Compino-TRIZ: A Software Package For Formulating And Solving Inventive Problems

1. Software Package Development Ideology

The main goal of the Compino-TRIZ (Compass for Innovations Based on TRIZ Tools) software package being developed by the authors is to create an automated system for formulating and solving inventive problems. Compino-TRIZ is based on the idea of integrating ARIZ, functional analysis and application roadmap for tools used in complex analysis of systems.

***Key words**: ARIZ, function of a system, contradictions of requirements, inventive problems outside the engineering field, ele-field analysis, evolutionary systemology.*

2. Compino-TRIZ Package Composition and Structure

At present, the Compino-TRIZ package includes:

- ARIZ-U-2014 – an updated version of ARIZ-Universal-2010 enabling the automation of major ARIZ steps.
- Updated (as of 14.06.2014) version of Universal Standards System for Inventive Problem Solving - 2010 based on ele-field analysis.
- Component, structural and functional analysis.
- Means to formulate a generalized function for transition to FOS.
- Fragments of physical effects catalog.
- Tools for formulating and reformulating problems and revealing secondary problems.

Some ARIZ-U-2014 algorithm steps have been updated to maximize the degree of automation when implemented in the software package.

ARIZ-U-2014 steps used for building a problem model provide a simple and logical transition to tools applied in component, structural and functional analysis. And conversely, a problem model is automatically formed in the process of functional analysis to provide a transition to next ARIZ-U-2014 steps of problem solving.

Based on the selected problem model, the Compino-TRIZ software package formulates an IFR, builds solution models and formulates requirements contradictions (technical contradictions).

3. Fundamental Principles of TRIZ Tools Integration in Compino-TRIZ

The integration of different tools for inventive problem formulating and solving proceeded from the following ideas:

- a set of system function models enables an automatic transition to problem models and IFR formulation;
- typical problem models enable an automatic transition to recommended principles and standards of inventive problem solving;
- parametric description of function models makes it possible to draw generalized function models to be used as a basis for transition to function oriented search and physical effects catalog.

4. ARIZ-U-2010 and ARIZ-U-2014 Concepts and Terms

There are terms and concepts in ARIZ-U-2010 and ARIZ-U-2014 that require preliminary clarification:

- A function model includes a Subject (carrier) of the function and an Action directed at the Object of the function. The action can be described either by a verb, or by modification of one or more parameters of the function Object. Five options of action that affects a function object parameter can be identified: increase – decrease, stabilization – variation, measurement.
- A set of system functions (system components relationships) is a set of function models or system relationships (1-5 or more) interconnected by elements and containing conflicting requirements.
- A system components relationship model includes Component 1, a verb, and Component 2. This description summarizes the function model. Here, the verb must not necessarily mean an action; it can describe the relations or states of the components.
- FOS, or function-oriented search, is aimed at revealing systems with analogous functions; the reverse FOS is a search for possible fields of application for a function.
- Function-field analysis is a system function model analysis supplemented with fields of interaction between function components. A function model of a system consists of models of functions. A function-field system consists of Element-Fields (Ele-Fields) of this system.
- Standard models (patterns) of conflicts are described in [2] and in Table 2.

- An Ele-Field (elements and their fields of interaction) is a generalized analog of a Su-Field and function model for material and non-material systems. An Ele-field can be regarded as a function model supplemented with a field of interaction between the function carrier and function object [7].

- Universal Standards System for Inventive Problem Solving 2010 [7] is designed to search for inventive problem solutions related to material and non-material systems.

- Contradictions of requirements represent a generalized analog of a technical contradiction for material and non-material systems. System requirements arise from the supersystem. A formulating option: IF...(indicate an introduced change)..., THEN (indicate the main requirement), BUT (indicate an undesirable requirement).

- Contradiction of features is a generalized analog of physical contradiction for material and non-material elements. It can be formulated for any features (aspects of analysis) of objects: physical, chemical, biological, aesthetic, artistic, etc. The features of a system are associated with its internal structure. A formulating option: an element of a conflicting pair must possess a feature X to meet the main requirement, and at the same time it must possess a feature "ANTI-X" to eliminate a harmful function associated with it.

- Aspects of system analysis: physical, chemical, biological, technical, social, psychological, legal, financial-and-economic, etc.

- Principles of resolving feature contradictions: in time, in space, through system transition, in relationships. Relationship contradictions can be resolved for both material and non-material systems.

- Principles of resolving requirement contradictions (40 major and 10 additional principles proposed by G.S.Altshuller [2] for engineering systems). Each principle of resolving feature contradictions can be correlated to a set of principles for resolving requirement contradictions. Some of the principles (25 out of 40 major ones), e.g., fragmentation, out-taking, integration, reversing, dynamicity, etc., can be applied to non-material systems.

- Functional IFR: An object (name) all by ITSELF does (describe what) during the period (indicate the period) under mandatory conditions (describe the constraints).
- Resource IFR: X-element from system resources ELIMINATES all by ITSELF harmful functions (name), while RETAINING useful functions (name).
- Feature IFR: The operational zone (indicate) must provide (indicate opposing macro- or micro-states or features) all by ITSELF during the operational period (indicate).
- SFR: substance-and-field and other resources.
- The principle of system operation is determined by three parts: system components, system of functions, and system “tissue” (what the system components consist of).

5. ARIZ-U-2014 Block Diagram

Fig.1 shows a block diagram of ARIZ-U-2014.

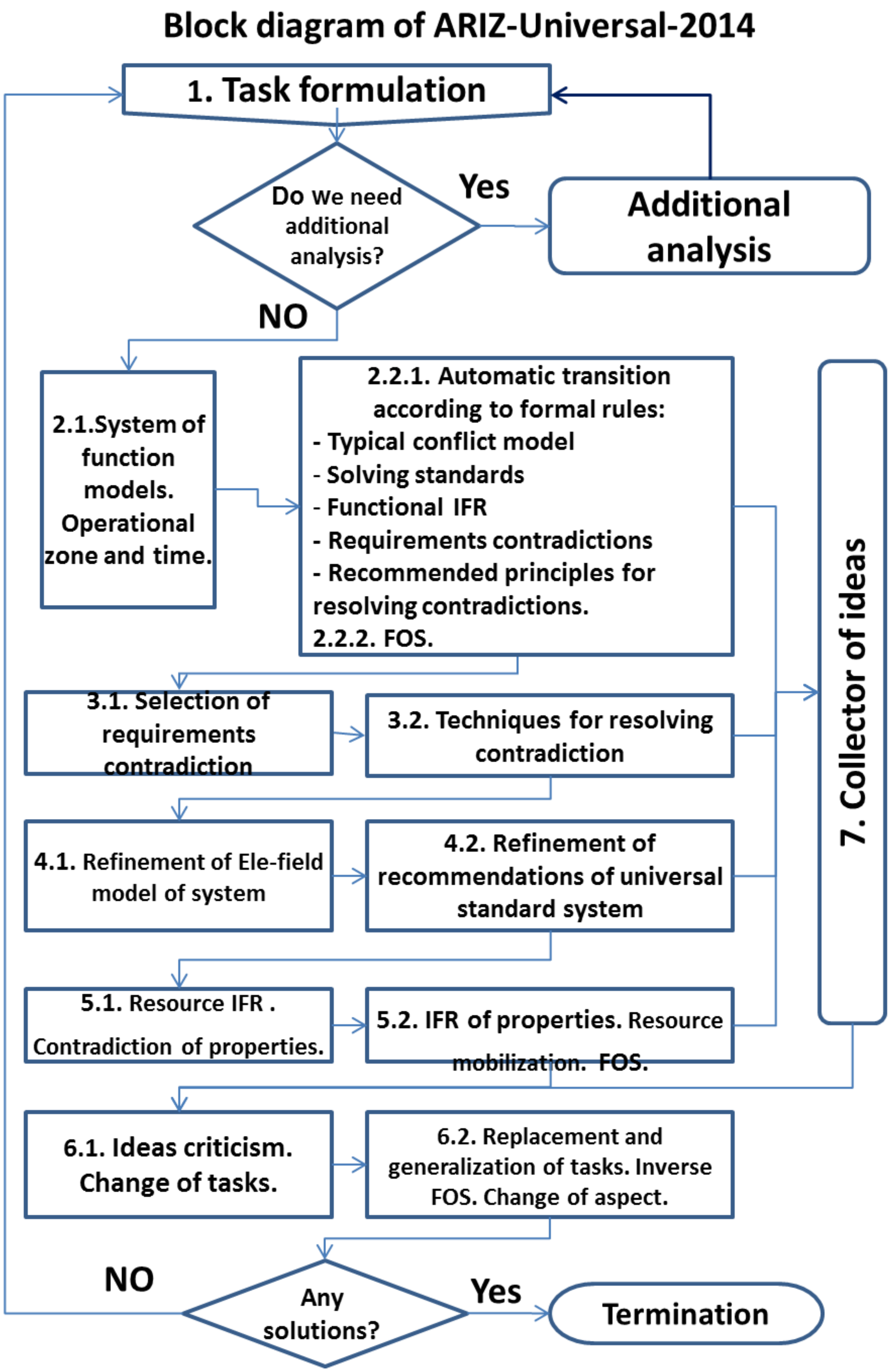


Fig. 1. ARIZ-U-2014 Block Diagram.

All ARIZ-U-2014 steps are divided into three groups: system analysis, synthesis of a new system, and evaluation and revision of proposed ideas. (Fig. 2)

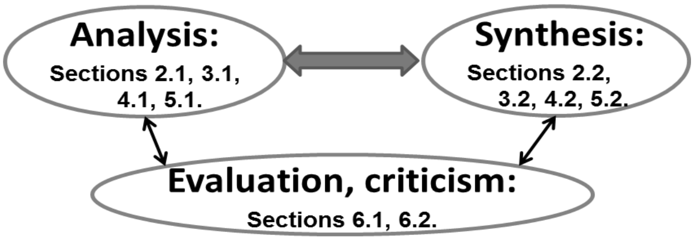


Fig. 2. Logics and Relationship of Individual ARIZ-U-2014 Sections.

Step 2.2.1 is performed following the formal rules described below and is fully automated in the COMPINO-TRIZ software package described below. Table 1 compares ARIZ-U-2010 to ARIZ-U-2014.

Table 1. ARIZ-U-2010 Versus ARIZ-U-2014 Comparison

Section	ARIZ-U-2010	ARIZ-U-2014
Com-ments	Analysis of systems: engineering, non-engineering, non-material (informational), statement and solving of inventive problems. For level 2-4 problems. Short description of ARIZ-U-2010 steps is given below.	Adapted for implementation in computer programs; problem model is stated as a system of functions, the main formulations are generated automatically. The changes listed below have been introduced.
1	Source problem. System elements and parameters. Problem template. Problem re-formulation. System analysis, function-and-field analysis, etc. Use of various methods of analysis and problem statement (key problems identification, etc.). Problem scale analysis template. Must the stated problem be solved?	The template of problem formulation has been transferred to step 2.1 and integrated with function template.
2.1.	Function model and constraints. Function template. Analysis of parameters and parametric model of functions. Information search in information databases. Operational time (OT). Operational zone (OZ).	Set of functions (1-3 or more) containing disadvantages and contradictions is described following the assigned template. Based on this system of functions, the type of conflict is determined automatically, standards for conflict elimination are offered, formulation of requirement contradictions and functional IFR are proposed.
2.2.	Functional IFR. FOS. Information search. Function model refinement. Source problem	Selection of functional IFR statement proposed by the algorithm.

	refinement.	
3.1.	Requirement contradictions: RC-1, RC-2. Refinement of algorithm for drawing up requirement contradictions.	Selection of requirement contradictions proposed by the algorithm.
3.2.	Table of principles. Conflict resolution principles.	Selection of principles the most suitable for a given problem becomes possible.
4.1.	Conflicting elements. Conflict pattern. OT. OZ. Variety of conflict models.	Automatically selected conflict model is refined if necessary.
4.2.	Ele-field problem model. System of standards.	Problem model and problem solving standards are refined if necessary.

Other sections of ARIZ-U-2014 remained unchanged as compared to ARIZ-U-2010:

5.1.	Resource IFR. Feature contradictions. Resource analysis. Micro-algorithm for feature contradiction formulation. Resource IFR of features. Micro-resource IFR of features.
5.2.	Feature IFR. Substance-and-field resources (SFR). Micro-algorithm for formulating IFR of features. Principles of solving feature contradictions (FC). Application of effect catalogs, FOS evolution lines in information databases.
6.1.	Analysis and revision. STC (size – time – cost) method for revealing potential problems. Secondary problems. Questions for revealing secondary problems. Super-effects resulting from changes introduced into the system.
6.2.	Problem generalization. Change of the problem and the aspect of its analysis. Analysis of operation principle and reverse FOS.
7.	Collector of ideas and problem solving roadmap.

6. Inventive Problem Solving Standards

The Universal Standards System for Inventive Problem Solving - 2010 [7] is designed for solving problems in engineering and non-engineering systems and is characterized by the following structure:

U1. Ele-fields Synthesis

U1.1. Creation of Ele-Field structure (new system)

U1.2. Elimination of harmful relationships in Ele-field

U1.2.1 Elimination of harmful relationships through replacement, change or addition of elements

U1.2.2 Elimination of harmful relationships through addition of fields

U2. Development of Ele-field structures

U2.1. Transition to complex Ele-field

U2.1.1. Enhancement of Ele-field efficiency through element introduction

U2.1.2. Setting limiting modes for fields.

U2.2. Creation of double Ele-field.

U2.2.1. Enhancement of Ele-field efficiency through field introduction.

U2.2.2. Setting minimum mode for an element.

U2.3. Creation of chain Ele-Field

U3. Synthesis and efficiency enhancement of systems for measurements and identification (systems possessing the features of interaction fields)

U3.1. Bypasses

U3.2. Synthesis and efficiency enhancement of systems for measurements and identification

U4. Lines of systems evolution

U.4.1. Line of components (substances) introduction

U.4.2. Line of interaction fields introduction and development

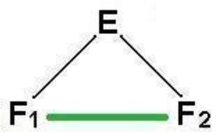
U.4.3. Line of fragmentation and dynamization

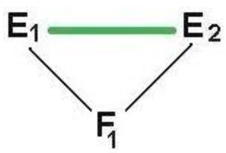
U.4.4. Lines of coordination-discoordination and structurization

U.4.5. Transition to supersystems and substystems (to micro-level)

U.4.6. Lines of collective and individual use of systems

U.4.7. Lines of systems evolution in accordance with S-curves.





The Universal Standards System for Inventive Problem Solving – 2010 is based on the language of Ele-field analysis. In particular, the Ele-field analysis proceeds from the fact that any component, subject to situation and its supersystem, can be viewed as an element/substance (inner Ele-field) in one case, and as a field (outer Ele-field) in another. For instance, a flow of liquid can be regarded as a substance (a liquid) and as a filed (flow-distributed velocities and pressures). As is known, an electron can display the features of a substance particle in one case and field properties in another. Such a behavioral duality of electrons and other microparticles is one of the general properties of the matter (as well as substance and field). So, the same elements of algorithms and software programs can be regarded as elements (subjects or objects of a function) in one case or as fields of interaction with other program components in another case.

7. Set of Functions and Transition to Typical Conflict Patterns

A set of functions consists of one or several interrelated function models that jointly describe one or several problem situations in a source problem. Each function model in the set can be characterized as: a useful function, harmful function, insufficient function, uncontrolled function, or nonexecutable function. Each of those functions can be marked as unchangeable from the viewpoint of a given system (problem).

This set of functions could be illustrated by the well-known TRIZ problem of vortex formation caused by a parachute mock-up:

“To study the formation of vortex, a mock-up of parachute (tower, etc.) is placed inside a glass tube through which water is pumped. The process is monitored visually. Colorless swirls, however, are poorly visible against the background of a colorless flow. If we dye the flow, the observation will become even more difficult: black swirls become absolutely invisible against the background of black water. To overcome this difficulty, a thin layer of water-soluble dye is applied over the parachute mock-up, thus making the colored swirls visible in colorless water. Unfortunately, the dye runs out quickly. If a thick layer of dye is applied, the size of the parachute mock-up gets distorted and any monitoring becomes senseless. What is to be done?"

Several sets of function models can be identified for solving this problem.

The first option of a set of functions can consist of one function only:

Function 1. The parachute mock-up Dyes (changes color of) Water swirls (useful, insufficient).

The second option of functions set is:

Function 2.1. The dyeing agent Dyes (changes color of) Swirls (useful).

Function 2.2. The water Dilutes (decreases thickness of) Dyeing agent (harmful).

Function 2.3. The dyeing agent Dyes (changes composition of) Water (useful).

Function 2.4. The water Generates (changes shape of) Swirls (useful).

Based on the algorithm given below and Table 2, the type of conflict is determined and a standard solution is recommended.

Conflict type identification algorithm based on set of functions

1.Major elements as well as functions associated with them (useful and harmful) are identified. These sets of functions must be referred to one of the six types of conflicts given in the Table 2.

2.If the required function is known, but the function subject is missing (X-element), **the 1-st type** of conflict is recommended.

3.If the described functions perform an insufficient useful function, **the 2-d type** of conflict is recommended.

4.If two useful functions acting upon the same object are present and one of these functions is performed insufficiently, **the 3-rd type** of conflict is recommended.

5.If a harmful function (harmful link) is found among the described functions, which has elements that are inseparably associated with the useful or unchangeable function, **the 4-th type** of conflict is recommended.

6.If an uncontrollable (poorly controllable) useful function is present among those described, **the 5-th type** of conflict is recommended.

7.If the described functions are associated with field measurement, identification or transformation, **the 6-th type** of conflict is recommended.

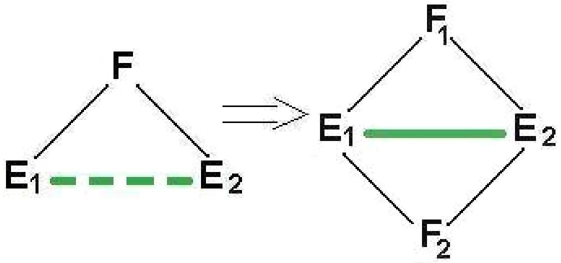
8.If a set of functions or component relationship models includes several types of conflicts, the problem situation is subdivided into several problems having the same type of conflict.

Table 2.Typical Conflict Patterns and Problem Models

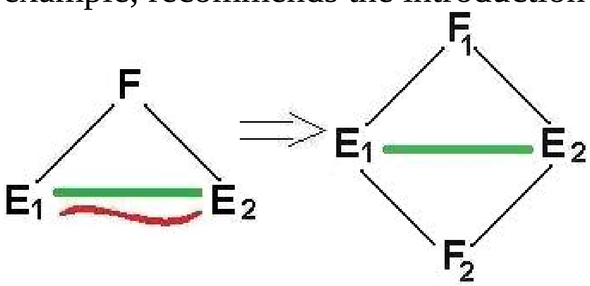
Number and type of conflict	**Description of typical conflicts**	**Recommended solving models**
1.Required useful action is missing	Useful action upon element B is missing	• Standard U1.1.
2.Useful action is insufficient (incomplete)	Element A performs a useful action in relation to element B incompletely or with insufficient quality	• Standards U2.1.1, U2.2.1, 2,3 or • Standard U1.1 (replace the element)
3.Useful actions are incompatible	One useful action of element A upon element B blocks the implementation of another useful action of element A upon element B	• Standards U2.1.1, U2.2.1, U2.3. • Eliminate the need to perform one of the two actions (trimming): no need for A-B (or C-B) function; the function is performed by a resource element instead of A (C) element; element B performs the function by itself
4.Harmful function	Counteraction: element A positively acts upon element B, while element B acts harmfully upon element A.	• Standard U1.2.1, U1.2.2. • Eliminate the need to perform one of the two actions (trimming): ✓ no need for A-B (or C-B) function; ✓ a resource element performs the action
	Conjugated action. Element A produces both positive and negative action upon element B. Or a useful action is produced upon one part of element B, while a harmful action is produced upon its other part. Or A produces a useful action upon B, and a harmful action upon C, which is associated	

	with B. Or A harmfully acts upon itself while performing a useful action upon B.	instead of A (C) function; ✓ element B performs this function by itself ✓ apply function analysis and trimming • Standard U1.1 (replace element A)
5.Unregulated action	Element A acts excessively or insufficiently upon element B.	Standards U2.1.2, U2.2.2.
6."Silence"	Measurement problems	Standard U3.1, U3.2.

For example, function 1 “The parachute mock-up Dyes (changes color of) Water swirls (useful, insufficient)” corresponds to the 2-d type of conflict, then standards U2.1.1, U2.2.1,2,3 or standard U1.1. from the Universal Standards System are recommended for its elimination. For example, standard U2.2.1 recommends introducing an additional interaction field between the parachute mock-up and water swirls.



The second option of functions set (as applied to the above example) emphasizes the conflict between function 2.2. “Water Dilutes (decreases thickness of) Dyeing agent (harmful)” and function 2.3. “Dyeing agent Dyes (changes composition of) Water (useful)”. In other words, the dyeing agent performs a useful action upon water, while water acts harmfully upon the dyeing agent. It corresponds to the 4th type of conflict: apply standards U1.2.1, U1.2.2, standard U1.1 or eliminate the need to perform one of the two functions (trimming). Standard U1.2.2, for example, recommends the introduction of an additional field (e.g., electric field).



The check solution for this problem, as is known, consists of using electrolysis to induce the emission of gas bubbles out of water, which replace the dyeing agent in making swirls visible. Trimming recommendations also prompt that the dyeing agent must be substituted with a system resource, for example, “emptiness” (bubbles).

8. Set of System Component Relationship Models

Of course, non-engineering and especially non-material systems have specific characteristics, which must be considered when using TRIZ tools for inventive problem solving. These specifics are taken into account one by one as they are revealed through the analysis of systems taken from various fields of human activity.

For example, the construction of a model set needed for stating a problem does not always provide for an opportunity to build a set of function models, if, for example, social, legal and financial systems are at hand. In such situations, sets of system component relationship models can be built instead of sets of function models.

A set of system component relationship models consists of the following triad:

Component 1 – Verb – Component 2

When the verb in the triad is an action verb describing the change of a parameter of Component 2, we obtain a function model. For example, Heat (increase temperature), Thicken (grow in weight and volume), Crush (decrease integrity).

If the verb in a component relationship model describes relationships (mutual attitude, condition) of two components, we obtain a component relationship model. For example:

- a company owns shares or shares belong to a company;
- a train consists of cars or cars are joined into a train;
- a suspect is victim's brother;
- a house has balconies**.**

Although these triads do not constitute functions, they possess the same properties as function models in terms of problem situation analysis and problem model construction. They may contain contradictions or provide a basis for formulating IFR; same algorithms can be used to generate recommendations for applying standards, principles and methods for resolving contradictions.

For example, the following problem is known:

At the end of the XIX century, the Duma of St. Petersburg and shareholders of the horse-drawn railway company "Konka" that operated the St. Petersburg horse-drawn tram network concluded an exclusive contract prohibiting any other operator from providing mass transportation for the population of St. Petersburg. The city authorities were given the right to redeem the horse-drawn tramways in 15 years. How can tram transportation be organized, if the authorities give no permission to other tram companies to lay rails in the streets of St. Petersburg?

A possible set of system component relationship models is:

1. Duma - concluded - Contract with Konka (harmful, unchangeable)
2. <u>Konka - owns - Contract for transportation (harmful)</u>

3. Tram company - lays - rail tracks (nonexecutable).
Selecting the 2-d model (Konka owns the contract) as a problem model means that only this contract is opposed. According to Table 2, it corresponds to the 4-th type of conflict, which recommends applying standards U1.2.1, U1.2.2., in other words, adding an element to Konka or introducing some supplements to the contract. Preferably, this element or supplement should be taken from the system resources, for example, from the modification of Konka itself.
Another recommendation is to introduce a new field. In other words, to use, alongside with St. Petersburg legislation, an arbitrary legislation or influence.
Standard U1.1 recommends creating an absolutely new Ele-field that contains neither Konka, nor the contract with this company.
At the same time, IFR can be automatically formulated. For example:
X-element BY ITSELF eliminates the need to fulfill the contract with Konka.
The formulation of requirement contradictions also becomes possible:
IF a tram company lays rails in St.Petersburg
THEN organizing tram traffic becomes possible
BUT it violates the contract with Konka.
If we address Table 3 below, we can find the sought-for recommendations in cell 2-3 (rails laid by Konka and the other tram company practically come into contact space-wise and overlap time-wise), namely, apply separation in space (lay rails within the city space not prohibited by the contract), system transition (individual rails are not located in the city, though collectively they are within the city), physico-chemical and phase transitions, change of relationship (tram rails are present relative to the city, but missing relative to the contract with Konka).
As is known, the Finnish Light Steamship Society laid a rail crossing over the frozen Neva in 1894 to provide tram transportation from the Senatskaya Square to the Vasilevsky Island and operated this crossing for two winters.

9. IFR and Requirement Contradiction Formulation

The list of function models offers options for formulating IFR in the course of problem statement. For example, two options of automatically synthesized IFR formulations can be offered for the above problem. One of them is: X-element BY ITSELF makes it unnecessary to perform the function "Dyes (changes color of) Swirls". The second one is: X-element BY ITSELF performs the function Dyes (changes color of) Swirls.
The IFR option selected by the user makes it possible to automate the formulation of options of requirement contradictions. For example, as regards the first IFR formulation: IF dyeing agent is used as X-element, THEN the function “X-element dyes the parachute mock-up" is performed, BUT the constraint “Mock-up shape must not be distorted” is violated.

10. Selection Peculiarities for Contradiction Resolving Principles

Transition from requirements contradiction (technical contradiction) to principles of their resolving is usually made with the help of Altshuller's Table [2]. However, the analysis of a set of function models makes it possible to avoid the use of this

Table, or refine the list of proposed principles based on the Table, or refine their priority. The following algorithm could be used for this purpose:

- Identify where the effects of useful and harmful function overlap in time and space by analyzing a set of functions.
- Identify from Table 3 the recommended principles for resolving those contradictions in the given situation.
- Compile a list of techniques that correspond to the selected contradiction resolving principles (such a list has been developed based on publication [9]); the list may include 40 major principles as well as additional ones [10].

  First of all, those techniques that simultaneously correspond to several contradiction resolving principles should be identified.
- Then, the list of recommended techniques is expanded using Altshuller's Table, in which case the techniques that coincided with the recommended ones prior to addressing Altshuller's Table are assigned higher ranks.
- If a problem refers to a non-engineering (non-material) system, then lines, columns and principles from Altshuller's Table that refer to engineering systems only (e.g., replacement of a mechanical scheme, thermal expansion, phase transitions, etc.) [11] are disregarded.

Table 3 Recommended Principles for Resolving Contradictions

Time of conflict and time of useful action / **Zone of useful action and zone of non-desirable effect**	**Don't overlap**	**Partly overlap**	**Fully coincide**
Don't overlap	• In time • In space (direction) • In relationship	• In space (direction) • In time • In relationship	• In space • System transition • Physico-chemical and phase transitions • In relationship
Coming into contact	• In time • In relationship	• In space (direction) • In time	• In space (direction) • System transition

	• System transition • Physico-chemical and phase transitions	• In relationship • Physico-chemical and phase transitions	• In relationship • Physico-chemical and phase transitions
Overlap	• In time • System transition • Physico-chemical and phase transitions • In relationship	• System transition • Physico-chemical and phase transitions • In relationship	• System transition • Physico-chemical and phase transitions • In relationship

For the example given above (the second option of functions set) two conflicting functions were identified: function 2.2. "Water dilutes (decreases thickness of) Dyeing agent (harmful)" and function 2.3. "Dyeing agent Dyes (changes composition of) Water (useful)". The operation time and operation zone of the harmful and useful actions coincide. This situation corresponds to cell 3-3 of Table 3 which recommends the following solving principles: system transition, physico-chemical and phase transitions, changes in relationships.

When comparing several dozens of techniques related to those three contradiction resolving principles, six techniques are found repeatedly: Fragmentation (1), Integration (5), Porous materials (31), Multistage action (42), Bi-principle (45), Dissociation-association (49). In other words, recommendations as to what techniques can be used for resolving contradictions can be offered even before addressing Altshuller's Table.

Different lines and columns can be selected from Altshuller's Table for the given problem. For example, one could select the following pair: line 8 (Volume of immobile object) and line 31 (Harmful factors of object proper). In this case the following four techniques are recommended: #30 (Use of flexible shells), #18 (Use of mechanical oscillations), #35 (Modification of object's physico-chemical parameters), and #4 (Asymmetry). If we take into account the recommendations based on the basic principles of resolving contradictions, only 2 out of 4 techniques will be left: #18 (Use of mechanical oscillations), and #35 (Modification of physico-chemical parameters of the object). And these recommendations should be regarded as final.

11. Compino-TRIZ Software Implementation

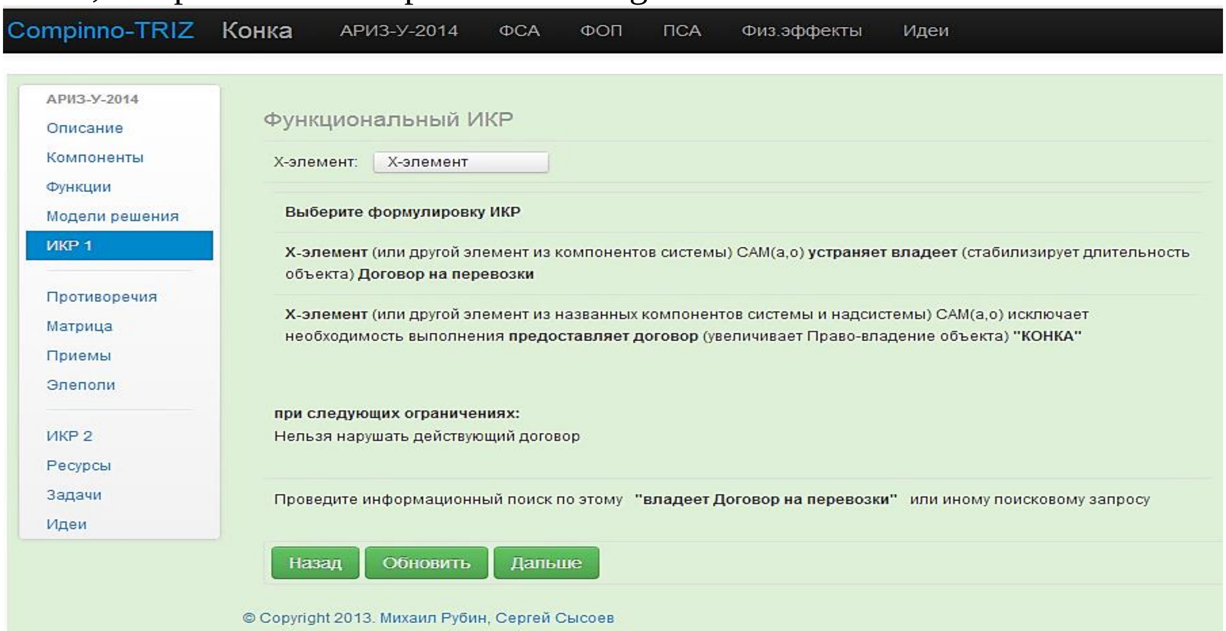
The Compino-TRIZ software package was developed on the basis of Python language and deployed in Google AppEngine infrastructure. Its user interface is implemented as a web application available with a restricted access at http://ariz-2010.appspot.com/.

Google AppEngine infrastructure provides:

1. High availability rates of application.

2. Automatic allocation of hardware resources in line with the current load on the application.

ARIZ-U 2014 and Compino-TRIZ have been used in practice since 2013 for inventive problem solving as well as for workshop training of students, teachers, scientists, and engineers. Experience shows that ARIZ-U-2014 is effective, easier to use, and provides faster problem solving.



Text in the screenshot:

<table>
<tr><td colspan="2">Compino-TRIZ Konka ARIZ-2014 FCA FOS CEA Physical Effects Ideas</td></tr>
<tr><td>ARIZ-U-2014
Description
Components
Functions
Solution models
IFR 1

Contradictions
Matrix
Techniques
Ele-fields

IFR 2
Resources
Problems
Ideas</td><td>Functional IFR
X-element X-element
Select IFR formulation
X-element (or another element of system component) by ITSELF eliminates holds (stabilizes duration of object) Contract for transportation
X-element (or another element of designated components of system or supersystem) by ITSELF eliminates the need to execute presents Contract (enhances ownership right of object) KONKA
under following restrictions:
valid contract must not be violated

Undertake an information search using "holds Contract for transportation" or another query

Back Refresh Next

© Copyright 2013 Michael Rubin, Sergey Sysoev</td></tr>
</table>

Further development of the Compino-TRIZ software package is supposed to follow such directions as:

- improving the existing algorithms and methods;
- developing and introducing new information assets and methods of system analysis;
- consideration of the specific area the problem belongs to, for example, information systems, software programming, etc.;
- diagnosis of user training level and adaptation of the software package to user's skills;
- introduction of training materials for mastering TRIZ tools;
- adaptation of Compino-TRIZ software package to the capabilities of users of different age: schoolchildren, students, and seniors;
- development of service functions and program design.

12. Conclusions

1. The functional approach to formulating problem and solution models made it possible to formalize the process of formulating contradictions, IFR, recommended standards of inventive problem solving, and other ARIZ steps.
2. The Compino-TRIZ software package developed on the basis of ARIZ-U-2014 allows formulating and solving inventive problems not only in the engineering sphere, but in non-engineering areas too. The Compino-TRIZ scope of use is continually expanding due to the fact that problem models are constructed based not only on function modeling, but also on generalized models of system component relationships.
3. The general trend of TRIZ tools development can be outlined as follows: further formalization and detailization of those tools to enable their software implementation, make easier their practical application for inventive problem solving and TRIZ training.
4. Automated formulation of ARIZ steps enhances their application efficiency in inventive problem solving, innovative design, and TRIZ training.

Bibliography:

1. Altshuller G.S., Shapiro R.B. Psikhologiya izobretatelskogo tvorchestva [Psychology of Inventive Creativity] – Voprosy psikhologii, 1956, No.6, pp.37-49

2. Altshuller G.S., Algoritm izobreteniya [Algorithm of Invention], M. "Moskovski rabochii', 1973

3. Altshuller G.S. ARIZ - znachit pobeda. Algoritm resheniya izobretatelskikh zadach ARIZ-85-V. Pravila igry bez pravil [ARIZ Means Victory. Algorithm of

Inventive Problem Solving ARIZ-85-V. Rules for Game without Rules]/Compiled by Selutsky A.B, - Petrozavodsk, Karelia, 1989. 280pp. (Tekhnika-molodezh-tvorchestvo), p. 11-50.

4. Algoritm resheniya izobretatelskikh zadach ARIZ-91 [Algorithm of Inventive Problem Solving ARIZ-91], St.Petersburg, 1997 http://www.triz-summit.ru/ru/205253/203840/204230/204231/.

5. Petrov V.M., Rubin M.S. Trebovaniya k razrabotke ARIZ novogo pokoleniya [Requirements to New Generation ARIZ Development]. TRIZ-Fest 2009 Scientific-Practical Conference: collected conference papers SPb, 2009. - 302 pp. (pp.159-165). http://www.triz-summit.ru/ru/section.php?docId=4201

6. Petrov V.M. Istoriya razvitiya algoritma resheniya izobretatelskikh zadach – ARIZ [Development History of Inventive Problem Solving Algorithm – ARIZ] 2-d edition, Tel-Aviv, 2008 http://www.triz-summit.ru/file.php/id/f3975/name/History%20of%20ARIZ-book.rar

7. Rubin M.S., Universalnaya sistema standartov na reshenie izobretatelskikh zadach [Universal Standards System for Inventive Problem Solving] - 2010. Version dd.18.02.2012 www.triz-summit.ru/ru/203798/205151/205266/205416/205422/

8. Rubin M.S., ARIZ-Universal-2010. Version dd.15.07.2012 http://www.triz-summit.ru/ru/203798/205151/205266/205416/205301/

9. Litvin S.S. Priemy razresheniya fizicheskikh protivorechii [Physical Contradictions Resolving Techniques] Leningrad, February 1981. – p.11; manuscript

10. Altshuller G.S., Dopolnitelnyi spisok priemov ustraneniya tekhnicheskikh protivorechii [Additional List of Techniques for Eliminating Tecnical Contradictions], Baku, 1973, http://www.altshuller.ru/triz/technique1a.asp

11. Rubin M.S. Osnovy TRIZ. Primenenie TRIZ v programmnykh i informatsionnykh sistemakh. [TRIZ Basics. Application of TRIZ to Software and Information Systems]: Training manual. – SPb: ATM Kniga, 2011

12. Rubin M.S. Parametricheskii TRIZ [Parametric TRIZ], 2009, www.temm.ru/ru/section.php?docId=4466

13. Rubin M.S. O razrabotke ARIZ-Universal-2014 [On ARIZ-Universal-2014 Development], TRIZ-Fest 2014, conference papers, Prague, 2014 http://triz-summit.ru/file.php/id/f300090-file-original.pdf