Principles and solutions for integrating computer algebra tools and applications based on open semantic technologies

Valery Taranchuk
Computer Applications and Systems
Belarisian State University
Minsk, Belarus
taranchuk@bsu.by, 0000-0003-2210-652X

Daniil Shunkevich
Intelligent Information Technologies
Belarusian State University of Informatics and Radioelectronics
Minsk, Belarus
shunkevich@bsuir.by, 0000-0003-4223-9389

***Abstract*—The paper discusses the principles of integration of computer algebra systems and the Ecosystem of Intelligent Computer Systems of a new generation (OSTIS Ecosystem). The feasibility of such integration, various options for such integration, the advantages and disadvantages of these options are shown. The implementation of the proposed approach is considered using the example of an intelligent learning system in discrete mathematics.**

***Index Terms*—OSTIS, ontology, knowledge base, OSTIS Ecosystem, OSTIS Technology, Wolfram Mathematica**

I. Introduction

Since the middle of the 20th century, the fundamental scientific direction ***computer algebra*** has been intensively developing – the science of effective algorithms for calculating mathematical objects. Synonyms for the term *"computer algebra"* are: *"symbolic calculations"*, *"analytic calculations"*, *"analytic transformations"*, and sometimes *"formal calculations"*. The direction "computer algebra" is represented by theory, technology, and software. Applied results include developed algorithms and software for solving problems using a computer, in which the initial data and results are in the form of mathematical expressions and formulas. The main product of computer algebra has become software *computer algebra systems* – ***CAS***. The range of mathematical problems that can be solved using *CAS* is constantly expanding.

At the current stage of development of information technology, the transition from modern computer systems to computer systems of a new generation, which, obviously, must have a fairly high level of ***intelligence***, is relevant. This, in turn, means a transition to a fundamentally new technological structure in the field of automation of various types of human activity and involves rethinking and using all the experience accumulated in the development and operation of various *computer systems*, and also makes it necessary to develop a complex *Technology for supporting the life cycle of new generation intelligent computer systems* (***OSTIS Technology*** [1]). Systems developed on the basis of this technology are called *ostis-systems* [1]. It is important that participation in the project to create such technology does not imply a radical change in the field of scientific interests of the relevant specialists. You just need to take into account additional requirements for formalizing your results. The main among these requirements is ***semantic compatibility*** with other (related) [1] results.

With the development of technology, the first place in the development and modernization of *intelligent computer systems* should be solutions, implementation of the integration of such systems and tools of *computer algebra systems*. It is important to transform the modern variety of tools (frameworks) for developing various components of intelligent computer systems into a single technology for integrated design and support of the full life cycle of these systems, guaranteeing the compatibility of all developed components, as well as the compatibility of the intelligent computer systems themselves as independent subjects interacting with each other within the framework of complex automation systems for complex types of collective human activity. There is a need for *convergence* and *unification* of new generation intelligent computer systems and their components. At the same time, convergent solutions generally mean optimized complexes that contain everything necessary to solve Artificial Intelligence problems, organized or configured for the efficient use of information resources, to simplify implementation processes, including the ability to solve specific problems with optimization requirements and achieve maximum performance, and in all implementations – optimized for ease of use. The above fully applies to the *OSTIS Ecosystem* concept [1].

Similar problems are also solved when developing, improving, systematically updating the content and expanding the capabilities of computer algebra systems. Thus, the integration of *CAS* and *OSTIS Ecosystem* is an important and urgent task.

II. Approaches to integrating computer algebra systems with applications within the OSTIS Ecosystem

One of the options for interaction between *OSTIS Ecosystem* and *CAS* may be approaches similar to those implemented within the framework of integration of artificial neural networks into ostis-systems [2] – [4]. The following methodological and technical solutions can be considered:

- Integration according to the "black box" principle, when in the *knowledge base ostis-system* there is a specification of the used function of the kernel of the computer algebra system, as well as a specification of the method of calling this function (for example, an indication through which program interface interacts with this external system). This integration option is the simplest to implement and generally has the advantages listed above. At the same time, this option also has a disadvantage due to the fact that the *ostis-system* does not contain tools for analyzing and explaining how a specific step of solving the problem, implemented by the used *CAS* function, was performed;
- Closer integration, in which a specific function still remains part of a third-party *CAS*, when not just the result of its execution, but also all its possible specifications are immersed in the *knowledge base* of the *ostis-system*, for example, an explanation of the progress of solving a problem, an indication of specific algorithms and formulas that may be involved in the solution, a description of possible alternative solutions, an assessment of the effectiveness of the solution, and so on. In this integration option, the *ostis-system* gets more opportunities to analyze and explain the progress of solving the problem. (It should be noted that specifically in *CAS Wolfram Mathematica* there are already detailed explanations of the progress of solving the problem and a step-by-step execution mode is acceptable);
- Full integration, in which the used functions of a computer algebra system are translated from the internal language of this system into the *ostis-system*. This option is the most labor-intensive and complex from the point of view of updating the implementation of the capabilities of computer algebra systems in the corresponding *ostis-systems*, taking into account their constant development. At the same time, this integration option, compared to the previous two, has an important advantage – it ensures the platform independence of the resulting solution and allows you to use all the advantages of the approaches offered within the framework of *OSTIS Technology* when solving a specific problem, in particular, the possibility of multi-user, parallel processing of knowledge and the ability to optimize the plan for solving a problem or its fragments directly during the solution.

The feasibility of using one or another integration option is determined primarily by the difference in their labor intensity, which, in turn, makes it advisable to implement these integration options in stages as the project develops. Thus, at the stage of testing the idea of integration, it is advisable to use the "black box" option as the least labor-intensive, but at the same time providing the opportunity to demonstrate certain advantages of such integration.

III. Integration of computer algebra systems and intelligent learning ostis systems

From a practical standpoint, at this stage of development and application of the *OSTIS Ecosystem*, it seems appropriate to integrate into a single set of capabilities *computer algebra systems* and intelligent teaching systems [5] built within the framework of the *OSTIS Ecosystem*. The corresponding implementations are provided by the content of *CAS*, which have an undoubted advantage and ample capabilities in solving problems that are relevant for educational systems in almost all natural science and technical disciplines that involve the use of complex mathematical apparatus.

On the other hand, despite the popularity of topics related to the automation and intellectualizing of educational activities in the natural sciences and the development of corresponding computer systems, at the moment there are practically no tested intelligent teaching systems in the public domain that implement the ability to independently generate and solve various problems, as well as check the correctness of the user's solution to the problem. As prototypes, we can cite individual systems in which non-trivial problems are solved, for example, in geometry [6], [7] and graph theory [8]. But it should be noted that in the mentioned systems there is no "intelligence" as such (in fact, only a specific set of actions is laid down; problems are not generated in the applications themselves), there are no means of checking solutions that have even minor deviations from the design rules.

An approach to solving problems of intellectualizing of educational activities, based on the integration of ostis-systems and computer algebra systems, has a number of advantages:

- When developing *ostis-systems*, the need to program many functions that have already been implemented, tested and approved in *CAS* is eliminated. This is important, since computer algebra systems are developed by highly qualified specialists in the relevant fields, the implementation of similar functions in ostis-systems may require significant financial and time costs.
- A specific *ostis-system* using individual *CAS* functions, thanks to the approach to the development of hybrid *problem solvers* in *OSTIS Technologies* [9] gets the opportunity independently plan the progress of solving problems, provided that some of its stages will be implemented using attached functions. From the point of view of the approach proposed within the framework of *OSTIS Technology*, each function of a computer algebra system becomes a *method* for solving problems of a certain class. This class of problems is described in the *knowledge base* of the *ostis-system* and allows it, when solving a specific problem, to independently draw a conclusion about the advisability of using a particular function of *CAS*. Such integration with *ostis-systems* will eliminate the previously formulated possible drawback of *computer algebra systems*.

We especially note that the indicated integration options are not mutually exclusive and can be combined. In addition, deepening integration can be carried out step by step, taking into account the listed advantages and disadvantages, as well as taking into account the relevance of using certain functions of computer algebra systems when solving specific problems within the framework of the OSTIS Ecosystem and the corresponding ostis-systems.

Step-by-step integration of *CAS* with *OSTIS Ecosystem* involves, at a minimum, a description of the specification of the main functions of the selected computer algebra system using *OSTIS Technologies*, in other words, the development of an ontology of external functions. In the case of systems of the *Wolfram Mathematica* family, the process of developing such an ontology can be automated due to the presence of the formal *Wolfram Language* and good documentation of the system functions.

Summarizing the above, we state that the integration of learning systems developed on the basis of *OSTIS Technology* and computer algebra systems will make it possible to create systems with a high level of intelligence in a shorter time, and using carefully (mathematically, algorithmically) developed and repeatedly tested instruments.

IV. An example of integration of a prototype OSTIS learning system for discrete mathematics and Wolfram Mathematica

The basic capabilities that ensure the achievement of the goals formulated above and the identified main technical solutions are outlined in [1]. Fundamental positions and capabilities are especially noted, in particular: working with mathematical expressions in symbolic form (with execution in analytical form, presenting results in mathematical notation), numerical operations of any specified accuracy, data types, interactive graphical visualization, presentation of results and preparation for publication, use of special-purpose extension packages, programming in an embedded language, program synthesis.

Let us give illustrations of the joint use when working with graphs of the prototype of the learning *ostis-system* in discrete mathematics, which is part of the *OSTIS Ecosystem*, and *Wolfram Mathematica*. The results noted below show the possibilities of using calculation and visualization results performed in *Wolfram Mathematica* in the ostis-system. Moreover, implementations are available using the appropriate software interface (you can execute code hosted in the Wolfram cloud in the Wolfram Language within a user program, for example, in *Python* or *C++* [10]) or import and export tools.

In the example below, the initial data (a specific graph) is received (imported) from the *learning ostis-system for discrete mathematics*, visualized using graphics *Wolfram Mathematica*, then a typical problem is solved and the preferred final results are exported back to educational *ostis-system* in discrete mathematics.

The following illustrations were obtained in *Wolfram Mathematica*, while the original sc.g-text from the ostis-system was used to define the graph.

For an imported graph in *Wolfram Mathematica*, you can obtain and display general information, for example: the number of vertices, arcs, list of edges; can be visualized in different ways. For example, 3 output options are shown below. Fig. *1* shows connections with directions (output without formatting, with default settings):

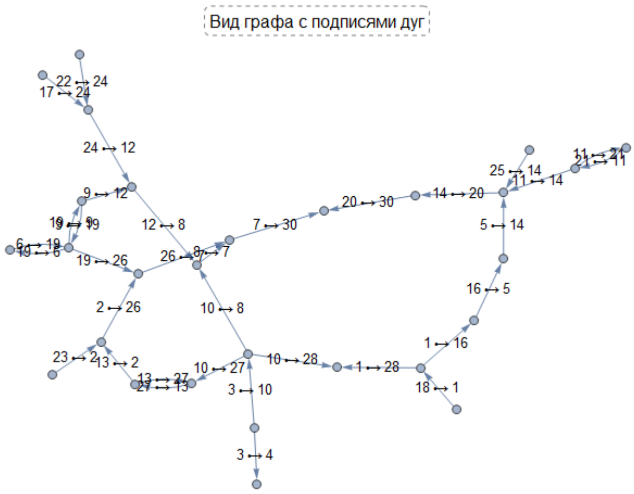

Figure 1. Graph used, arcs (output in Wolfram Mathematica)

Fig. *2* shows implemented output indicating vertex styles and their numbers, it is specified that all arcs from a node with a higher number to a node with a lower number are displayed as dotted red lines, and the rest as solid green:

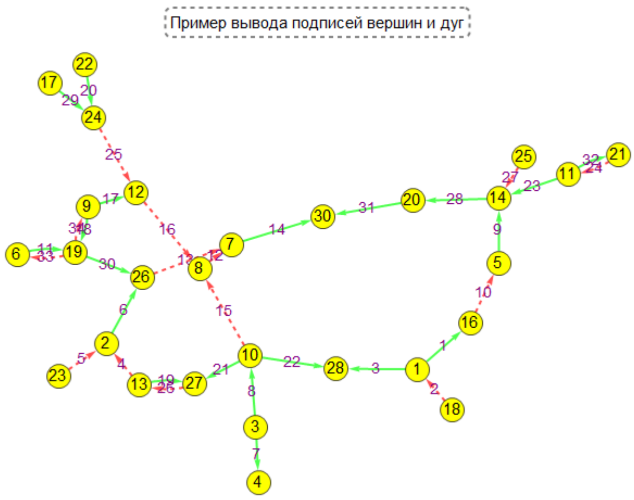

Figure 2. Used graph with rules design (output in Wolfram Mathematica)

An example of output with setting the format for laying out the DiscreteSpiralEmbedding graph and using vertex design options is shown on Fig. *3*

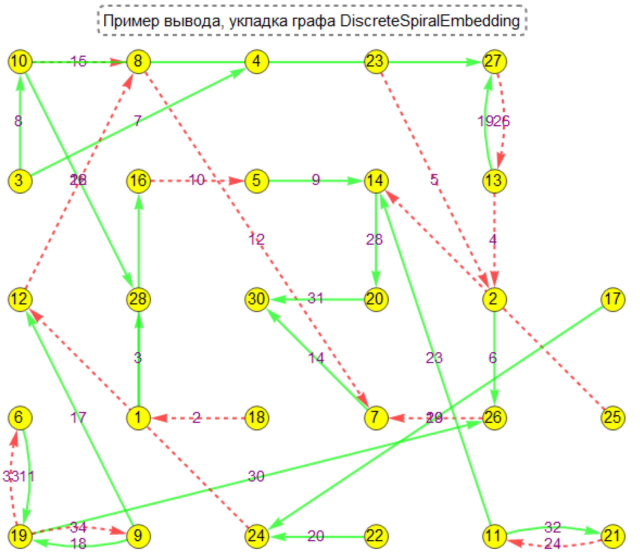

Figure 3. Used graph with DiscreteSpiralEmbedding (output in Wolfram Mathematica)

An example of solving the problem of finding the shortest path between two vertices is illustrated in Fig. *4* – *5*. The following functions were used in the solution: *Wolfram Mathematica*: GraphDistance, NeighborhoodGraph, Sow, DirectedEdge, Placed, Union, Flatten. The output is done using SpringEmbedding layout.

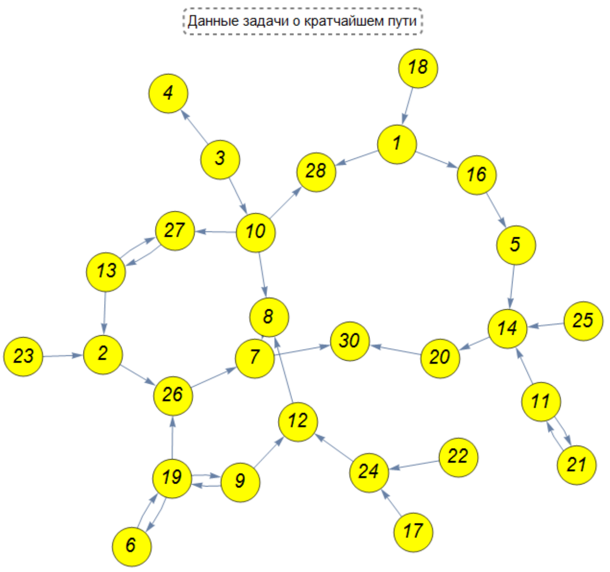

Figure 4. Solving the problem of finding the shortest path between two vertices (output in Wolfram Mathematica)

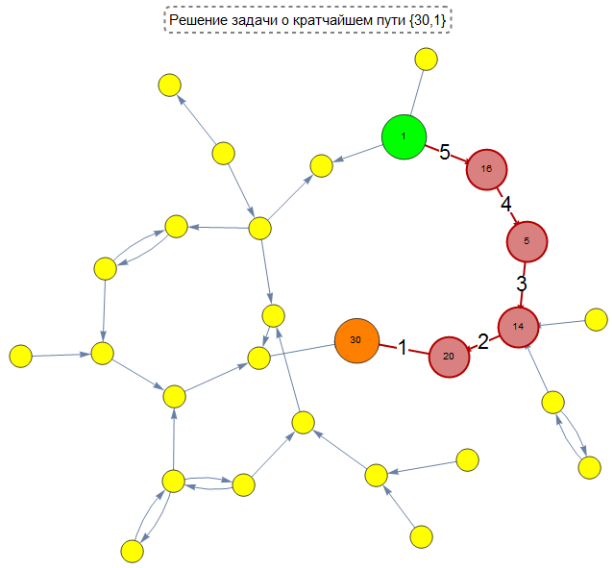

Figure 5. Solving the problem of finding the shortest path between two vertices (output in Wolfram Mathematica)

Fig. *6* – *7* shows an example of solving the problem of finding a minimum spanning tree.

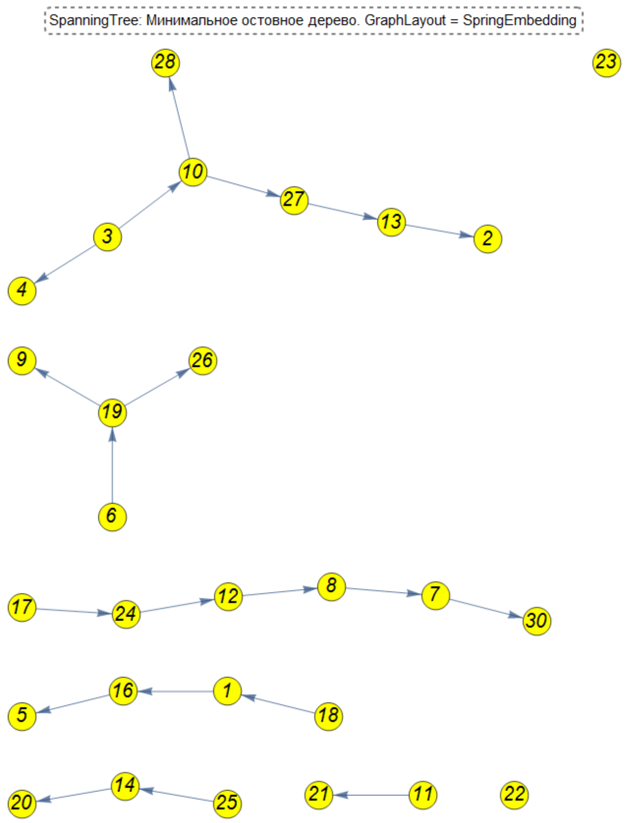

Figure 6. Solving the minimum spanning tree problem (output in Wolfram Mathematica, part1)

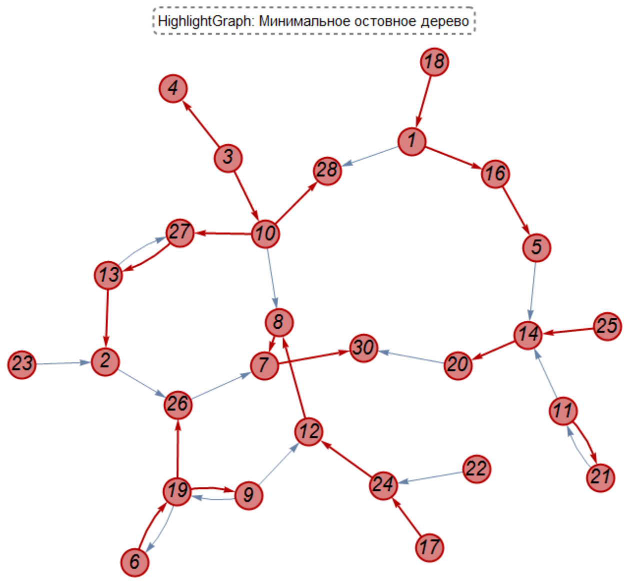

Figure 7. Solving the minimum spanning tree problem (output in Wolfram Mathematica, part2)

It should be noted that the results obtained and discussed include graphics tasks that are labor-intensive to implement in programming languages, as well as mathematically and algorithmically complex domain problems. The presented options for visualization and finding a solution require only a careful study of examples of the *Wolfram Mathematica* help system and initial programming skills, that is, they are accessible to most software engineers. Transferring results to other software applications is also not difficult, because *Wolfram Mathematica* provides export capabilities to any standard formats.

V. Conclusion

Computer algebra systems currently represent powerful instrumental complexes, the capabilities of which have long gone beyond the scope of algebraic calculations and even classical mathematics in general. *CAS* provide many computational capabilities, processing algorithms, analysis, and visualization. One of the leaders is the *Wolfram Mathematica* system, the core of which contains more than 6000 functions. The *Wolfram* company has also developed many unique projects, which, in addition to the *Wolfram Mathematica* system, include the *Wolfram Alpha* [11] question-and-answer system, which contains an extensive knowledge base and a set of computational algorithms.

The basis for the representation of factual, logical and procedural knowledge for systems of the *Wolfram* family is the multi-paradigm programming language *Wolfram Language*. The presence of such an internal language for describing the functions of *Wolfram* systems and, in general, a high level of documentation of these functions, distinguishes *Wolfram* systems from other services that allow solving general and specific problems. In many cases, *Wolfram* can not only solve a problem, but also explain the progress of the solution, and also help the user in choosing a particular function suitable for solving his problem, or offer a set of functions that can be applied to data obtained as a result of solving the original problem. Another advantage of the *Wolfram* family of systems is their complexity, which allows solving quite complex problems within a single application and without the need to integrate heterogeneous services.

Considering the above, we can conclude that it is advisable to integrate the computer algebra system *Wolfram Mathematica* with *ostis-systems* that are part of the *OSTIS Ecosystem*. This paper examines the principles and possible implementations of such integration and provides relevant examples.

Acknowledgment

The authors express gratitude for their assistance in the work and valuable comments to colleagues of the Department of Computer Technologies and Systems of the Belarusian State University and the Department of Intelligent Information Technologies of the Belarusian State University of Informatics and Radioelectronics.

References

[1] "Technology for comprehensive support of the life cycle of semantically compatible computer intelligent systems of the new generation" [Tehnologija kompleksnoj podderzhki zhiznennogo cikla semanticheski sovmestimyh intellektual'nyh komp'juternyh sistem novogo pokolenija], V. Golenkov, Ed. Minsk: Bestprint, 2023.

[2] M. Kovalev, A. Kroshchanka, V. Golovko, "Convergence and integration of artificial neural networks with knowledge bases in next-generation intelligent computer systems", Open semantic technologies for intelligent systems, no. 6, pp. 173–186, 2022.

[3] V. Golenkov, V. Taranchuk, M. Kovalev, "Problems and prospects of automating varios types and fields of humen activity with the help of next-generation intelligent computer systems", Open semantic technologies for intelligent systems, no. 6, pp. 327–346, 2022.

[4] V. Taranchuk, "Integration of computer algebra tools into OSTIS applications", Open semantic technologies for intelligent systems, no. 6, pp. 369–374, 2022.

[5] N. Gulyakina, A. Kazlova, "Semantically compatible ostis educational automative systems", Open semantic technologies for intelligent systems, no. 6, pp. 375–380, 2022.

[6] "GeoS", mode of access: https://geometry.allenai.org/. — Date of access: 18.09.2023.

[7] "Mathpix", mode of access: https://mathpix.com/handwriting-recognition. — Date of access: 18.09.2023.

[8] "Graph Online", mode of access: https://graphonline.ru/ — Date of access: 18.09.2023.

[9] D. Shunkevich, "Hybrid problem solvers of intelligent computer systems of a new generation", Open semantic technologies for intelligent systems, no. 6, pp. 119–144, 2022.

[10] "Wolfram API", mode of access: https://reference.wolfram.com/language/ref/APIFunction.html — Date of access: 18.09.2023.

[11] "Wolfram Alpha", mode of access: http://www.wolframalpha.com/ — Date of access: 18.09.2023.