SHORT GRAPH-SCHEME OF A SUCCESSFUL SYSTEM IDEA

Nikolay Kosovskiy

Abstract*: Notions of a successful system and an ecologically acceptable successful system are proposed. These notions are convenient for formalization of a received inventive problem solution idea by means of TRIZ. This name was proposed by G.S. Altshuller [1] as abbreviation of the Theory of Inventive Problem Solution. Such a formalization is useful for training a creator of a successful system or an ecologically acceptable successful system for the intensification of his creative efforts.*

Keywords*: inventive graph-scheme idea,system, sub-system, ,graph, algorithmic heuristic, TRIZ.*

ACM Classification Keywords*: H.1.1 MODELS AND PRINCIPLES Systems and Information Theory – General systems theory; I.6.1 SIMULATION AND MODELING Simulation Theory – Systems theory*

Introduction

At our days a heuristic technology named in Russian TRIZ (Theory of Inventive Problem Solution) is widely spread. A mathematical model of an invention is introduced in this paper. Below the term “system” is used for a description of inventive scheme idea aimed to the solution of a practical problem. For a formal description of the result of a practical invention problem solution two notions are proposed: a successful system and an ecologically acceptable successful system [2].

Preliminary considerations

A successful system is a pair (invention scheme; numerical characteristic of this scheme outcome). Every successful system may be represented by a graph of relations between its elements names. Every element of a successful system may be also regarded as a successful system. The term “successful sub-system” will be used for such an element.

An ecologically acceptable successful system is a successful system with sufficiently precisely measurable parameters and their bounds which are ecologically admissible for the nature.

A set of a system elements for which a numerical result of its work as well as its productivity may be pointed out is called a successful sub-system.

A graph-scheme of system is a graph with nodes and edges marked by means of short names. A successful scheme of system requires also the presence of sufficiently precisely measurable system output.

It is useful to point out actions (named fields in TRIZ) upon some system elements (named objects). Such a graph-scheme of a system is called in TRIZ “OBACT” (as abbreviation of the words OBject and ACTion). In Russian the term “VEPOL” is used instead of OBACT. So, a graph-scheme of a successful system uses two sorts of elements: objects and actions.

Consider, for example, the **problem of saving a skier** who breaks thin ice not far from his fellows. The fellows are on sufficiently stable ice. The skier in the water may be regarded as an object. The pulling by a stick may be

considered as an action. Unfortunately such an action directly does not allow to pull the skier in wet clothes from the polynia. The edge of stable ice and the weight of the skier prevent this.

The problem may be solved by means of an additional object which is the skier who put his hand with a stick and HIS FOOT WITH A SKI on the stable ice. In such a case the foot with a ski allows the skier to get out on the stable ice.

Here an action directly upon the object (which does not solve the problem) is replaced by an action upon an additional object. So we replace one object by an additional one in OBACT.

While solving an inventive problem it is important to detect a key (the most important) conflict (contradiction). Criteria of primari importance of a necessity may be used.

It is needed to mark that the design of OBACT is a creative one. The main essence of a successful system must be represented in OBACT. Nevertheless the unnecessary detailing of some its sub-systems must be avoid. It is convenient to consider such a sub-system as an object or as an action.

OBACT may be regarded as a set of atomic formulas i.e. predicates of objects which are the ends of oriented edges beginning in an action. In such a case every predicate and every object is presented in OBACT only once.

Somebody can see in OBACT some similarity with a mathematical notion of category using objects and morphisms. Another ones can consider OBACT as an ontological approach to the knowledge representation.

Algorithmic heuristic

As a sub-system and an above-system is considered simultaneously with a system, so it is useful to consider a sub-object and an above-object simultaneously with an object as well as a sub-action and an above- action simultaneously with an action.

The edges of scheme of system graph may be oriented and non-oriented, soft and strong, initial and final, insufficient and excess, etc.

Comparison of two successful systems with each other may be done first of all on the base of the efficiency (success) of their outputs.

In the opposite to the system of mathematical problem solving offered by G. Polya [3, 4] and aimed to stimulation of appearance of ideas useful for the solving of a problem, TRIZ uses trends in the technical system development. Such a trend must be formulated as a low, for example, the low of system transformation to a micro-system.

The described here approach may be called an algorithmic heuristic. But such a name is rather conflicting. If an algorithm exists then there is no place for heuristic. If it is only heuristic then we can't say about an algorithm. More precise name for such an approach would be “an algorithmically organized set of heuristics on the different levels of successful system creating and, first of all, on the level of its idea description” .

Namely, I speak about some ways of human creative ability intensification aimed to the improvement of a successful system. The algorithmic heuristic helps to receive an answer for the question: ”How to guess hidden but sometimes evident solution of a problem?”. But such an “evidence” often becomes clear only after the solving of a problem. The decreasing of heuristic exhaustive search may be done, in particular, by means of heuristics structuring, set and making exhaustive search on the deeper level. I use also some modification of brain storming in student seminars.

Briefly the described algorithmic approach may be characterized as one oriented to the nonstandard solving of a practical problem for the improvement of successful system by means of exhaustive search of heuristics organized as a hierarchy structure. It may be represented as a heuristic technology.

More detailed exposition of educational technology for innovative solution of an inventive problem is presented in [2].

Solving of mathematical problems

The notion of successful system may be extended to a mathematical proof. Detailed algorithm as well as every other result received while solving a nonstandard mathematical problem beginning from the school level and up to the university one. The term “nonstandard mathematical problem” is used for such a problem which has no formal algorithm of its decision. For example, the main table from the book [3] has a two levels. The higher level contains only four useful questions:

- Are you ready to understand more exactly the problem statement?
- Are you ready to make a decision plan?
- Are you ready to implement the formulated decision plan?
- Are you ready to investigate the received solution?

Below the sequence of useful recommendations helping to answer the second question is formulated. This recommendations were formulated by me according to similar ones from different authors. The maximal degree of recommendation and the maximal degree of creative application of recommendation are very importante while using this sequence. The offered recommendations are the most useful for the solving of nonstandard problems. The sequence of such recommendations may be named an algorithmic heuristic one.

As a rule, the solving of every problem is not an isolated process but is united with the previous experience. In such a case the integrity of the solving process perception appears.

The first recommendation may be the following.

- Wait while the solution would come into your mind itself.

Only if you wait too long then you can go to the next recommendation.

The next step of decision making is a switch-over to an over-goal.

- Why I try to solve exactly this problem? May be it is better to read some other book or to go for a walk.

Then the following two alternatives are possible.

- To leave alone this problem and to find a new one.
- Has the problem in such a setting some solution? Is it probable that I would find a solution?

If nevertheless you are planning to solve the problem then begin the decision with the end.

- Present obviously what must you do.

Use the next recommendation until you feel that it can bring some new idea of a decision.

- Use the language of draughts, formulas, algorithms or programs which allows to reformulate the problem. What does become more clear after the reformulation?
- Simplify the problem.

Increase the volume of input.

Add an additional premise to the conditions.

Decrease the number of unknowns.

Weaken the conclusion.

Specialize the problem.

- Formulate an intermediate goal or problem. It is desirable that the intermediate one be sufficiently far from the data and unknowns of the main problem.

Actually such an intermediate problem is a “mathematical brick” named so by V.A. Ufnarovskiy [5].

- Find an almost solved similar problem, a solved problem with similar output or conclusion, a solved problem with similar input.
- Sort the information about the problem according to its importance and usefulness.
- Order the discovered difficulties according their significance.
- Introduce auxiliary elements or new dimension, pass to an over-problem. Generalize!

Sometimes persistence in going to the goal does not permit to see slight circumstances which are key ones in reaching the goal. Moreover, sometimes a key role of some circumstance is already in the subconsciousness. Hence, there appear the next recommendation.

- Fix the appeared thoughts. If you fill something like “This detail promises something”, then it is useful to write down a short denotation of this detail and its connection with the proper of investigation.

As a rule it is not needed to make big efforts in this direction just now. Further sharpening of the direction may appear itself.

- Formulate the problem in metafora language! Search for analogy! It may lead to discovering new facts.
- Make a list of the simplest particular and limiting cases. Order them according to their clarity and studying!
- Look for discovering of some often repeated fact or scheme.
- Control your observation by means of a luck thought or idea.
- Check your supposing by means of particular cases and facts that follow from it.
- Use the symmetry of the problem. For example, use the principle of sufficient basis proposed in the book [4].
- Make the problem more precise! Or make it less precise! (Change the degree of precision.) Change terms by their definitions!
- Reduce the problem to the one from another mathematical theory. Formulate it in the language of logic, algorithms, programs. In the language of variables and functions. Pick out a parameter and a proposition for mathematical induction implementation.
- Reduce the problem to itself. Use the recursion or the induction.
- Reduce the problem to the conjunctive normal form only with universal quantifiers before it. Remove surplus elementary disjunctions. Find locus for every elementary disjunction and then find an intersection of all these loci. In the other words decide the problem as a search one.
- Reduce the problem to the disjunctive normal form only with existence quantifiers before it. Remove surplus elementary conjunctions. In the other words, try to decide the problem as a proof one. Locate the main case (main elementary conjunction).
- Beginning with the main case try to receive the general solution with the help of superposition of special cases.

The decision of a search problem is provided by more formalized objects than a decision of an existence one.

- Introduce a common notation for a sub-problem which appears at least twice (may be with different parameters).
- Apply trial and error method of G. Polya [4] for searching among small number of formalized objects with easy enough checking if it is a solution.
- Check a consistency of some conditions. Of all conditions!
- Check an independence of every condition from the other problem conditions.

- Sort out and use your knowledge.
- Sort out and use your abilities.
- Adapt abilities to the deciding problem.
- If you have no idea of a decision then return to definitions (point 20).

Of course, there is a lot of problems which does not satisfy any of these recommendation. If you solve such a problem try to formulate a new recommendation.

- Include the idea of the solved problem into your experience (base of tools, base of knowledge, base of solution search).

What is the main sense of the problem?

What was the most important idea in the process of decision?

What was the main difficulty?

What could I do better?

This detail I have mist. What must be the peculiarity of an intellect permitting to see this detail?

Whether there is a method which I can use the next time in a similar situation?

What else approaches to this problem decision are there?

Bibliography

Altshuller G.S. Creativity as an exact science. M.: Soviet radio. 1979. 175 p. (In Russian)

Kosovsky N.K. Technology of innovation development. St.Petersburg: St.Petersburg state university. 2013. 186 p. (In Russian)

Polya G. How to solve it. Princeton University Press. ISBN 0-691-08097-6.

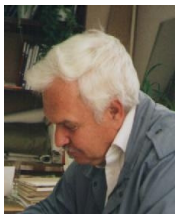
Polya G. Mathematical discovery.

Ufnarovsky V.A. Mathematical aquarium. Kishinev: Shtiintsa. 1987. 216 p. (In Russian)

Acknowledgement

The paper is published with financial support of the project ITHEA XXI of the Institute of Information Theories and Applications FOI ITHEA (www.ithea.org) and the Association of Developers and Users of Intelligent Systems ADUIS Ukraine (www.aduis.com.ua).

Authors' Information



Nikolay Kosovskiy – Dr., Professor, Head of Computer Science Chair of St.Petersburg State University, University av., 28, Stary Petergof, St.Petersburg, 198504, Russia,

e-mail: kosov@NK1022.spb.edu

Major Fields of Scientific Research: Mathematical Logic, Theory of Complexity of Algorithms.