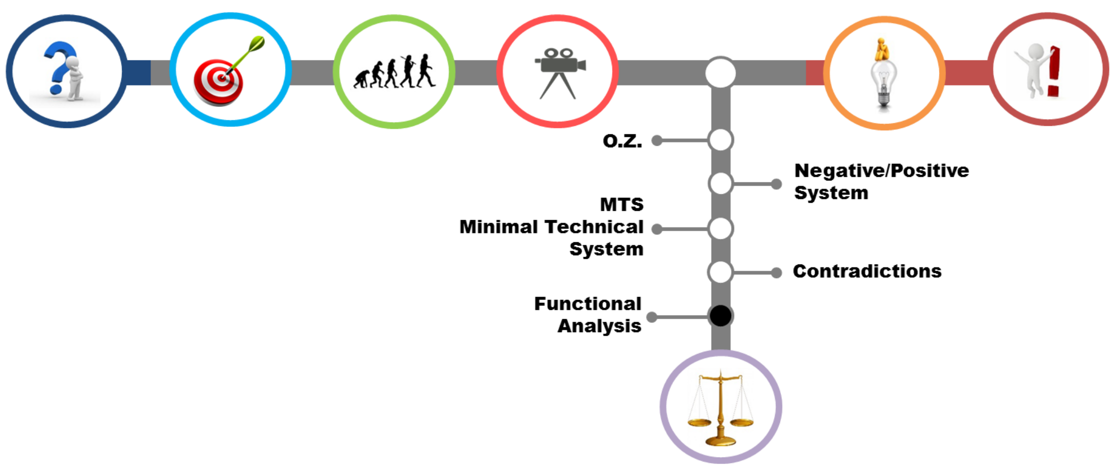


Figure 20. Tools for Problem formulation.

TRIZ functional analysis is used to formalize the problems contained in a selected frame in terms of satisfactory or unsatisfactory interactions and actions among elements. An example, compiled by a student, for a frame of the "aseptic valve" is reported in figure 21.

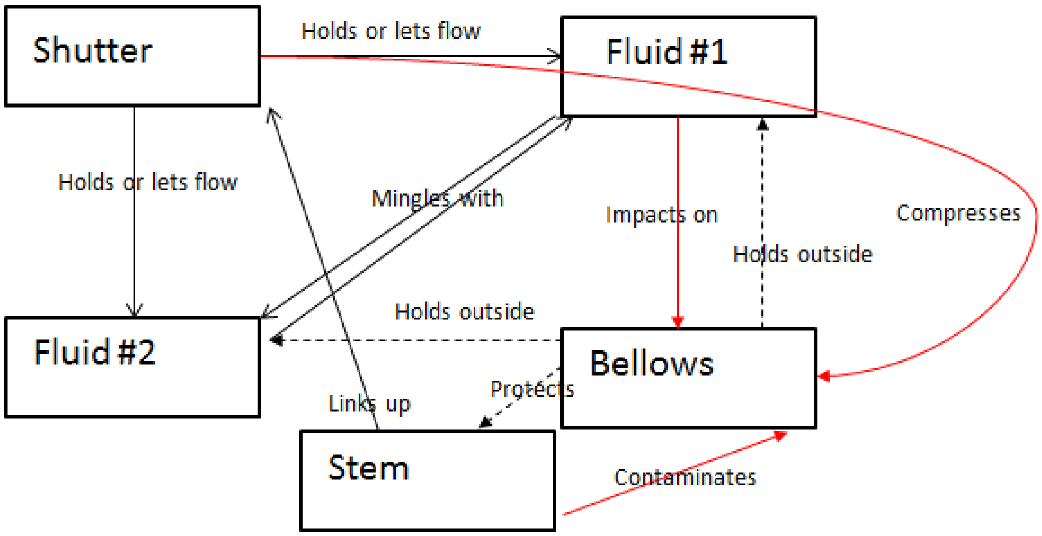


Figure 21. Example of functional analysis compiled by a student for a frame of the "aseptic valve".

Contradictions are defined when solution paths contain shortcomings or harmful effects when compared with the initial system. At this point of the methodology we should have:

- SYSTEM A: A system selected in step 2
- Evaluation parameter 1 (EP1): The requirement that has to be improved identified in step 2
- SYSTEM B: A description of another system with an improved EP1

In order to formulate a physical contradiction, we need:

- Evaluation parameter 2 (EP2): the requirement that is worsened when passing from SYSTEM A to SYSTEM B.
- Control parameter (CP): the parameter that is changed when passing from SYSTEM A to SYSTEM B.

It is clear that the control parameter should be conceptually different from the evaluation parameters to avoid the creation of tautological phrases. The selection of EP2 and CP can follow these simple heuristics: if more than one requirement is worsening by passing from SYSTEM A to SYSTEM B, start from the one with the highest innovation potential (see the table of requirements in step 2); prefer technical requirements instead of the requirement "costs"; in order to find control parameters, list what changes from SYSTEM A to SYSTEM B, select the ones that are clearly related with the evaluation parameters.

Physical contradictions can be graphically represented as in figure 22. The formulation of the contradiction is double:

PhC#1: we want a high value of CP because EP1 is realized but in this manner we do not satisfy EP2.

PhC#2: we want a low value of CP because EP2 is realized but in this manner we do not satisfy EP1.

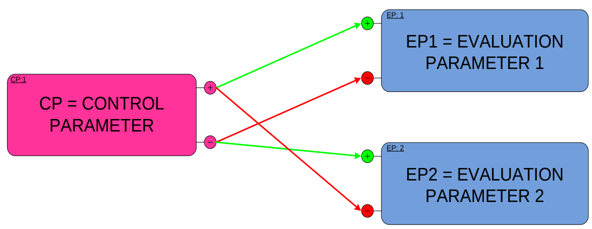


Figure 22. Graphical representation of physical contradictions.

Outputs: The outputs of this step are one or more contradictions to be solved.

3.6 Step 5: Idea generation

The inputs of the "idea generation" step is a formulated contradiction.

This step contains tools for generating new ideas and solving problems, such as Separation Principles, 40 Inventive principles (here used as part of the separation principles tool) and Standard Solutions. In addition, techniques and tools for patent searching and functional search are presented.

The outputs of this step can be partial solutions or completed solutions. Partial solution has to be further elaborated to remove secondary problems. In these cases, the partial solutions will iteratively be the input of the "Problem Formulation" step.

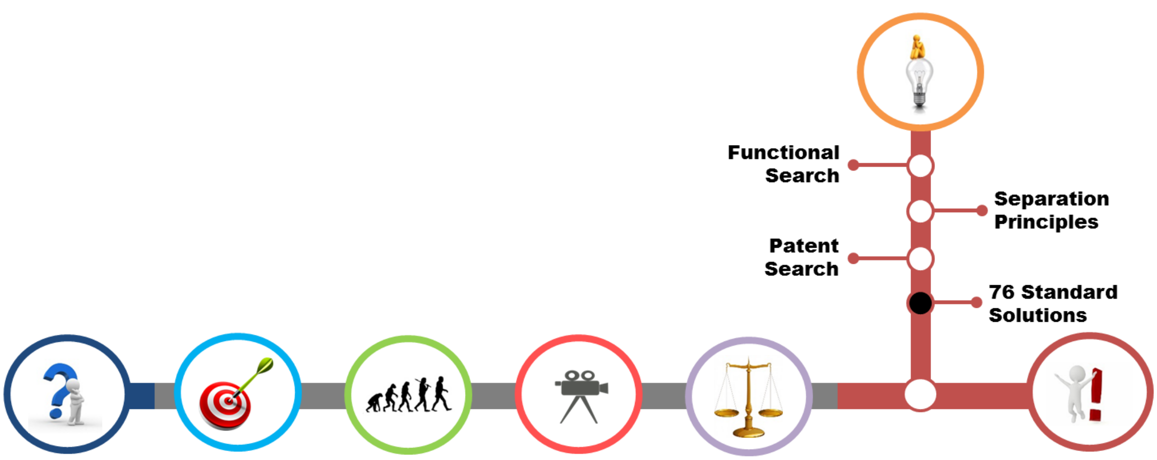


Figure 23. Tools for Idea generation.

Functional searches of solutions are supported by recent information retrieval techniques and tools, such as FOS (function oriented search) [50] or KOM (Knowledge Organizing Module) [51–53]. **Patents** are a huge container of available technical information.

Separation principles, 40 Inventive Principles and 76 Standard solutions are classical TRIZ tools (see [24–26]). They are used in Spark only after the identification and formulation of a physical contradiction. There is not a specific way to select one tool instead of another.

Outputs: The outputs of this step can be partial solutions or completed solutions. Partial solution has to be further elaborated to remove secondary problems. In these cases, the partial solutions will iteratively be the input of the "Problem Formulation" step.

3.7 IMPLICATIONS OF SPARK

Spark has been developed with the objective of systematizing the formulation of contradictions and has become a sorted methodology for product improvements. Some tools, such as filmmaker, O.Z., and ENV model stimulates an out-of-the-box way of thinking that enlarges the problem spaces on areas that are not naturally explored. In order to prove these assumptions, qualitative evaluations on students' exams are studied every year at the University of Bergamo. One of these studies is here presented.

The problem solving activity involved two groups:

- S: 4 students, after 40 hours of course about Spark.
- C: 4 control group of consultants with three years of experience on problem solving activities.

The proposed problem was the one of the "Aseptic valve", already presented in section 3.1. Both students and consultants had four days to solve the problem. Students were constrained to follow Spark methodology while consultants were left free to use whatever they wanted. Both groups had access to internet connection.

Solutions have been classified in accordance with a FBES ontology, differentiating between function, behavior, physical effect and structure. Solutions have been counted in accordance to the created classification in tables 9 and 10. It is easy to notice that there are differences on the type of solutions that has been generated by students and consultants. Although numbers are not suited for a statistical analysis, we can qualitatively state that:

- Both consultants and students have worked mainly on how the seal could be moved, suggesting many alternative solutions involving different physical effects. Consultants identified more suitable effects than students.

- While consultants have not worked on maintaining the fluid uncontaminated, students provided many solutions also in that direction. After qualitative analysis of the students' project, these solutions seem to be a consequence of the analysis of the present a system in space and time with the tools FilmMaker and OZ.

The strong differences between the two groups are sufficient to consider the aforementioned statements valid. Instead, there is not yet a direct way to objectively judge solutions with numerical approaches. According to a qualitative evaluation of the author, consultants tended to present solutions that are more feasible, while students suggested also incomplete paths for solutions. In some cases, these paths are peculiar and should be considered for the evaluation of the methodology. Let's take for example "The fluid changes its viscosity when it has to stop". Alost half of the students mentioned it by applying the ENV model or by applying the IFR. It is a valuable example of how Spark helps in thinking out of the box, although students did not actually search for physical effects that would make that solution practicable.

Table 9. Classified solutions of the "Aseptic Valve" test (part 1).

Classified solution	Students	Consultants
Stopping and unstopping a fluid in a conduit		
The fluid itself blocks and unblocks the flowing liquid	1	0
The fluid changes its viscosity when it has to stop	1	0
The tube itself blocks and unblocks the flowing of the fluid	1	1
The tube is flexible and bends to block the fluid	0	1
The tube changes its roughness (difficult to imagine)	1	0
Moving a seal to block the flow of the fluid	7	12
Moving only the internal part of the stem	7	11
Termally	3	3
Electromagnetic	3	3
Piezoelectric	0	2
Magnetostriction	0	1
Memory shape material	0	2
Stem with inflatable part	1	0
Moving the seal with the actuation inside	0	1
Deforming a seal	1	2
Using variation of porosity of a seal	0	2
Piezoelectric effect to change porosity	0	1
Thermal expansion to change porosity	0	1
Radially	1	0
Using water or air	1	0
Inflatable baloon	0	1
Using Pressure waves to stop the flow	1	0
Generating pressure waves with ultrasound	1	0

Neither students nor consultant actually tried to answer the main question about "how an aseptic valve will look like in the future". This can be interpreted as a limitation of Spark in selecting the best solution among many. Students as well as consultant concentrated on generating many solutions instead of justifying which one would have been the best. Further considerations on the implications of Spark for product improvements will be presented in future publications.

Table 10. Classified solutions of the "Aseptic Valve" test (part 2).

Classified solution	Students	Consultants
Mantaining the fluid incontaminated		
Avoiding chemicals residual in intertices	1	0
Avoiding contamination of the environment	3	0
Sterilizing the environment around the valve	3	0
Removing air	1	0
Avoiding the contamination on the external part of the stem	1	1
Insulating the stem from the external environment	1	1
Insulating the stem with a flexible membrane	0	1
Avoiding contamination of the internal part of the stem	3	2
Avoiding that the stem goes inside/out	1	2
Using rotation instead of linear motion	1	2
Avoiding the movements of bacterias	6	0
Acting on the miscibility of bacterias into the fluid	1	0
Holding bacteria on the stem	3	0
With electrostatic fields	1	0
With porous materials that traps bacteria	1	0
Trapping bacterias in filters after they have incresed in dinensi	1	0
Reducing the space between the seal and the stem	1	0
Eliminating/killing bacteria	2	0
Sterilizing the part of the stem which is entering	1	0
Using biocide material only on the the external part of the ster	1	0
The fluid itself kills bacteria	1	0
Deactivating bacteria	1	0
Using vibrations	1	0
Avoiding failures of the bellow	3	0
Non eterogeneus bellow (thikness only where needed)	1	0
The bellow is termally expandible and support stem movemer	1	0
The bellow is an elastic thick material	1	0
The bellow elasticity is increased with porosity	1	0
Allowing failures but the bellow will not break since it fixes itself before bacterias have time to contraminate the fluid	1	0
Avoiding that small parts of contamined fluid is harmful	1	0
Sterilized the contamined part of the fluid	1	0

3.8 Boundaries and limitations of Spark

Spark is grown by slightly adjusting the TRIZ methodology on real industrial case studies in SME and students' responses in an academic context. Although Spark is now a robust and systematized method to creatively improve products, it lacks a robust decision-making procedure. When a multinational company has to change its products, decision-making can be more important than generating solutions. This situation can be better understood if we think that big companies have entire R&D departments working every day to produce new ideas. It is easy to imagine that ideas are not missing, the problem will be how to select the best one among them. In Spark, decision-making is somehow supported by the table of requirements, which is usually performed in less than one hour to identify the main customers' needs. As is, the table of requirements cannot be considered a robust tool for decision-making. In the previous section, I have mentioned a problem solving activity performed by student where the main task was to identify "how an aseptic valve of the future will look like". Students using Spark preferred to generate as many solutions as possible but they actually missed the evaluation of these alternatives to suggest one or another. This behavior confirms the lack of a proper methodology or tools to compare and select one solution instead of another.

Furthermore, Spark inherits TRIZ tools for idea generation, such as the 76 standard solutions, the 40 inventive principles and the separation principles. Spark uses these tools with the purpose of triggering solutions, after the identification of a contradiction. It is not clear how to select one tool instead of another, furthermore, these tools are sometimes overlapping and inconsistent.

Other limitations of Spark are present for problems about "finding the unknown causes and effects", where some tools such as FilmMaker and OZ can (see section 3.4) can be useful, but no specific adaptations of the methodology is yet available. "Forecasting" problems may take advantage of tools which are present on the second step of the methodology but the procedure would not be guided.

Table 11 summarizes the evaluation on the limitations of Spark with respect to the typology of problems. The table shows how much Spark is adapted (specificity) for a specific type of problem. Furthermore, three lines are highlighted since they represent the problems for which this dissertation tries to improve the effectiveness of Spark (chapter 4 and chapter 5). In chapter 4, step 2 is reorganized to make decision-making a robust process in Spark. In chapter 5, the idea generation tools (contained in step 5) are adapted for problems of unsatisfactory actions.

Table 11. Boundaries of Spark according to the unified classification of problems.

Groups	Problem types	Specificity of Spark
Design problems	Creating a new system, artifact, process or product	Low
Conceptual Design problems	Finding how to perform an action or function	Medium
	Improving an existing product	High
	Improving requirement "X" of a product	High
	Improving an existing process	Low
	Improving requirement "X" of a process	Low
	Solving technical contradictions	Medium
	Solving physical contradictions	High
	Improving unsatisfactory actions or interactions	Low
Finding the unknown causes and effects	Finding the unknown causes of something that happens (emerges) in a system.	Low
	Finding an unknown function of an element in a system	
	Finding unknown information about physical and chemical processes.	
	Finding possible undesired unknown effects and causes in a system (failure prevention).	
Forecasting (Predicting)	Predicting the future characteristics of a system	Low
	Finding possible new applications for an existing system	Low
Decision-making Problems (ranking or selecting)	Selecting or ranking a series of alternative systems	Low
	Selecting or ranking the requirements of a product or process that should be changed	
	Deciding if a product should continue to stay on the market or it should be abandoned	
	Selecting the number of versions/sizes that should be part of a product family	
	Selecting on which product in a company to invest and innovate	

4 PROPOSAL: A METHODOLOGY TO DEFINE THE INNOVATION STRATEGY IN SPARK

In the previous chapter, limitations and present boundaries of Spark have been highlighted. Among them, decision-making problems about "selecting or ranking a series of alternative systems" and "selecting or ranking the requirements of a product or process that should be changed" has been identified as valuable points for improvements. This proposal presents an adaptation of Spark for the definition of a robust Innovation strategy for product improvements and for the selection of the most appropriate system or technology. This proposal influences the second step of the overall methodology and improves the management of requirements in Spark.

The launch of a new product usually involves conspicuous investments and risks. Best practice companies have implemented structured processes to effectively manage the development of a new product and increase the probability of success. The stage-gate process was introduced for this purpose [123] and some big firms implemented it for their product development activities [124,125]. Although many customized gate-models are practically applied in companies, the basic idea is composed of five gates [123]: initial screen, second screen, decision on business case, post-development review, pre-commercialization business analysis. This dissertation presents a method that is specially adapted for the first stage of the stage-gate process, i.e. between the first and the second gates. Of course, this stage can be supported also with other methodologies, which can be found in literature [126].

QFD is the acronym of Quality Function Deployment [127], an approach to define customer needs or requirements, with a structured procedure to translate them into technical parameters and plans to produce products that essentially meet those needs.

Benchmarking is one of the first form of structured product development [128]; products are compared with industry bests from other companies and improvements are planned to cover possible gaps or to overcome competitor's performances. An integration between QFD and Benchmarking was experimented by Kumar [129], but also many other integration involving QFD with other methodologies are recorded in literature [130,131].

QFD and other similar approaches start from the definition of a list of requirements (or desired and undesired features) that are usually related to the costumers' perception of "how the product should be". This task is usually performed with the help of a voice of customer [132] and through structured interviews with sales department personnel and factory representatives [133]. However, with a voice of customer, clients' ideas are often vague and ambiguous and they provide suggestions that are strongly related with what already exists. Furthermore, clients are not aware of new emerging technologies. In order to obtain new functions, some methodologies involve the extraction of functions from patents [134], or from other online sources [135], such as forum or social networks.

When a big firm must invest in the development of a new product, the allocation of resources should be concentrated on the most important requirements. The identification of the main requirements can be associated with the formulation of the right problems to be solved. If the evaluation has been done correctly, the problems will involve the requirements with the maximum priority and customers will perceive the added value of innovation. As different problem formulations [9] can lead to different solutions, a different selection of requirements will lead to different problems and therefore to different solutions.

In order to improve the ranking of requirements, a series of methods that are called Importance-Performance Analysis (IPA) have been developed [133,136]. The ranking of requirements is done with two parameters: importance and performance. In one of the first experimentation of this method a questionnaire was sent to many people asking "how important a feature is?" and "how well the dealer perform?" [133]. The result was a series of opinions that could be visually represented in an Importance-Performance Cartesian graph. After some years, a similar approach was introduced with different terminology, "performance" is now called "satisfaction" [137]. Furthermore, a third parameter was added and was called "opportunity". This parameter combined importance and satisfaction to create an overall quantitative evaluation of each requirement. Another way of calculating "opportunity" has been presented. In the TRIZ world, Livotov 2008 suggested a new formula to calculate the "opportunity" value (or innovation potential) which was theoretically based on the concept of ideality extracted from TRIZ.

Despite the differences on the mathematical formula that lead to the ranking of requirements, the idea of IPA analyses is to concentrate innovation efforts on requirements which have high importance and low satisfaction. In fact, customers will perceive improvements just if their satisfaction is not already maximum, and this perception will be more determinant if the requirement is important for them.

Although the assessment of requirements has been recognized as a valuable process, its implementation is not easy and without complications. During the evaluation of requirements, different people will have different opinions. These opinions are influenced by many factors involving personal background and knowledge. These differences can be found in both customers and experts, also among departments of the same firm. For instance, when creating a new product and making interviews, knowledge of previous experience may influence the evaluation; production managers may be more focused on requirements that simplify the manufacturing problems, sale personnel will request lower costs, engineers would ask for better technical performances and so on.

Furthermore, requirements are subjected to language ambiguities, which are especially evident in abstract requirements such as “pleasing to see” or “that give the perception of a green product”.

In this dissertation, an overall methodology is presented to make the evaluation and ranking of requirements a more robust process. The same rank of requirements is used to select the most appropriate system or technology to improve.

Interviews, patent information, marketing information and problem solving are not just retrieved, but presented during interviews to influence the evaluation of requirements. Specifically, infographics and maps are used to summarize information and give a comprehensive overview at glance.

The methodology has been applied in a big multinational firm for the definition of an innovation strategy of several products that were on the market from longtime. The methodology includes the first step of Spark as explained in section 3.2, while it divides the second step “Innovation strategy” in five sub-steps (see figure 24).

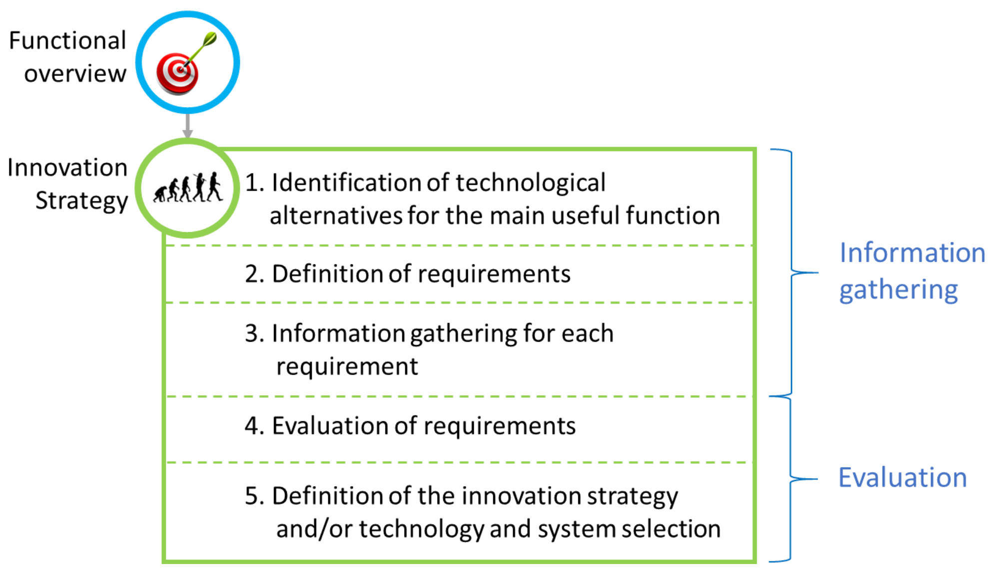


Figure 24. A step-by-step methodology for the “Innovation strategy”.

4.1 INFORMATION GATHERING

The goal of this phase is to collect quantitative data that will be used to influence the second phase of requirement evaluation. The outputs are a report and an infographic map for each requirement. The first one is meant to provide a similar knowledge to everyone that involved the interviews; the second one to provide an overview of the present scenario and possible future scenarios at glance.

4.1.1 Identification of technological alternatives for the main useful function

The identification of the possible technologies or systems are mainly performed with the tool called Technology landscaping, a systematic search of all the possible technologies that are alternatives for a specific function. Specifically, the tools and methods used in Spark are explained in [51,52,119]. An example is shown in figure 25.

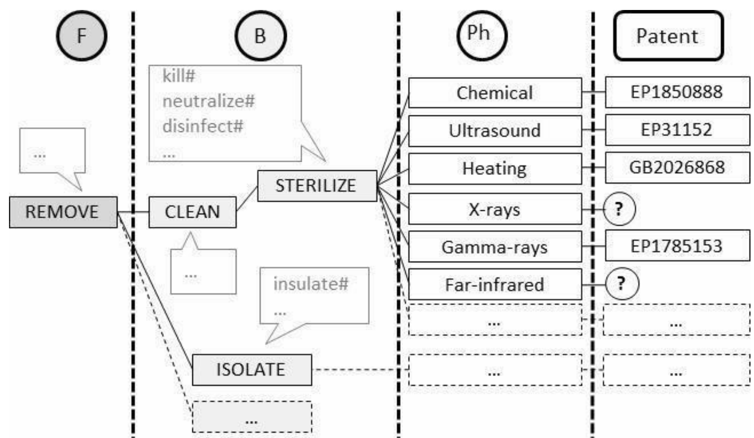


Figure 25. Technology landscaping for sterilizing lens [52].

4.1.2 Definition of requirements

The standard procedure starts from a first audit with the "project manager", i.e. the "owner" of the project. During this meeting, the main function and the name of the product are identified, as well as the name of the persons that can be useful to collaborate in the definition of the list of requirements.

Knowing the name and function of the product, a fast screening of patent literature is used to identify competitors, patent density, and common problems of the product. The patent search can be integrated also with the reading of a commercial catalogues of the product. A series of interviews are scheduled with at least a member for each department: experts from the marketing area, experts from sale area, experts of the technical area, experts of the manufacturing process and experts of quality. It is especially important to interview people that collect complaints of the customers. Interviews are performed to extract information on "how the product is" and "how the product should be". The concept of ideal final result is explained to each person to imagine the new product without any constrain.

Collected data are structured in a list of requirements that is proposed to the "project manager" for confirmation. The provided list of requirement has a nested structure, with more general requirements that form groups containing more specific requirements. For instance, "compactness" can be decomposed in sub-requirements such as “height, width, depth, footprint and volume”. Note that sub-requirements are not necessarily technical parameters.

This grouping operation is very important for not everyone is expert enough to provide detailed evaluations on all the specific aspects of the product, but they will usually be capable of assessing the importance of the more general requirement.

Along with the definition of requirements, the identification of main competitors is part of this phase. Known competitors provided by the company are eventually added to the new ones coming from the patent searches.

4.1.3 Gathering information for each requirement

Information gathering is performed for each requirement. This information contains both state of the art and new potential technologies. The state of the art is built on four sources of knowledge: patents, scientific literature, product catalogues and the web. Where web searches are performed in competitors’ websites and available web-search engines. These knowledge searches aim at positioning the product in comparison with the best solutions at the state of the art.

Note that searches are performed by looking for the improvements of a certain requirement for a certain product. For instance, if the analyzed product is an engine and the requirement is “compactness”, the searches will include all possible solutions that improve or reduce the engine compactness.

Once a pool of document for a specific requirement is defined, the following information are extracted:

- Patent density: the number of patents applications during the years.
- Cooperation: the name of the companies that appear as co-applicants in patents.
- Regulation aspects and market structure: this is an interpretation of data about economic situation.
- Technological alternatives: this is a classification of solutions available on literature. Differences are made if the technology is coming from scientific literature, patent literature or other sources. The presence of the technology on the market is also checked.
- Technological trends: as part of the previous point. Technological alternatives are represented in a timeline to identify technological trends of the past years.
- Performances: performances of commercial products of competitors are extracted.
- Investments: based on data availability, the investments of the company during the years, associated with a specific requirement are included.

Once the state of the art search is finished, a short problem solving activity is necessary to understand “what may come next”. In fact, decisions cannot be completed without a proper knowledge of possible future scenarios. The problem solving activity allows to include the following information for the evaluation:

- Technological new alternatives: this is a classification of new solutions not already explored in literature.
- Future technological trends: new and state of the art technological alternatives are analyzed from an evolutionary perspective, with the help of TRIZ laws of system evolution. The results are one or more descriptions of possible future scenarios.
- Ideality: using the concept of ideality, the IFR (ideal final result) for the considered requirement is generated.

At the end, each requirement will be accompanied with a big amount of information, which can be difficult to manage during interviews. An infographic overview on each requirement is built to provide ready to use information during the evaluation. The overview is printed in maximum two A4 pages. An example of infographic overview is presented for the requirement "monitoring pressure" in figure 26 and its use will be clarified in the next section.

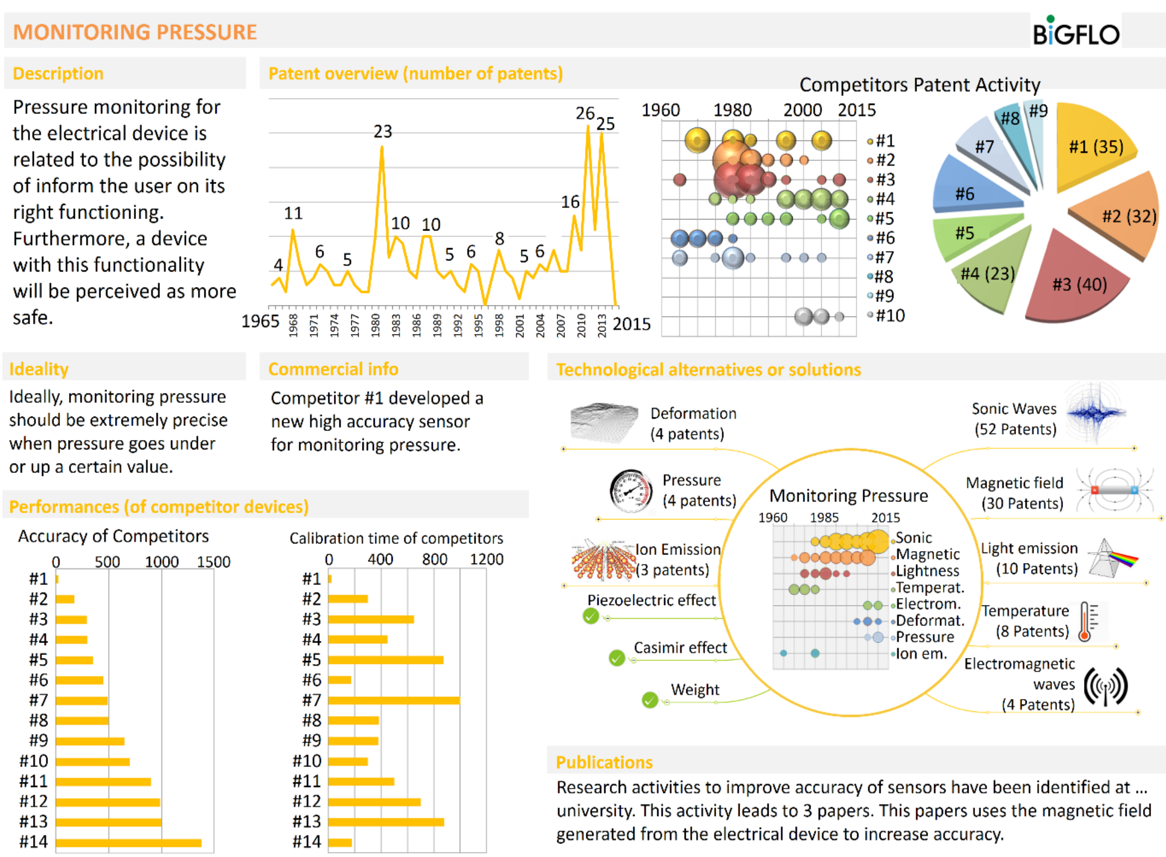


Figure 26. Example of infographic for the requirement monitoring pressure.

4.2 REQUIREMENTS EVALUATION

The goal of this phase is to weight all requirements in terms of importance and satisfaction, eventually providing a ranking based on the market potential of each requirement. The output of this phase is an importance-satisfaction graph containing the weighted average values of experts' opinions.

4.2.1 What importance and satisfaction are

Jacoby [139] argues that importance is reflected in goal-oriented search attributes that consumers actively look for in the target product and consider when making a purchase decision. In a simpler and practical way, we define the degree of importance as a quantification of the influence of the considered requirement on the sales volume. In a scale from 0% to 100%, 100% means very strong influence while 0% means no influence at all.

Satisfaction is a measure of how products and services supplied by a company meet or surpass customer expectation [140]. In a scale from 0% to 100%, 0% is the absence of the feature or great dissatisfaction, 100% is maximum satisfaction. The aforementioned definitions are given to experts before the interviews for the evaluation, which are described in the following paragraph.

4.2.2 Evaluation of importance and satisfaction

The "project manager" will schedule a series of meeting with the experts. Also in this phase, as in the definition of requirement, an interview with a person of each department is necessary. If possible, more than one experts from the marketing area are required for they are the one with more knowledge of the customers. Some days before the interviews, the report and the infographics are sent to the experts to make them acquire knowledge. Interviews can be performed with one or two person at a time. The infographic overview of each requirement is shown and used for discussion during the interviews.

First, the evaluation of importance is performed. The evaluation of importance is usually more reliable through a comparison approach than asking for absolute values [141]. For this reason, the interviews are performed with the following rules, placing post-it on a graduated scale:

- The leader explains the most interesting result of the analysis with the help of the summary infographic.
- For the first requirement: he asks to select a value from 0% to 100%.
- For the requirements that follow the first one: he asks to select a value from 0% to 100%. Then we ask if the requirements with the closest rate are really more important or less important. If the opinion changes, we repeat this step.

The evaluation of satisfaction is performed simply by asking to impersonate the customer and imagine its degree of satisfaction for the product, comprised from 0 to 100.

After the interviews, data are elaborated to provide the final weighted values of importance and satisfaction. A simple weighted average can be used, as well as an average based on ranking considerations [141]. Regardless to the specific formula that can be used, more weight is usually given to raters that are considered closer to the customer. The final result of the evaluation is an Importance-Satisfaction graph as shown in figure 27. Improvements on requirements on the bottom-right corner of the graph will likely to be perceived by the customer. Although the ranking of requirements can be presented with the aforementioned graph, the list of requirements can be sorted with one of the formulas to calculate the market potential (or opportunity index) [134,137]. Specifically, in our approach, we used a normalized version of the opportunity index (inspired by [137]):

$$MP=(100+I+(I-S))/10 \qquad (1)$$

where MP is the market potential index, I is importance and S is satisfaction of a certain requirement. In this way, the market potential is a positive number from 0 to 30. The formula is studied to give a higher rank to a requirement that is considered very important with very low satisfaction, lower rank to a requirement that is considered not important with high satisfaction.

4.3 Defining the innovation strategy

The graph of figure 27 represents a fast and easily understandable overview on the current client's satisfaction and importance values of the product. The next step is called "definition of the innovation strategy" and is created to define the identikit of the future product. The new product will be depicted using the same importance-satisfaction graph, where importance or satisfaction may change:

- Importance changes: habits of customers can change during time, and feature with little importance can become of high importance. In some rare cases, such as when the company has the monopoly on a certain product, importance can be manipulated through sensitization or desensitization campaigns.
- Satisfaction changes: the satisfaction value can be changed by changing the performances associated with a certain requirement, or the perception of that requirement.

The definition of the innovation strategy is represented in the same importance-satisfaction graph as arrows. The length of these arrows provide a quantitative idea on the changes. In this phase, problem solving information (which are present on the infographic sheet) is critical to suggest improvements that are feasible, at least from a conceptual point of view.

4.3.1 Case study

In this case study, we consider the development of a new electrical device for a multinational company. Data are partially hidden and suitably modified to hide any confidential information.

4.3.1.1 Information gathering

In accordance with the first step of the methodology, the list of requirements has been defined through a series of interviews with at least one person for each department of the firm. The collected information was reorganized in a series of requirements. The nested structure of requirements can be seen in table 12. Patents about the product and web searches help in finding competitors.

Table 12. Nested requirements list.

Energy Consumption	Phase 1 power consumption
	Phase 2 power consumption
Peak power	Peak power consumption
Compactness	Height (version 1 and 2)
	Width (version 1)
	Width (version 2)
	Depth (version 1 and 2)
…	…
	…
	…

For each requirement, a comprehensive search and a problem solving activity are performed. Reports of this activity were sent to the "project manager" for approval. After this, reports were sent to the persons who were going to participate in the following interviews.

Here follows an example of information that was provided for the requirement "Presence pressure monitoring". The following information were used for the evaluation:

- Patent distribution during the years: the patent activity revealed a strong increment on the last years and an increasing diversification of the agents/companies.
- Performance comparison: we compared performances about the peak power and continuous power consumption.
- IFR: the minimum amount of energy necessary to perform the function has been identified with ideality, along with a conceptual solution to reach it.
- Problem solving: a classification of all the solutions derived from the problem solving phase is presented as a map. This map is developed by TRIZ experts with the help of a functional based search [52].

- …

Figure 26 shows the summary of this information in the overall infographic representation.

4.3.1.2 Requirements evaluation.

Interviews were performed with several experts from marketing, two experts from the technical area and one manager from sales area. Given the time limits, other people were interrogated through a questionnaire sent through e-mail. The interviews were performed following the rules of paragraph 3.2, with outputs such as the one of figure 28. As it can be seen, evaluation is divided into areas for the markets are different for each country (zone). An average value for importance and satisfaction has been extracted. In the average calculus, e-mail's questionnaires were considered with less weight than proper interviews and the example for the final outcome of one zone is shown in figure 27. Results from phase 1 are used by experts to conduct the evaluation process dealing with the market potential of each requirement.

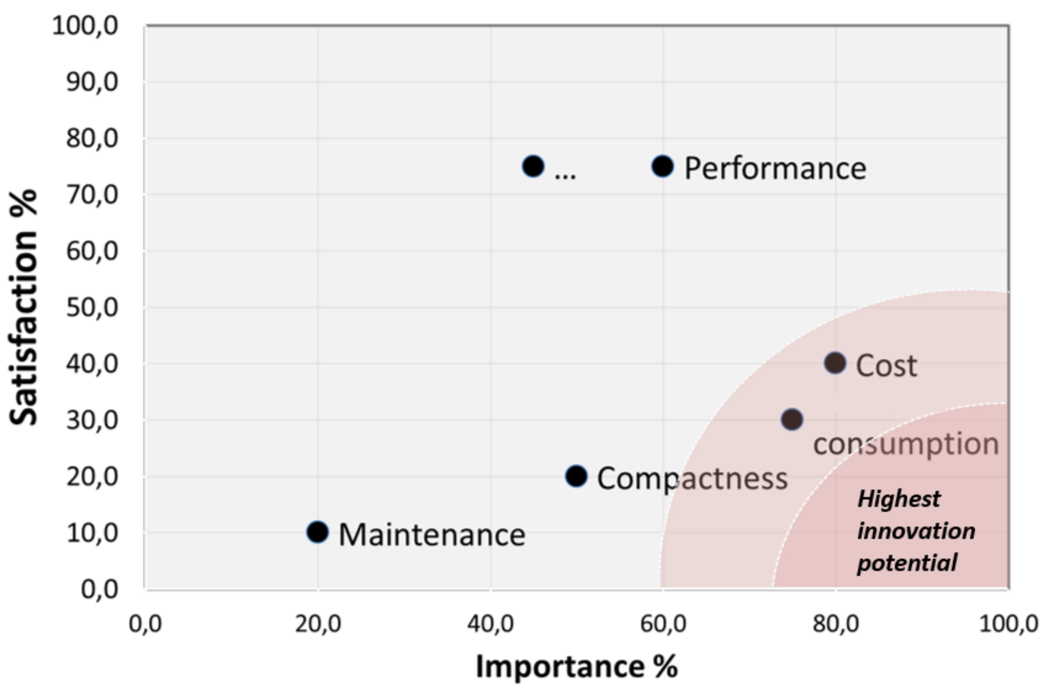


Figure. 27. Requirements ranking mapped on the Importance-Satisfaction diagram.

	Asia			Europe			Geographical zone 3			Geographical zone 4		
	Overall Importance %	Overall Satisfaction %	Market Potential	Overall Importance %	Overall Satisfaction %	Market Potential	Overall Importance %	Overall Satisfaction %	Market Potential	Overall Importance %	Overall Satisfaction %	Market Potential
Energy consumption	15	60	7	50	40	16	50	5	19,5	70	80	16
Compactness	10	75	4,5	25	20	13	15	25	10,5	30	50	11
Requirement 3	10	50	7	10	50	7	90	5	27,5	60	50	17
Requirement 4	75	20	23	75	80	17	80	50	21	90	70	21
Requirement 5	50	40	16	90	50	23	50	40	16	90	50	23
Requirement 6	50	60	14	40	50	13	10	80	4	55	60	15
Requirement 7	50	40	16	60	40	18	60	40	18	65	40	19
Requiremqnt 8	60	60	16	60	40	18	80	40	22	55	50	16
...	50	80	12	5	95	1,5	90	25	25,5	55	70	14

Figure 28. Output of interviews divided into different groups of customers, one for each global area.

4.3.1.3 Innovation strategy

The example in figure 29 shows the innovation strategy for a new product. Arrows indicate the changes that will characterize the new product. The general objective is to have a greater satisfaction for each requirement, preferably for the ones with the highest innovation potential (see figure 27). In the example of figure 29, a strong cost reduction is used to strongly increase the satisfaction of the product. To do that, it is also acceptable to slightly reduce satisfaction for other requirements with lower market potential and lower importance.

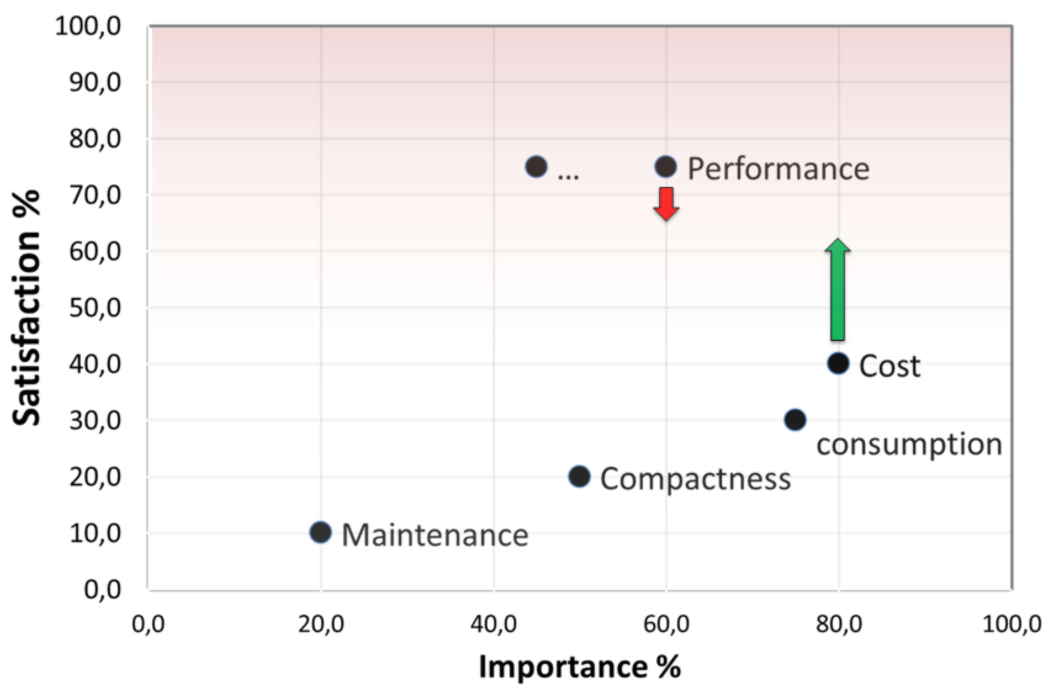


Figure 29. An example of innovation strategy.

One or more innovation strategies have been proposed to the "project manager" and the other persons that were responsible for investments. Although this phase can be organized with interviews and further evaluation, in this case study, the final decision was discussed in two meetings with R&D managers from the different departments.

4.4 DECISION-MAKING – SELECTION OF THE TECHNICAL SYSTEM

Every time we need to develop a new technical system or improve an existing one, the evaluation and the identification of potential alternative technologies assume a central role. In conventional decision-making strategies, the decision to invest in one or more alternative technologies is performed by weighting benefits and risks of each alternative. Both the identification and weighting of alternative technologies are usually performed without a systematic methodology, relying on the experts' knowledge and somehow on the unquestionable judgment of leaders. The main risks of these approaches are the strong subjectivity of the evaluation and the strong dependence of alternative technologies from experts' knowledge. In the worst case scenario, this situation may lead to an ineffective investment. The procedure is the same to the one for the definition of the innovation strategy.

In this exemplary case study, the author considered sensors for monitoring the level of pressure in a gas circuit breaker. First, we list all the requirements of the sensor: cost, maintenance, compactness, performance, sensitivity, calibration, selectivity, response time, data transmission, data security, precision, power consumption.

Using KOMPAT software we perform a technological landscaping; a list of potential physical effects/technologies that can be used to measure the pressure, as shown in figure 30. This output is then converted into research targets for planning the info gathering phase in patent, non-patent literature and brochures. Figure 30 shows a partial list of alternative technologies for measuring pressure inside gas insulated circuit breakers. In yellow nine technologies already present into the gas breakers field, and in green three potential new physical effect to be transferred from other fields.

4.4.1 Case study

4.4.1.1 Information gathering

The aim of this phase is to deeply analyze all requirements (by combining knowledge search and problem solving activity), collecting all information useful to compare technologies looking at a single requirement at time

For example, in the "performance" requirement analysis, we identified over than 400 patents. Figure 31 shows the acceleration in the last 5 years of electrical and vibrating devices. Time distribution of main players allows understanding who is still working in very recent years (e.g. Hitachi and Siemens), and who has peak activity only in the past (Meidensha, Toshiba and GE).

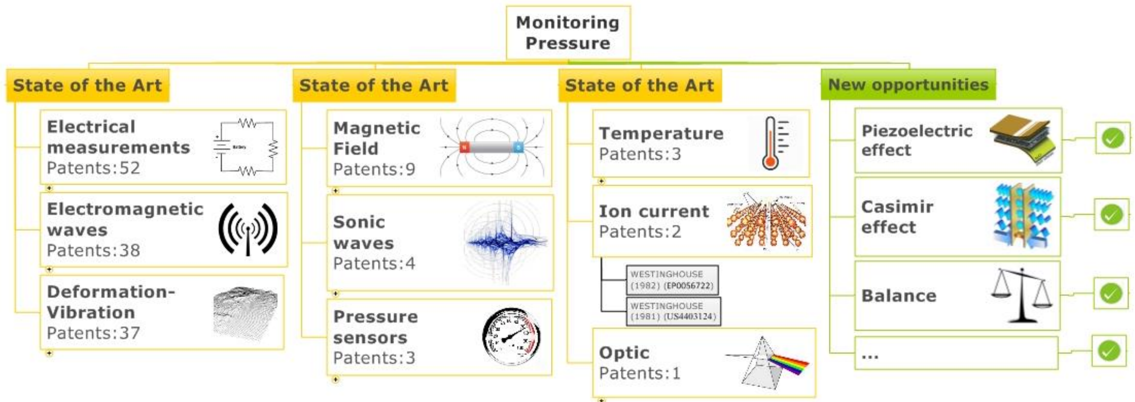


Figure 30. Pressure sensors – State of the art and future technologies.

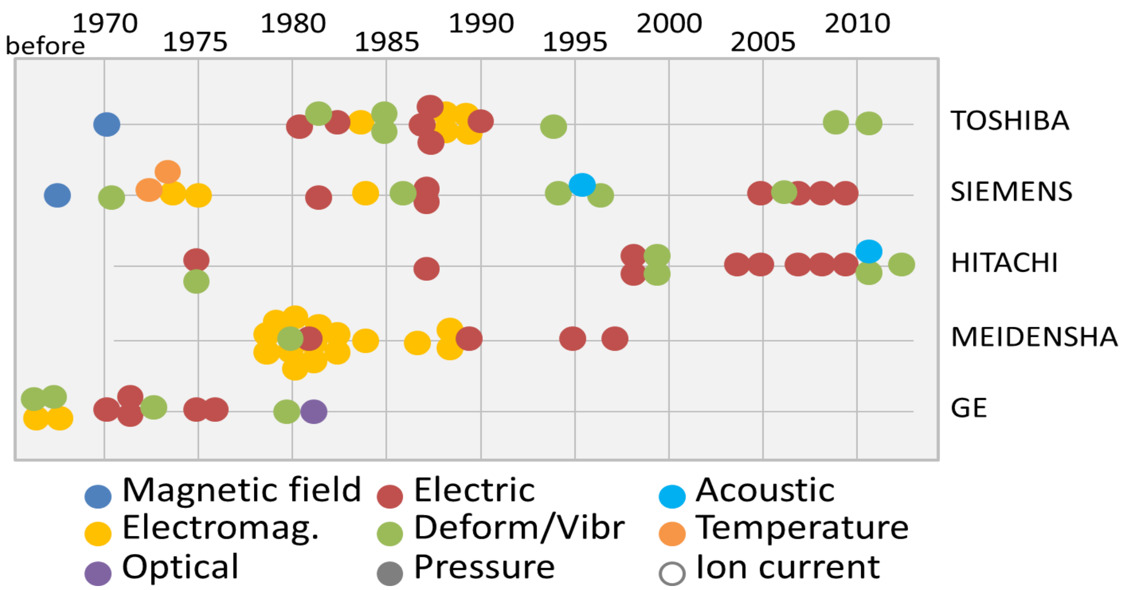


Figure 31. Performance of pressure sensors - Time distribution of technologies for the TOP 5 players

A different analysis has been provided for compactness. Patents and brochures have been scanned in order to identify the size of different devices proposed in the different circuit breakers size (rating currents: 400A, 600A, 800A, 1200A, 1600A, 2500A, 3000A, 4000A, 8000A and rating voltages: 3,6kV, 7,2kV, 12kV, 15kV, 17,5kV, 24kV, 27kV, 36kV, 40kV). Figure 32 suggests results only for 7,2kV, 2500 A.

Gas insulated CB - 7,2kV, 2500 A					
Electric	Elecrtomag.	Deformat.	Magnet.	Acoustic	Optical
5-10 cm^3	2-3 cm^3	8-12 cm^3	2-3 cm^3	3-5 cm^3	6-7 cm^3
Ion	Temperat.	Pressure	Piezoelec.	Casimir ef.	Balance
6-7 cm^3	<1 cm^3	3-5 cm^3	< 1 cm3	2-4 cm^3	6-7 cm^3

Figure 32 Compactness of pressure sensors (partial) for version 7,2kV – 2500A.

Figures 32 and 31 are just an example, many other graphs and tables have been provided for giving a concise perspective on the position of each technology. Similar analysis has been repeated for all others requirements.

4.4.1.2 Requirements evaluation

An importance-satisfaction map can be generated for the technology used as the reference technology. All others alternative technologies are then mapped in a comparison table, as shown in table 13, and then quantitatively evaluated by a unique score calculated by a weighted average between satisfaction and market potential. For privacy reasons, values in table 13 are only indicatives.

Table 13 Comparison between pressure sensors technologies, values are modified for confidential reasons

Requirements	Market Potent.	Electric Measur.	Electrom. Wave	Deform. - Vibration	Magnetic field	Sonic waves	...
1 Power Consump.	3,0	80,0	70,0	90,0	50,0	80,0	...
2 Cost	4,5	75,0	75,0	75,0	80,0	85,0	...
3 Maintenance	10,0	100,0	80,0	60,0	70,0	70,0	...
4 Compactness	23,0	20,0	90,0	40,0	90,0	80,0	...
5 Sensitivity	16,0	40,0	80,0	60,0	60,0	70,0	...
6 Calibration	14,0	60,0	30,0	40,0	80,0	30,0	...
7 ...	...	...	...	...	...	...	...
PRODUCT VALUE	77,0	86,0	78,0	82,0	69,0	73,0	...

4.5 Discussion

Existing methodologies rank requirements on the base of importance and satisfaction values. Both the identification and weighting of requirements are usually performed without a systematic methodology, relying on the experts' knowledge and somehow on the unquestionable judgment of leaders. The main risks of these approaches are the strong subjectivity of the evaluation and the strong dependence of the evaluation with experts' knowledge.

The presented methodology combines knowledge search and problem solving. First, knowledge of experts is integrated with knowledge extracted from patents, market analysis, scientific literature and commercial literature. Second, the generation of new alternative technologies is supported with a systematic theory of problem solving and knowledge transfer. Third, decision-making and the definition of an innovation strategy are supported with a concise diagram that summarizes the gathered knowledge and facilitates the assessment of each requirement. Gathered knowledge and problem solving foster the ability of experts to identify and rank requirements. In some cases, completely new technologies or solutions are identified to be suitable with the considered application. The graphical summary allows experts and leaders to have a comprehensive and fast overview on the situation, increasing awareness and consistency of decision-making during the interviews. Some typical situations have been recorded on how the provided information changed the judgment of a requirement.

Although not specifically demonstrated, subjectivity is likely to be reduced thanks to shared knowledge. A general problem solving activity allows the construction of different scenarios that are useful for the evaluation, while the laws of technical evolution allows a better understanding of future drivers. The selection of personnel for the interviews is structured to facilitate communication between different departments of the same firm.

A first limitation of this analysis is similar to many information retrieval processes. It implies the availability of the needed information. Mentioning the direct experience coming from the aforementioned case study, selling volumes would have been very useful to improve the analysis, as well as other marketing searches that were not included for confidential problems. Among them, a structured voice of customers and interviews with customers were missing. In this way, information is somewhat filtered through the company's personnel.

Another limit of the evaluation is to consider requirements as independent from each other to facilitate the judgment of experts. This limitation is partially overcome by the visualization of requirements in the importance-satisfaction graph, that permits to easily identify trade-off and indirectly consider influences among requirements. However, there is not a specific way to effectively manage the complexity of requirement's correlations.

After several academic case studies, the proposed methodology has been applied in a big multinational firm for three different products, with encouraging results. As side effect, these activities encouraged the spread of theories for systematic innovation inside the company, with a series of planned TRIZ and Spark courses.

5 PROPOSAL: A NEW SET OF GUIDELINES FOR IDEA GENERATION IN SPARK

In this chapter I propose an adaptation of the idea generation step of Spark for problems of unsatisfactory actions. I analyze guidelines for the generation of conceptual solutions. Then, I define a practical ontology that allows the organization of the information contained in existing guidelines in accordance with a type of problem and make them suitable for a software implementation. This ontology is used to reorganize the 76 standard solutions for problems of unsatisfactory actions and may allow the unification of idea generation tools, avoiding overlapping suggestions and improving completeness.

Several studies on idea generation involve visual stimuli [79,80], a combination of textual and visual stimuli [81] and textual stimuli [76,83]. In general, they proved a positive effect on creativity. Alongside purely linguistic methods for idea-generation, there are several more structured methods, which provide complex guidelines or checklists. In this class of stimuli, it is difficult to make overall considerations due to the big number of factors involved. However, these methods are largely used in industry and their effectiveness has been proved in several case studies.

Among them, there are highly free methods, such as the checklist of Osborn [49], elaborated by Eberle [142] with the name of SCAMPER (acronym of Substitute-Combine-Adapt-Modify-Put to other use-Eliminate-Reverse); and there are strictly guided methods, such as the 76 standard solutions [97], the 40 Inventive Principles [94], Synectics [34], ASIT (Advanced Systematic Inventive Thinking) [86,104], and many others [88].

5.1 FEATURES OF GUIDELINES

A practical set of parameters have been used to highlight differences among different set of guidelines: where they act onto the problem space, their level of detail and complexity, the presence of examples, if they are "problem-based" or "problem free" and if they are supported by graphical models.

5.1.1 Goal state and current state

The activity of a human problem solver can be explained with the mental reconstruction of a problem space and an activity-based manipulation of the problem space [109,143]. Since a problem can be defined as the difference between a goal state and a current state [109], there will be no problem space without discrepancies between "what I want" and "what I have". Furthermore, there will not be a problem solving process without a manipulation of the problem space. A guideline may influence the mental representation of current state and goal state, as well as the transition from current state to goal state. Thus, we define three space of intervention.

- Goal state (GS): a guideline acting on the goal state identifies, defines or changes goals. Typical guidelines acting on the goal state are "What if you combined purposes or objectives?" or "Define the desired action". Thus, it is clear that designers are stimulated on generating or identifying objectives.
- Current state (CS): a guideline acting on the current state identifies new entities or attributes of entities to make designers aware of their existence. Typical guidelines acting on the current state are "What can be blended, mixed, or included?" or "Define the problem objects". In these cases, it is clear that designers are stimulated on generating or identifying new problem elements.
- Transition current state->goal state (TCG): a guideline acting on the transition from current state to goal state describes how the goal state can be achieved through the manipulation of known entities of the current state. A typical guidelines acting on this transition is "imagine the object X performing the wanted action Y". Thus, designers are stimulated on using entities already present in the current state to reach a goal state.

5.1.2 Structuredness and complexity

Guidelines can somehow inherit the characteristics of problems, such as structuredness, complexity and abstractness. Excluding random guidelines, we may suppose that a guideline will be designed to decrease its complexity and increase its structuredness, while they can be more or less abstract depending on the type of problem that they want to address. I did not find an objective way to keep the aforementioned three characteristics as separated, therefore, they are evaluated together. On one hand there are complex and structured guidelines, on the other hand there are simple and unstructured guidelines. No evaluation is considered for abstractness.

A minimal guideline is a "trigger word", where a verb, adjective, concrete or abstract nouns can be provided. For instance, "modify", "length", "pen" or "high". A more detailed guideline can be in the form of Verb-Object, without specifying particular features (such as "modify the pen"). A higher level of detail can be achieved by mentioning a specific feature or a way to do what you need. For instance: "Modify the shape of the pen to perform the wanted action" or "Modify the pen to perform the wanted action, adding a substance inside the pen".

A more structured guideline helps in avoiding ambiguities and finding solutions, but it is not always good for creativity since it can lead to design fixation [144]. Thus, there is a conflict about structuredness; i.e., a guideline should be structured to prevent tedious ambiguities, but it should not be structured to allow lateral thinking and creativity.

The advantages of un-structured methods are the freedom of interpretation, allowing a divergent thinking that may lead to a solution with a high level of novelty. However, this kind of approach may not be effective in practical problem solving; the risk of solving the wrong problem is always present. This statement comes from a generalization of [145] conclusions, where brainstorming, SCAMPER and Functional Analysis have been compared: "intuitive methods provide more novel outcomes, while the most useful outcomes are achieved with the use of more structured methods".

The description of an example may be considered as a very structured and complex guideline; however, the author considered examples as a different part of a guideline, and the possibility of using examples in an idea-generation method has been chosen as a separated parameter of evaluation. Usually, guidelines are intended to be abstract to prevent fixation [144] and obtain more novel results [146], but an example to clarify a guideline may be unavoidable to achieve a better understanding.

Belski et al. [147] compared the outputs of different methods based on suggestions with different levels of structuredness and complexity (completeness). The considered methods were a set of 8 random words, the 8 fields of MATCEMIB (Mechanical, Acoustic, Thermal, Chemical, Electrical, Magnetic, Intermolecular and Biological) and a more detailed extended MATCEMIB classification, called MATCEMIB+ (Belski et al. [148]). The experiment was conducted with the first year engineering students in Australia, Czech Republic, Finland and Russian Federation. The first interesting result of the test is that students that used a more structured set of guidelines like MATCEM statistically propose more creative solutions than the students that did not use any tool (control group). Students that applied random suggestions produced similar results if compared with the control group. Students that used MATCEMIB+ did not appreciably improve results if compared with students that use MATCEMIB. The test seems to confirm that a major degree of complexity (completeness) and detail of the suggestions do not necessarily improve results, at least if they are used with the same limits of time.

5.1.3 Problem types and variety

A set of guidelines can be developed for a specific type of problem or it can try to address many types of problem. To explain this concept we adapted the meaning of variety used for the evaluation of creative solutions of idea generation activities [149]. As variety for solutions is a measure of the explored solution space during the idea generation process [149]. Variety for a set of guidelines is a measure of the extension of problem typologies that can be addressed.

After many studies about a general schema for problem solving, many authors seem to agree that effectiveness of schemas for problem solving depends on the type of problem [109], and although there are methods which can be used more frequently than others [150], each type of problem should have its customized set of guidelines, and a set of guidelines is more or less effective depending on the type of problem to be solved. In general, a set of guidelines for idea-generation can be:

- Problem-free: it addresses a general technical problem and boundaries are not clearly specified;
- Single Problem-based: it is contextualized for a specific type of problem;
- Multiple Problem-based: it is contextualized for more than one kind of problem;

For example, the 76 standard solutions use "If-Then" conditions to identify the type of problem, and they are expressed in one or more sentences, such as "If there is a SFM (Su-Field model) which is not easy to change as required…". The first part identifies the SFM as the involved entity in the problem. The second part specifies a transformation, i.e. a required change on the SFM. In this way, initial state and main goal are respectively defined as "a SFM" and "a modified SFM". Generalizing this approach, the identification of the problem can be obtained defining the necessary entities for the existence of a problem and their required transformations.

Other methods do not explicitly mention a condition; however, some conditions are implicitly present. For instance, a quite intuitive method such as SCAMPER cannot be applied if there is not a product to improve. Thus, the existence of a product and the need to improve it are necessary conditions for the application of the method.

In the author's opinion, the initial state and the main goal should be explicitly clear. For each type of problem there can be different guidelines, although the structure with which they are suggested may be similar.

5.1.4 Models and schemas

The use of models, schemas, or merely external representations is a good support for design in general. There are several reasoning schema for design, such as function-behaviour-state [22], TOP model, Energy-Material-Signal model [29] and others. Although there are many of them, they are concentrated on the analysis phase, and they rarely support guidelines for idea-generation; moreover, just a few of them allow a graphical representation of the solutions. In this category, the Su-field model [24] has been used to support the 76 standard solutions; the Energy-Material-Signal (EMS) has been used for a set of compacted standard [31–33], and TRIZ functional analysis has been used along with some guidelines of "Oxford Creativity", to facilitate "trimming" [27]. Furthermore, a graphical representation of the solution is an important factor, in reducing the cognitive load and facilitating memorization of a guideline.

5.2 A COMPARISON BETWEEN SETS OF GUIDELINES

A comparison between different methods for idea-generation has been addressed. SCAMPER, ASIT, 76 Standard Solutions, 40 Inventive Principles and trigger verbs have been assessed through the aforementioned parameters. These methods have been chosen because they are used in industry, and they are very different from each other.

The main features of each method are summarized in table 14. Each method has its strength and shortcomings. It is clear that a set of guidelines is a compromise between several parameters. Thus, there are some guidelines, such as the 76 standard solutions, which are very complex and very detailed; while others, such as "trigger verbs", are very simple and with a low level of detail. A very detailed guideline is important to give contextualized triggers for an invention; however, a very detailed guideline may lead to design fixation.

Table 14 Comparison between different methods for idea-generation. CS: Current state, GS: Goal State, TGS: Transition, from CS to GS.

	Where they act	**Examples**	**Problem Based**	**Model support**	**Structuredness and Complexity**
SCAMPER	CS+GS	Yes/No	Problem-free	No	2
ASIT	CS+GS+TGS	Yes	Single Problem-based	No	3
Standard Solutions	CS+GS+TGS	Yes	Multiple Problem-based	Yes	4
40 Inventive Principles	CS	Yes	Problem-free	No	3
Trigger verb	TGS	No	Single Problem-based	No	1

Some methods guide all the idea-generation process (CS+GS+TGS), while others guide just one part of it. Just the 76 standard solutions are contextualized for more than one type of problem, and supported by a graphical representation of the problem and solution (model support).

SCAMPER is a method for idea-generation derived from the checklist of [49], the precursor of brainstorming. It is used as a series of questions to provoke designers' creativity and they are grouped into seven verbs to facilitate memorization: Substitute, Combine, Adapt, Modify, Put to other use, Eliminate, Reverse. Some typical questions of SCAMPER are:

- What can be blended, mixed, or included?
- What if you combined purposes or objectives?
- How could you change the shape, look, or feel of your product?
- What could you add to modify this product?

SCAMPER aims at finding new entities of the current state and the goal state. In fact, these questions stimulate the creation of new entities, the identification of hidden entities and the elaboration of different objectives. There are not sentences such as "do this to obtain this"; thus, the transition from current state to goal state is not guided. SCAMPER is a problem-free methodology and it has a wide range of applicability. However, when designers address an idea generation session with a specific objective, where the problem objects and the goal are defined, they may find SCAMPER difficult to use. Thus, SCAMPER is used for a very divergent thinking and when "you do not know what you want". SCAMPER will not give a very detailed indication, resulting in a quite simple sentence. However, since objectives are not well defined, there is freedom of interpretation and ambiguities. This freedom may sometimes be useful to inspire creativity, but surely it is not suited for designers, who must think of an overall goal and contextualize the guideline in their problem. Usually, SCAMPER is not provided along with examples, but its structure would allow examples to be easily implemented.

ASIT [86] is a method for idea generation designed to address a problem of undesired effect. It was derived from SIT (Systematic Inventive Thinking) and it is composed of five thinking tools: unification, multiplication, division and breaking symmetry and object removal. These tools have a similar structure of guidelines. For instance, the main part of unification is:

- Make a list of problem objects
- Define the undesired effect
- Define the desired action
- Imagine the selected object performing the wanted action

ASIT results to be a well-structured and highly guided method. The designer is guided throughout all the ideation process, from goal and current state definition to the transition from current state to goal state. The method is circumscribed to problems of undesired effect and it gives quite detailed descriptions of the conceptual solutions, with medium complexity. Furthermore, it is provided along with examples.

"Trigger verbs" is a method for idea-generation tested by [151]. It consists in the generation of verbs, related or oppositely-related to the functional description of the problem.

Chiu and Shu discussed about the effectiveness of different types of verbs, providing some considerations: lower level (more specific) verbs are more effective, while higher level general verbs are used successfully in conjunction with lower verbs [151]; intransitive verbs are less likely to be used successfully in the development of concepts [151]; verbs similar to the functional description of the problem are less effective than verbs oppositely related with the functional description of the problem [152]. In order to explain this last sentence, it seems that an increased level of novelty, granted by opposite verbs, may be due to the introduction of new entities; not directly related to the problem, but recalled from designer's memory [152].

"Trigger verbs" helps designers to match current state and goal state. However, goal state and current state definitions are quite free and not guided. The level of detail is surely minimal and complexity is very low. "Trigger words" is contextualized through language links and it needs a functional description of the problem; so, it can be considered as a problem-based approach.

The 40 Inventive Principles [94] were born as a limited set of mechanisms that are used by inventors in many patents. They are used as suggestions to reach solutions. They are often used with the matrix of contradictions [24–26] to solve technical contradictions. The matrix of contradiction is used to select the proper set of principles for the identified contradiction, but they can also be used as an independent idea generation tool. Mostly, inventive principles suggest a modification of the current situation, e.g. "Segmentation": divide your object into independent parts; divide your object into parts so that some its part can be easily taken away; increase the degree of the object's fragmentation.

As independent tool of TRIZ, the 76 Standard Solutions were created by G. Altshuller [24] between 1975 and 1985, as solutions for common inventive problems, extracted from the studies of patents. A typical standard (classified as the 1.1.2) has been reported:

"If there is a SFM (Su-Field model) which is not easy to change as required, and the conditions contain limitations on the introduction of additives into the existing substances, the problem can be solved by a transition (permanent or temporary) to an external complex SFM, attaching to one substance of these substances an external substance, which improves controllability or brings the required properties to the SFM".

The 76 standard solutions are the only method, among those analyzed, to be supported by a model (Su-Field model). Thanks to the model, the identification of current state and goal state is very easy, and the match between current state and goal state is guided. The method provides a problem-based approach with very detailed information, examples and graphical representations of the solution.

5.2.1 Type of suggestions

An extrapolation of the ways with which guidelines act on the problem space is here proposed:

- Creating goals: guidelines that are suggesting how to create new goals. E.g. SCAMPER suggest to "Put to other uses", i.e. using existing elements for other purposes.

- Modifications and adaptations of goals: manipulation of goals already present in the problem space. E.g. separations principles are suggested on TRIZ [24] to limit the goals in time, space or under specific conditions.
- Modifications and adaptations of existing elements: manipulation of existing elements of the problem space. E.g. modifying a substance or modifying a field. Detailed guidelines also mention the properties of the substances or fields that should be changed (such as porosity, degree of fragmentation, state, frequency).
- Adding: adding new elements in the problem space. E.g. adding fields and substances. Detailed guidelines also present the properties of the substance (e.g. porosity, degree of fragmentation, geometrical properties) or field (see physical and chemical effects). More specific guidelines can also specify where substances and fields can be found (resources).
- Eliminating elements: elimination of elements that are present in the problem space, such as elimination of substances or fields.
- Combining elements: combine one system with another or a field with another.
- Substituting elements: a sequential combination of eliminating an element and adding a new one.

5.3 ANALYSIS OF THE 76 STANDARD SOLUTIONS

This analysis starts from the 76 Standard Solutions, which has been identified as the most complete and complex set of guidelines. The 76 standard solutions are a very powerful tool of the well-known theory of inventive problem solving. They were created by G. Altshuller [30] between 1975 and 1985 as solutions for common inventive problems extracted from the studies of patents. The set of standards directly derives from the laws of technical systems evolution, guiding the synthesis and transformation of these systems by implicitly eliminating technical contradictions [25]. In practice, it is usually used in ARIZ (the algorithm for inventive problem solving) [97] as part of the Substance-field analysis, after the Su-field model has been built and any constraints on the solution have been identified [153].

Although both Su-field model and standard solutions are unquestionably effective tools for innovation, their use has been somewhat limited, especially in companies and Western universities. The reasons are different, starting from the influence of translators' interpretation [154] to the shortcomings of the instruments themselves [26,155].

Many authors have attempted to fill these gaps, pointing out some difficulties in applying the standards properly and in making them attractive for less experienced users. Therefore, the need to modify this powerful tool arises. Many examples from the TRIZ community (such as V. Petrov, V. Souchkov, N. Khomenko, A. Smirnov, I. Belski, Z. Royzen, S. Savransky and others) worked on this path. Some of them defined guidelines or a new classification while others built new methods starting from the standards or joining them with other innovation tools.

5.3.1 The Su-field analysis

The Substance-Field analysis is a TRIZ methodology composed of a substance-field modeling phase, an abstract solving phase and an interpretation phase [156]. This approach is consistent with the TRIZ way of thinking: it proposes to solve the problem with a high level of abstraction, regardless of the particular technical field and exploiting the knowledge of previous similar problems.

As part of the method, Su-field model is a graphical representation of a technical system. It can be presented through three concepts (see figure 33) substance, field, and mutual interaction.

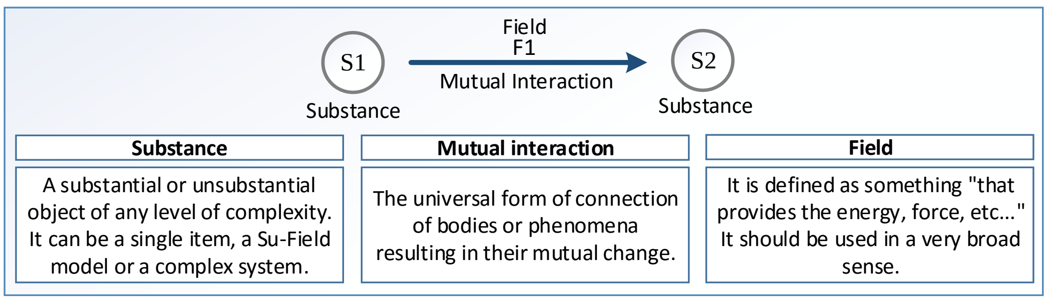


Figure 33. A part of Su-Field model ontology [24,156,157].

The original notation of Su-field model is quite different from the notation used in this thesis, which reflects the common interpretation of functional analysis and Tool-Object-Product model [28]. Since the Su-field model is used to explain the 76 standard solutions, these differences may lead to some ambiguities. In particular, for the original notation (see figure 34 on the left) the dashed line means "action (or interaction) which should be introduced according to the specification of the problem" while the curved line means "unsatisfactory action (or interaction) which according to the specifications of the problem has to be replaced" [24]. Following these definitions, excessive action, insufficient action and harmful action of the functional-analysis notation (see figure 34 on the right) seem to be a subclass of the original curved line.

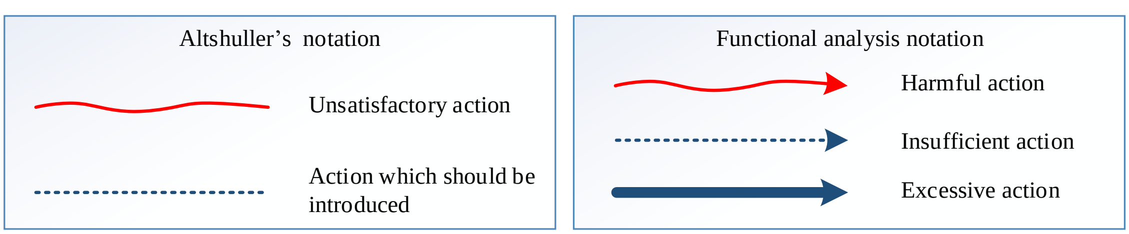


Figure 34. Original Su-field model notation (on the left) and functional analysis notation (on the right).

More confusion arises on the definition of a substance-field triad (or complete Su-Field). In the original concept of triad Altshuller depicted a field and two substances but he also implied a second field for the mutual interaction between the two substances. In fact, an example from [24] shows a wedging device for cladding consisting of a wedge S1 and a wedge cladding S2 designed to facilitate the removal of the wedge; the wedge is made in two parts, one of which is easily melted. The heat field F1 acts on S2 by altering the mechanical interaction F2 between S1 and S2. According to Altshuller's approach mutual interaction between substances is indicated without detailing the form of mutual interaction (see figure 35).

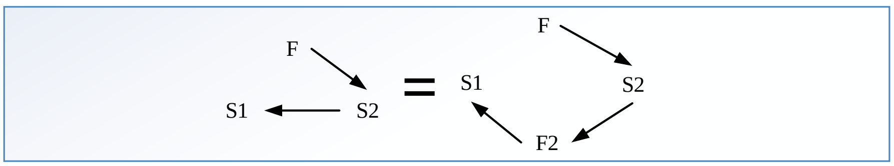


*Figure 35. Altshuller's triad notation **[24]**.*

In this way he focused on the field that could be directly controlled, without mentioning how to generate it. In the last decade, some TRIZ experts have used the Su-field triad in a way more congruent to functional analysis [27,157], using the field F1 as the one between the substances (see figure 33). In this thesis, the adopted notation is similar to the one of functional analysis, but the interpretation of the original standards has taken into account both points of view.

5.3.2 Discussion on Altshuller's Standard Solutions

The first five standards were developed in 1975 by G. Altshuller [158]. They were presented with a theoretical explanation without a classification. In approximately two years the number of standards increased from five to ten [24].

A qualitative leap in the development of standards can be identified in 1979, when a system of 28 standards was published [98]. The system consisted of three classes: standards to change systems; standards for detection and measurement; standards for the use of standards.

A second significant improvement in 1981 led the number of inventive standards to 50, preserving the same three classes but arranging the subclasses in a more logical way. Meanwhile, the current system of numbering was adopted, moving from sequential to structured. The system was organized in three digits: the number of the class, the number of the subclass and the number of the standard for each subclass. The number of classes was not changed until 1985, when the current system of 76 standard solutions was presented [30]. It is structured in five classes [25]:

- Composition and decomposition of SFMs
- Evolution of SFMs

- Transitions to supersystem and microlevel
- Measurement and detection standards
- Helpers

5.3.3 Criticisms and improvements after G. Altshuller

Since 1985, the system of standards was criticized from the TRIZ community and some shortcomings were highlighted. The structure of the standards with five classes complicates their use and appears less logical than the system with just three classes [155,158]. Many standards are equal to the trends of technique evolution, but they are not a consequence of all the laws of system development [26,158]. People learning TRIZ still must do a lot of case studies that illustrate the principles of TRIZ using terms and technologies before using Inventive Standard correctly [159]. "Although inventive standards are more specific than 40 Inventive Principles, their application requires more learning and practice" [160]. The system of standards is not applied in all fields of the known physical, chemical, biological and geometrical effects [158] and is difficult to expand. It is inhomogeneous [26]: some standards consist in a special case of more general standards leading to frequent repetition. In particular, too much attention is given to the introduction of a magnetic field, which is a special case of the introduction of a field. Most standards can be modeled with Su-Field model, but some of them are beyond the scope of symbolic description of Su-Field. Besides, the formulation and structure of various standards are different and inconsistent. Finally, this kind of classification is not always suitable to guide the choice of the proper standard for a specific problem. A new classification should use the classes to identify the problem instead of classifying the type of solution.

Starting from these assumptions, a great effort has been made in order to improve the system of 76 standard solutions. The problem has been addressed in several ways from a lot of TRIZ experts. At first, some guidelines in the form of flowcharts were developed to facilitate the use of the standards in general or particular situations [161,162]. Afterwards, some more radical proposals were made. For instance, V. Petrov has worked on a system of generalized models [96], by interpreting the standards as the mechanisms of the laws of evolution of systems. Other authors [31,33,155] have generated their own set of compacted standards. In particular, they reduced the number of standards as well as the number of classes, which became three: improving the system with little or no change; improving the system by changing the solution; detection and measurement. Ogot also suggested a different modeling tool: the Energy-Material-Signal model [32].

In order to replace the ineffective numbering, Kim has suggested a new notation to represent both problematic situation and solutions [163].

Meanwhile, Gadd [27] has worked on the reformulation and reclassification of the standards in three classes: harm 24 solutions; insufficiency 35 solutions; measurement 17 solutions. In a similar way, Mann [120] has proposed four classes of standards:

incomplete Su-Field; measurement/detection problem; harmful effect; insufficient/excessive relationship.

All the aforementioned proposals are very effective improvements of the classical classification and formulation of the standards, however, some of them strayed far from the original Altshuller's idea. They reduced the strong link between standards and Su-field model, which was part of the greatest Altshuller's discovery [164]. Besides, in order to simplify the use of the standards, they reduced the amount of information they contained or the flexibility of this innovation tool.

5.3.4 Information on the Standard solutions

From an analysis of the standards we can observe that they contain information to perform both problem description and problem solving. The part of problem description defines a generic problem in terms of objectives, constraints and initial conditions while the part of problem solving contains suggestions for the solution. The information can be found in form of text or graphical Su-Field. All this information is structured in the original system of standards (see figure 36).

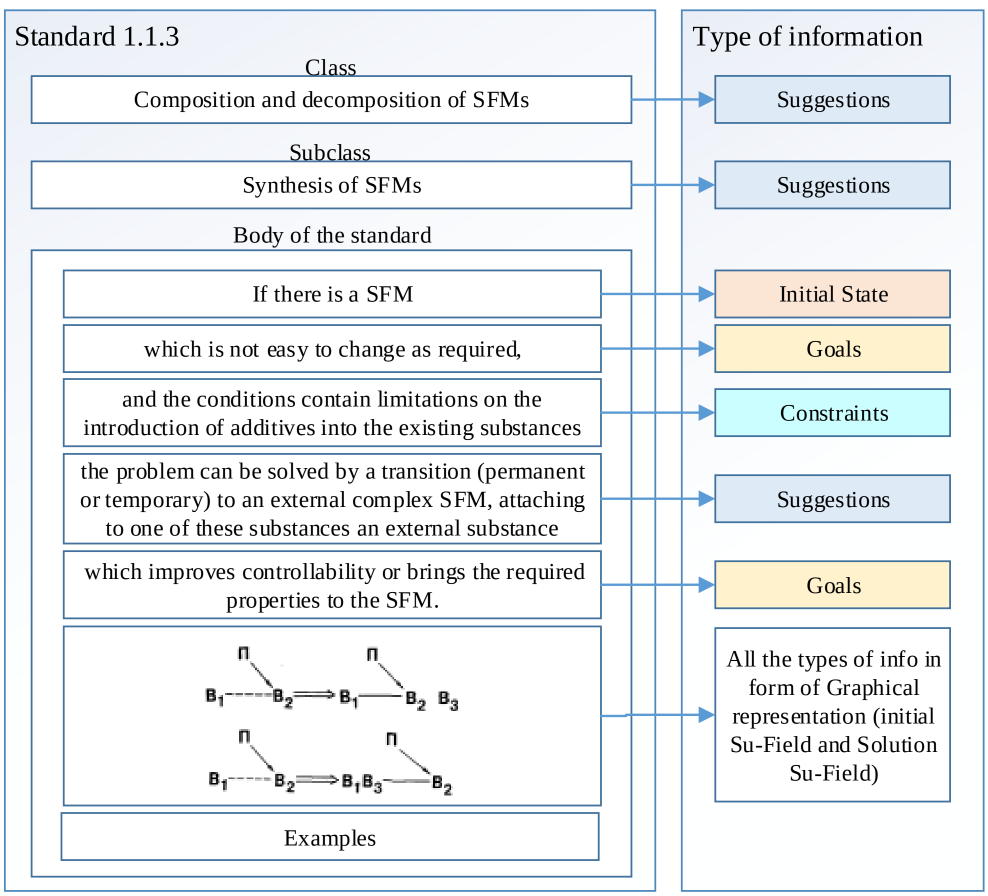


Figure 36. Type of information in the standard 1.1.3 of the original system of standard solutions.

A standard can be identified with its class, subclass and body. With the exception of the class "measurement and detection standards", all classes and subclasses are named according to the type of solution they lead. So, they can be considered as textual suggestions for the solution.

The body of a standard contains a first condition to identify objectives and/or initial conditions. A second condition and eventually a third one are used to identify constraints of the problem. The rest of the body contains suggestions for the solution, objectives, a graphical representation of initial Su-Field and solution Su-Field model and some examples.

The information contained in the standards is fragmented and complicated. Consequently, interpretation and reading of a standard are difficult. The use of a standard is given according to the initial conditions, which are expressed as if <condition> then. This approach is inefficient and tedious. In fact, many authors have proposed approaches to simplify the application of standards, proposing an external schema based on a functional diagram [165]. However, many difficulties have been reported to organize all the information contained in the original system and to manage the correct use of the class "Helpers".

5.4 An ontological framework to structure design guidelines

A framework to systematically structure a set of guidelines has been extrapolated from the discussion of the first three sections and is shown in figure 37. The attempt of the author reflects the willingness to use what can be deduced from a logical point of view (Sub-goals) with what can be derived only empirically (Suggestions).

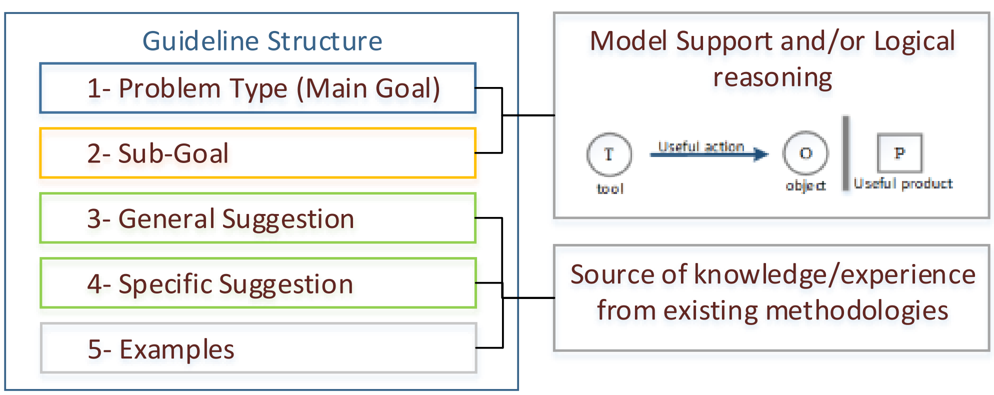


Figure 37. The structure of a guideline for problem solving.

5.4.1 Problem Type

When a set of guidelines is being analyzed, we must understand the necessary conditions to use it. Accordingly, the first box represents the identification of the problem, i.e. the identification of the initial state and the main goal. For instance, we can use the TOP model [28] to identify the problem. A TOP model can represent a problematic situation (see figure 38) and can be easily adapted to symbolically represent solutions. By using a model, the initial state is well defined and the identification of the problem is more intuitive.

Practically, the TOP model limits the boundaries of applicability to problems that can be described with an Action, a Tool (substance that generates the Action), an Object (substance that is subjected to the Action) and a Product (substance that is generated from the object after the application of the Action). I.e., the model visually describes a sentence such as "T acts on O obtaining P".

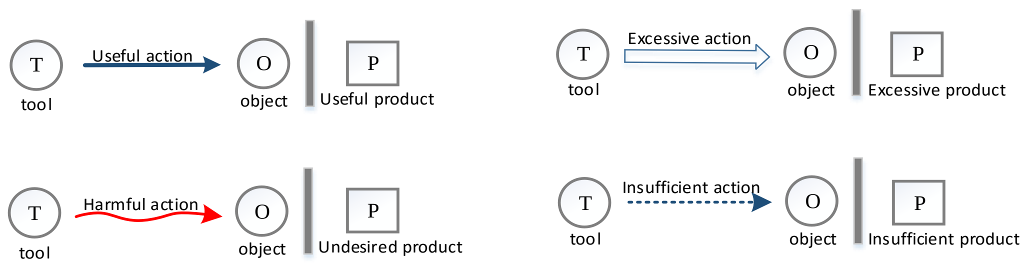


Figure 38. Identifying problems with TOP model.

Depending on the type of product, TOP model can describe three types of problem:

- Excessive action (when the product of the action is excessive)
- Insufficient action (when the product of the action is insufficient)
- Harmful action (when the product of the action is undesired)

These problems are related with an explicit main goal:

- Excessive action->I want to reduce the product of an action
- Insufficient action->I want to increase the product of an action
- Harmful action->I want to avoid the undesired product of an action

Actually, TOP model may be used to describe problems of missing actions, but this is not discussed on this thesis. Thanks to the identification of a problematic TOP model, the initial state (in the form of necessary entities for the existence of the problem) and the main goal of each problem are clearly defined.

5.4.2 Sub-goals

The second box of figure 38 represents the Sub-goals of the identified problem. Sub-goals are defined as an elaboration of the main goal and represent conceptual solutions to the given problem. They are valuable alternatives that can be found through a better understanding of the goals or a cause-effect analysis on the identified problem. Thus, the Sub-goals are a more precise way to describe "what I want" and can be interpreted as an elaboration of the goal state.

A simple cause-effect analysis has been performed to understand necessary and sufficient changes to solve the problem. When dealing with harmful actions, the main goal is "avoiding the generation of the undesired product", thus, there can be several logical Sub-goals:

- Prevent the action to be generated -> make the tool unable to produce the harmful action;
- Prevent the action to propagate -> block the harmful action or deflect the harmful action;
- Prevent the action to produce the undesired product -> make the object insensitive to the harmful action;
- Prevent the product to be harmful -> make the product useful or not harmful.

For insufficient action, the main goal is "increasing the product of an action":

- Improve the generation of the action -> make the tool more effective on producing the action;
- Improve the propagation of the action -> enhance the action;
- Improve the effect of the action -> make the object more sensitive to the action;
- Improve the insufficient product -> make the product sufficient.

For excessive effect, the main goal is "reducing the product of an action":

- Reduce the generation of the action -> make the tool less effective on producing the action;
- Reduce the propagation of the action -> reduce the action;
- Reduce the effect of the action -> make the object less sensitive to the action;
- Reduce the excessive product -> make the product sufficient.

Using the TOP model, the Sub-goals can be supported by a symbolic representation, reducing cognitive load and facilitating memorization.

5.4.3 Suggestions

Suggestions (third and fourth boxes of figure 38) explain the feasible manipulations of the current state. The main difference between a Sub-goal and a Suggestion is that a Suggestion is not a conceptual solution, but merely a possible change of the current situation. While a Sub-goal logically follows from the type of problem, a suggestion can statistically or intuitively help in solving a problem. For instance, if we want to increase the acceleration of a body, we can use the Newton's formula $F = ma$ to define two Sub-goals: "reduce the mass of the body without changing the force" and "increase the force without changing the mass of the body". These are directly and logically related to the main goal of increasing the acceleration of the body. Instead, a suggestion can be "reduce the volume of the body" or "change the temperature of the air", which are not logically related to an increased acceleration of the body, but they may work as triggers for a practical solution.

Suggestions explain manipulation of the current state: i.e., manipulation of the problematic TOP model.

In order to manage a big number of suggestions they are grouped in General Suggestions. A General Suggestion takes the form of verb and object, in order to identify all the operations that can be performed on the elements of the problem, such as “modify a substance”, “add a substance”, “merge a substance with another substance” and so on.

Each General Suggestion is supported by a group of Specific Suggestions. A Specific Suggestion answers to the question “how can I carry out the General Suggestion?”. Thus, for the General Suggestion “modify substance”, the Specific Suggestions will reveal some ways of “modifying a substance”, such as “divide the substance in more parts”, “make the substance flexible” or “change the form of the substance”.

Since the Suggestions are simply triggers, their content may be filled with every source of knowledge; in the case study of the next section, this knowledge has been taken from the 76 standard solutions (see figure 39).

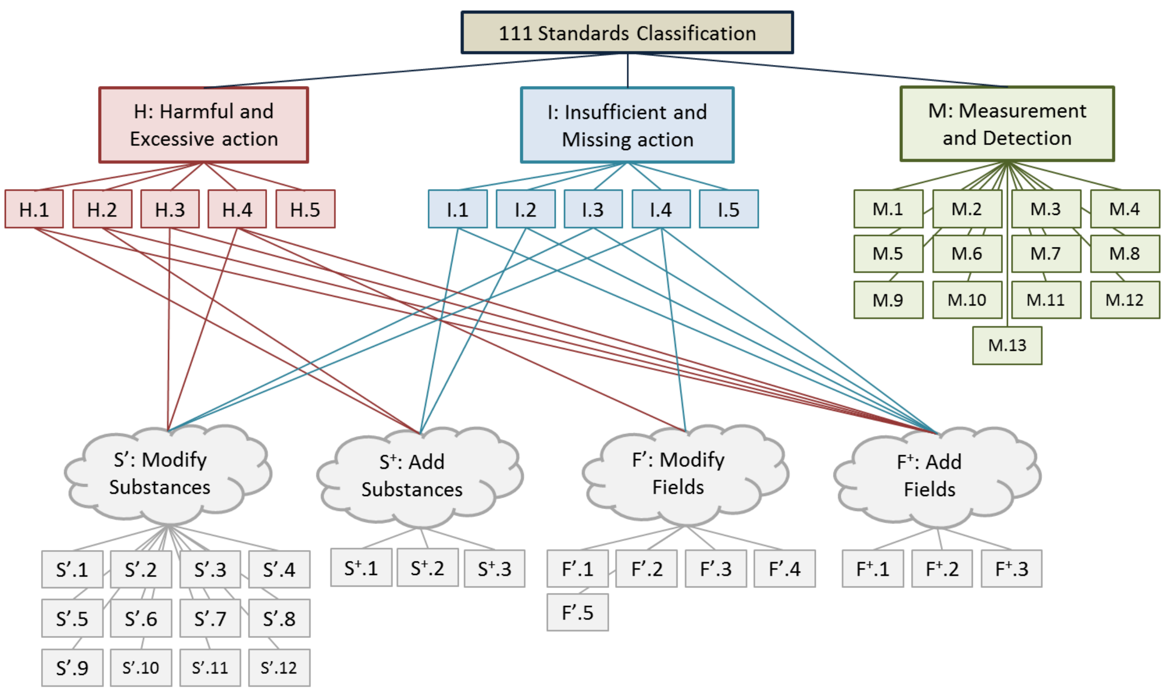


Figure 39. Schema of the 111 Standards.

5.5 A NEW SET OF GUIDELINES: 111 STANDARDS

The 76 standard solutions do not contain Sub-goals, and the identification of the problem is not adapted for a functional approach. For this reason, the 76 standard solutions can be ameliorated with the new proposed structure, using the TOP model to identify the problem and the Sub-goals to improve efficacy, clarify objectives and possibly increasing the paths for the solution.

The 76 standard solutions exploit the substance-field ontology, which is composed of two main concepts: substance and field [156]. Thus, we can extrapolate four types of General Suggestions: "add a new substance", "add a new field", "modify a substance" and "modify a field".

Since Tool and Object of the TOP model can be considered as substances, the General Suggestions can be easily introduced in the proposed structure. Thus, the General Suggestions and the Sub-goals are combined to form an almost complete guideline. From a logical point of view, some General Suggestions cannot be associated with all the Sub-goals; consequently, each Sub-goal is supported by a maximum of three General Suggestions. The resulted Sub-goals and their General Suggestions for a problem of harmful action are reported in figure 40.

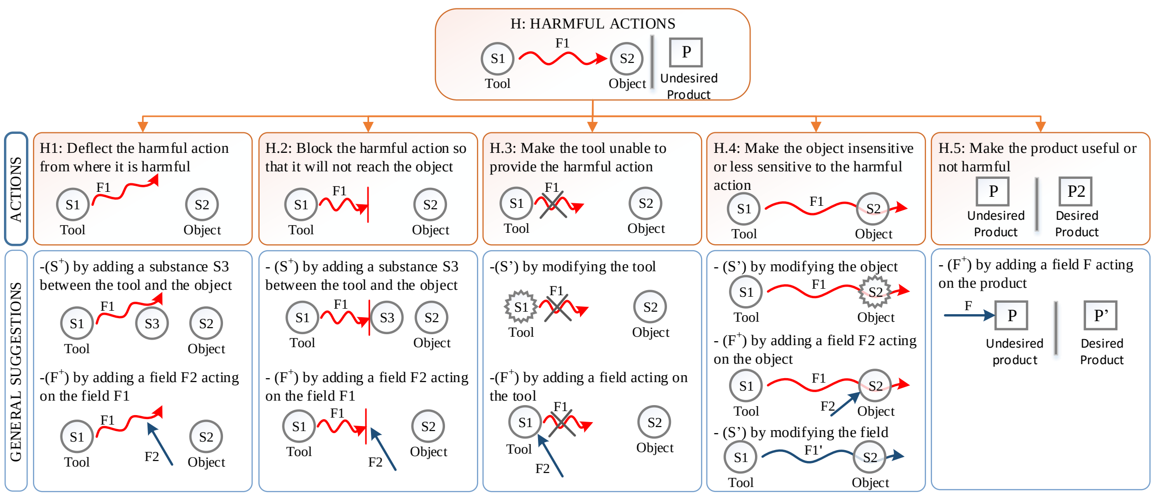
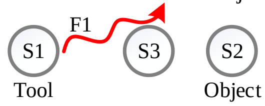


Figure 40. Sub-goals and General Suggestions for a problem of harmful actions.

Each General Suggestion can be supported by one or more Specific Suggestions, which answer to the question "how can I carry out the General Suggestion?". Thus, extrapolating the knowledge from the 76 standard solutions, selecting the feasible Specific Suggestions for each General Suggestion (see the example of figure 41) we filled the fourth part of the guideline.

GENERAL SUGGESTION

$-(S^+)$ by adding a substance S3 between the tool and the object



SPECIFIC SUGGESTIONS

- Use a substance of the system or the external environment
- Use a substance created by modifying another substance of the system or the external environment
- Use a substance created from the existing substances by exposure to the present fields
- Decompose the external environment, the object itself or the tool, for instance, by electrolysis, or by changing the aggregate state of a part of the object or external environment
- Use substance particles by decomposing a substance of a higher structural level. It is easier to use the nearest higher element
- Use substance particles (e.g molecules) by combining particles of a lower structural level (e.g ions). It is easier to use the nearest higher element
- Add a chemical compound which can be later decomposed
- An additive can be introduced in very small quantities, and concentrated in certain parts
- A substance can disappear or become indistinguishable from a substance that was on the system or in the external environment before
- Add void, it can be also a gaseous substance, like air, or empty space formed in a solid object. Void can be formed by other substances, such as liquids or loose bodies.
- Add void in the form of inflatable substances
- Use a very little amount of a very effective substance.

Figure 41. Specific Suggestions for the General Suggestions "Add a new substance" extrapolated and reorganized from the 76 standard solutions.

Finally, a Sub-goal, a General Suggestion and a Specific Suggestion can be supported by an example. An example can be taken from patents or from the everyday life, and it is presented with natural language and images.

Eventually, since a guideline is a composition of more parts, it can be presented with a flexible structure: for a selected problem, there are several Sub-goals, for a selected Sub-goal there can be several General Suggestions and so on. In figure 42 a complete guideline is reported as a combination of hints of different levels of detail. In this way, the user has a comprehensive overview of the feasible Sub-goals and he can choose the most appropriate path to solve the problem. Moreover, he can use the Suggestions to get more knowledge or triggers.

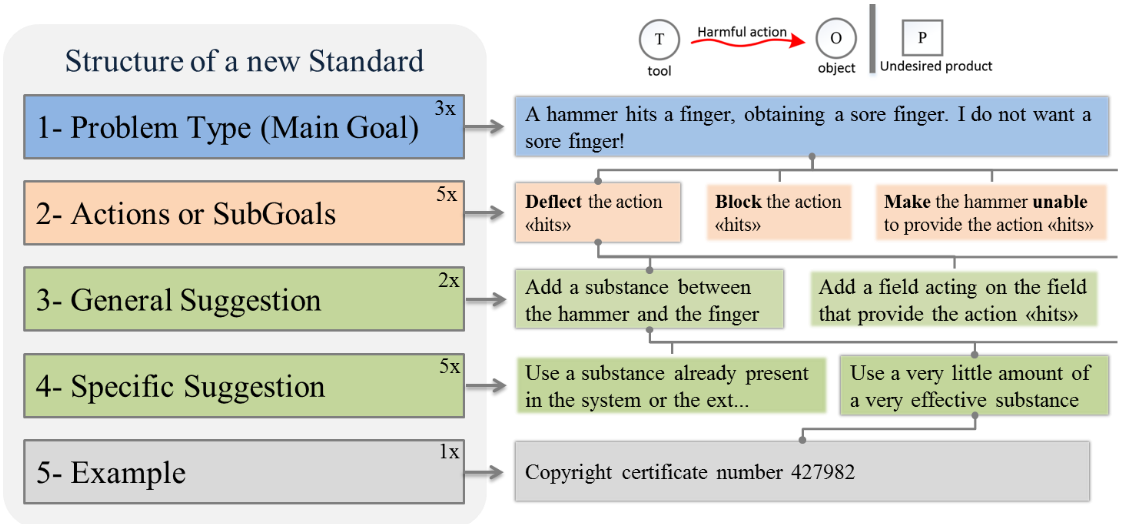


Figure 42. Guidelines as a combination of hints.

The structure of the new set of guideline was used to create a web-based platform (see figure 43).

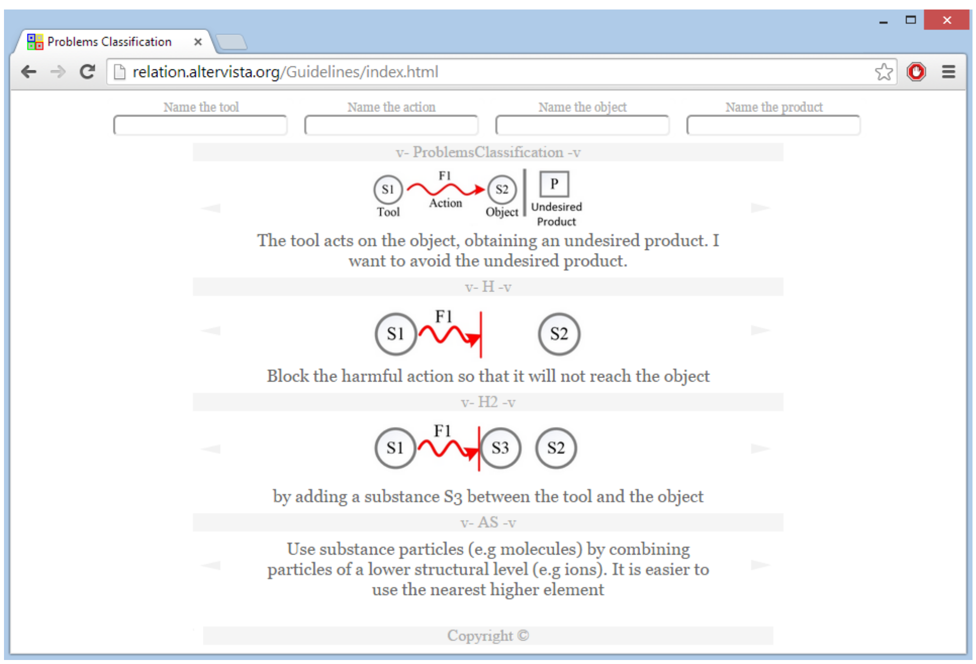


Figure 43. The web-based software for guidelines.

5.6 Test

A test has been designed to understand the possible implications of a set of guidelines that have different levels of detail. The test involved 32 students of the fourth year of mechanical engineering without any background on ill-structured problem solving. The test was composed of three sequential parts:

1. The problem is given without any guideline (time limit: 1 hour).
2. A composition of Sub-goals and general suggestions is given (time limit: 1 hour).
3. Specific suggestions are given (time limit: 1 hour).

The problem was relatively simple so that one hour was even excessive to generate all the alternative solutions. Indeed, with the exception of four students, all finished the first part of the test in less than 50 minutes. All the students finished the second part in less than 45 minutes and all the students finished the third part in less than 50 minutes. Furthermore, the problem was selected so that the formulation of it as a problematic TOP model was explicitly stated in the problem statement (see figure 44).

Figure 44. The problem of the motorized seat.

About the number of solutions:

- During the first phase, the most of conceptual solution were already identified. The average number of solutions was 4.4.
- During the second phase, many students argued that some suggestions were directly referring to some solutions (1.6 in average) they already developed in the first phase.
- During the second phase, new conceptual solutions, which were not present in the first phase, have been identified (3.2 in average).
- During the third phase, there are not new conceptual solutions, Instead, there are refinements of solutions of the first two phases (1.8 improvements in average).

Students opinions were collected with a simple anonymous open question: "what do you think of the guidelines that you have used for this test?". Just a few students answered to the question properly. Four of them highlighted the usefulness of the first set of more general guidelines in order to complete their range of solutions. At the same time, they have considered the second set of guidelines as tedious and ineffective. Specifically, they have mentioned the fact that many specific suggestions did not provide any hint for a solution or to improve a solution. This opinion was shared by many students when asked to the whole class.

5.7 THE 111 STANDARDS USED FOR PROBLEM FORMULATION

In this section, the 111 standards are used in the third and fourth steps of Spark to systematically formulate contradictions. Thus, starting from a perceived problem, the proposed methodology follows a step-by-step procedure to widen the perspectives on the current situation and support a systematic generation of solutions and partial solutions. Afterwards, it helps the formulation of a technical contradiction and suggests the use of ARIZ-85C to overcome it.

This methodology is composed of four steps and its schema is shown in figure 45.

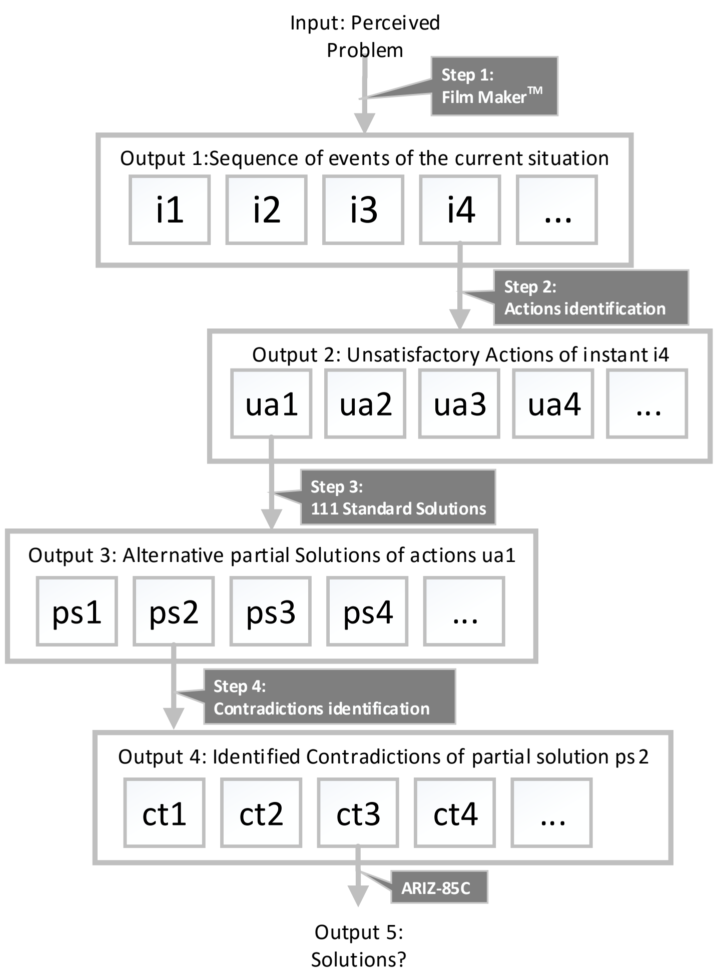


Figure 45. An overall schema of the proposed step-by-step methodology.

Each step is supported by the corresponding tool:

- Film Maker [§4.1]: build a Film Maker about the problem, describing a sequence of instants (i1, i2, …).
- Actions identification [§4.2]: identify one or more unsatisfactory actions for one or more selected frames (ua1, au2, …).
- 111 Standards [§4.3]: generate alternative partial solutions, with the 111 standards, for one or more selected unsatisfactory action (ps1, ps2, …).
- Contradictions identification [§4.4]: check for drawbacks about one or more selected partial solution and define contradictions (ct1, ct2, …). Use ARIZ-85C to overcome them.

In the following sections, the aforementioned tools are explained in detail to allow an easy reproduction of the entire methodology.

5.7.1 Film Maker

The Film Maker tool is used to describe the dynamics of the current situation, representing the complexity of the problem as a sequence of events.

This tool is studied to highlight the cause-effect relationships that involve time. In this sense, each state represents a picture of what is happening in a specific instant of time.

A frame (or instant) is divided into an upper part and a lower part. The first one specifies the effect of the actions which has been described in the previous frame. The second one describes the actions between elements on the current instant.

Usually, a Film Maker is completed starting from the instants where the problem solver has his own perception of the problem. Afterwards, new frames are added in the past and in the future in order to create a film. Each frame should contain an image or a drawing and/or a textual description.

In figure 46, the perceived problem "I cut the weeds of my garden with a sickle, but the sickle rapidly wear" has been analyzed with this tool.

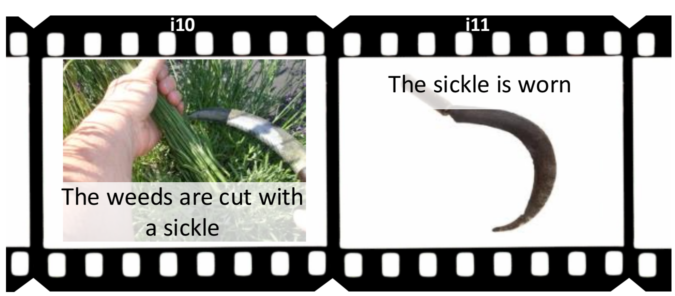


Figure 46. Film Maker: the representation of the perceived problem in frames.

The first two frames will be a representation of the perceived problem, which has to be placed in a certain instant of time. First, we write about the current interactions in the bottom part of a frame, then, moving into the upper part of the next frame we write the effect of the previous interactions (see figure 46).

Now that the perceived problem has been described, we can fill the frames in the past and in the future. Thus, before "the weeds are cut with a sickle" the weeds must grow; in order to grow, the weeds must be fed from the ground, they must be exposed to the sun etc. In figure 47 not all the conditions are mentioned for simplicity of representation.

Figure 47. Film Maker: filling the past and the future frames.

Moving again toward the past, the weeds were seeds, and seeds should be transported by the wind and so on. The same concept can be applied into the future, where the sickle blade has been sharpened, polished and so on.

The output of this tool is a sequence of frames with the aforementioned structure. From practical experience, a completed Film Maker should contain at least seven frames, highlighting every state in which a condition is changed.

The Film Maker has some similarities with the known TRIZ tool called Multiscreen [24], but it has just one row and its purpose is quite different. Multiscreen (or 9 windows) itself has been used in some different ways from other authors [120], but in this sense, a Film Maker is more similar to the Domino Theory [121] where a long series of events can result in an unexpected situation.

5.7.2 Actions identification

Thanks to a completed Film Maker, the user has a comprehensive overview on the changing conditions in time, and he can choose the most appropriate moment to intervene.

In this step, the user is invited to choose a specific frame where an unsatisfactory product is present. Then, considering the bottom part of the previous frame, he must relate the unsatisfactory product with one or more unsatisfactory actions. Thus, the relationships must be formalized in the form of a problematic TOP model: the <tool> acts on the <object>, obtaining a <product>. There are three types of unsatisfactory actions that can be used for this purpose, and they are shown in figures 48, 49 and 50.

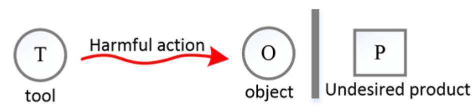


Figure 48. TOP model for harmful actions.

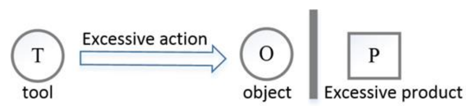


Figure 49. TOP model for excessive actions.

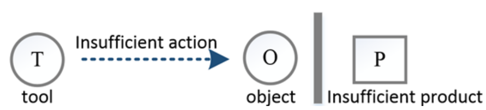


Figure 50. TOP model for insufficient actions.

Completing the example of section 4.1, we choose the frame where the weeds are grown; thus, the unsatisfactory product is "the grown weeds". Therefore, we search the previous frame to find unsatisfactory actions, which lead to this product using available resources in that moment (see figure 51):

- The ground feeds the weeds, obtaining grown weeds.
- The sun provides energy for the weeds, obtaining grown weeds.
- The sun heats insufficiently the weeds, obtaining grown weeds (insufficiently burned).
- …

The output of this step will be a list of sub-problems, in the form of TOP models, related to a specific couple of frames.

5.7.3 The 111 standards

After the problematic TOP models of a specific instant have been represented, the user must select one of them and solve it with the help of the 111 Standards. The set of 111 standards has been developed to systematically provide several ways to solve a problem. A standard has a structure which comprises 5 parts:

- Identification of the problem: the first part of the guideline is a description of the problem in form of TOP model.
- A Sub-goal (or Actions): they are conceptual solution to the identified problem.
- A General Suggestion: they are trigger to reach the selected Sub-goal.

- A Specific Suggestion: they provide different ways to follow a General Suggestion.
- An example

For each problem there are several Sub-goals, for each Sub-goal there are several General Suggestions and so on. The main idea on the use of these guidelines is that the user may just read the Sub-goals, and use the Suggestions and examples only if needed.

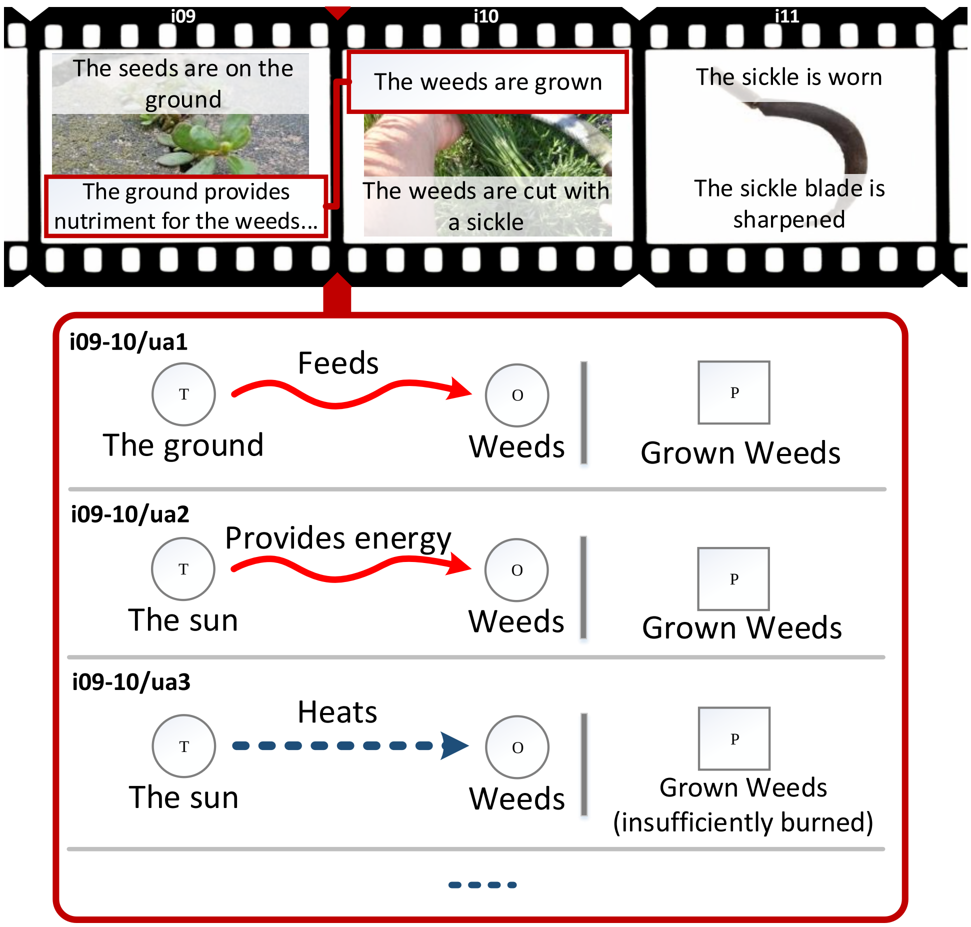


Figure 51. Actions Identification: the problematic TOP models for a selected couple of frames.

Completing the aforementioned example, we choose to solve the second problematic TOP model, in which "the sun provides energy for the weeds, obtaining grown weeds". This is a problem of harmful action, thus, the Sub-goals automatically generated by the 111 standards are as follows (see figure 47):

- Deflect the action "to provide energy" from where it is harmful.
- Block the action "to provides energy" so that it will not reach the weeds.

- Make the sun unable to provide the harmful action "to provide energy".
- Make the weeds insensitive or less sensitive to the harmful action "to provide energy".
- Make the "grown weeds" useful or not harmful.

Although the third Sub-goal is not reasonable, the other ones are description of different kind of solution. Deepening the second sub-goal, the description of the solution can be more detailed. The 111 standards can provide a General Suggestion, so that the description becomes: block the action "to provides energy" so that it will not reach the weeds, by adding a substance between the sun and the weeds. Thus, a partial solution to this problem can be a simple roof, which avoids the sunlight to reach the weeds.

The output of this step is a series of partial solutions that solve the identified problematic TOP model.

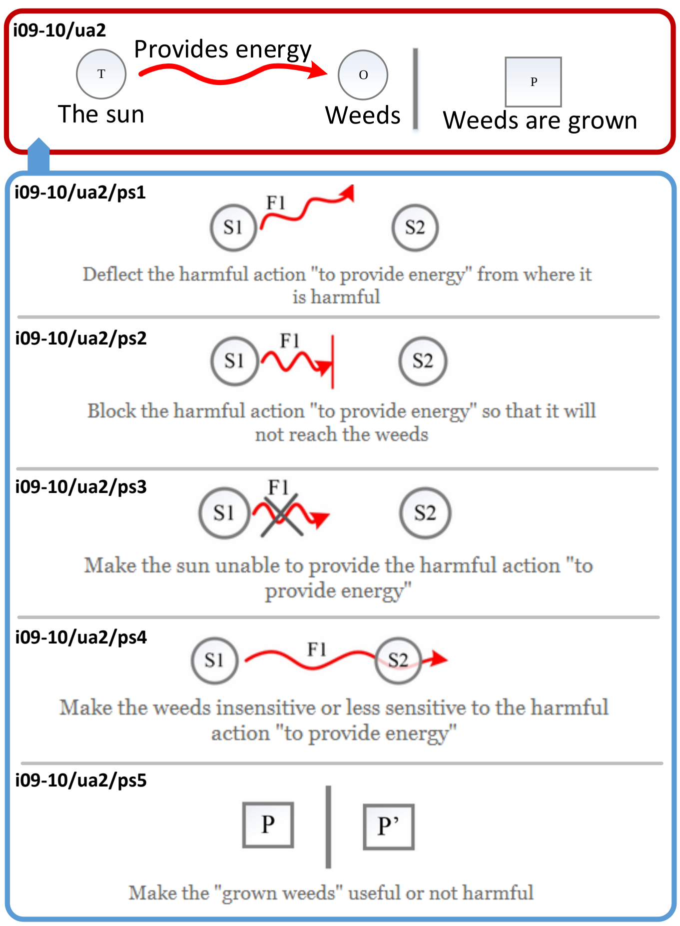


Figure 52. The 111 Standards: automatically generated Sub-goals for a selected problematic TOP model.

5.7.4 Contradictions identification

After the solutions of a specific TOP model are defined, the user must choose one of them and evaluate it. The chosen partial solution solves the perceived problem, which was the input of the entire methodology; however, other problems will probably arise.

The contradictions identification phase is the more creative one. Inside it, the user must imagine the selected solution and find drawbacks in its implementation. If necessary, he should define a new Film Maker to describe the dynamics of the new solution, but it is usually enough to answer the following questions:

- Compared to the current situation, does the new solution involve new unsatisfactory products?
- Compared to the current situation, does the new solution involve new harmful actions?
- Compared to the current situation, does the new solution involve new undesired consequences?

If one or more drawbacks are identified, we can formulate a technical contradiction. Specifically, ARIZ presents the formulation of a technical contradiction as follows:

> *A technical system for <state the purpose of the system> includes <list the main parts of the system>.*
>
> *Technical contradiction 1 (TC-1): (to be identified).*
>
> *Technical contradiction 2 (TC-2): (to be identified).*
>
> *It is necessary, with minimum changes to the system, to <state the required result>.*

In accordance with the proposed methodology, TC-1 will describe the current situation, while TC-2 will describe the new generated system. Furthermore, the understanding of the dynamics of the problem can result in a more aware compilation of <purpose of the system>, <main parts of the system>, as well as <the required result>.

Completing the example of the previous paragraphs, we answer the first aforementioned questions with: yes, the new solution "prevents the sun to provide energy for plants to grow". Therefore, a contradiction can be defined as:

> *A technical system for "growing plants" includes "plants, weeds, ground, sun".*
>
> *Technical contradiction 1 (TC-1): without using a roof to block the sun, the plants grow, but the weeds infest the ground.*

Technical contradiction 2 (TC-2): using a roof to block the sun, the weed does not grow, but the plants do not grow too.

It is necessary, with minimum changes to the system, to prevent the weeds to grow, allowing the plants to grow.

A graphical representation of this contradiction is shown in figure 53.

Here, the proposed methodology suggests the use of ARIZ-85C to overcome the technical contradiction. Since ARIZ-85C is a relatively mature and known methodology, we will not describe it and we will present just one of the solutions that can arise from separating in space the presence of the roof. Thus, completing the example of the previous paragraph, the most interesting solution, found with ARIZ-85C, has been a special configuration of plants in order to block the sun from reaching the weeds. This solution is depicted in figure 54, where plants themselves grow in such a way that they make a roof to block the sunlight, preventing the weeds to grow.

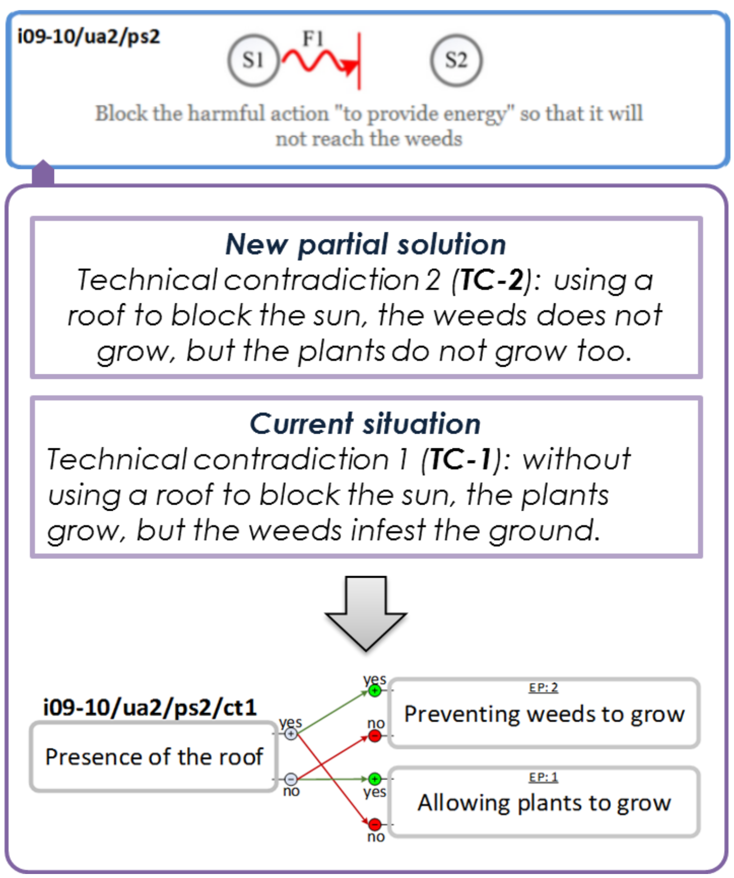


Figure 53. Contradictions identification: the representation of a contradiction for a selected partial solution (i09-10/ua2/ps2/ct1).

Figure 54. A solution for an identified contradiction. A roof which is made of plants with grapes that get the sunlight and shade the ground.

Finally, the output of these last two steps is respectively composed of one or more contradictions and one or more solutions for each contradiction.

5.8 Discussion

The 111 standard solutions are designed by organizing the information contained inside Altshuller's standard solutions. Thus, all the 76 standard solutions (except one) are included in a new system of standards.

The original system of standards often provides a suggestion for only a type of problem. The new system splits a suggestion according to the purpose of the problem, reformulating the hints using different verbal forms. For instance, the original standard 1.2.1 is defined for a harmful action, while in the new standards we also suggest to use it for insufficient action. So, if a substance can be used "to block" an action, a substance can also be used "to enhance" it (amplifier).

Some hints of the old set of standards are merged together following similarity of meaning. For instance, suggestions such as "introducing a substance which is a modification of the present substances" (1.2.2) and "introducing the external environment as substance" (1.1.4) are part of a more general new Suggestion "Finding and creating substances".

Some old standards that provide very specific suggestions have been included in the new system in form of examples or notes in order to uniform the validity level of standards and reach more homogeneous suggestions. For example, standards of the 2.4 subclass refer to the use of ferromagnetic substances and ferromagnetic fields which are obviously too specific recommendations to be maintained as general paths for the solution.

In Altshuller's standard solutions the user is supposed to take into account all the constraints from the beginning. The use of a standard is closely related to the conditions of applicability of the standard itself (if <condition> then). In this way, the user needs to browse the entire system of standards finding the most suitable suggestion to solve his problem. In the new proposed system of standards, the conditions on the constraints are removed. In particular, the classification itself replaces the conditions. The user enters the system of standard according to the functional model of his/her problem and refers directly to the proper set of standards.

Although the number of standards is increased from 76 to 111, they are grouped in only 25 Actions ensuring completeness and simplicity at the same time. The Actions allow a fast overall overview of the possible path to find a solution. Only if needed, the user can refer to more detailed suggestions by reading the Clouds. A study of the correspondence between the old and new standards ensures the presence of all the information from the original standards. The only exception is the standard 1.1.1. This standard suggests the creation of a new function and do not fit with the new system of standards. For this type of problem, several whole methodologies were developed [119].

Nowadays, although the set of 111 Standards has preserved its name, it has been expanded and completed. A software version has been developed (24) and a first evaluation has been performed on its effects on creativity outcomes.

The new 111 standards are also suitable for the step of problem definition. In fact, they can be used as a tool to structure the identification of a contradiction through a step by step procedure (see section 5.7). This methodology has been applied within a set of projects aimed at promoting and strengthening the competitive growth of micro, small and medium Italian enterprises (SMEs).

Future works: The unification of different set of guidelines can be performed by classifying their information into the structure presented in the previous paragraphs. Nowadays, a study for implementing more problem types is under development at the University of Bergamo and results will be presented in further publication. The basic idea is to unify problem solving guidelines in one framework in accordance with the proposed ontology. Therefore, separation principles, 40 inventive principles, 76 standard solutions and other guidelines may be integrated together. At the same time, problem types such as "solving contradictions" or "measurement problem" may be integrated in a single interface and use the same specific suggestions. Of course, as for the 111 standards, Sub-goals and general suggestion should be customized on the type of problem.

6 CONCLUSIONS

The main outcomes of this dissertation are a unified classification of technical problems, a systematic methodology for the definition of an innovation strategy, and a new set of guidelines to support the generation of alternative solutions for problems of unsatisfactory actions.

The first contribution is a new classification for problems that is elaborated by analyzing existing literature in design education, TRIZ and problem solving in industry. The classification includes 24 types of problems that are grouped under design problems, conceptual design problems, finding the unknown causes and effects, forecasting (Predicting), decision-making (ranking or selecting). The classification and the description of the types of problems are based on the characteristics of the main objective of the problem solving activity.

This classification is propaedeutic for a critical analysis of a problem solving methodology called Spark. Spark was determined to be well suited for problems that involve the generation of conceptual alternatives for the improvement of existing products, while it lacks specificity for decision-making and problems of unsatisfactory actions. Thus two solutions to improve the efficiency of Spark in the aforementioned types of problems have been presented.

The second contribution of this dissertation deals with the adaptation of the second step of Spark for two decision-making problems: the definition of an innovation strategy and the selection of a technology among a series of alternatives. For both types of problems, the management of requirements has a central role. Specifically, I treated the aspects related to the reconciliation of R&D and problem solving, with marketing requirements. The proposed methodology is composed of five steps: a) Identification of technological alternatives; b) definition of requirements; c) gathering information for each requirement; evaluation of importance and satisfaction; d) definition of the innovation strategy (or selection of the technical system). Knowledge is extracted from patents, market analysis, scientific literature, commercial literature and expert's interviews. Problem solving tools of Spark are used to identify alternative scenarios and new technological alternatives. The selection of a technical system and/or the definition of an innovation strategy are supported with a concise diagram that summarizes the gathered knowledge and allows experts and leaders to have a comprehensive and fast overview of the current scenario.

Consequently, subjectivity of the evaluations is reduced by increasing awareness and consistency of the decision-making process. Some typical situations have been recorded when the provided information influenced the numerical evaluation of importance and satisfaction of a requirement. A limit of the evaluation is to consider requirements as independent from each other to facilitate the judgment of experts.

This limitation is partially overcome by the visualization of requirements in the importance-satisfaction graph, which allows to easily identify trade-offs and indirectly consider influences among requirements. However, there is no specific way to effectively manage the complexity of requirements' correlations.

After several academic case studies, the proposed methodology has been applied to support the first stages of a stage-gate process, in a big multinational firm, for three different products.

A third contribution involves an adaptation of the idea generation step of Spark for problems of unsatisfactory actions. I analyzed the methodologies for the generation of conceptual alternatives, with special attention to those involving a problem solving activity. Among them, I identified the 76 standard solutions, coming from TRIZ (theory of inventive problem solving), as the most suitable starting point. A new system of 111 Standards is proposed. Special attention is given to how the guidelines can be structured. As a result of this analysis, the ontology of a standard now has a fixed structure, which comprises five parts: a) the identification of the problem with a description of the problem that defines if the guideline can be used or not in the present situation; b) a Sub-goal (or Actions) that contains a conceptual solution to the identified problem; c) a General Suggestion, which is a trigger to reach the selected Sub-goal; d) a Specific Suggestion, which provides different ways to follow a General Suggestion; e) an example. For each problem there can be several Sub-goals, for each Sub-goal there are multiple General Suggestions and so on. Given the proposed structure, the use of these guidelines is radically different from the original 76 standard solutions and makes them suitable for the identification of contradictions in the problem formulation step.

This dissertation supported the assumption that an ideal problem solving methodology should be able to address most types of problems without losing specificity. Accordingly, the basic idea was to select a method among those existent, clearly set its boundaries and extend them with proper customizations and adaptations. Following this approach, a problem solving methodology called Spark has been improved and adapted for two types of decision-making problems and for problems of unsatisfactory actions. Effectiveness of these approaches has been tested on real case studies in both small and big enterprises with encouraging results.

7 APPENDIX

This chapter contains a more detailed state of the art about problems, dealing with the meaning of knowledge (section 7.1), the meaning of problem solving (section 7.2) and presenting a collection of problem classifications elaborated by the author (section 7.3). This chapter contains useful but not fundamental discussions for the reading of this dissertation.

7.1 KNOWLEDGE

Classical definitions of knowledge comes from a Platonic distinction between true knowledge and mere belief [166]. At the heart of this view is the conception of knowledge as “justified trues belief”. This is also known as the tripartite theory:

- Belief: Unless one believes a thing, one cannot know it. Even if something is true, and one has excellent reasons for believing that it is true, one cannot know it without believing it.
- Truth: If one knows a thing then it must be true. No matter how well justified or sincere a belief, if it is not true that it cannot constitute knowledge. If a long-held belief is discovered to be false, then one must concede that what was thought to be known was in fact not known.
- Justification: In order to know a thing, it is not enough to merely correctly believe it to be true; one must also have a good reason for doing so. Lucky guesses cannot constitute knowledge; we can only know what we have good reason to believe.

Therefore, a sentence of the form “S knows that p” has the following truth conditions: S knows that p if and only if

- ‘p’ is true;
- S believes that p;
- S has adequate evidence for believing that p.

The tripartite theory of knowledge is intuitively very plausible. Since Edmund Gettier’s critique of it in the 60s [167], however, using thought-experiments now known as Gettier cases, it has been generally rejected.

In the author’s opinion, this definition is not helpful to be used for practical purposes, such as cognitive sciences or artificial intelligence. The main limits on the use of this definition is that the “p is true” assumption cannot be tested, since no one has completely or even partly accessed to “what is true”. What we can use is a personal (or shared) interpretation of a perceived reality, i.e. “what we think to be true”. It is also interesting to notice that knowledge is strongly time dependent for the first two assumptions: “S knows that p” only in the moments where “belief that p” and “p is true” are both true. Therefore, when a person changes his belief about p, from true to false, he's jumping from "S knows that p" to "S does not know that p". This condition cannot be so clearly defined, in fact, a person is not always sure of what he/she believes. Finally, the third assumption is strongly subjective, since an adequate evidence can be adequate for some people while not adequate for others.

A more practical definition of knowledge has been given by Sloman [54]:

Knowledge is a set of beliefs about changes in the world and the mechanisms that support those changes. Knowledge is about how changes in some things lead to changes in other things. In other words, what we know about the world is how things could have been otherwise. Representations of causality allow us to describe how the world would have been (that is, another possible world) if some cause had had a different value, for then its effects would have been different.

Sloman greatly simplifies the concept of knowledge making it more understandable and practical. In this way, whatever we can use to predict the future or imagining alternatives present or past realities is knowledge.

Another definition of knowledge or at least a definition of "knowledge space", has been used to define "design" in what is known as C-K theory [168]:

- We call K, a "knowledge space", the space of propositions that have a logical status for a designer D. This space is always neglected in the literature, yet it is impossible to define design without such referring space.
- We call "logical status of a proposition", an attribute that defines the degree of confidence that D assigns to a proposition. In standard logic, propositions are "true or false". In non-standard logic, propositions may be "true, false, or undecidable" or have a fuzzy value. A Designer D may use several logics. What matters in our approach is that we assume that all propositions of K have a logical status whatever it is, and we include here as a logical status all non-standard logical systems. In the following, we will assume for simplicity reasons that in K we have a classic "true or false" logic. But the theory holds independently of the logic retained.
- We call "concept", a proposition, or a group of propositions that have no logical status in K. This means that when a concept is formulated it is impossible to prove that it is a proposition of K. In Design, a concept usually expresses a group of properties qualifying one or several entities. If there is no "concept" Design is reduced to past knowledge2.
- Definition 1 of Design: assuming a space of concepts C and a space of knowledge K, we define Design as the process by which a concept generates other concepts or is transformed into knowledge, i.e. propositions in K.

But, there really exist concepts without any clue about their logical status? Although the theory doesn't clearly define what the degree of confidence is, this confidence is here associated with the estimated probability that "p" is true. Therefore, the confidence that "p" is true must be different from 50%. So, it seems that there is only one instable (instable equilibrium) point where a belief cannot be considered knowledge, i.e. when this belief can be either be true or false with equal probability. This is an unstable situation and it seems to appear only in an ideal reasoning. In fact, in the moment we are aware or we perceive or we address "p", we probably change our opinion towards the true or false statement, or at least we oscillate from one part to the other.

In the author's opinion, that's why student and experienced Designers may be not comfortable with the distinction between concepts and knowledge. Immediately after a concept is considered (i.e. something that has 50% of probability to be true), we often already transform it towards true or false on the basis of our experience.

An alternative to this interpretation is that a concept is considered a concept when the degree of confidence is comprised between a range (such as 40-60% on the truthfulness of the preposition), but again it is quite difficult to identify the boundaries of this assumptions; is it 40% to 60%? 30% to 70%. These boundaries vary from person to person, and it is also difficult to fix them during time.

Thus, instead of keeping the distinction between knowledge and concepts, we provide a new definition of knowledge.

Our definition of knowledge takes the main corpus from Sloman [54] definition and makes it more rigorous to be used in a problem solving or design context.

Knowledge is a set of propositions that can be used from a thinking entity (it can be either a human or an AI) to imagine alternative future, past or present worlds with a certain degree of confidence. Each of these propositions have the following properties:

- They are associated with a certain logical status, which represent the degree of confidence on the truthfulness of the considered preposition.
- The degree of confidence must be different from totally uncertain (50%), otherwise it cannot be useful to predict or imagining alternatives.
- The degree of confidence on a certain proposition must be justified (i.e. not guessed)

Notes: this new tripartite is not meant to give the absolute meaning of the world "knowledge". Instead, it is meant to provide a simple definition which is not far from its practical implications in computer sciences. The definition is strictly related with the theories of causalities. In fact, representations of causality allow us to describe how the world would have been (that is, another possible world) if some cause had had a different value, for then its effects would have been different [54].

7.2 PROBLEM SOLVING

The most accepted definitions of problems involve two main concepts. First, the problem involves a difficult action, a blocked goal, a valuable goal. Second, the problem is (or involves) an unknown entity.

Karl Duncker [169] stated that "A problem arises when a living creature has a goal but does not know how this goal is to be reached. Whenever one cannot go from the given situation to the desired situation simply by action, then there has to be recourse to thinking. Such thinking has the task of devising some action, which may mediate between the existing and desired situations". According to Maier [170], "A problem exists when a response to a given situation is blocked". McDermott [171] defined a problem as "just a difficult action". Jonassen [109] defined it as "an unknown entity in some situation (the difference between a goal state and a current state)", where "finding or solving for the unknown must have some social, cultural, or intellectual value." Similarly, Mayer [172] stated that "a problem exists when a problem solver has a goal but does not know how to accomplish it. When a situation is in a given state, a problem solver wants the situation to be in a goal state, and the problem solver is not aware of an obvious way to transform the situation from the given state to the goal state."

As part of the innovation process, problem solving is surely a very important activity. It was defined by Anderson [173] as “any goal-directed sequence of operations”. According to Jonassen [109], problem solving can be described as the process of finding the unknown. Mayer and Wittrock [172] described problem solving as “cognitive processing directed at achieving a goal when no solution method is obvious to the problem solver”. Lynch [174] defined problem solving as "the practical application of reasoning and other types of skills in a process that involves the identification and use of relevant information".

Although there is little agreement on the cognitive processes of problem solving, Jonassen [109] highlighted two necessary attributes. First, problem solving requires the mental representation of the problem, known as problem space [143]. Second, problem solving requires some activity-based manipulation of problem space [143], be it an internal representation or an external physical representation [109].

Some confusion arises in literature when searching for "problem solving" and "design". Is problem solving part of design or is design part of problem solving? Simon [175] claimed that “design theory was nothing else than problem solving theory”. On the other hand, in recent publications about the C-K theory (concept-knowledge theory), Hatchuel [168] states that problem solving theory can be considered as a special and restricted case of Design theory. Jonassen [109] identified "design problems" as a type of "problems", somehow supporting the first hypothesis. In this thesis, I do not distinguish between "design theory" and "problem solving theory", and at the same time we consider "design" a specific type of problem.

Problem solving is related with other terms, such as thinking, reasoning, decision-making, critical thinking and creative thinking [172]. A very important aspect of problem solving is the criteria with which we evaluate the success of a problem solving process. In Ill-structured design problems, these criteria are not easily defined [109]. In this cases, since there is not a right or wrong answer, the evaluation of a problem solving activity becomes an entire research field. Creativity is probably the most used metric. Its definition is yet an arguable point in the scientific literature. "Creativity occurs through a process by which an agent uses its ability to generate ideas, solutions and products that are novel and useful" [145]. It seems that the metrics used to evaluate creativity are becoming the definition of creativity itself. A widely used method to assess creativity were defined by [149,176] and it is composed of four metrics: novelty, variety, and quality, quantity. Novelty is a measure of how unusual or unexpected an idea is as compared to other ideas. Variety is a measure of the explored solution space during the idea generation process. Quality is a measure of the feasibility of an idea and how close it comes to meet the design specifications. Quantity is the total number of ideas generated by a group when it uses a certain idea generation method. Some considerations and refinements on this kind of metrics have been conducted by several researchers [177–179].

The path to solutions requires creativity for working in a complex and dynamic system in which there are more “messes” than neat problems (9, 10). Therefore, creativity and problem solving has to be studied together when addressing different problem solving processes and methods. In accordance with the C-K theory [168], the definition of creativity should even be embedded in the definition of design.

A method for problem solving should produce creative results (relatively to another method or to no-method). The description of the creative reasoning were effectively collected by Horowitz in its dissertation [104], and are here summarized:

- Creativity as Divergent thinking [182];
- Creativity as Remote Associations [183];
- Creativity as Bisociations [184];
- Creativity as Search [143];
- Creativity as Heuristic Search with criteria for interestingness [185];
- Creativity as search in a “Klondike space” [186,187];
- Creativity as Variations on a Theme [188];
- Creativity as Exploring and Transforming a Conceptual Space [189];
- Creativity as Process of ‘Function Follows Form’ [190];
- Creativity as Process of Preparation, Incubation, Illumination and Elaboration [191];

- Creativity as Mechanical Process [192,193];
- Creativity as ‘Nothing Special’ [194].

Problems vary in their nature, in the way they are presented or represented, and in their elements and interactions among them (see appendix in section 7.2.1). Since problems strongly differ from one another [105], the identification of the type of problem can help in the customization of the tools and procedures that are used to solve them.

The understanding of the nature and characteristics of problems are the first step to define criteria to distinguish between different types of problems.

7.2.1 External factors in problem solving

Context or external factors are also called "problem representation" by some authors (such as [109]). Characteristics of a problem are not sufficient to describe the complexity of a problem solving activity. Many other aspects then those embedded in the problem itself should be considered. A same problem can be presented to the same solver in different ways, from different persons and in different contexts. A problem solving activity with rigid time constraints will be different from another one with no time limits. A problem solving activity where the committee is a big multinational company will be different from a problem solving activity where the committee is a small company and yet a problem solving activity will be different if the committee is the solver himself.

Simply stated, we defined "context" as what can influence the solver's perception of the problem. There is not a clear set of feature to identify different contexts in literature. Therefore, we tried to summarize some of the most important one, some extracted from literature reviews and others extracted from our consulting experience in problem solving:

- Evaluators: I called "evaluators" those responsible for the evaluation of the solver's activity. Since there are not objective ways to evaluate problem solving outcomes, the individual differences of the "evaluators" will indirectly influence the context of a problem solving activity. In academic contexts the evaluator is usually a professor, in a professional context the evaluators are all the persons involved or interested in the consulting activity. In a personal context the "evaluator" is usually the solver himself.
- Competition level: if the problem solver is involved a competition [109], a cooperation or a competition among groups that should internally cooperates.
- Who propounds the problem [106], we can differ between:
    - a presented problem situation: when the problem exists and it is propounded to the problem solver;
    - a discovered problem situation: when the problem exists and is discovered by oneself;

    - a created problem situation: when the problem does not exist until someone invents it.
- Modality (or problem representation): simply stated, if a problem is presented, modality is the way with which the problem is represented to the problem solver. The "presenter" will decide the modality and medium to represent a problem to the problem solver [109]. In the author opinion, somewhat in contrast with what Jonassen meant [109], "problem representation" is here intended as synonym of "modality".
- Clues: the "presenter" often includes clues [109] or other information that think relevant for the problem solving activity.
- Other historical, social and cultural factors [109]. Such as the "holiness" of the current political/economic system, prejudice to change, the occidental/oriental differing views of goals, Bias against recognition of the current paradigm limit [26].

Savransky [26] identified some other major obstacles to innovation: belief in some official functional method (often called a scientific), money constraints, decision-making and leadership styles, time restraints. All the aforementioned factors, and probably many others, determine how the problem will be perceived by the problem solver and how the outcomes of the problem solving activity will satisfy the expectations of committees. Among contextual factors,

7.2.2 Humans' ability to solve problems

Factors that affect the problem solving skills, and more in general creativity, can be searched in both single individuals and teams. Depending on the problem solving context, the problem solving tasks can be performed individually or in groups of two or more people.

Individual differences surely influence problem solving activities. Mascitelli [195] recognized tacit technical skills and tacit cognitive skills as being prerequisites for innovative abilities for technological innovation. According to Amabile [196], creativity is supported by expertise or domain-relevant skills, creative thinking skills or creativity-relevant skills, as well as motivation. Hoover [197] listed memory organization and facilitation; problem-specific knowledge and; general problem-solving skills. Savransky [26] identified some obstacles to Innovation: stereotyped thinking and/or lack of creativity and/or psychological inertia [24], risk of failure, lack of knowledge and/or faulty memory, Self-imposed constraints (e.g. taboo, fear of questioning). Jonassen [109] summarized a series of characteristics of individuals that are here used as reference: domain knowledge (familiarity, perplexity, experience); structural knowledge; procedural knowledge; systemic/conceptual knowledge; domain-specific reasoning; cognitive styles; general problem-solving strategies; self-confidence and motivation/perseverance.

Perhaps the strongest predictor of problem solving ability is the solver's familiarity with the problem type [109]. Familiarity is a parameter to consider which is related to the problem solver or to the problem solving team. A problem will be more familiar if the problem solver has experienced similar problems in the past. It seems that problem solvers have better developed schemas, which can be employed more automatically [198]; however, familiarity among problems seldom transfer to other kind of problems or even to the same kind of problem represented in another way [199].

Mayer and Wittrock [200] distinguish between routine and non-routine problems. This distinction is similar, until a certain extent, the aforementioned concept of familiarity. "What makes problems either routine or non-routine depends on the knowledge of the problem solver because the same problem can be routine for one person and non-routine for another" [172].

The differences between familiar and not familiar problems were also implicitly argued in a presentation about the introduction of TRIZ in Samsung [113]. Specifically, the presentation mentioned three different types of problems: standard problems, non-standard problems and research and development problems. Although not explicitly stated, this differentiation seems to resemble an increased degree of familiarity. However, the terminology "standard/non standard problems" as well as "common/non common problems" seem to leave a bit of ambiguity, since It is difficult to separate the aspect of familiarity (as described above) from frequency and probability (how many times a problem appear or is likely to appear).

In the TRIZ community, the obstacle that prevent individuals to innovate and be creative is called psychological inertia [24].

Kowalick [117] states that "**psychological inertia** implies an indisposition to change, a certain "stuckness" due to creativity human programming". Psychological Inertia (PI) represents the many barriers to personal creativity and problem-solving ability, barriers that have as their roots "the way that I am used to doing it." Rigid mindsets of individuals hinder creativity or innovation [201], being for excessive familiarity with a given domain, of an expert's biased view on a problem. Rigid mindsets cause the phenomenon of design fixation in design problem solving [144].

Different types of psychological inertia were identified by TRIZ masters and some of them are reported in [117]. Psychological inertia can be the habit to associate a function with the object that performs it, the tendency to extend partial restrictions to the whole object, the tendency to rely on past ways of doing something without actually know the reason, the tendency to associate common properties to words, the tendency to consider given information as valid (such as the information contained in physics books), and so on.

The mechanisms to overcome psychological inertia were explicitly studied by the TRIZ community and implicitly addressed by any kind of problem-solving methodology.

Team differences. Problem solving is even more complicated when teams are involved. A team is a group of two or more individuals who interact over a certain time in order to achieve a common goal or objective. One of the most advanced, although not recent, theory to improve performances of problem solving in teams is Synectics [34]. Many practical factors that affect team effectiveness in problem solving are well described in [85]. The author summarized the factors affecting team performances in problem-solving in the following points:

- Communication, collaboration and coordination [202].
- Organizational structures, task structures, team processes and team outcomes [203]
- Informational Diversity, which relies on differences in terms of education, experience and expertise. It describes the degree to which team members differ in terms of knowledge bases and perspectives [204].
- Social category diversity: 'race', gender and ethnicity [204].
- Value diversity: differences related to individual opinions on the goal of the task and the way these goals should be obtained [204].
- Functional diversity: diversity in terms of the team members' organizational occupation (such as marketing, research and development), can cause both informational and value diversity within a team [205].
- Disciplinarity diversity: composed of persons trained in different fields of knowledge (disciplines).

7.2.3 Structuredness of problems

Structuredness is probably the most discussed characteristic of a problem. It was firstly proposed by Newell and Simon [17,143,206] and further investigated by Jonassen [105,109,207]. Structuredness was addressed by many other authors, often using different names or even without mentioning the name of the characteristic itself [174,208–210]. Despite slight differences on the meaning of their classifications, it seems reasonable to assume that they are talking about the same concepts.

In the seventies, Newell and Simon distinguished between well-structured and ill-structured problems [17,143]. Instead of a clear distinction between them, they supported the idea of a continuum of problems that moves from well-structured to ill-structured ones [206].

According to Simon [206] it is impossible to construct a formal definition of "well structured problem"; however, he provided a list of requirements to identify a "well structured problem", where it is not clear, as declared by Simon, when one or more requirements are sufficient or necessary. The following [206]:

1. There is a definite criterion for testing any proposed solution, and a mechanizable process for applying the criterion.
2. There is at least one problem space in which can be represented the initial problem state, the goal state, and all other states that may be reached, or considered, in the course of attempting a solution of the problem.
3. Attainable state changes (legal moves) can be represented in a problem space, as transitions from given states to the states directly attainable from them. But considerable moves, whether legal or not, can also be represented-- that is, all transitions from one considerable state to another.
4. Any knowledge that the problem solver can acquire about the problem can be represented in one or more problem spaces.
5. If the actual problem involves acting upon the external world, then the definition of state changes and of the effects upon the state of applying any operator reflect with complete accuracy in one or more problem spaces the laws (laws of nature) that govern the external world.
6. All of these conditions hold in the strong sense that the basic processes postulated require only practicable amounts of computation, and the information postulated is effectively available to the processes-i.e., available with the help of only practicable amounts of search.

These requirements are judged by the same Simon as indefinite and relative. In fact, criteria are not absolute, "but generally express a relation between characteristics of a problem domain, on the one hand, and the characteristics and power of an implicit or explicit problem solving mechanism, on the other" [206]. This undefined and uncomfortable way of identifying well-defined problems reflect "the continuum of degrees of definiteness between the well-structured and ill structured ends of the problem spectrum" [206]. Differentiating problems between well-structured and ill-structured is very important for education and learning. Many scientists complain on the tendency of educational programs to propose well-structured problems instead of ill-structured ones, especially because ill-structured problems are the most frequently encountered in real life [109,200].

Structurdness is strongly related with the availability of knowledge. According to Wood [211], structuredness of a problem is the degree to which the ideas in the problem are known to the problem solver. This definition was then used by Jonassen [109] in his further studies on the structurdeness of problems. The new eleboration does not contradict the original distinction, but it is expressed differently. According to Jonassen [109], a well-defined problem owns the following requirements: present all elements of the problem to the learners; require the application of a limited number of regular and well-structured rules and principles that are organized in predictive and prescriptive ways; have knowable, comprehensible solutions where the relationship between decision choices and all problem states is known or probabilistic [211]. Ill-structured problems, instead: possess problem elements that are unknown or not known with any degree of confidence [211]; possess multiple solutions, solution paths, or no solutions at all [212]; possess multiple criteria for evaluating solutions, so there is uncertainty about which concepts, rules, and principles are necessary for the solution and how they are organized; often require learners to make judgments and express personal opinions or beliefs about the problem, so ill-structured problems are uniquely human interpersonal activities [213]. Where with problem elements we can refer to initial or beginning state, operators or actions, end state or goal state and constraints.

Similar distinctions among problems was performed in the field of mathematical learning. The terms well-structured and ill-structured problems are respectively substituted by closed problems and open-ended problems [208]. Open ended problems [208] are generally considered as problems with more than one correct solution. According to Lynch [174], the following requirements are generally used to identify them: they cannot be described completely; they have more than one solution option; they generate controversy, even among experts; they have incomplete information that is subject to a variety of interpretations; they have a variety of solution options with unknown outcomes; they often need to be addressed repeatedly over time as conditions change and better information becomes available; they can be addressed through a problem solving process that uses information in increasingly complex ways. This elaborated definition of open ended problems makes them very similar to ill-structured problems.

Ill-structured problems was also addressed by Coyne [209], with the name of "wicked problems", in the book entitled Dilemmas in a General Theory of Planning [210]. According to Coyne, "wicked problems" are [209]: only loosely formulated; there is not a rule to stop the problem solving process; they are subjected to redefinition and resolution in different ways over time; there are given differently since the formulation changes with different points of view; there is no ultimate test of the validity of a solution; the testing of solutions takes place in some practical context; the solutions are not easily undone.

Also Wittrock [172,200] considered the distinction between well-defined and ill-defined problems. "A well-defined problem has a clearly specified given state, a clearly specified goal state, and a clearly specified set of allowable operations. An ill-defined problem lacks a clearly specified given state, goal state, and/or set of allowable operators".

The last attempt to formalize the difference between well-defined and ill-defined problems was yet addressed by Jonassen [105]. The distinction is now supported with a sub-set of parameters: intransparency, heterogeneity of interpretations, interdisciplinarity, dynamicity, and legitimacy of competing alternatives. These parameters, in the authors opinion, clarify the concept of structuredness to a level never reached before and they deserve special attention.

The degree of **intransparency** grows with the uncertainties about the problem. "In order to solve a problem that contains unknowns in the problem space, the problem solver must solve the problem based on assumptions or guesswork. These assumptions or guesswork inevitably reduce the problem solver's confidence level in successfully solving a problem" [105]. Also Getzel [106] uses knowledge as a parameter to classify different problems, i.e. he considers if the problem has a known formulation, a known method of solution, or a known solution to the problem solver of to others.

Heterogeneity of interpretation is "The number of possible interpretations and perspectives for understanding or solving the problem" [105]. We can also distinguish two types of interpretation. The first if it is vaguely defined in one of its problem elements (initial state, goal state, constraints). The second one is related with the concept viable or acceptable solutions.

The degree of **interdisciplinarity** is here simplified as the number of disciplines that are necessary to solve a problem. The more disciplines are involved, the less luckily is to have considered all the facet of the problem. Furthermore, interdisciplinarity can cause problems of comprehension [105].

About **dynamicity**, dynamic nature of variables or operators increases the ill-structuredness of the problem. A variable or operator is dynamic if it is not fixed during time, it changes during, before or after the problem solving process.

In some cases, "there are emergent properties that only appear in response to the changes of other related variables or states of the problem or certain actions taken by the problem solver" [8].

About **legitimacy of competing alternatives, ill-structuredness** increases if the number of conceivable options for executing operators in various states and solution paths exist within the problem space [105].

The effects on the problem solver are a reduction of the confidence in selecting the best solution that consequently increase the number of tasks and time to evaluate solution and select solution paths.

7.2.4 Complexity of problems

It is useful to emphasize that complexity and difficulty are related but different. Difficulty is dependent on complexity and many other factors [105]. This difference is not shared in literature and there are different opinions on the meaning of complexity.

According to Funke [214], noncomplex problems are characterized by the following factors: availability of information about the problem. That is transparency of the problem situation; precision of goal definition, that is, whether a goal is defined, and whether there are multiple goals, some of which can be contradictory; "Complexity" of the problem as defined by the number of variables, the degree of connectivity among the variables, and the type of functional relationship (linear vs non linear); stability properties of the problem, that is, time dependencies in the course of the problem-solving process ("eigendynamic"); "Richness" of the problem's semantic embedding. Rich semantic embedding often reduces the uncertainty to a large degree. With these criteria, Funke elaborated three characteristics of complex systems: intertransparency, polytely (many goals), complexity of the situation.

If the reader has already checked the previous section on structuredness, it appears that some characteristics are repeated. In the author's opinion, the ambiguity can be partially solved if we think that complexity is often a synonym of difficulty (as intended in [105]).

Although there is not a clear line between complexity and structuredness [109], Jonassen offered a different definition [109], involving: the number of issues, functions, or variables involved in the problem; the degree of connectivity among those properties; the type of functional relationships among those properties; stability among the properties of the problem over time (dynamicity). Excluding the availability of information and contextual richness which were already included in structuredness. In later publications, also the last factor (dynamicity) seemed to be included in structuredness [105].

In this thesis we share the definition of complexity described in [105], which includes four parameters: the breadth of knowledge required, Attainment level of Domain knowledge, Intricacy of problem solution procedures and relational complexity. These parameters are better analyzed in the following paragraphs.

Breath of knowledge required. This is the most intuitive factor, i.e. the degree of complexity will increase with the size of the problem space (see definition of problem space in [17]). Simply stated, complexity will increase with the amount of information and interrelationships. In a more practical sense, the degree of complexity was considered in design theories. Koller [110] described the complexity of a technical system with examples divided in ten degree of complexity, from simple needles to really complex design such as airplanes. Similarly, adapting the scale for students Ponn [215] divided designs into four level of complexity: low complexity systems such as nutcrackers; medium complexity systems such as vacuum cleaners, high complexity systems such as bicycles, and very high complexity systems such as automobiles.

Attainment level of Domain knowledge. "When the concepts involved in solving one particular problem are difficult for learners to grasp, most likely, the problem is more difficult to solve" [105].

Intricacy of problem solution procedures. This parameter participates in the degree of complexity since it is the intricacy of the problem-solution process. Simply stated, it increases with the number of steps to be executed in a solution path and the extent of complexity of the tasks and procedures in these steps [105].

Relational complexity. Simply stated, it increases with the number of relations that need to be processed in parallel during a problem solving process. "The more complex the relations in a problem, the more processing load is required during problem solving, and as a result, the more complex the problem is" [105].

7.2.5 Abstractness of problems

"Problem-solving activities are situated, embedded, and therefore dependent on the nature of the context or domain" [109]. It seems that cognitive operations are related to the domain of that problem. In fact, students of engineering will be more skilled in solving engineering problems, statistical students will be more skilled in solving problems involving statistic and so on. We can make a distinction between weak strategies (domain independent) and strong strategies (domain-specific). Experts effectively use strong strategies, and some researches has shown that less experienced solvers can also learn to use them [200]. There are different opinions on the efficacy of strong and weak methods in solving problems. On one hand, strong methods seem to work better since they are especially "designed" for problems that share the same domain, and therefore some characteristics. "General heuristics like means-ends analysis that can be applied across domains, generally fair no better than those who do not" ([216] as cited in [109]). On the other hand, weak methods seem to work better in scientific discovery [217], or when the knowledge contained within strong methods is not sufficient or even misleading.

The concept of abstractness is very important for some problem-solving methods and theories, such as TRIZ [24], design by analogy (biomimetic included) [36,45,218,219], functional modeling [220] and technology transfer [52]. In accordance with these methodologies, a problem can be abstracted (or generalized) to match solved problems in different domains and eventually apply the same principles of resolution. This means that problems which appear to be totally different can be more similar if depurated from their domain specificity.

7.2.6 Problem elements

Problem elements has been introduced by the author to justify other classifications of problems that do not just imply structuredness, complexity and domain specificity but also the subjects that are involved in problem itself. Actually, also Jonassen in its general approach on problem solving classification [109] had to specify the characteristics of some elements of the problem. Specifically: inputs, success criteria and constraints. More in general we can say that problems significantly change where the problem space changes (initial state, goal state).

In design problems a first distinction is the type of entity on which design is directed: a product, a process, an action or a function (section 7.2.1). Another parameter is, for example, the dimension of the change: minimum change, totally new product.

Of course, the diversity of problem elements is infinite. In section 7.2.1 I reported a literature review on the classification of problems, which would allow the identification of many other problem characteristics.

7.3 Problem classifications

This chapter summarizes the classifications of problems that are present in literature. It is important to provide a valuable starting point for the unified classification for technical problems that has been proposed in chapter 2.

7.3.1 Jonassen, Newell and Simon's classification of all problems

Newell and Simon [143] distinguished among different types of problems after defining structuredness (see paragraph 7.2.1). They provided a differentiation among problems based on the ill-structuredness of problem elements (initial state, goal state and operators). Jonassen expanded work started by Newell and Simon and defined its set of problem classes [8,109] proposed a classification for different types of problems, which is reported in table 15. The differentiation is done considering structuredness, abstractness, problem elements and context. Among problem elements we can find: inputs (or initial state), success criteria (goal state) and learning activity (operators that the solver should know to solve the problem). Complexity, although mentioned in Jonassen work, is too variable in the same problem category that is not defined [109]. It is not clear, in Jonassen work, if characteristics define the type of problems or the type of problems are first defined and then associated with the set of corresponding characteristics. As Jonassen stated, this classification is probably not completed and other types of problems may be added. However, the identification and classification of problems needs big effort, hundreds or thousands of problems should be analyzed [109].

Table 15. Table, without modification, of Jonassen's classification of problems (from [109]).

	Logical Problems	*Algorithmic Problems*	*Story Problems*	*Rule-Using Problems*	*Decision making Problems*	*Trouble-shooting Problems*	*Diagnosis-Solution Problems*	*Strategic Performance Problems*	*Case Analysis Problems*	*Design Problems*	*Dilemmas*
Learning Activity	logical control and manipulation of limited variables; solve puzzle	procedural sequence of manipulations; algorithmic process applied to similar sets of variables; Calculating or producing correct answer	disambiguate variables; select and apply algorithm to produce correct answer using prescribed method	procedural process constrained by rules; select and apply rules to produce system-constrained answers or products	identifying benefits and limitations; weighting options; selecting alternative and justifying	examine system; run tests; evaluate results; hypothesize and confirm fault states using strategies (replace, serial elimination, space split)	troubleshoot system faults; select and evaluate treatment options and monitor; apply problem schemas	applying tactics to meet strategy in real-time, complex performance maintaining situational awareness	solution identification, alternative actions, argue position	acting on goals to produce artifact; problem structuring & articulation	reconciling complex, non-predictive, vexing decision with no solution; perspectives irreconcilable
Inputs	puzzle	formula or procedure	story with formula or procedure embedded	situation in constrained system; finite rules	decision situation with limited alternative outcomes	malfunctioning system with one or more faults	complex system with faults and numerous optional solutions	real-time, complex performance with competing needs	complex, leisure-time system with multiple ill-defined goals	vague goal statement with few constraints; requires structuring	situation with antinomous positions
Success Criteria	efficient manipulation; number of moves or manipulations required	answer or product matches in values and form	answer or product matches in values and form; correct algorithm used	productivity (number of relevant or useful answers or products	answer or product matches in values and form	fault(s) identification; efficiency of fault isolation;	strategy used; effectiveness and efficiency of treatment; justification of treatment selected	achieving strategic objective	multiple, unclear	multiple, undefined criteria; no right or wrong —only better or worse	articulated preference with some justification
Context	abstract task	abstract, formulaic	constrained to pre-defined elements, shallow context	purposeful academic, real world, constrained	life decisions	closed system real world	real world, technical, mostly closed system	real-time performance	real world, constrained	complex, real world; degrees of freedom; limited input & feedback	topical, complex, inter-disciplinary
Stucturedness	discovered	procedural predictable	well-defined problem classes; procedural predictable	unpredicted outcome	finite outcomes	finite faults & outcomes	finite faults & outcomes	ill-structured strategies; well-structured tactics	ill-structured	ill-structured	finite outcomes, multiple reasoning
Abstractness	abstract, discovery	abstract, procedural	limited simulation	need-based	personally situated	problem situated	problem situated	contxtually situated	case situated	problem situated	issue situated

Table continu

7.3.2 Presented, created and discovered problems (J.W. Getzels)

J.W. Getzel [106] used characteristics that are almost entirely related with context and individual differences (see sections 7.2.1 and 7.2.2): whether the problem already exists, who propounds it, and whether it has a known formulation, a known method of solution, or a known solution. He then defined ten types of problems:

- The problem is given (is known) and there is a standard method for solving it, known to the would-be problem solver (for example, experimental subject, student) and to others (for example, experimenter, teacher), guaranteeing a solution in a finite number of steps.
- The problem is given but no method for solving it is known to the problem solver, although it is known to others.
- The problem is given but no method for solving it is known to the problem solver or to others.
- The problem itself exists but remains to be identified (become known) by the problem solver, although it is known to others.
- The problem exists but remains to be identified by the problem solver and by the others.
- The problem exists but remains to be identified (as in 4 and 5) and there is a standard method for solving it, once the problem is dis- covered known to the problem solver and to the others.
- The problem exists but remains to be identified, and no standard method for solving it is known to the problem solver, although known to others (as in 2).
- The problem exists but remains to be identified, and no method for solving it is known to the problem solver or to others (as in 3).
- The problem does not yet exist but is invented or conceived, and a method for solving it is known or becomes known once the problem is formulated.
- The problem does not yet exist but is invented or conceived, and a method for solving it is not known.

He also states that the typology does not exhaust the possibilities. After this classification, Getzel defined three problem situations (or classes), which are:

- a presented problem situation: when the problem exists and it is propounded to the problem solver;
- a discovered problem situation: when the problem exists and is discovered by oneself;
- a created problem situation: when the problem does not exist until someone invents it.

This classification has been used by the author to extrapolate contextual factors influences the problem solving activity, but it does not provide clues on how to differ the methodologies on the mentioned classes of problems.

7.3.3 Problems in everyday life (Daniel Theyagu)

Daniel Theyagu defined four types of problems [221]:

- Question based problems: problems that involve a question that needs an answer. In some declinations, this category is quite similar to case analysis problems and dilemmas (see section 7.3.1); however the definition of Theyagu is so broad that it becomes difficult to make more precise correlations.
- Situation-based problems: problems which involves personal decisions in constrained situations. These problems resembles decision-making classes (see section 7.3.1).
- Convincing-based problems: where the solver's have to convince a person to think the same way that he thinks. It happened where the solver possesses information that the other person does not possess. This is a special type of problem, hardly associable with other classes in literature. A slight similarity has been perceived by the author with situated case-policy problems (see section 7.3.1).
- Solving-based problems: "solving based problems usually will involve you being in a current non-desirous state of being and the need to move to an ideal state of being where the problem is resolved or if that is not possible at least in a state where the problem can be minimized" [221]. This class seems to include design problems and troubleshooting problems (see section 7.3.1).

Although this classification is somewhat confusing, it contains some interesting insights, and differently from others, it strictly involves the personal aspects of problem solving. Problems should be solved to improve our quality of life or should have positive implications in our life.

7.3.4 Classifying complexity of design problems (Koller)

Jonassen considered complexity too variant among problem types to distinguish problems in term of complexity [109]. Other authors, especially those from the world of design, classified problems according to their complexity, or more specifically, with their breath of knowledge. Koller [110] described ten complexity levels in technical systems from simple geometrical features, such as needlepoints, to extremely complex technical systems such as manufacturing plant, ships or buildings.

A similar but simplified classification was performed in [215], by defining four levels of complexity, according to the number of parts of the system: low complexity, medium complexity, high complexity, very high complexity. Where with low complexity they meant something like nutcrackers and very high complexity something like automobiles.

7.3.5 Classifying industrial problems

In this section, I grouped classifications based on common problems in industry.

Ivanov and Barkan differentiated between technical problems [107]. This classification is limited to problems involving technical products or technical processes. Mainly, problems are divided in terms of the problem elements (initial state, goal state). In particular, on the subject of the changes (tool, object), if the subject exists (existent or not existent).

- Manufacturing process problems: glitches, stops, non-rhythmical character and ineffectiveness of the main technological process. Inability to keep manufacturing process within established parameters; increase in number of rejects, unfavorable impact on environment.
- Design problems: Low productivity of the existing technical system, high energy consumption, large overall dimensions (mass), unreliability, short life and complexity of structure. The design problems' subtypes:
    - Development of the existing systems: all components of the system are changed with the exception of the tool, which remains the same. (This is based on the "Tool-action-object" formulation of a technical system, in which the tool takes some action that affects some parameter of the object.)
    - Creation of new systems: changes in the tool, which uses a completely new principle of action.
- Creating a new technical system to satisfy new requirements.
- Emergency problems: emergence of self-developing, uncontrollable processes, resulting in the destruction of technical system and its environment.
- Science and research problems: lack of information about physical and chemical processes, disparity between expected and real results, emergence of a previously unknown phenomenon or event.

If we compare this classification with the one proposed by Jonassen (see section 7.3.1) they all seem to be classifiable under "design problems", with the exception of emergency problems and science and research problems that seems more related to "troubleshooting problem solving".

The institute for learning TRIZ in Irkoutsk [no reference], in Russia, proposed a customized set of problems (translated), similar to those described by Barkan [107]:

- Commercial problems: such as problems of selling volumes;
- Production/manufacturing problems;
    - Modification of an existing system, i.e. the elimination of undesired effect in the present production system.

    - Measuring/monitoring the parameters of the system and using the gathered information.
- Design problems
    - Improvements on the existing system
    - Creation of a new system
- Maintenance problems
    - Failure prevention: identifying the zones that are subjected to failures.
    - Identification and study causes of failures.
- Research problems
    - Identification of new harmful elements.
    - Evaluation/measurement of undesired effects.
    - Identification of an interaction.

In the last years, many examples on the use of systematic methodologies for problem solving have spread in big multinational companies. This company slightly adapted the methodology to meet their specific requirements.

In a presentation on the use of TRIZ in Intel [108], three types of problems were mentioned:

- Corrective: Problems where a standard previously achieved is not being met.
- Improvement: The current system or process performance, as it was designed, is expected to be improved.
- Preventive: Problems are where the goal is to add robustness and prevent systems or processes from failure or falling below baseline.

Another more articulated example has been found in Samsung [113]. In the roadmap for the problem solving process of figure 55 we can easily see two different levels of classification, one based on the familiarity of the problem, the other one with the problem elements. The first one distinguished between:

- "Standard" Engineering problems: they contain obvious technical contradiction that can be expressed by the expert.
- "Non standard" Engineering problems: they contain implicit contradictions and these problems cannot be solved with application of Principles or Standards alone.
- Research and development problems: prediction and application of modern scientific and technical effects do not usually contain an open contradiction.

7. The second one divide problems as follows:

- Existing product improvement: for product/process improvement and inexpensive engineering (without additional researches) problem solving;
- New product improvement: for product/process improvement;
- Manufacturing technology Improvement: for cost reduction of manufacturing;

- Patent overcoming and patent development: for cost reduction by avoiding competitor patents and development new patents ("umbrella patenting");
- Short and long term forecasting: for forecasting and development of new concepts for existing product design;
- Scientific research engineering: for development future brand new core technologies.

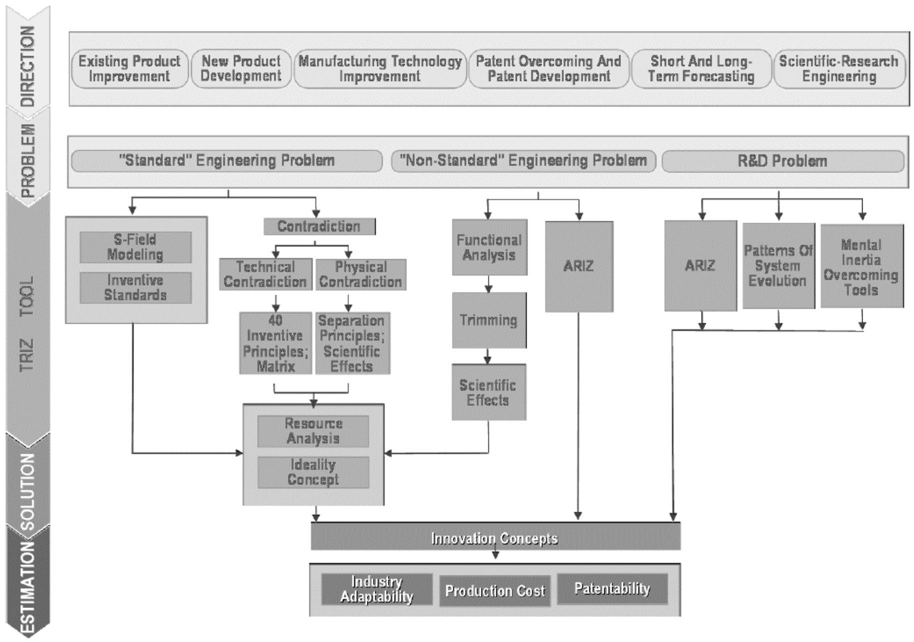


Figure 55. Road Map of TRIZ Problem Solving Process

According to Roggel [108], another important question would be related with "the integration of TRIZ utilization for technology or product prediction in concert with marketing and customer's needs".

7.3.6 Diagnostic problems (Gregory Frenklach)

Gregory Frenklach distinguished different types of diagnostic problems [112] (understanding and checking hypothesis). There are two types of the research (diagnostic) problems:

1. Something happens in our system and we don't know the reason.
2. The function of the specific element of the system is unknown. I.e. there is a system which is done in a certain way and we want to understand why.

7.3.7 Classifying improvements of technical parameters

Zlotin developed a **system of operators** [222], identifying a series of parameters that are typical subjects of technical improvements. "Specialized blocks address specific types of problems related to a particular function to be performed or drawback to be eliminated" [222]. The operators are reported in table 16.

Table 16. Zlotin system of operators [222].

Useful parameters:	**Harmful parameters:**
• Reliability	• Weight
• Speed of action	• Overall dimension
• Mechanical strength	• Energy consumption
• Composition stability	• Complexity
• Convenience	• Time wasted
• Productivity	• Energy wasted
• Local (selective) mode	
• Manufacturing accuracy	
• Dispensing accuracy	
• Shape	
• Universality	
• Degree of automation	
• Degree of adaptation	

One of the most known tool of TRIZ is the **matrix of contradictions** [24]. This matrix contains pointers to forty inventive principles [94] that should contain the mechanisms to overcome technical contradictions. Only a set of the forty principles is suggested when given a combination of two technical parameters. As a matter of fact, the technical parameters (see table 17) identify different types of problems.

*Table 17. List of Altshuller technical parameters (from **[24]**).*

1. Weight of moving object	21. Power
2. Weight of binding	22. Waste of energy
3. Length of object	23. Waste of substance
4. Length of binding object	24. Loss of information
5. Area of moving object	25. Waste of time moving object
6. Area of binding object	26. Amount of substance
7. Volume of moving object	27. Reliability
8. Volume of binding object	28. Accuracy of measurement
9. Speed	29. Accuracy of manufacturing
10. Force	30. Harmful factors acting on object
11. Tension, pressure	31. Harmful side effects
12. Shape	32. Manufacturability
13. Stability of object	33. Convenience of use
14. Strength	34. Reparability
15. Durability of moving object	35. Adaptability
16. Durability of binding object	36. Complexity of a system
17. Temperature	37. Complexity of control
18. Brightness	38. Level of automation
19. Energy spent by moving object	39. Productivity
20. Energy spent by binding object	

	Undesired Result (Conflict) → Feature to Improve ↓	Weight of moving object	Weight of nonmoving object	Length of moving object	Length of nonmoving object	Area of moving object	Area nonmoving object	Volume of moving object	Volume of nonmoving object
No		1	2	3	4	5	6	7	8
1	Weight of moving object	+		15, 8, 29,34		29, 17, 38, 34		29, 2, 40, 28	
2	Weight of nonmoving object		+		10, 1, 29, 35		35, 30, 13, 2		5, 35, 14, 2
3	Length of moving object	8, 15, 29, 34		+		15, 17, 4		7, 17, 4, 35	
4	Length of nonmoving object		35, 28, 40, 29		+		17, 7, 10, 40		35, 8, 2,14
5	Area of moving object	2, 17, 29, 4		14, 15, 18, 4		+		7, 14, 17, 4	
6	Area of nonmoving object		30, 2, 14, 18		26, 7, 9, 39		+		

Figure 56. Matrix of contradictions (from [101]*)*

In addition to the 39 technical parameters of Altshuller, Savransky [26] added (not complete) those of table :

• Safety	• Electrical impedance
• Stability of parameters	• Optical transparency
• Accuracy of operation	• Viscosity
• Information	• Friction
• Tolerances	• Corrosion
• Susceptibility	• Resistance
• Ergonomics	• Noise
• Aesthetics, etc.	• Transient processes in condensed matter

7.3.8 Classification based on system changes

A classification based on evolution were the choice of Altshuller to organize his 76 standard solutions [30,97,158]. This idea were further expanded by Petrov [96] in its system of generalized models. The original classification involved five classes based on an evolutionary perspective: composition and decomposition of SFMs; evolution of SFMs; transitions to supersystem and microlevel; measurement and detection standards; helpers.

Khomenko and Terninko [153,161] as well other TRIZ experts [158] defined a flow chart to use the 76 standard solutions. Problems were classified as follows: problems of Measurement, problems of System change, problems of Minimum change; problems of system improvement or system synthesis.

Also Smirnov [No reference], from TRIZ France, developed his own flow chart to use the 76 standard solutions, which was more articulated then the aforementioned ones. Other authors [31,33,155] have generated their own set of standards (suggestions) to solve problems. Specifically, they grouped them in three classes considering the entity of the intervention on the system: improving the system with little or no change; improving the system by changing the solution; detection and measurement.

In the field of design, similar classifications divide problems in terms of how much the system is different from the existing ones [223] (see table 18).

Table 18: Typology of Design Problems [223]

Design problem type	Sub type	Characteristics
Routine design	-	Derived from common prototypes with same set of variables or features; structure does not change; design plan exists prototypical solutions known from the start
Redesign	Adaptive, configurational or transitional	Adaptations of known systems to changed tasks; solution principle remains unchanged; can include detail refinements
	Variant, extensional or parametric	Design by extra- or interpolation; generation of geometrically similar variants of differing capacities based on proven design
Non-routine design	Innovative	Based on new variables or features which still resemble to existing ones; known problem decomposition but sub-problems and their solutions must be synthesized; solving the same problem in different ways OR solving different problem in the same way
	Creative	Based on variable or features which are completely different from previous prototypes; design has very little resemblance to existing ones; no a priori known design plan

7.3.9 Classification on Su-field conditions

Valeri Souchkov and Savransky [26] studied the 76 standard solutions [30,97] and found a general pattern to unify the standard. In fact, standards are mainly written in the form IF-THEN:

- IF "a problem of goal is given as Su-Field conditions and constraints according to the problem circumstances"
- THEN "such problems are solved by action"

The author of this thesis collected the possible Su-field conditions (identified in [26]) under the "IF" statements, avoiding to consider constraints, which were quite inhomogeneous. The result is reported in table 19.

Table 19 Unified set of goals (extracted from ***[26]****).*

Optimization of Su-Fields: • Minimal (dosed, optimal) mode • Useful function maximal mode • Selective mode Destruction of Su-Fields: • Both useful function and harmful function take place Between substances in Su-Field • Harmful function of a field on substance exists Construction of Su-Fields • The given substance is hardly changeable in the needed direction Increase the Su-Field Efficiency due too resources • Su-Field is weakly controllable and its efficiency should increase Growth of Su-fields Efficiency by Phase Transitions • Contradictory requirements to introduce S and F can be met only by using phase transitions • Formation of Su-Fields for Measurement	• Poorly measurable or detectable incomplete Su-field • Poorly measurable or detactable complete Su-Field Substances management in Su-Field: • Complete Su-field • Add a lot of substance Add fields in Su-fields: • Complete Su-field Forcing of measuring Su-fields • Complete Su-field Growth of efficiency for Physical Effects application • Su-Field's component must be in various states • Su-field has a "weak" input (constraint cannot increase input, but a "strong" output is needed

7.3.10 Classifications based on language and functional models

A functional model is a description of a product or process in terms of the elementary functions that are required to achieve its overall function or purpose [114]. A functional model generally involves the description of functions and sub-functions. When dealing with a product or technology, a function can be described by focusing on the goal, on how the product achieves the goal, on the performed transformation, on the changes between inputs and outputs, and so on. A plurality of definitions was given by design experts and none of them is comprehensive of all the faceted aspects of the overall means of function.

Stone and Wood [114] presented a consistent definition of functions and sub-functions for a functional model. A product function is the general input/output relationship of a product having the purpose of performing an overall task, typically stated in verb-object form. A sub-function is a description of part of a product's overall task (product function), stated in verb- object form. Sub-functions are decomposed from the product function and represent the more elementary tasks of the product.

In literature, a function is typically described with a couple verb and object. In this way, products which perform functions can be described by a single Subject-Action-Object SAO triad (e.g., a hammer hits a nail). However, in many other cases the situation can be much more complex (e.g., windows that transmit the light, and depending on their configurations, block sound, UV or IR light, block/allow the passage of water, air or dust). Although a description with just verb and object has its own limits, it practically became the standard for knowledge transfer and functional modelling.

An entire field of research is dedicated on the definition of a series of classified verbs to model functions in design. This idea is not a recent concept. Koller [110] defined a set of elementary functions: emitting, conducting, collecting, guiding, transforming, enlarging, direction-changing, directing, coupling, connecting, adding, and storing. According to Modarres and Cheon [225], a function in design can be associated to one of the functional primitives: generate, destroy, maintain, control, transform and transport. A following work of Stone and Wood [114] presented a large verbs vocabulary based on the possible transformations of mass, energy and signal flows. Two years later Hirtz et al. [220] continued this work reconciling and evolving the previous version of the so called functional basis. The functional basis is now a widely accepted standardized representation of engineering product functionality [226]. It consists of generic taxonomies of engineering functions, defined as function sets, and associated flows to describe product functionality [227].

The introduction of a functional basis for design represented a fundamental step towards the standardization of the design language and the reuse of knowledge. According to Pahl et al. [20] a system with a clear and easily reproduced relationship between inputs and outputs is necessary to solve technical problems. Furthermore, all technical systems involve the conversion of energy, material and signal.

Nagel et al. [228] highlighted the advantages on the use of functional basis in the modelling of the flow transformations within a biological system. With these models, the system is an abstraction of its true form. These abstractions can facilitate the creation of connections such as analogies or metaphors that lead to creative leaps

According to Stone and Wood [114] the adoption of the functional basis will allow different designers to share information at the same level of detail, to generate repeatable function structures, and to compare functionality of different products for idea generation purposes. All of these features contribute to an overall goal of formulating engineering design as a set of systematic and repeatable principles and as a teachable content area.

In addition to the list of verbs provided by Hirtz et al. [220], engineering flows were decomposed into an established set of physical parameters to be used as criteria to gauge the engineering capabilities.

After the development of TRIZ [24], many researchers have worked to summarize knowledge in databases of physical, chemical, biological and geometrical effects. These databases are widely used from TRIZ experts [101] to support the generation of new ideas during problem solving sessions and they have become attractive also for their integration in several design approaches. For instance, Russo et al. [53] proposed the use of TRIZ functional analysis along with effect databases to support idea generation. Cascini et al. [229] proposed an integration of the EMS model and effects in a step-by-step method to generate a Network of Evolutionary Trends.

A database of effects was firstly proposed by Altshuller [24]. After that, many TRIZniks extended it and tried to simplify the search of an effect relating it with a required function or property change [26]. Thus, TRIZ effect databases are usually used with the so called pointers to effects (In the TRIZ word, also a set of chemical (table 20) and geometrical effects (table 22) were developed, along with their pointers to effects [26]. In the same context, other effect databases were developed but not yet completed, such as biological databases, material effects databases and mathematical databases.

An extensive work was performed to extend the list of effects and adapt their use for a software implementation. TechOptimizer (Invention Machine, Inc.) used a classification of effects (see Table 23) based on the substance-field ontology [24]. In accordance with this ontology, a substance is a substantial or unsubstantial object of any level of complexity. In can be a single item, or a more complex system. A field is something that provides the energy, force, etc. It should be used in a very broad sense. The resulting set of pointers is reported in table and are classified in terms of the type of object (field, parameters, substance) and the type of verb (Accumulate, detect, etc…).

In the growing field of biomimetic design, databases have been developed. Among them, AskNature [47] has been developed with a structured taxonomy (Table 24), allowing the search of biological data with functions.

AskNature possesses an articulated classification. Mainly, its structure is composed of three levels, where the first and the second are verbs; the third is an object or a more specific couple verb-object (Table 24). The classification is quite inhomogeneous if compared with the functional basis, but it is very detailed and intuitive to navigate.). A pointer to effect is usually a couple verb and object that recall a specific set of effects. For example [24], in order to “move a liquid” (pointer) some of the following effects can be used: centrifugal force, thermal expansion, pressure of light, capillary force, etc. With the use of pointers, effects can be filtered in accordance with the required action (function). In this way, there is no need to explore all the effects but just the most likely to be useful.

The choice of the verbs and objects to group effects is not that distant from the attempt of standardizing design knowledge with functional basis. Altshuller [24] defined a set of 30 verb-object couples for linking physical effects.

In the TRIZ word, also a set of chemical (table 20) and geometrical effects (table 22) were developed, along with their pointers to effects [26]. In the same context, other effect databases were developed but not yet completed, such as biological databases, material effects databases and mathematical databases.

An extensive work was performed to extend the list of effects and adapt their use for a software implementation. TechOptimizer (Invention Machine, Inc.) used a classification of effects (see Table 23) based on the substance-field ontology [24]. In accordance with this ontology, a substance is a substantial or unsubstantial object of any level of complexity. In can be a single item, or a more complex system. A field is something that provides the energy, force, etc. It should be used in a very broad sense. The resulting set of pointers is reported in table and are classified in terms of the type of object (field, parameters, substance) and the type of verb (Accumulate, detect, etc…).

In the growing field of biomimetic design, databases have been developed. Among them, AskNature [47] has been developed with a structured taxonomy (Table 24), allowing the search of biological data with functions.

AskNature possesses an articulated classification. Mainly, its structure is composed of three levels, where the first and the second are verbs; the third is an object or a more specific couple verb-object (Table 24). The classification is quite inhomogeneous if compared with the functional basis, but it is very detailed and intuitive to navigate.

Table 20. Extract of pointers to effect for TRIZ chemical effects

Transform Substance	**Transform Energy**
• Carry in space • Change of mass • Change of concentration • Change of specific weight • Change of volume • The changed forms • Change of electrical properties • Change of optical properties • Change of magnetic properties • Change of biological properties	• Reception of heat (input of thermal energy in system) • Reception of cold (conclusion of thermal energy from system) • Reception of mechanical pressure • Generation of light radiation • Storage of heat • Storage of cold • Storage of light energy

• Change of chemical properties • Change of a phase condition • Disposal (destruction) • Stabilization (temporary reduction of activity) • Transformation of two and more substances into one • Protection of one substance from penetration by another • Drawing one substance on a surface of another • Connection of diverse substances (condensation, congestion) • Division of substances (allocation of one from another) • Destruction of substance • Accommodation of one substance in the friend • Reception of new substances (synthesis) • Organization of a closed cycle on substance (absorption-allocation) • Assembly of substance from atoms • Reception of substances with well-organized structure (reception of pure substances) • Transport of one substance through other	• Transport of thermal energy • Transport (drain) static electricity • Regulation of light energy • Power influence on substance **Transform Information** • Indication of the current information about substance • Indication of the information about energy

Table 21. Extract from pointers to effect for TRIZ physical effects ***[24]****.*

• Measure temperature • Reduce temperature • Increase temperature • Stabilize temperature • Locate an object • Move an object • Move a liquid or gas • Move an aerosol • Produce mixtures • Separate mixtures • Stabilize an object's position • Generation and/or manipulation of force • Change friction • Destroy object • Accumulate a mechanical and/or thermal energy • Transfer energy	• Influence on a moving object • Measure a dimension, change a dimension • Detect surface properties and/or conditions • Detect surface properties and/or conditions • Vary surface properties • Detect volume properties and/or conditions • Vary volume properties • Develop certain structures • Structure stabilization • Detect electrical and/or magnetic fields • Detect radiation • Generate electromagnetic radiation • Control electromagnetic field • Control light, light modulation • Initiate and/or intensify chemical reaction

Table 22. Pointers to effect for TRIZ geometrical effects

• Reduce or increase |the volume| of a (body) |at constant weight| • Reduce or increase |the area or length| of a (body) |at same weight| • Transform |one kind of a movement into another| • Concentrate (energy/particles flow) • Intensify |process|	• Decrease (loss of energy or substance) • Increase |process accuracy| • Increase |control| • Decrease |control| • Increase |lifetime, service reliability| • Decrease |expenses|

Table 23. Groups of pointers to effect for geometrical effects in TechOptimizer 3.0.

Field: Accumulate	Parameters: Change	Substance: Accumulate
Field: Detect	Parameters: Decrease	Substance: Combine
Field: Prevent	Parameters: Increase	Substance: Detect
Field: Produce	Parameters: Measure	Substance: Eliminate
	Parameters: Stabilize	Substance: Form
		Substance: Move
		Substance: Phase change
		Substance: Preserve
		Substance: Produce
		Substance: Separate

Table 24. Elaborated extraction of AskNature taxonomy.

Break down	Chemically break down	Catalyze (chemical reactions), Cleave (halogens from organic compounds), Cleave (heavy metals from organic compounds), -(Other inorganic compounds), -(Other organic compounds), -(Polymers)
	Physically break down	-(Abiotic materials), -(Biotic materials)
Get, store, or distribute resources	Capture, absorb or filter	-(Bulk solids), -(Chemical entities), -(Energy), -(Gases), -(Liquids), , -(Organisms), -(Solid particles)
	Distribute	-(Energy), -(Gases), -(Liquids), -(Solids)
	Expel	-(Gases), -(Liquids), -(Solids)
	Store	-(Bulk solids), -(Chemical entities), -(Energy), -(Gases), -(Liquids), -(Solid particles)
Maintain community	Cooperate and compete	-(Between (eco)systems), -(Between different species), -(Within a (eco)system), -(Within the same species)
	Coordinate	-(Activities), -(Groups (self-organize)), -(Systems)
	Provide (ecosystem services)	Biological control (of populations, pests, diseases), Control (erosion and sediment), Cycle (nutriments), Disperse (seeds), Generate (soil/renew fertility), Maintain (biodiversity), Pollinate, Regulate:, -(atmospheric composition), -(climate), -(habitat response to disturbance), -(hydrological flows), -(water storage)
Maintain (physical integrity)	Manage (structural forces)	-(Chemical wear), -(Compression), -(Creep), -(Impact), -(Mechanical wear), -(Shear), -(Tension), -(Thermal shock), -(Turbulence)

	Prevent	-(Buckling), -(Deformation), -(Fatigue), -(Fracture (rupture)), -(Melting)
	Protect (from abiotic factors)	-(Chemical), -(Dirt/solids), -(Excess liquids), -(Fire), -(Gases), -(Ice), -(Light), -(Loss of gases), -(Loss of liquids), -(Nuclear radiation), -(Temperature), -(Wind)
	Protect (from biotic factors)	-(Animals), -(Fungi), -(Microbes), -(Plants)
	Regulate (physiological processes)	-(Cellular processes), -(Homeostasis), -(Reproduction or growth)
Make	Chemically assemble	Attach (a functional group), Catalyze (chemical reactions), Detach (a functional group), -(Inorganic compounds), -(Metal-based compounds), -(Mineral crystals), -(Molecular devices), -(On demand), -(Organic compounds), -(Polymers), -(Specific stereoisomers)
	Generate/convert (energy)	-(Chemical energy), -(Electrical energy), -(Magnetic energy), -(Mechanical energy), -(Radiant energy (light)), -(Thermal energy)
	Physically assemble	-(Structure)
	Reproduce	-(Self-replicate)
Modify	Adapt/optimize	Adapt (behaviors), Adapt (genotype), Adapt (phenotype), Coevolve, Optimize (space/materials)
	Modify (chemical/electrical state)	-(Chemical potential), -(Chemically generate flow of electrons (redox)), -(Concentration), -(Conductivity), -(Electric charge), -(Electron transport), -(Energy state), -(Free radical reactivity), -(Oxidation state), -(pH), -(Reactivity with water), -(Solubility), -(Surface tension)
	Modify physical state	-(Buoyancy), -(Density), -(Light/color), -(Material characteristics), -(Number of), -(Phase), -(Position), -(Pressure), -(Size/shape/mass/volume), -(Speed)
Move or stay put	Attach	[permanently], [temporarily]
	Move	[in gases], [in/on liquids], [in/on solids]
Process information	Compute	
	Encode/decode	
	Learn	
	Navigate	-Over land, -Through air, -Through solids, -Through water
	Process (signals)	Differentiate (signal from noise), Respond to signals, Transduce/convert (signals)
	Send (signals)	-(Chemical (odor, taste, etc.)), -(Electrical/magnetic), -(Light - non-visible spectrum), -(Light - visible spectrum), -(Sound), -(Tactile), -(Vibratory)
	Sense (signals/environmental cues)	-(Atmospheric conditions), -(Balance/gravity/orientation), -(Body awareness), -(Chemicals (odor, taste, etc.)), -(Disease), -(Electricity/magnetism), -(Light - non-visible spectrum), -(Light - visible spectrum), -(Motion), -(Pain), -(Shape and pattern), -(Sound and other vibrations), -(Temperature), -(Time and day length), -(Touch and mechanical forces)

7.3.10.1 Classification of verbs, flows and parameters

The set of verbs of the functional basis is expressed with a table of four columns. The verbs are organized in a classification involving primary, secondary and tertiary functions. TRIZ pointers to effect and AskNature taxonomy use a different set of verbs to describe functions, and they have been related by the author in table 25.

Similarly, a flow is classified in functional basis with primary, secondary and tertiary classes. A flow is a material, a form of energy or a form of signal. In the TRIZ word, flows are not usually used. Instead, a classification based on the substance-field ontology can be found. Although there is not an exact correspondence of meaning between the two classifications, a field can be broadly associated with an energy flow while a substance with a material flow.

AskNature uses a wide set of objects. Some of these are not easily classifiable into the functional basis classes, e.g. biological organisms as well as chemical compounds are surely neither a gas, nor a solid and nor a liquid. The whole set of flow has been developed by the author and summarized in table 26.

While the function set (verbs) and the flows (objects) of the functional basis are consistent with our approach, the performance parameters in Hirtz et al. [220] are not suited for our purposes. Therefore, a different classification is proposed for them. Parameters are intended in a very broad sense, i.e. a parameter is generally intended as a performance or characteristic of a flow, but also as a condition around the flow. A parameter may be associate to an object (flow) or a verb (functional verb); it can be a quantifiable characteristic of the flow (e.g. volume is a parameter of a liquid substance) or a distinctive quality (e.g. wettability, chemical composition, index of refraction). The proposed classification of parameters and the correspondences between them are reported in 27. The present classification is suitable for products but is not complete for parameters of processes, such as productivity, glitches, and so on.

Table 25. Verbs extracted from functional basis, TRIZ effects (from TechOptimizer 3.0 1998) and AskNature function sets.

Class (Primary)	**Secondary**	**Tertiary**	**TRIZ effects (from TechOptimizer 3.0)**	**AskNature**
Branch	Separate		Separate, Disassemble, Break Down	Break Down, Chemically Break Down, Physically Break Down, Catalyze, Cleave
		Divide	Decompose	
		Extract	Extract	Filter
		Remove	Clean, Dry	
	Distribute			Distribute, Disperse
Channel	Import			
	Export		Eliminate, Destroy, Remove	Expel
	Transfer			
		Transport	Move, Lift, Vibrate	Move, Transport
		Transmit		Navigate
	Guide			

		Translate	Vibrate	
		Rotate		
		Allow DOF		
Connect	Couple			Catalyze, Detach
		Join	Assemble	Chemically Assemble, Physically Assemble
		Link	Embed	Attach
	Mix		Combine, Mix	
Control Magnitude	Actuate			
	Regulate			Maintain, Regulate
		Increase	Increase	
		Decrease	Decrease	
	Change		Change	Modify, Optimize
		Increment		
		Decrement		
		Shape	Bend, Flatten	
		Condition		Adapt, Coevolve
	Stop			Stay Put, Attach (Stay Put)
		Prevent	Prevent	
		Inhibit		Protect
Convert	Convert		Produce, Melt, Condense, Evaporate, Sublimate, Synthesize	Generate, Convert, Encode, Decode, Transduce
Provision	Store		Accumulate	
		Contain		Capture
		Collect	Absorb, Preserve	Absorb
	Supply			
Signal	Sense			Sense
		Detect	Detect	
		Measure	Measure	
	Indicate			
		Track		
		Display		
	Process			Process, Compute, Encode, Decode, Differentiate
Support	Stabilize		Stabilize	
	Secure		Embed	
	Position		Orient	
Others				Cooperate, Compete, Coordinate, Reproduce, Learn, Respond To Signals

Table 26. Objects (flows) extracted from functional basis, TRIZ effects and AskNature flows.

Primary	Secondary	Tertiary	TRIZ from Tech-Optimizer	AskNature
Material			Device, molecular particles, sub-molecular particles, chemical compounds, structured substances, technical objects, loose substances, flow of substance, substance	Biotic Materials, Abiotic Material, Chemicals, Halogens From Organic Compounds, Heavy Metals From Organic Compounds, Polymers, Other Inorganic Compounds, Other Organic Compounds, Chemical Entities, Systems, Chemical, Animals, Fungi, Microbes, Plants, Functional Group, Metal-Based Compounds, Mineral Crystals, Molecular Devices, Specific Stereoisomers, Self-Replicate, Space, Flow Of Electrons, Organisms, Nutriments, Sediment, Pests, Electrons, Cells
	Human			
	Gas		Gas	Gases, Fire, Loss Of Gases, Wind
	Liquid		Fluid flow, liquid substance	Liquids, Hydrological Flows, Water Storage, Excess Liquids, Loss Of Liquids
	Solid	Object	Solid substances, substance, geometric objects, porous substances	Bulk Solids, Solids, Seeds, Soil, Dirt/Solids, Ice, Structure
		Particulate	Elements of solid substances, particles	Solid Particles
		Composite		
	Plasma		Plasma	
	Mixture	Gas-Gas		Abiotic Materials
		Liquid-Liquid		
		Solid-Solid		
		Solid-Liquid		
		Liquid-Gas		
		Solid-Gas		
		Solid-Liquid-Gas		
		Colloidal		
Signal	Status	Auditory		Sound
		Olfactory		Chemical (Odor)
		Tactile		Tactile (Signal), Vibratory
		Taste		Chemical (Taste)
		Visual	Image, color	
	Control	Analog		
		Discrete		
Energy			Energy, field	Energy
	Human			
	Acoustic		Sound wave	
	Biological			
	Chemical		Energy of sub-molecular particles, Chemical Reactions,	Chemical Energy
	Electrical		Electrical energy, electric discharge	Electrical Energy, Electrical (Signal)
	Electromagnetic	Optical	Electromagnetic wave, light, birefringence, laser radiation, light propagation	Light, Radiant Energy
		Solar		
	Hydraulic			
	Magnetic			Magnetic Energy, Magnetic (Signal)

	Mechanical	Rotational	Mechanical wave	Mechanical Energy
		Translational	Mechanical wave	Mechanical Energy
	Pneumatic			
	Radioactive/Nuclear			Nuclear Radiation
	Thermal		Thermal energy	Thermal Energy
Other				(Eco)Systems, Species, Activities, Groups (Self-Organize), Populations, Diseases, Day

Table 27. Parameters extracted from functional basis, TRIZ effects and AskNature parameters.

Parameters Classes	Hirtz quantified performance metric	TRIZ from Tech-Optimizer	AskNature
(Optical)		Index of refraction, color, optical devices parameters, reflection coefficient	Color
(Surface)	Surface Area	Friction parameters, concentration of defects, surface parameters, wettability	Surface Tension
(Volume)	Weight	Solids parameters, weight	Mass, Buoyancy
(Physic)	Density, Temperature	fluid parameters, substance density, temperature, viscosity	Density, Material Characteristics, Temperature,
(Geometric)	Volume	Geometric parameters, dimension, configuration	Shape, Length, Volume, Size, Compression
(Cinematic)	Motion, Displacement, Angle, Angular Velocity, Amplitude, Frequency, Position	Deformation, deformation parameters, disposition of object	Orientation, Position, Balance, Motion, Speed
(Dynamic)	Pressure, Force, Torque	energy of moving object, energy parameters, motion and vibration parameters, Force, momentum, force parameters, momentum parameters, mechanical force, moment of force, pressure, mechanical waves	Creep, Impact, Mechanical Wear, Erosion, Gravity Mechanical Forces, Pressure, Touch, Shear, Tension,
(Thermo And Fluid-Dynamic)	Volumetric Flow, Flowrate, Mass Flowrate, Heat Transfer Rate, Heat Flow	Thermal parameters	Thermal Shock, Turbulence
(Acoustic)	Tone, Volume, Intensity	Sound waves parameters	
(Chemical)	Composition, Mixing Ratio, Reaction Rate	Chemical parameters, concentration parameters, concentration of charged particles, concentration of sub-molecular particles, humidity	Chemical Potential, Reactivity With Water, Solubility, Energy State, pH, Oxidation State, Free Radical Reactivity, Chemical Wear, Concentration, Phase,
(Electromagnetic)	Tone, Intensity, Visual	Absorption of electromagnetic waves, electromagnetic induction parameters, electromagnetic waves parameters, intensity of electromagnetic waves, intensity of light, light parameters, frequency shift of electromagnetic waves, phase of electromagnetic waves, polarization of electromagnetic waves, wavelength of electromagnetic waves, radioactivity parameters	Electric Charge, Conductivity
(Electrical)	Intensity , Velocity, Current, Voltage,	Electric field, electric current, electrical parameters, electrical resistance,	

	Electromotive Force	frequency of alternating current, electric field parameters	
(Magnetic)	Magnetomotive Force, Flux Rate	Magnetic field, Magnetic field parameters	
(Timing)		Speed of response	Temporarily, On Demand, Permanently, Time
(Environment Conditions)			Over Land, Through Air, In Solids, In Gases, Through Solid, In Liquids, Through Water, Atmospheric Conditions, Atmospheric Composition,
(Biological)			Awareness (Of The Body), Biodiversity, Fertility, Genotype, Phenotype, Homeostasis, Reproduction Or Growth, Behaviours, Pain, Disease,
Others		Interference pattern parameters, processes parameters, quantity parameters, reliability	Number Of, Processes, Frame Of Reference, Pattern,

7.3.11 Classifications based on TRIZ functional analysis

TRIZ functional analysis is derived from the Substance-Field analysis of TRIZ and is a graphical way to represent interactions and actions among elements. Interactions and actions are also classified in useful, insufficient, and excessive (see figure 57).

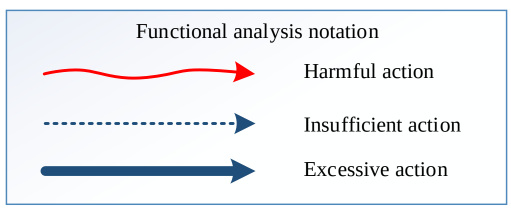


Figure 57. Type of actions in a functional analysis.

Pinyayev contributed on the classification of problems with its thesis [230] and another article on the TRIZ Journal [231]. In these publications he explained what he called "system of functional clues". This system was developed to address sub-problems that comes from a TRIZ functional analysis ([24,28]) with the proper set of suggestions. Pinyayev identified fourteen typical application conditions, that we interpreted here as type of problems:

1. U1: How to perform the function?: These problems are all about finding a Subject which can perform the required function.
2. U2: How to improve the function?: the difference of this class from U1 is that here we know how to perform the function and would like to keep using the same Subject, we just need to improve the interaction between the Subject and Object.
3. U3: The same action is both insufficient and excessive on the same object.
4. U4: Subject can be optimized for one function or another but not both of them together, where the object of the functions is the same.
5. U5: Subject can be optimized for one function or another but not both of them together, where the objects of the functions are distinct.

6. U6: Excessive action: where the action exceeds the optimal.
7. U7: Insufficient action caused by variations of Subject, Object or Action: when the Subject is optimized for a certain combination of Object's parameters and becomes sub-optimal when these parameters change.
8. H1: Harmful action: an action which is not usually designed but it is found as collateral effect. This action usually causes an undesired product.
9. H2: Subject performs both useful and harmful actions on the same object.
10. H3: Subject performs both useful and harmful actions on different objects.
11. H4: Concurring useful and harmful actions: this type of problems involves two elements (A and B), where A is performing a useful action on B and B is performing a harmful action on A. Therefore, the elements are both subjects and objects.
12. H5: Interfering object: where an interfering subject act on the object of an action that result diminished.
13. H6: Interfering object: where an interfering subject act on the subject of an action that result diminished.
14. H7: Interfering subjects: where the subjects of two actions interact among them reducing the entity of the actions.

Problems U3, U4, U5, U7, H2, H3, H4 resembles the concept of contradictions [24].

From the TRIZ community, Royzen [111] more or less explicitly defined a set of problems:

1. Reveal the Cause: when the problem is to identify causes of a failure or effect.
2. Detection or measurement: when the problem is about measuring or detecting a parameter.
3. Conflict: when the problem is in form of contradiction.
4. Harmful action: when the problem is a harmful action acting on an object.
5. Absent or insufficient action: when the action does not exist or its too weak.
6. Useful action: when I have an action which is performing well and I want to predict future improvements.

Gadd (from Oxford Creativity) stated that "a function is delivered by a complete S-a-O (an action between two components or two or more components interacting with each other)" [27]. Therefore, Gadd classified suggestions in accordance with the product that is obtained from this action:

1. Insufficient or weak;
2. Excessive or harmful;
3. Measurement and detection.

Similarly, Mann [120] has proposed four classes of standards, that are here intended more generally as types of problems:

1. incomplete Su-Field;
2. measurement/detection problem;
3. harmful effect;
4. insufficient/excessive relationship.

Valery Souchkov [165] developed a classification for the use of the 76 standard solutions [30,97] mainly based on TRIZ functional analysis (see figure 58)

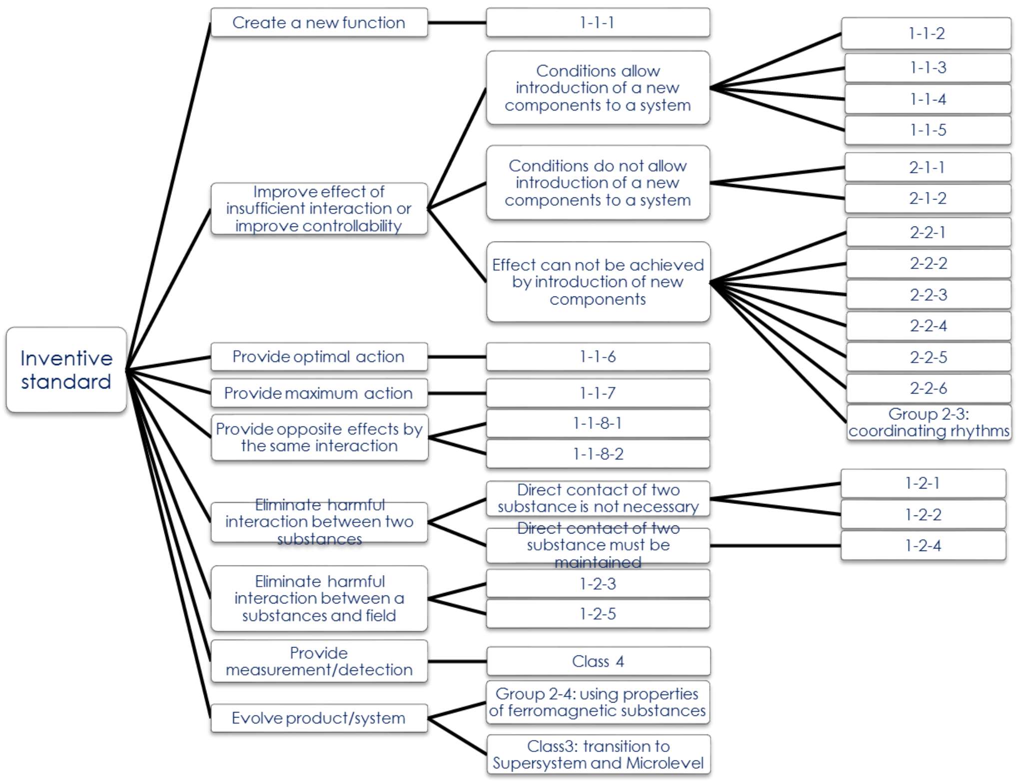


Figure 58. Souchkov's classification of the 76 standard solutions.

7.3.12 Contradictions as a type of problem

A contradiction literally means "No" but it is more generally referred as a proposition that assert apparently incompatible or opposite things [26]. Contradictions are a types of problems that are very famous in the TRIZ community. Altshuller [24] distinguished between three types of contradictions, although we may argue if the first one is or not a contradiction:

1. Administrative contradiction (AC): something has to be done, but how to do it is unknown.
2. Technical contradictions (TC): if one part (or one parameter) of a technical system is improved by any known method, some other part (or some other parameter) will be inadmissibly impaired.

3. Physical contradiction (PC): mutually opposing demands are placed upon one and the same system. The formulation can be "to be able to perform a specific function, given area should possess A property. At the same time, to comply with problem requirements, it should possess non-A property".

Savransky [26] defined a physical contradiction in the form:

The key subsystem (name) should be or has ("positive" parameter), in order to (the first requirement for the tool), the key subsystem (name) should not be or not have ("negative" parameter), in order to (the second requirement for the tool).

This formulation allows to identify different types of physical contradictions:

1. The key subsystem (or its parameter) A must exist and A must not exist [26].
2. A has to have the characteristic B and the characteristic –B (the opposite characteristics) [26].
3. The field-substance interaction must be strong and must be weak [26].
4. A must be in a phase state C and in another state C; e.g., C is plasma and C′ is solid; or C is gas and C′ is liquid [26].
5. A must be at a time period D and must not be at time period E (if E = D, see type 1) [26].
6. A must be constant (i.e., time independent) and A must change in time [26].
7. A parameter of a key subsystem has a spatially or temporally distributed value [115].
8. Two spatially or temporally related subsystems of a system have two values of the same parameters in the same and/or different space elements or at the same and/or different time (for example, different phases of matter) [115].
9. A key subsystem has the same parameter as one of its elements or as another subsystem of the system or whole system [115].
10. Performing the key function is necessary to achieve useful functions (UF), and not performing this function is necessary to avoid harmful functions (HF). [116].
11. The characteristic of the key subsystem must be of one value (big, infinite) to achieve one UF (useful function) and must be the opposite (small, zero) one to avoid HF (harmful function) or to achieve another UF. [116].
12. The key subsystem must be present to achieve one UF and must be absent to avoid HF or to achieve another UF [116].

BIBLIOGRAPHY

[1] Wang Y, Chiew V. On the cognitive process of human problem solving. Cogn Syst Res 2010;11:81–92. doi:10.1016/j.cogsys.2008.08.003.

[2] Sternberg RJ, Frensch PA. Complex problem solving: Principles and mechanisms. Psychology Press; 2014.

[3] Groner R, Groner M, Bischof WF. Methods of heuristics. Routledge; 2014.

[4] Pearl J. Heuristics: intelligent search strategies for computer problem solving 1984.

[5] Polya G. How to Solve It: A New Aspect of Mathematical Method: A New Aspect of Mathematical Method. Princeton university press; 2014.

[6] Israel P. Edison: A life of invention. John Wiley; 1998.

[7] Bransford JD. Who ya gonna call? Thoughts about teaching problem solving. Cogn Perspect Educ Leadersh 1993:171–91.

[8] Jonassen DH. Instructional Design Models for Well-Structured and Ill-Structured Problem-Solving Learning Outcomes. Educ Technol Res Dev 1997;45:65–90. doi:10.1007/bf02299613.

[9] Voss JF, Post TA. On the solving of ill-structured problems. Nat. Expert., 1988, p. 261–85.

[10] Csikszentmihalyi M, Getzels J. Discovery-oriented behavior and the originality of creative products: a study with artists. J Pers Soc Psychol 1971;19:47–52.

[11] Basadur M. Managing the creative process in organizations. Probl. Find. Probl. solving, Creat., 1994, p. 237–68.

[12] Runco M, Chand I. Cognition and creativity. Educ Psychol Rev 1995;7:243–67. doi:10.1007/BF02213373.

[13] Sternberg RJ. A three-facet model of creativity. Nat. Creat. Contemp. Psychol. Perspect., 1988, p. 454.

[14] Treffinger DJ. Creative Problem Solving: Overview and educational implications. Educ Psychol Rev 1995;7:301–12. doi:10.1007/BF02213375.

[15] John D. How we think. Heat New York 1910.

[16] Zlotin B, Zusman A. ARIZ on the Move. TRIZ Journal, March 1999.

[17] Simon HA, Newell A. Human problem solving: The state of the theory in 1970. Am Psychol 1971;26:145–59. doi:10.1037/h0030806.

[18] Manktelow JN. Mind tools: essentials skills for an excellent career. 2003.

[19] Shetty D. Design for Product Success. 2002.

[20] Pahl G, Beitz W, Feldhusen J, Grote K. Engineering design: a systematic approach. 2007.

[21] Umeda Y, Takeda H. Function, behaviour, and structure. Appl Artif Intell Eng V 1990;V:177–94.

[22] Umeda Y, Tomiyama T. FBS modeling: modeling scheme of function for conceptual design. Proc 9th Int Work Qual Reason 1995:271–8.

[23] Russo D, Spreafico C, Duci S. On the co-existance of FBS and TRIZ for simplifying design process in an iterative way. Int. Conf. Eng. Des. 2013, n.d.

[24] Altshuller GS. Creativity as an exact science. vol. 16. 1984.

[25] Salamatov Y. TRIZ: The Right Solution At The Right Time. 2005.

[26] Savransky SD. Engineering of creativity. 2000.

[27] Gadd K. TRIZ for Engineers: Enabling Inventive Problem Solving. Chichester, UK: John Wiley & Sons, Ltd; 2011. doi:10.1002/9780470684320.

[28] Royzen Z. Tool , Object , Product (TOP) Function Analysis. TRIZ J 1999.

[29] Rodenacker WG. Methodisches Konstruieren. Springer; 1971.
[30] Authors C, Altshuller GS. A Thread in the Labyrinth. 1988.
[31] Ogot M, Kremer G. Engineering design: a practical guide. 2004.
[32] Ogot M. EMS models: adaptation of engineering design black-box models for use in TRIZ. Proc ETRIA TRIZ Futur Conf 2004.
[33] Orloff M. Inventive thinking through TRIZ: a practical guide. 2006.
[34] Gordon W. Synectics: The development of creative capacity. 1961;11:180.
[35] Holyoak KJ, Thagard P. Mental Leaps: Analogy in Creative Thought. Am Polit Sci Rev 1995;89:745. doi:10.1207/s15327868ms1201_8.
[36] Verhaegen P-A, D'hondt J, Vandevenne D, Dewulf S, Duflou JR. Identifying candidates for design-by-analogy. Comput Ind 2011;62:446–59. doi:10.1016/j.compind.2010.12.007.
[37] Casakin H, Goldschmidt G, Planning T. Expertise and the use of visual analogy: implications for design education 1999;20:153–75.
[38] Chan J, Fu K, Schunn C, Cagan J, Wood K, Kotovsky K. On the Benefits and Pitfalls of Analogies for Innovative Design: Ideation Performance Based on Analogical Distance, Commonness, and Modality of Examples. J Mech Des 2011;133:081004. doi:10.1115/1.4004396.
[39] McAdams D a., Wood KL. A Quantitative Similarity Metric for Design-by-Analogy. J Mech Des 2002;124:173. doi:10.1115/1.1475317.
[40] Christensen BT, Schunn CD. The relationship of analogical distance to analogical function and preinventive structure: the case of engineering design. Mem Cognit 2007;35:29–38. doi:10.3758/BF03195939.
[41] Kolodner J. Educational implications of analogy: A view from case-based reasoning. Am Psychol 1997.
[42] Vattam S, Goel A. Seeking bioinspiration online: A descriptive account. DS 75-7 Proc. 19th Int. Conf. Eng. Des. (ICED13), Des. Harmon. Vol. 7 Hum. Behav. Des. Seoul, Korea, 19-22.08. 2013, 2013.
[43] Nagel JKSJ, Stone RB, Mcadams DA. An engineering-to-biology thesaurus for engineering design. Proc ASME 2010 Int Des Eng Tech Conf Comput Inf Eng Conf 2010:1–12.
[44] Nagel RL, Midha P a., Tinsley A, Stone RB, McAdams D a., Shu LH. Exploring the Use of Functional Models in Biomimetic Conceptual Design. J Mech Des 2008;130:121102. doi:10.1115/1.2992062.
[45] Linsey J. Design-by-analogy and representation in innovative engineering concept generation. 2007.
[46] Miller G a. WordNet: a lexical database for English. Commun ACM 1995;38:39–41. doi:10.1145/219717.219748.
[47] Deldin J-M, Schuknecht M. The AskNature Database: Enabling Solutions in Biomimetic Design. In: Goel AK, McAdams DA, Stone RB, editors. Biol. Inspired Des. SE - 2, Springer London; 2014, p. 17–27. doi:10.1007/978-1-4471-5248-4_2.
[48] Allen MS. Morphological creativity. Englewood Cliffs, NJ 1962.
[49] Osborn A. Applied imagination 1953.
[50] Litvin S. New TRIZ-Based Tool—Function-Oriented Search (FOS). TRIZ Journal, August 2005.
[51] Montecchi T, Russo D. Knowledge based approach for formulating TRIZ contradictions. TRIZ Futur. Conf. 2012, Lisbon, Port., 2012.
[52] Russo D, Montecchi T, Liu Y. Functional-based search for patent technology transfer. ASME 2012 Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf., American Society of Mechanical Engineers; 2012, p. 529–39. doi:10.1115/DETC2012-70833.

[53] Russo D, Montecchi T, Regazzoni D. A systematic exploration for conceiving function and behaviour of a new technical system. Proceeding IDMME Virtual Concept 2010 2010:1–6.
[54] Sloman S. Causal Models: How People Think about the World and Its Alternatives. vol. 117. 2005. doi:10.1093/acprof.
[55] Pearl J. Reasoning with Cause and Effect. AI Mag 2002;23:95–111. doi:10.1016/0888-613X(90)90013-R.
[56] Stamatis DH. Failure mode and effect analysis: FMEA from theory to execution. ASQ Quality Press; 2003.
[57] Heuvel LN Vanden. Root cause analysis handbook: a guide to effective incident investigation. Rothstein Associates Inc; 2005.
[58] Dettmer HW. Goldratt's theory of constraints: a systems approach to continuous improvement. ASQ Quality Press; 1997.
[59] Ishikawa K, Lu DJ. What is total quality control?: the Japanese way. vol. 215. Prentice-Hall Englewood Cliffs, NJ; 1985.
[60] Kepner CH, Tregoe BB. The new rational manager: An updated edition for a new world. Kepner-Tregoe Incorporated; 1997.
[61] Lee W-S, Grosh DL, Tillman FA, Lie CH. Fault Tree Analysis, Methods, and Applications ⍰ A Review. Reliab IEEE Trans 1985;34:194–203.
[62] Serrat O. The five whys technique 2009.
[63] Regel TL. The Root Cause Analysis handbook: A Simplified Approach to Identifying, Correcting, and Reporting Workplace Errors. Qual Prog 1998;31:116.
[64] Latino RJ, Latino KC. Root cause analysis 2006.
[65] Khomenko N, De Guio R, Lelait L, Kaikov I. A framework for OTSM? TRIZ-based computer support to be used in complex problem management. Int J Comput Appl Technol 2007;30:88–104.
[66] Cavallucci D, Khomenko N. From TRIZ to OTSM-TRIZ: addressing complexity challenges in inventive design. Int J Prod Dev 2007;4:4–21.
[67] Yezersky G. General Theory of Innovation. Trends Comput. Aided Innov., Springer; 2007, p. 45–55.
[68] Suh NP. Axiomatic design theory for systems. Res Eng Des 1998;10:189–209.
[69] Einstein A, Infeld L. The evolution of physics: the growth of ideas from early concepts to relativity and quanta. CUP Archive; 1961.
[70] Morgan M. Creating workforce innovation: turning individual creativity into organisational innovation. Business & Professional Publishing; 1993.
[71] De Bono E. Lateral Thinking; Creativity Step by Step. 1970.
[72] De Bono E. Six thinking hats. Taylor & Francis; 1989.
[73] De Bono E. Serious creativity: Using the power of lateral thinking to create new ideas. HarperCollins; 1992.
[74] Fey V, Rivin E. Innovation on demand: New product development using TRIZ. Cambridge University Press; 2005.
[75] Walker BA, Olson JC. Means-end chains: Connecting products with self. J Bus Res 1991;22:111–8.
[76] Goldschmidt G, Sever AL. Inspiring design ideas with texts. Des Stud 2011;32:139–55. doi:10.1016/j.destud.2010.09.006.
[77] Gonçalves M, Cardoso C, Badke-Schaub P. What inspires designers? Preferences on inspirational approaches during idea generation. Des Stud 2013;35:1–25.

doi:10.1016/j.destud.2013.09.001.
[78] Chulvi V, González-Cruz MC, Mulet E, Aguilar-Zambrano J. Influence of the type of idea-generation method on the creativity of solutions. Res Eng Des 2012;24:33–41. doi:10.1007/s00163-012-0134-0.
[79] Goldschmidt G, Smolkov M. Variances in the impact of visual stimuli on design problem solving performance. Des Stud 2006;27:549–69.
[80] van der Lugt R. How sketching can affect the idea generation process in design group meetings. Des Stud 2005;26:101–22. doi:10.1016/j.destud.2004.08.003.
[81] Schnotz W. Commentary: Towards an integrated view of learning from text and visual displays. Educ Psychol Rev 2002;14:101–20.
[82] Sarkar P, Chakrabarti A. The effect of representation of triggers on design outcomes. AI EDAM 2008.
[83] Chiu I, Shu L. Investigating effects of oppositely related semantic stimuli on design concept creativity. J Eng Des 2012:37–41.
[84] Fantoni G, Taviani C, Santoro R. Design by functional synonyms and antonyms: a structured creative technique based on functional analysis. Proc Inst Mech Eng Part B J Eng Manuf 2007;221:673–83. doi:10.1243/09544054JEM635.
[85] Prince GM. The practice of creativity. Harper; 1970.
[86] Horowitz R. Creative problem solving in engineering design 1999.
[87] Michalko M. Thinkertoys.. 1998.
[88] Smith G. Idea Generation Techniques: A Formulary of Active Ingredients. J Creat Behav 1998;32.
[89] Rader M, Porter AL. FTA Assumptions : Methods and Approaches in the Context of Achieving Outcomes The Development of FTA. Second Int. Sev. Semin. Futur. Technol. Anal., 2006.
[90] Porter AL. Technology foresight: types and methods. Int J Foresight Innov Policy 2010;6:36. doi:10.1504/IJFIP.2010.032664.
[91] Russo D, Spreafico C. TRIZ industrial case study: a critical survey. TRIZ Futur Conf 2015.
[92] Cavallucci D. World Wide status of TRIZ perceptions and uses; a survey of results. Rep. TRIZ Futur. 2009 Conf., 2009.
[93] Ilevbare IM, Probert D, Phaal R. A review of TRIZ, and its benefits and challenges in practice. Technovation 2013;33:30–7.
[94] Altshuller GS. 40 Principles: TRIZ Keys to Technical Innovation. Technical Innovation Center, INC.; 1 edition; 1998.
[95] Altshuller G. The Innovation Algorithm: TRIZ. 2000.
[96] Petrov V. System of generalized models 2008.
[97] Altshuller G. To find an Idea. Publ Nauk Novosib 1986.
[98] Altshuller GS, Selutskii AB. Wings of Icarus. 1980.
[99] RUSSO D, DUCI S. From Altshuller's 76 Standard Solutions to a new Set of 111 Standards. 13TH ETRIA WORLD TRIZ Futur Conf 2013 2013;131:305–15. doi:10.1016/j.proeng.2015.12.369.
[100] Altshuller G. The history of ARIZ development. J TRIZ 1986.
[101] Altshuller GS. Algorithm of inventive Problem Solving (ARIZ-85C). 1985.
[102] Cameron G. Trizics. 2010.
[103] Sickafus E. Unified structured inventive thinking: How to invent. Ntelleck; 1997.
[104] Turner S. ASIT-a problem solving strategy for education and eco-friendly sustainable design. Int J Technol Des Educ 2008;19:221–35. doi:10.1007/s10798-008-9080-6.

[105] Jonassen DH, Hung W. All Problems are Not Equal: Implications for Problem-Based Learning. Interdiscip J Probl Learn 2008;2. doi:10.7771/1541-5015.1080.
[106] Getzels JW. The Problem of the Problem. In: Hogarth R, editor. New Dir. Methodol. Soc. Behav. Sci. Quest. Fram. response consistency, San Francisco: Jossey-Bass; 1982.
[107] Barkan M. Process Management Using Systemic Thought Process : Identification and Formulation of Creative Tasks. TRIZ-Journal 2006.
[108] Amir Roggel. TRIZ Development at Intel Corporation. Japan TRIZ Symp., 2008.
[109] Jonassen DH. Toward a design theory of problem solving 2000;48:63–85. doi:10.1007/BF02300500.
[110] Koller R. Konstruktionsmethode für den Maschinen, Geraete und Apparatebau. Springer-Verlag Berlin; 1976. doi:10.1163/_q3_SIM_00374.
[111] Royzen Z. Solving Problems Using TOP-TRIZ. Www.aitriz.org 2008.
[112] Frenklach G. The Research (Diagnostic) Problems' Classification. TRIZ J 1998.
[113] Krasnoslobodtsev V, Langevin R. Applied TRIZ in high-tech industry. Proc TRIZCON2006, Milwaukee, USA 2006.
[114] Stone R, Wood K. Development of a functional basis for design. J Mech Des 2000;122.
[115] Glazunov VN. Parametric Method for Resolution of Contradictions in Technique, Rechnoj Transport. Moscow: 1990.
[116] Terninko J, Zusman A, Zlotin B. Innovazione Sistematica Un'introduzione a TRIZ. 1996.
[117] Kowalick J. Psychological inertia. TRIZ J 1998:4–7.
[118] Borgianni Y, Frillici FS, Rotini F. Integration of OTSM-TRIZ and Analytic Hierarchy Process for choosing the right solution 2012.
[119] Russo D, Montecchi T, Regazzoni D. A systematic exploration for conceiving function and behaviour of a new technical system. Extras.springer.com n.d.
[120] Mann D. Hands-on systematic innovation. 2002.
[121] Heinrich H, Petersen D, Roos N. Industrial accident prevention. 1950.
[122] Ikovenko S. Basic TRIZ Workshop n.d.
[123] Cooper RG. Stage-Gate Systems: A New Tool for Managing New Products. Bus Horiz 1990;33:44–54. doi:10.1016/0007-6813(90)90040-I.
[124] Cooper RG. Perspective: The stage-gate Idea-to-launch process - Update, what's new, and NexGen Systems. J Prod Innov Manag 2008;25:213–32. doi:10.1111/j.1540-5885.2008.00296.x.
[125] Tingström J, Swanström L, Karlsson R. Sustainability management in product development projects - the ABB experience. J Clean Prod 2006;14:1377–85. doi:10.1016/j.jclepro.2005.11.027.
[126] Araujo CS, Benedetto-Neto H. The utilization of product development methods: A survey of UK industry. J Eng Des 1996;7:13.
[127] Akao Y. QFD: Quality Function Deployment - Integrating Customer Requirements into Product Design. 2004.
[128] Camp RC. Benchmarking: The Search for Industry Best Practices that Lead to Superior Performance. 1989.
[129] Kumar A, Antony J, Dhakar TS. Integrating quality function deployment and benchmarking to achieve greater profitability. Benchmarking An Int J 2006;13:290–310. doi:10.1108/14635770610668794.
[130] Chan L-K, Wu M-L. Quality function deployment: A literature review. Eur J Oper Res 2002;143:463–97. doi:10.1016/S0377-2217(02)00178-9.

[131] Shi G-L, She Y-G. Study on problem-solving theory based on QFD, TRIZ and Taguchi methods. Xi Tong Gong Cheng Yu Dian Zi Ji Shu/Systems Eng Electron 2008;30:851–7.
[132] Griffin A, Hauser JR. The voice of the customer. Mark Sci 1993;12:1–27.
[133] Martilla JA, James JC. Importance-performance analysis. J Mark 1977:77–9.
[134] Livotov P. Using patent information for identification of new product features with high market potential. Triz Futur. Conf. 2014, 2014.
[135] Jin J, Ji P, Liu Y. Translating online customer opinions into engineering characteristics in QFD: A probabilistic language analysis approach. Eng Appl Artif Intell 2015;41:115–27. doi:10.1016/j.engappai.2015.02.006.
[136] Oh H. Revisiting importance–performance analysis. Tour Manag 2001;22:617–27. doi:10.1016/S0261-5177(01)00036-X.
[137] Ulwick AW. Turn customer input into innovation. Harv Bus Rev 2002;80:91–7. doi:citeulike-article-id:850877.
[138] Livotov P, Gmbh TE. Method for Quantitative Evaluation of Innovation Tasks for Technical Systems, Products and Processes. Proc ETRIA World Conf 2008:197–9.
[139] Jacoby J. The emerging behavioral process technology in consumer decision making research. Adv Consum Res 1977;4.
[140] Farris P, Bendle N, Pfeifer P. Marketing metrics: the definitive guide to measuring marketing performance. vol. 22. 2010. doi:10.1016/j.emj.2004.04.012.
[141] Abalo J, Varela J, Manzano V. Importance values for Importance–Performance Analysis: A formula for spreading out values derived from preference rankings. J Bus Res 2007;60:115–21. doi:10.1016/j.jbusres.2006.10.009.
[142] Eberle R. Developing Imagination Through Scamper*. J Creat Behav 1972;6:199–203. doi:10.1002/j.2162-6057.1972.tb00929.x.
[143] Newell A, Simon H. Human problem solving. 1972.
[144] Jansson DG, Smith SM. Design fixation. Des Stud 1991;12:3–11.
[145] Ramos VC, Escrig EM. Comparison of the degree of creativity in the design outcomes using different design methods 2012:37–41.
[146] Ward TB, Patterson MJ, Sifonis CM. The Role of Specificity and Abstraction in Creative Idea Generation. Creat Res J 2004;16:1–9. doi:10.1207/s15326934crj1601_1.
[147] Belski Iouri, Belski Anne, Berdonosov Victor, Busov Bohuslav, Bartlova Milada, Malashevskaya Elena, Kässi Tuomo, Kutvonen Antero TN. Can simple ideation techniques influence idea generation: comparing results from Australia, Czech Republic, Finland and Russian Federation. In: 2015 AA for EEC, editor., 2015.
[148] Belski I. Improve your thinking: substance-field analysis. TRIZ4U; 2007.
[149] Shah JJ, Kulkarni S V., Vargas-Hernandez N. Evaluation of Idea Generation Methods for Conceptual Design: Effectiveness Metrics and Design of Experiments. J Mech Des 2000;122:377. doi:10.1115/1.1315592.
[150] Lin C, Hong J. A study of the applicability of idea generation techniques. Think Ski Creat 2006:1–21.
[151] Chiu I, Shu L, Burg J. Using language as related stimuli for concept generation. AI EDAM Artif Intell Eng Des Anal Manuf 2007;21:103–21.
[152] Chiu I, Shu LH. Use of opposite-relation lexical stimuli in concept generation. CIRP Ann - Manuf Technol 2008;57:149–52. doi:10.1016/j.cirp.2008.03.023.
[153] Terninko J, Domb E, Miller J. The seventy-six standard solutions, with examples (Classes from 1 to 5). TRIZ J 2000.

[154] Cavallucci D, Lutz P, Kucharavy D. Converging in problem formulation: a different path in design. 2002 ASME Des Eng Tech Conf 2002:279–86. doi:10.1115/DETC2002/DTM-34025.
[155] Soderlin P. Thoughts on Substance-Field Models and 76 Standards Do we need all of the Standards? TRIZ J 2003.
[156] Bultey A, De Beuvron FB De, Rousselot F. A substance-field ontology to support the TRIZ thinking approach. Int J Comput Appl Technol 2007;30:113–24. doi:10.1504/IJCAT.2007.015702.
[157] Terninko J. Su-field analysis. TRIZ J 2000.
[158] Petrov V. The history of development of standards. 2003.
[159] Kim S. Innovative Design of Substance-Field Notations for Reformulating the Seventy-Six Standard Solutions in TRIZ. Int J Syst Innov 2012;26.
[160] Souchkov V. Annotated List of Key TRIZ Components and Techniques 2011:1–7.
[161] Khomenko N. How to use Standard Solution. Www.triz.co.kr 2002.
[162] Souchkov V. Accelerate Innovation with TRIZ 1997.
[163] Kim S. Conceptual Design Based on Substance-Field Model in Theory of Inventive Problem Solving. Int J Innov Manag Technol 2012;3:306–9.
[164] Kowalick J. Altshuller's Greatest Discovery - And Beyond. TRIZ J 1997.
[165] Souchkov V. Systematic Innovation for technology applications. Course Slides (Bergamo) 2005.
[166] Plato, Cornford FM. Plato's Theory of Knowledge: the Theaetetus and the Sophist of Plato 1935:xiv, 336.
[167] Gettier E. Edmund Gettier , " Is Justified True Belief Knowledge ?" Analysis 1963;23:444–6.
[168] Hatchuel A, Weil B. A New Approach of Innovative Design : an Introduction To C-K Theory. Int Conf Eng Des 2003:1–15. doi:citeulike-article-id:4891368.
[169] Duncker K. On problem solving. Psychol Monogr 1945;58:270.
[170] Maier NRF. Problem Solving and Creativity; In Individuals and Groups. 1970.
[171] McDermott D. Planning and Acting*. Cogn Sci 1978;2:71–100. doi:10.1207/s15516709cog0202_1.
[172] Mayer RE, Wittrock MC. Problem Solving . Handb. Educ. Psychol., 2006, p. 287–304.
[173] Anderson J. Cognitive psychology and its implications 2009.
[174] Lynch CL, Wolcott SK, Huber GE. Tutorial for optimizing and documenting open-ended problem solving skills 2000.
[175] Simon HA. The Sciences of the Artificial. vol. 1. 1969. doi:10.1016/S0898-1221(97)82941-0.
[176] Shah JJ, Smith SM, Vargas-Hernandez N. Metrics for measuring ideation effectiveness. Des Stud 2003;24:111–34. doi:10.1016/S0142-694X(02)00034-0.
[177] Oman SK, Tumer IY, Wood K, Seepersad C. A comparison of creativity and innovation metrics and sample validation through in-class design projects. Res Eng Des 2012;24:65–92. doi:10.1007/s00163-012-0138-9.
[178] Nelson BA, Wilson JO, Rosen D, Yen J. Refined metrics for measuring ideation effectiveness. Des Stud 2009;30:737–43. doi:10.1016/j.destud.2009.07.002.
[179] Verhaegen P, Vandevenne D, Peeters J, Duflou JR. A variety metric accounting for unbalanced idea space distributions. TRIZ Futur. 2011, 2011, p. 205–14.
[180] Lester S. Beyond knowledge and competence towards a framework for professional education. Capability 1995;1:44–52.
[181] Murphy E. Identifying and Measuring Ill-Structured Problem Formulation and Resolution in Online Asynchronous Discussions. Can J Learn Technol 2004;30.

[182] Guilford JP. Three faces of intellect. Am Psychol 1959;14:469.
[183] Mednick S. The associative basis of the creative process. Psychol Rev 1962;69:220.
[184] Koestler A. The act of creation 1964.
[185] Lenat DB. The ubiquity of discovery. Artif Intell 1977;9:257–85.
[186] Perkins D. An unfair review of Margaret Boden's The Creative Mind from the perspective of creative systems:(Basic Books, New York, 1991); 303 pages 1995.
[187] Perkins DN. Insight in minds and genes. 1995.
[188] Hofstadter DR. Metamagical Themes: Questioning for the Essence of Mind and Pattern 1985.
[189] Boden MA. The creative mind: Myths and mechanisms. Psychology Press; 2004.
[190] Finke RA, Ward TB, Smith SM. Creative cognition: Theory, research, and applications 1992.
[191] Wallas G. The art of thought. 1926.
[192] Schank RC. 9 Creativity as a mechanical process. Nat Creat Contemp Psychol Perspect 1988:220.
[193] Schank RC, Cleary C. Making machines creative. Creat Cogn Approach 1995:229–47.
[194] Weisberg RW. Creativity: Beyond the myth of genius. WH Freeman New York; 1993.
[195] Mascitelli R. From experience: harnessing tacit knowledge to achieve breakthrough innovation. J Prod Innov Manag 2000;17:179–93. doi:10.1016/S0737-6782(00)00038-2.
[196] Collins MA, Amabile TM. Motivation and creativity. Handb Creat 1999:297–312.
[197] Hoover SM, Feldhusen JF. Scientific problem solving and problem finding: A theoretical model. Probl Find Probl Solving, Creat 1994:201–19.
[198] Sweller J. Cognitive load during problem solving: Effects on learning. Cogn Sci 1988;12:257–85. doi:10.1016/0364-0213(88)90023-7.
[199] Gick ML, Holyoak KJ. Schema induction and analogical transfer. Cogn Psychol 1983;15:1–38. doi:10.1016/0010-0285(83)90002-6.
[200] Mayer RE, Wittrock MC. Problem-solving transfer. Handb. Educ. Psychol., 1996, p. 47–62.
[201] Williams WM, Yang LT. Organizational creativity. Handb Creat 1999:373–91.
[202] Jackson SE, Chuang CH, Harden EE, Jiang Y. Toward Developing Human Resource Management Systems for Knowledge-Intensive Teamwork. Res Pers Hum Resour Manag 2006;25:27–70. doi:10.1016/S0742-7301(06)25002-3.
[203] Paletz SBF, Schunn CD. A social-cognitive framework of multidisciplinary team innovation. Top Cogn Sci 2010;2:73–95. doi:10.1111/j.1756-8765.2009.01029.x.
[204] Jehn K a, Northcraft GB, Neale M a. Why Differences Make a Difference: A Field Study of Diversity, Conflict, and Performance in Workgroups. Adm Sci Q 1999;44:741–63. doi:10.2307/2667054.
[205] Gebert D, Boerner S, Kearney E. Cross-functionality and innovation in new product development teams: A dilemmatic structure and its consequences for the management of diversity. Eur J Work Organ Psychol 2006;15:431–58. doi:10.1080/13594320600826314.
[206] Simon H. The structure of ill structured problems. Artif Intell 1973;4:181–201.
[207] Jonassen DH. Instructional Design Models for Well-Structured and Ill-Structured Problem-Solving Learning Outcomes. Educ Technol Res Dev 1997;45:65–90. doi:10.1007/BF02299613.
[208] Becker JP, Shimada S. The Open-Ended Approach: A New Proposal for Teaching Mathematics. ERIC; 1997.
[209] Coyne R. Wicked problems revisited. Des Stud 2005;26:5–17. doi:10.1016/j.destud.2004.06.005.
[210] Rittel HWJ, Webber MM. Dilemmas in a General Theory of Planning. Policy Sci 1973;4:155–69.

[211] Wood PK. Inquiring systems and problem structure: Implications for cognitive development. Hum Dev 1983;26:249–65.
[212] Kitchener KS. Cognition, metacognition, and Epistemic cognition. Hum Dev 1983;26:222–32. doi:10.1159/000272885.
[213] Meacham JA, Emont NC. The interpersonal basis of everyday problem solving. 1989.
[214] Funke J. Solving complex problems: Exploration and control of complex systems. Complex Probl Solving Princ Mech 1991:185–222.
[215] Ponn J, Lindemann U. Characterization of design situations and processes and a process module set for product development. Int Conf Eng Des 2005.
[216] Singley MK, Anderson JR. The transfer of cognitive skill. Harvard University Press; 1989.
[217] Eysenck MW, Keane MT. Cognitive Psychology: A Student's Handbook. 2005.
[218] Fantoni G, Gabelloni D, Tilli J. Concept design of new grippers using abstraction and analogy. Proc Inst Mech Eng Part B J Eng Manuf 2013;227:1521–32. doi:10.1177/0954405413489967.
[219] Fu K, Moreno D, Yang M, Wood KL. Bio-Inspired Design: An Overview Investigating Open Questions From the Broader Field of Design-by-Analogy. J Mech Des 2014;136. doi:10.1115/1.4028289.
[220] Hirtz J, Stone RB, Mcadams DA, Szykman S, Wood KL. A Functional Basis for Engineering Deisgn: Reconciling and Evolving Previous Efforts. Res Eng Des 2002;13:65–82. doi:DOI: 10.1007/s00163-001-0008-3.
[221] Daniel Theyagu. Gravitating Toward Success. 2012.
[222] Zlotin B, Zusman A. An integrated Operational Knowledge Base (System of Operator) and the Innovation Workbench System Software. TRIZ J 1999.
[223] Evbuomwan NFO, Sivaloganathan S, Jebb A. A survey of design philosophies, models, methods and systems. Proc Inst Mech Eng Part B J Eng Manuf 1996;210:301–20.
[224] Miller J, Domb E, MacGran E, Terninko J. Using the 76 Standard Solutions: A case study for improving the world food supply. TRIZ J 2001:1–18.
[225] Modarres M, Cheon SW. Function-centered modeling of engineering systems using the goal tree–success tree technique and functional primitives. Reliab Eng Syst Saf 1999;64:181–200. doi:10.1016/S0951-8320(98)00062-3.
[226] Cheong H, Chiu I, Shu LH, Stone RB, McAdams D a. Biologically Meaningful Keywords for Functional Terms of the Functional Basis. J Mech Des 2011;133:021007. doi:10.1115/1.4003249.
[227] Hirtz J, Stone RB, Mcadams DA, Szykman S, Wood KL. A Functional Basis for Engineering Previous Efforts A Functional Basis for Engineering Deisgn: Reconciling and Evolving Previous Efforts. Res Eng Des n.d.
[228] Nagel JKS, Nagel RL, Stone RB, McAdams D a. Function-based, biologically inspired concept generation. Artif Intell Eng Des Anal Manuf 2010;24:521–35. doi:10.1017/S0890060410000375.
[229] Cascini G, Rotini F, Russo D. Functional modeling for TRIZ-based evolutionary analyses. DS 58-5 Proc. ICED 09, 17th Int. Conf. Eng. Des. Vol. 5, Des. Methods Tools (pt. 1), Palo Alto, CA, USA, 24.-27.08. 2009, 2009, p. 371–84.
[230] Pinyayev AM. A Method for Inventive Problem Analysis and Solution Based On Why-Why Analysis and Functional Clues. St Petersburg, Russ Int TRIZ Assoc TRIZ Master Thesis 2007.
[231] Pinyayev AM. Trevozhnyi chemodanchik izobretatelya. TRIZ J 1995;1.