# SOLUTION OF APPLIED PROBLEMS: FORMALIZATION, METHODOLOGY, JUSTIFICATION

KRASNOPROSHIN VICTOR

*Belarus State University, 4 Nezavisimosti Avenue, Minsk, 220050 Belarus, e-mail: krasnoproshin@ bsu.by*

OBRAZTSOV VLADIMIR

*Belarus State University, 4 Nezavisimosti Avenue, Minsk, 220050 Belarus, e-mail: obraztsov@bsu.by*

VISSIA HERMAN

*Byelex Multimedia Products BV Argon 1 – 4751 XC Oud Gastel, the Netherlands, e-mail: h.vissia@byelex.com*

Nowadays there is a tendency of the growth of applied problems for the solution of which formal methods and computers are used. Naturally, there arise problems of correspondence between theory and practice, the key one being the assessment of the quality of the obtained results.

The paper deals with questions of management of the applied problem solution. Three groups of problems have been considered. The problems concern the formalization, selection and construction of the model, method and algorithm of the solution as well as the justification of the obtained results.

Features for characterizing the class of applied problems have been determined. General principles of the transition from the meaningful description of a problem to its formal statement have been proposed. Main methodological aspects of the problem solution have been singled out. The result justification scheme, based on the inductive nature of problems, has been proposed. All the fundamental notions and results are exemplified by applied problems from pattern recognition and artificial intelligence.

## Introduction

With the development of the society, the role of theoretical knowledge and computer technology in solving applied problems is constantly rising. However, the theory and practice are developing in some sense independently of each other. Each of them has its own specific features and priorities. Using the theory for practice usually occurs on formal grounds, without posing questions of the justification, solvability, the intended aim, etc. All this has a negative impact on the final result when solving applied problems.

For a correct understanding of trends and directions in the development of any applied theory it is necessary to:

- define an applied problem and the essence of its solution;
- establish a correspondence between the theoretical and applied results;
- coordinate the development of the theory and practice, to define their dependence and mutual influence.

The solution of these problems would help to evaluate the relation between theory and practice in general, identify the challenges of specific theories, software and computer technology, the decision of which would satisfy the needs of practice.

Such problems have long been of interest to mathematicians. Briefly they are formulated quite simply [1-4]: can we justify the application of mathematics? It is known that with mathematics means to respond positively to this question is impossible even for the problems outlined in the mathematical formalization [1]. And what should we do when solving applied problems?

For such problems, according to authors, the global formulation of the justification problem is not quite legitimate. The problems just need to be solved, defining the direction of motion and the expected result [11]. The forefront in this case are issues of methodology.

Why is it important to know how to solve applied problems in particular? Because such problems form the majority, and the accumulation and standardization of the solution means result in a possibility to automate the solution in general. Finally, a correct understanding of the problem is a substantial and important part of the solution [11].

## 1. Definition of an applied problem

We define first the notion of an “arbitrary problem”. For this we consider the basic components that are commonly used in its formulation and do not depend on the subject area, informal meaning of information, etc. The components are shown in Fig. 1.

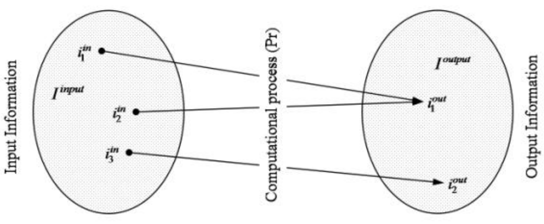


Fig. 1. The main components of a problem

The first element is the Cartesian product $I^{input} \times I^{output}$. The second element is the computational process $\Pr: I^{input} \to I^{output}$, which, generally speaking, can be unknown.

Thus, by a "problem" we mean a certain relation $T \subseteq I^{input} \times I^{output}$ for which at least two elements are defined $i_i^{in} \in I^{input}, i_k^{out} \in I^{output}$ ($i, k \in N$).

In other words, for stating an arbitrary problem $T$, it is necessary to explicitly specify elements $(i_i^{in}, i_k^{out}) \in T$, which define restrictions to the process $\Pr$. For example, in Fig. 1 the restrictions are represented as: $\Pr(i_1^{in}) = i_1^{out}, \Pr(i_2^{in}) = i_1^{out}, \Pr(i_3^{in}) = i_2^{out}$.

Note that not all "intuitively" understood problems are "problems" in the sense of $T$.

*Example* 1 [7]. In astrophysics, there is a problem related to the calculation of the diameter of the Universe. It is clear that the introduced definition of the problem $T$ does not work in this case, because it is impossible to find a pair $(i_i^{in}, i_k^{out})$, which would not cause doubts of the subject area specialists. Besides, despite the fact that the Cartesian product $I^{input} \times I^{outpu}$ is actually defined, computational mechanisms, forming the basis of the process $\Pr$, are unknown.

Let $T_0$ denote a set of explicitly defined elements $(i_i^{in}, i_k^{out}) \in T$. It is easy to see that $T_0 \subseteq T$. In this case a "problem" can be defined through the relation $T \subseteq I^{input} \times I^{output}$ for which there is the implication: $T \Rightarrow T_0$.

Depending on the method of forming the Cartesian product $I^{input} \times I^{output}$, the class of problems $T$ can be divided into two non-overlapping subsets [2, 4]. If the components $I^{input}, I^{output}$ are associated with any real objects, then the corresponding problem will be called an "applied" one ($T^{ap}$). Otherwise, it is a theoretical problem ($T^{th}$).

We can certainly say that $T = T^{ap} \cup T^{th}$, but semantic relationship between these classes, apparently, does not exist.

*Example* 2 [7]. The string theory [8] was proposed as a "unifying" physical theory in the 80s of the 20th century. For this theory $T^{th} \neq \varnothing$, but $T^{ap} = \varnothing$.

*Example* 3 [1]. Ptolemaic geocentric system of the world is an example of the theory, for which $T^{th} \neq \varnothing$ is also true, but $T^{ap} = \varnothing$. However, it should be noted that in solving applied problems of astronomy, the theory gave quite acceptable accuracy results.

As noted by several authors [1, 3, 4], applied problems are primary in relation to the theory, and therefore they are of particular interest. To establish a relationship between classes $T^{ap}$ and $T^{th}$, it is necessary to determine the characteristic properties of each of them. This will establish the limits of solvability and methodological features of the problems from the class $T^{ap}$.

## 2. Formalization

In general, the computational process for the problem $T \in T^{ap}$ is unknown. Therefore, a model of the process is used for its solution.

It is reasonable to provide two levels [4] i.e. an informal level, at which the problem is formed, and a formal one, intended for building the model. Relationship between the levels and their specific content are universal. In the general scheme of problem solution the relationship is shown in Fig. 2.

At the formal level, problems are formed from the set $T^{th}$. Here, $T^{th} \subseteq X \times Y$, and $p_2 : X \rightarrow Y$. In this case the condition $T^{th} \neq \varnothing$ is ensured by the presence of the problem on the informal level with some additional conditions of consistency on coding ($p_1$) and interpretation ($p_3$) mapping.

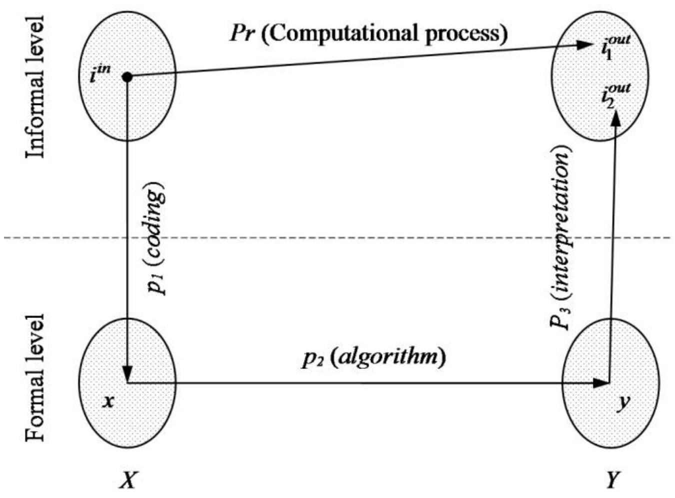


Fig. 2. Problem solution scheme

Really, this means that a formal level for a new problem can be regarded as an informal one. Thus, the general scheme of problem solution is not limited to the single (primary) scheme shown in Fig.1. In fact, it can be considered as a superposition of the primary schemes. In this case the recursion [9] can be used for constructing a formal model:

$$\begin{aligned} &h((x,y),0) \cong f(i^{in}, i^{out}), \\ &h((x,y),i+1) \cong g((x,y),i,h((x,y),i)), \end{aligned} \tag{1}$$

where the two-place function $f(i^{in}, i^{out})$ is the process $\Pr(i^{in}) = i^{out}$, and $i \in N$. In regard to the three-place function $g$, its arguments are defined by the following relations

$$h((x,y),i) \cong h_{\tilde{k}_i}((x,y),i),\ (x,y) \in X_{\tilde{k}_i} \times Y_{\tilde{k}_i},$$
$$\tilde{k}_i \cong \tilde{k}_{i-1} \times \tilde{N}, \tilde{k}_0 = \{(0)\}, \tilde{N} \subset N, |\tilde{N}| < \infty.$$

Besides,

$$\forall i > 0\, \exists p_1, p_2, p_3\ p_1 : X_{\tilde{k}_{i-1}} \to X_{\tilde{k}_i}, p_2 : X_{\tilde{k}_i} \to Y_{\tilde{k}_i}, p_3 : Y_{\tilde{k}_i} \to Y_{\tilde{k}_{i-1}},$$
$$h_{\tilde{k}_i}((x,y),i) \cong p_3 \circ p_2 \circ p_1(x).$$

In this scheme there is only a unique informal level, at which the set $T^{ap}$ is formed, and the final (in the limit - countable) set of formal levels, at which the corresponding sets $T^{th}$ are formed.

*Example* 4 [10]. Image recognition is a part of pattern recognition theory. This problem precisely belongs to $T^{ap}$ set. To construct suitable formal levels, within the solution of the problem, various ways of coding can be used, e.g. raster or vector conversions. As a result of formalization it is possible to state a problem of constructing correct algorithms. In this case, the general scheme consists of two formal and one informal levels. Moreover, the second formal level is associated with the algebra, and, in fact, unrelated to the original informal level.

Construction of a formal model with the use of the recursion (1), can be considered as a complete scheme of formalization necessary for solving any problem within $T$. Regardless of whether a problem belongs to the set $T^{ap}$ or $T^{th}$, it is necessary either to find a suitable scheme with the existing theoretical work, or somehow to build it.

For any scheme there is a problem, which is thematically close to the problem of the algorithm statement [9]: how many levels is it necessary to build, and in what sense can we speak of the original problem solution? In terms of mathematical formalization, the latter question is directly related to the problem of justification [1, 5, 6].

## 3. Justification

Let's consider the problem solution scheme from $T^{ap}$ in Fig. 2.

We can write

$$\begin{aligned} &\Pr(i^{in}) = i_1^{out} \in I^{output}, \\ &p_3 \circ p_2 \circ p_1(i^{in}) = i_2^{out} \in I^{output}. \end{aligned} \qquad (2)$$

Realization of the condition (2) is associated with the computability [9] of

the function (1). We call such a problem the algorithmically solvable one.
That is, we can state that for the problem there exists an algorithm that realizes the function (1) under all conditions associated with the implementation (1) and the fulfillment of the condition (2). But this is not enough, it is also necessary to establish a relationship between $i_1^{out}$ and $i_2^{out}$. Ideally, it could take the form

$$\forall i^{in} \in I^{input}\ ((\Pr(i^{in}) = p_3 \circ p_2 \circ p_1(i^{in})) \Leftrightarrow (i_1^{out} = i_2^{out})) . \qquad (3)$$

Condition (3) makes sense on the entire set of the set $T$ .

If we assume that we can formally prove the fulfillment of condition (3) on the set $T$ , then the resulting solution we'll call the "justified" one. For example, justified are solutions of propositional calculus problems, but the propositional calculus this property no longer has [1, 5, 6]. These are, in general, the problems that arise in the process of problem solution and construction of mathematical formalisms [1].

The condition (3) is stringent enough and for applied problems it is not appropriate [1, 4]. To be able to describe the nature of the solvability of problems with unjustified solutions we'll modify the condition.

Initially, we note that condition (3) is a consequence of one of the types of relationships, which can be introduced on the Cartesian product $I^{output} \times I^{output}$. Namely, the equivalence relation generated by the function of equality in this product. In general, any relationship $I^{out} \subseteq I^{output} \times I^{output}$ can be described by the function

$$\psi : I^{output} \times I^{output} \rightarrow R^{+} ,$$

where $R^{+}$ is a subset of non-negative real numbers. In this case $\psi$ realizes a variant of "similarity" of elements in $I^{output}$. This, for example, may be proximity ( $\psi(i^{out}, i^{out}) = 0$ ), similarity ( $\psi(i^{out}, i^{out}) = 1$ ) or some other variant.

Let's introduce a function of the type

$$\varphi : R^{+} \rightarrow [0,1] ,$$

and require that it be monotonic and satisfy the condition ( $r, r_1, r_2 \in R^{+}$ )

$$\varphi(r) = \begin{cases} 1, r = r_1, \\ 0, r = r_2. \end{cases} \qquad (4)$$

The selection of the function $\varphi$, corresponding to the condition (4), is determined by the nature of $\psi$ mapping. In case of proximity $r_1 = 0, r_2 = +\infty$ , and for similarity $r_1 = 1, r_2 = 0 \vee +\infty$ . Obviously with such a choice, the superposition $\varphi \circ \psi$ makes sense. The superposition is the basis for introducing the following function

$$\Phi(I^{input}) = \sum_{i^{in} \in I^{input}} \varphi(\psi(\Pr(i^{in}), p_3 \circ p_2 \circ p_1(i^{in}))) \cdot | I^{input} |^{-1}. \qquad (5)$$

It is easy to see that (3) is a special case of (5). With a suitable choice of $\varphi$ and $\psi$ mapping, the condition (3) can be written as: $\Phi(I^{input}) = 1$.

When calculating $\Phi(I^{input})$ there are only two possibilities

$$\begin{cases} \exists i^{in} \in I^{input} \forall \alpha \in [0,1] \ \Phi(i^{in}) \neq \alpha, & (6) \\ \forall i^{in} \in I^{input} \exists \alpha \in [0,1] \ \Phi(i^{in}) = \alpha. & (7) \end{cases}$$

Upon fulfillment of (6) $\Phi$ is noncomputable [9]. The reasons of noncomputability may be different. In the above formalization no restrictions are placed on the structure of sets $I^{input}$, $I^{output}$. And since they can be infinite, this can lead to a situation when resources are not sufficient to calculate $\Phi$.

The case (7) deals with the computability of $\Phi$ on the whole set $I^{input}$. And, specifically obtained number $\alpha \in [0,1]$ is in principle irrelevant.

*Example* 5. In the theory of pattern recognition known is the following result [12]:

- upon $\Phi(I^{input}) = 0$ the necessary condition $\Phi(I^{input}) = 1$ is obtained by a simple logical treatment of algorithms;
- for arbitrary $\Phi(I^{input}) = \alpha \in ]0,1[$ it is also possible to specify an algorithm, for which $\Phi(I^{input}) = 1$ will take place.

Of course, a generalization of the result, obtained in the theory of recognition, on the computability (5), requires a separate study. In the context of this work, we introduce two classes of problems.

As stated above, upon $\Phi(I^{input}) = 1$ the solution of problem $T$ is justified. It is known [1, 5], that the logical basis for this conclusion is the principle of deduction. And because the condition $\Phi(I^{input}) = 1$ is logically connected with computability at the whole interval $[0,1]$, then the principle of deduction can be extended to the whole class of problems for which there is the condition (7). Therefore, the relevant class of problems $T$ can be called "deductively solvable" (or "algorithmically deductively solvable").

For the class of problems $T$ with a noncomputable function $\Phi$, in accordance with the problem definition, it is still possible to specify a subset $T_0$ : $\Phi(I_0^{input}) = 1$ (where $I_0^{input}$ is the projection of the set $I^{input}$ onto the subset $T_0$ ). Otherwise, the problem does not exist.

Between conditions $\Phi(I^{input}) = 1$ and $\Phi(I_0^{input}) = 1$ there is an obvious link

$$\Phi(I^{input}) = 1 \Rightarrow \Phi(I_0^{input}) = 1.$$

However, the converse implication is interesting. Therefore, when trying to build something opposite to deductive solvability, it is appropriate to call this class of problems $T$ "inductively solvable". Although, unlike the first case, no single principle of induction exists.

Is it possible to build a reverse implication? It is very difficult to answer this question. But we can talk about the fundamental possibility of its realization. For example, in [12], such an implication is built for the so-called representative problem $T_0$. However, this is done only for the problems of pattern recognition. But this result is easily generalized to the case of the problem of recognizing the truth. Other generalizations or similar results are unknown.

We now turn to the characterization of the set of problems $T^{ap}$. It contains problems of two classes: inductively solvable and insolvable ones (for which there is (6)). For the latter class there is always a possibility of transition to the inductively solvable class. This possibility appears as a result of cognition. And problems in the class of inductively solvable can never pass*) to the class of deductively solvable.

In turn, a set of problems $T^{th}$, can include problems of all three classes: deductively solvable, inductively solvable and insolvable. Moreover, the fundamental difference between $T^{th}$ and $T^{ap}$ is that for $T^{th}$ and only for it, any problem can pass to a class of deductively solvable ones.

## 4. Methodological aspects

So, it was found that the characteristic features of a class of applied problems are:

- position in the solution scheme (only for such problems the informal level is directly connected with the reality);
- algorithmic inductive solvability/insolvability.

It is clear that the above features should influence the methods and way of organizing such a solution. These issues relate to the field of methodology [4], we consider them in more detail.

Let's make the corresponding particularization and specific filling of the solution shown in Fig. 2. At the informal level, there are several components that make up the solution process. Any problem is described by information, and obtaining a solution is associated with its processing. Therefore, the problem and the corresponding process have informational and computational components. Transformation of information is implemented in a certain environment, which naturally affects the informational and computational components. Therefore, we can single out another one, the so-called

*) For solvable problems the transition means that in the recursion (1) at some level $i$, a deductively solvable problem is obtained. This does not mean (in logical sense) that the original problem will be the same. For insolvable problems such a transition is informal.

"technological" component.

These components are naturally carried over to the level of formal constructions. The essence of management in this case is the solution of problems of representation and evaluation [6]. The problem of representation is typical for mathematical formalization. The concept of goal is associated with the two sides of the solution process: it is necessary to ensure the availability of a solution and to assess its quality from the standpoint of validity. These goals are also associated with the result and are characteristic of the process.

Finally, the implementation of any process is concerned with the recursion (1). At each step, a standard universal procedure is performed, the scheme of which is shown in Fig. 3. According to the results of evaluation, the problem can be corrected through its components. Schematically, such an adjustment is presented in the form of feedback. The process continues for as long as the achievement of local goals (at the level of an individual problem) is possible. Then, among all the problems the one is chosen for which the best results are obtained. Further development of the theory occurs in the direction of such a problem as long as possible, or simply feasible.

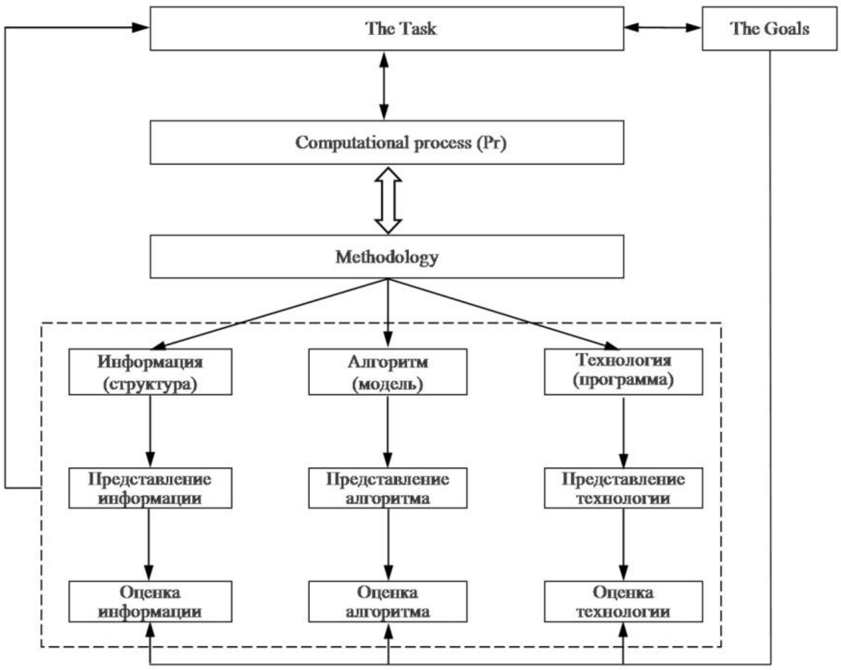


Fig. 3. Scheme of the general step of recursion

Заменить The Task на The Problem
Информация (структура) – Information (structure)
Алгоритм (модель) – Algorithm (model)
Технология (программа) – Technology (program)
Предствление информации - Information representation
Предствление алгоритма – Algorithm representation
Предствление технологии – Technology representation
Оценка информации – Information evaluation
Оценка алгоритма – Algorithm evaluation
Оценка технологии – Technology evaluation

The most general description of the methodology for problem solution is given above. It also applies to the inductively solvable problems from $T^{th}$.

## Conclusion

Returning to the questions set forth in the introduction, we can draw the following conclusions:

- the notion of a "practical problem" is largely philosophical. Each researcher can have his/her own understanding of it. However, its specific content should be associated with the organization and management of the solution process (in the sense of the correctness of the process). And any assessments of accuracy should lead to the desired result (in terms of the subject);
- theory and practice are developing in some sense independently of each other. In order to coordinate their development it is necessary to use principles of methodology, which are universal in nature. On the one hand they define a common set of problems that should be solved within any theory (and which establishes the relationship between $T^{th}$ and $T^{ap}$). As well as goals that may be associated with the solution of a particular problem. On the other hand they explain the essence of the result, which is obtained in the process of solution. Consideration of this complex of issues suggests some coordination of theory and practice;
- correlation between the results at the theoretical and practical levels is established by the relation between the solvability of the problem in an informal sense and validity of its solution within the theory.

The described approach to a practical problem does not pretend to any finality or completeness. Our aim is to show that investigating even such a complex issue, it is possible to set some base points, but in the ideal case - the border.

**References**


1. M.Kline. MATHEMATICS. The Loss of Certainty. – New York: Oxford University Press, 1980. – 447 p.
2. V.Turchin. - The Phenomenon of Science. - New York: Columbia University Press, 1977. - 368 p.
3. A.Foss. The Essence of Mathematics. - M.: Knizny Dom “Librokom”, 2009. – 120 p.
4. Blekhman I.I., Myshkis A.D., Panovko Ya.G. – Mechanics and Applied Mathematics. Logics and Peculiarities of Mathematics Application. - M.: Nauka, 1983. – 328 p.
5. V.Ya. Perminov. Philosophy and Mathematics Justification. – M.: Progress-Tradition, 2001. – 320 p.
6. Yu.A.Gastev. Homomorphisms and Models. Logico-algebraic Aspects of Modeling. - M.: Nauka, 1975. – 152 p.
7. Petrov Yu.A., Zakharov A.A. Practical Methodology. – Ozersk: OTI MIFI, 2001. - 107 p.
8. S.Hawking. The Universe in a Nutshell. – London: Bantam Press, 2001. - 222 p.
9. N.Cutland. Computability. An introduction to recursive function theory. – London: Cambridge University Press, 1980. – 256 p.
10. Yu. I. Zhuravlev. Selected Scientific Works. — M.: Maguister, 1998. — 420 p.
11. G.Polya How to Solve It: A New Aspect of Mathematical Method. – 2nd ed., New York: Princeton University Press, Princeton, 1957.
12. V.V.Krasnoproshin, V.A. Obraztsov Problem of Solvability and Choice of Algorithms for Decision Making by Precedence // Pattern Recognition and Image Analysis. 2006. Vol. 16. no 2.- p.p.155-169