Mathematical Models and Methods of Supporting the Solution of the Geometry Tasks in Systems of Computer Mathematics for Educational Purposes

Michael Lvov[0000-0002-0876-9928], Ludmila Shishko[0000-0001-9346-4338], Irina Chernenko[0000-0003-2445-7366], Evgen Kozlovsky[0000-0001-6630-083X]

Kherson State University, 27, University St., 73000 Kherson, Ukraine
Lvov@ksu.ks.ua, Shishko@ksu.ks.ua, Chernenko@ksu.ks.ua, Evgen@ksu.ks.ua

Abstract. The article is devoted to the problem of supporting the course of solving tasks in geometry in systems of computer mathematics of educational purposes. In the work: - the mathematical model of the learning geometric task is defined; - the object-oriented approach to the description of mathematical models of geometric training modules is presented; - the methods of supporting step-by-step solving of learning geometric task are proposed; - the classification of elementary transformations in geometric subject modules is proposed; - the implementation of the concept of support for the solution of geometric tasks in the systems of computer mathematics of educational purposes is illustrated. Object-oriented analysis of the problem revealed three major classes of transformations of geometric objects. These are constructors, selectors, and converters (elementary geometric tasks).

Keywords: Systems of computer mathematics for educational purposes, learning geometric task, computer software, support of learning processes.

1 Introduction

The quality of mastering mathematical knowledge largely depends on the student's practical mathematical activity. This is the main form of educational activity in the study of disciplines based on mathematical models and methods, and is to solve learning mathematical tasks.

Review most domestic and foreign software for educational purposes in mathematics (GRAN, DG, Geometer's Sketchpad etc.) reveals the lecture part of the course is the maximally advanced one from both methodical and technical points of view [16, 17]. But practical functionality of these software tools is limited. Educational purpose practical mathematical activity is to construct the course of solving the learning mathematical task, but not to receive an answer [9].

The function of supporting the process of solving the learning mathematical task (*LMT*) is realized in the concept of systems of computer mathematics for educational purposes (SCMEP).

SCMEP is a programmed educational system for exact and natural educational disciplines that uses mathematical models and methods of subject areas based on technologies of symbolic transformations and methods of computer algebra.

The general theoretical and methodological foundations, the formulation of functional requirements for SCMEP and the development of a model of SCMEP as a system for supporting learning processes based on the analysis of actual forms and peculiarities of learning processes in precise disciplines are described in [3-5].

2 The outline of the problem

The implementation of the tasks of supporting the solution of *LMT* requires the definition of a mathematical model of *LMT* in the framework of a mathematical model of the training module and the construction of appropriate algorithms of computer algebra.

The SD curriculum is defined by the quadrants $SD = <\Sigma, MM, ET, Task>$, where Σ - own signature of the training module, *MM* - list of models of the training module, *ET* - list of own elementary transformations of the training module, *Task* - a class of learning tasks, which defines the content of the *SD*.

The educational task *P* is determined by the list of mathematical models of the training module *MM*, the relation of the dependence φ between the models and their elements (the condition of the task) and the questions of the task *Q*:

$$P = <MM, \varphi, Q>. \tag{1}$$

The scope of application of models of the module is mathematical discipline. Each of the mathematical disciplines has its own class of *LMT*. Of course, mathematical models of *LMT*, depending on the mathematical discipline, have their own peculiarities.

Mathematical models and methods for solving algebraic tasks in computer science mathematical systems are described in [6, 15]:

- functional requirements for activity environments supporting the solving of educational tasks in algebra are developed;
- definition of the concept of a training module in school algebra (signature, list of mathematical models, list of own elementary transformations);
- definition of the concept of *LMT* in algebra, algebraic object, types of *LMT* in algebra;
- the main specific tasks of supporting the step-by-step solution of the National Academy of Sciences are described.

The problem of this study can be formulated as a study of the specifici6y of the construction of mathematical models of learning geometric task, formal tasks supporting the process of solving the learning geometric task in the SCMEP and its implementation in the SCMEP.

3 Results

Model of educational geometric task. Under the learning geometric task (*LGT*), unlike school algebraic tasks, we understand the task that is formulated in terms of geometric objects, which is the subject of study and is supported by SCMEP. Learning geometric task as well as *LMT* in algebra are determined and solved by analytical methods. The peculiarity of *LGT* support is that the geometric object and the elementary transformations of the *LGT* can be interpreted geometrically, and therefore should be reflected in the corresponding graphical interpretation.

In order to implement the support of the solution of the learning geometric task, it is expedient to introduce the notion of a mathematical object. A mathematical object is: an algebraic object (*AO*) and a geometric object (*GO*).

Algebraic objects are numbers, variables, numerical and symbolic expressions, determinants, matrices, equality, inequalities, systems, sets of equalities or inequalities.

Primitive geometric objects (*PGO*) is the point of the plane and space, lines, curves 2-order curves in polar coordinates, surface 2nd order.

Primitive geometric object (*PGO*) is defined identifier (*ID*) and algebraic object for this syntax: *PGO*:: = <*ID*> (<*AO*>).

Algebraic objects that determine the *PGO* are equations, inequalities, systems of equations or inequalities.

General definitions of *PGO*, except for variables, include alphabetic designations of *AO* coefficients - its general parameters. The general parameters of the *PGO* are the alphanumeric coefficients of the algebraic object that determines it.

Mathematical models of geometric modules. The structure of the geometric learning module is defined in the framework of the object design paradigm. Each *PGO* is an instance of the class.

For analytic geometry on a plane, these are the classes *Point, Line, Curve2, Circle, Ellipse, Parabola, Hyperbola*. The Curve2 class defines the *PGO* curve of the 2nd order. The *PGO* Hierarchy of Inheritance allows to distinguish general and specific signatures, models, elementary transformations and standard learning tasks.

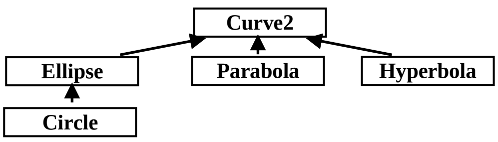

Fig. 1. Fragment of the tree of the classes of the module "Curves of the 2nd order"

Definition of the *PGO* class contains, in particular, the *AO*, which determine it.

```
Class PrimitiveGObject(
        CoordinateSpace Varset;            // (x, y);
        Variable ID;                       // l
        AlgObject F(x,y);
        . . .
        Virtual CartesianSpace Draw();
        . . .
        );
```

Here is an example of a straight line class definition.

```
Class Line :: PrimitiveAnalGeomObject (
    Canonicalforms (
        Genequ a * x + b * y = c,          // general equation
        Canequ y = k * x + b,              // canonical equation
        Segmequ x / a + y / b = 1,         // the equation in the segments
        ); . . .
        );
```

Thus, the classes determine, in particular, various algebraic objects (general canonical forms), which, in turn, determine the geometric objects of the class.

Definition of the class of *PGO* allows you to list the various forms of algebraic representation of the *GO*, to indicate the parameters in letters, thereby defining the functions of access to the parameters, as well as their geometrical predictions. In addition, the class defines the specifications of the functions of algebraic transformations *PGO*.

A composite geometric object (*CGO*) is determined by the name, a set of *PGO* and the relationships that determine them. *CGO* are, in particular, punctures, directed segments, angles formed by rays, triangles formed by point-vertices, etc. The ray is determined through straight line, point and inequality. Directional segment is an ordered pair of points. The angle is a pair of beams with a common point. A triangle is a triple point.

From the point of view of the object-oriented programming paradigm [10], the *CGO* is defined by the aggregation classes and, possibly, the relationships between them and their parameters. Example:

Class Segment = (Point A, Point B);

Class SemiLine = (Line l, Point A)((A in l)&(x >= x_A)).

Parameters of a composite object are marked by qualified identifiers. For example, if D is a triangle, the coordinate x of its vertex A has the $x_{D.A}$ identifier.

Classes of the *CGO* contain the definition of class member functions (transformations) that characterize the corresponding *CGO*. For example, in the Segment class, you should define the segment's characteristic as its length.

Length(A,B) = Sqrt(Sqr($x_B - x_A$) + Sqr($y_B - y_A$))

Apart from the classes of primitive and complex objects, the domain analytic geometry also defines elementary transformations - operations on objects. Example,

$$LineAB(A(x_A, y_A), B(x_B, y_B), l(\frac{y - y_A}{y_B - y_A} = \frac{x - x_A}{x_B - x_A}))$$

is a transformation that defines a direct plane passing through two points.

The result of an elementary transformation may be several objects. So, the intersection of a circle and a straight line determines either two points, or one point, or none.

The names of the *GO*, whose mathematical models are defined, are used in the *CGO* constructors as variables whose values are the corresponding mathematical models.

Each specific *LGT* can be formulated in terms of model, condition and question in the form (1):

Given: list of geometric objects; list of relations between them.
Find: a list of objects (geometric, algebraic, logical).

The *LGT* model is a *GO* or a set of *GO* and *AO*. The task *ID* is the service word Task with the task number in parentheses. Example:

Task 1. The distance between the points A (-2; 5) and M (x; y) is equal to three units of scale. Determine the coordinates of the point M if A and M are located on a straight line parallel to the abscissa.

Given: Points *A*, *M*, with $|AM| = 3$, $x_A = x_M$.

Find: Point *M*.

Hence the formal definition *LGT*: *Task*(1) = (*MM*, *φ*, *Q*), where *MM* = (*A* (-2; 5) & *M* (*x*; *y*)) – mathematical model, $\varphi = (|AM| = 3)\&(y_A = y_M)$ - condition, *Q* = *M* - question (denoted as ?*M*).

To solve this task, it is necessary to determine the length of the segment:

$$|AB| = Length(A, B) = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2} .$$

Substituting an algebraic object instead of its name into a mathematical model of a task defines a complete algebraic model of the task:

$$\begin{cases} \begin{cases} x_A = -2 \\ y_A = 5 \end{cases} \\ \begin{cases} x_M = x \\ y_M = y \end{cases} \\ \sqrt{(x_M - x_A)^2 + (y_M - y_A)^2} = 3 \\ y_A = y_M \end{cases}$$

The solution of the task can now be obtained by solving the systems of algebraic equations.

That is, the solution of *LGT* is carried out in terms of the simplest tasks, which are elementary transformations of analytic geometry, and algebraic transformations.

Support for a step-by-step solution to LGT. The introduced concept of the mathematical model of *LGT* allows us to investigate the problem of supporting the course of solving *LGT* in SCMEP.

LMTs are used to support the step-by-step solution in SCMEP using equivalence inference - an example based on the application of the rules of rewriting [1,2], is investigated in [6]. This type of inference naturally represents the course of the solution for *LMT* in algebra.

In order to support the course of solving *LGT*, it is proposed to use both an equivalence inference and a logical inference, since the mathematical models of *LGT* essentially use logical and algebraic means.

The course of the solution of *LGT* has two stages: the stage of compilation of the mathematical model and the stage of transformation of the model.

At the first stage the user must enter into the program the condition of the task. For *LGT*, the first stage plays a methodically important role. At this stage, support for user actions is to verify the mathematical model of the task [14]. Technologically, this is implemented in a separate window "Building mathematical model", which opens with the *Start Solution* command (Fig. 2). The *Verify* command has the function of verifying the correctness of the model.

Given: $A(-2;5)$
$M(x; y)$
$|AM| = 3$
$y = 5$
Find : x

Compose **Verify**

Fig. 2. Window "Building mathematical model" (schematic)

The implementation of this function requires:

1) the presence in the condition of the task of the correct model of this task and the answer (hidden from the user of the formal model (1));

2) implementation of the algorithm for comparing the model constructed by the user with the model or answer given in the condition of the task.

These requirements, in turn, require that the text of the task conditions all the symbols needed to formulate the mathematical model of the task.

The presence in the condition of tasks hidden from the user model of the task allows to automate the process of testing and debugging the text of the software module "*TaskBook*", as well as implement the function of composing the task model in software system (command *Compose*). Thus, if the user can not make a model of *LGT* independently, the system will perform this action itself.

The second stage - the stage of step-by-step solving is to form the course of the solution in the form of a sequence of transformations of the model of the task. The inference is a sequence of triples

$$((M_1, t_1, M_1'), (M_2, t_2, M_2'), \ldots, (M_j, t_j, M_j'), \ldots, (M_n, t_n, M_n')) \quad (2)$$

where M_j, M_j' – mathematical object , t_j - their transformation.

The problem we will discuss below is to define a complete, consistent and methodically correct list of transformations that support inference (2) and implemented as a structure of commands (references) that form the contents of the software module (*SM*) "*Guide*". This module, in turn, is used in the *SM* "*Medium of Solving*".

According to the definition of the *GO*, the *SM* "*Guide*" should contain both geometric and algebraic transformations, that is, the section "Transformation of algebraic objects", which contains equivalent algebraic transformations, the classification problem of which was investigated in [6,13], and the section "Transformation of geometric objects", which, in fact, contains the transformation of geometric objects.

Elementary transformations of geometric objects. Analyzing the problem from the point of view of the object-oriented programming paradigm, we distinguish transformation-constructors and transformation-selectors.

Transformation-constructor *PGO* builds *PGO* for its algebraic definition. Thus, the corresponding transformation has the specification $t : AO \Rightarrow PGO$.

In sequence (2), this transformation is represented by a triple (*AO*, *t*, *PGO*). If the *AO* conversion argument is allocated during the solution, the *PGO* is entered into the solution as its last (new) row. In Fig. 3 shows the selected equation in the 5th line of the solution and the transformation-constructor of a straight line by its equation, the result of which is entered into the course of the solution as the 9th row.

The course of the solution	*Reference*
… 5. Convert the equation: $y = 2 \cdot x + 1$ … 9. Construct a straight line l by the equation (5): $l(y = 2 \cdot x + 1)$	◙ Construct a straight line for its equation ▭ Highlight the equation of the line $F(x, y) \Leftrightarrow l(F(x, y))$

Fig. 3. A fragment of the course of the solution of the *LGT*

Transformation-selector allocates one or more objects that are included in the definition of the *GO*. Constructors and selectors, as a rule, can be interpreted as one reference that contains mutually inverse transformations (Fig. 4).

Reference

◙ Construct a circle according to its parameters ▢ Highlight circle settings $l((x - a)^2 + (y - b)^2 = r^2) \Leftrightarrow \begin{cases} a_l = a \\ b_l = b \\ r_l = r \end{cases}$

Fig. 4. Reference – constructor and selector circle parameters

Transformation-definition introduces *AO* - definition of *GO*. For example, in the course of the solution of the *LGT*, we can include the definition of the tangent line to the function $y = f(x)$ at point A (Fig. 5).

Reference

Equation of tangent line L to function graph G $(y = f(x))$ at the point A $L\ (y - y_A) = f'(x_A)\ (x - x_A)$

Fig. 5. Transformation-definition of a tangent line to the function graph

The transformation of the *CGO*, whose classes contain the definition of additional objects, are described by the *reference-selectors of additional objects*.

We separately note the necessity of the constructor and selectors of such algebraic objects as the equation, the system of equations, and the aggregate of equations. These transformations should be part of the Algebraic section of the *SM* "*Guide*".

In addition to the transformations of the classes of *GO*, one must identify elementary transformations - operations on objects, so-called elementary tasks of analytic geometry. For example, a *Point* × *Point* ⇒ *Line* transform, which defines a straight line passing through two points (Fig. 6).

Reference

Calculate a straight line equation that passes through two points $A, B \Rightarrow l(\frac{x - x_A}{x_B - x_A} = \frac{y - y_A}{y_B - y_A})$

Fig. 6. Conversion – basic task of analytical geometry

Logical transformations - the transformations of the type $GO \times GO \Rightarrow Bool$ return the value True, False. These transformations solve the tasks of the mutual position of the GO - the parallelism (the perpendicularity) of the straight lines, etc.

A special particular type of transformation is the *isomorphic transformation* of GO. They include: elementary transformations of the Cartesian plane or space, transformation of the transition to a polar coordinate system on a plane or transformation of the transition to spherical or cylindrical coordinate systems in space, transformation of the transition to vector algebra.

In the analytic geometry on the plane the following elementary transformations are used: parallel transfer, turning to the angle, stretching / compressing.

Both methodically and technically, these transformations should be realized in two forms: as an elementary transformation of a geometric object and as an elementary transformation of a plane. The transformation of the GO is to construct a new GO in the "old" xOy coordinate system. For example, a parallel transfer of a GO is defined by the transformation

$$l(F(x, y) = 0) \Rightarrow m(F(x + a, y + b) = 0).$$

The parallel transfer of the xOy plane is determined by the transformation

$$l(F(x, y) = 0) \Rightarrow l(F(x' + a, y' + b) = 0) \;\&\; x' = x - a, y' = y - b.$$

In the first form the result is a new object. Consequently, it is determined by the new ID and the new AO. The coordinate system remains "old". The second form changes the coordinate system and the AO. ID of geometric object is advisable to leave. The graphic illustration of this transformation form is to create a new Cartesian plane - the plane $x'Oy'$ and the reflection of the GO in a new coordinate system.

The transformation of the transition to the polar coordinate system associated with the xOy system determines the transformation

$$l(F(x, y) = 0) \Rightarrow l(F(\rho \cdot \cos(\varphi), \rho \cdot \sin(\varphi)) = 0).$$

The inverse transform has the form

$$l(F(\rho, \varphi) = 0) \Rightarrow l(F(\sqrt{x^2 + y^2}\ , arctg(y/x) = 0).$$

The transformation of the transition to vector algebra. The solution of a LGT by a vector method consists in constructing a task model in the form of a formula in the signature of the Euclidean space and solving the task by algebraic transformations of this formula. Conversion of the transition to the vector method consists in the application the type of a reference (Fig. 7).

Reference

Consider the vector $A, B \Rightarrow a = AB$

Fig. 7. Reference - Transformation of the transition to vector algebra

The further course of the solution of the LGT should be based on the list of transformations-definitions of vector geometric formulas and transformations-formulas in the signature of the Euclidean space. The list of definitions of vector geometric objects determines the completeness of the "*Guide"* from the section "Vector method in geometry". Here is an example of this section of the *Guide*:

<u>*Vector method in geometry / Vector properties of geometric shapes*</u>

• Axiom of a directed section: $AB = -BA$

• Axiom of the triangle ABC: $AB + BC = AC$, etc.

<u>*Transformation in the Euclidean Space*</u>

I. *Axiom Euclidean vector space*

II *Theorems of geometry*

• Definition of the scalar product: $(a,b) = |a|\cdot|b|\cdot\cos(a,b)$.

• The cosine theorem: $|AB|^2 = |BC|^2 + |AC|^2 + 2\cdot|BC|\cdot|AC|\cdot\cos(BC,AC)$, etc.

A more detailed analysis can be found in [11]. The basis of the formation of sections of the "*Guide"* was chosen [12].

Inference in geometric modules. If the model of *LGT* is represented in the form of one formula, the course of its solution, in principle, can be obtained as a result of the equivalence inference (2). However, the submission of the terms in this form is not accepted. It is generally acceptable to formulate the condition in the form of a list of primitive conjuncts that specify the relationship between the PGO parameters of the condition of the task. Here is an example of the application of the mathematical methods and models for a geometric task.

Task 2. Compose the equation of the tangent to the graph of the function $y = x^2 + 1$ *at the point A with abscissa* $x = 1$.

Given: 1. Graph $F(y = x^2 + 1)$.

2. The point $A(1, y_A)$.

This condition must be supplemented by definitions by using the following elementary transformations-definitions:

3. Point A belongs to the graph F: $y_A = x_A^2 + 1$.

4. Equation of tangent line L to graph F in point A: $L(y - y_A) = f'(x_A)(x - x_A)$.

Rows 1-4 of the solution are a model of the task:

$$F(y = x^2 + 1)\&A(1, y_A)\&(y_A = x_A^2 + 1)\&L(y - y_A) = f'(x_A)(x - x_A).$$

Consequently, the step of inference (solving) depends on the data presented in the previous steps. To execute it, the user must specify the appropriate lines, find the transformation in the "*Guide"* and apply it.

Realization. The scientists of the Department of Informatics, Program Engineering and Economic Cybernetics of the Kherson State University under the guidance of the professor M. Lvov are engaged the implementation of the concept SCMEP.

Here are some SCMEP created by order and recommended by the Ministry of Education and Science of Ukraine (copyright certificates [7-8]):

- Program-methodical complex «TerM VII» of the support of a practical mathematical learning activity;
- Software tool "Library of Electronic Visual Aids Algebra 7-9 grade for secondary schools in Ukraine";
- Pedagogical software tool "Algebra, Grade 7";
- Software tool for educational purposes "Algebra, Grade 8";
- Integrated environment for the study of the course "Analytical geometry" for higher education institutions; and others.

Features of the support of the course of solving LMT in algebra and the principles of classification of elementary algebraic transformations for school algebra on the example of the program module (*PM*) "*Guide*" of TerM VII is considered in [6].

The concept of supporting the course of solving geometric tasks in SCMEP is partially implemented in TerM VII-IX, since the content of the course of school algebra presents educational tasks that contain elements of analytic geometry and are formulated in terms of geometric objects.

In TerM VII-IX there is a program module "*Graphs*", which is intended for the solution of *LGT* and *LMT* in algebra by graphic method. In it, it is possible to formulate an algorithm for solving a task in the form of a sequence of commands defined by the references of the *PM* "*Guide*" of the *PM* "*Graphs*". Its content is a structured list of elementary transformations of this mathematical model of the training module.

The *PM* "*Guide*" contains the following references:

Section Formula-Graph

1. Construct a point $A(x; y)$ by its coordinates,
2. Construct a straight line $ax + by + c = 0$, etc.

Section Graph-graph

1. Conduct a straight line passing through two given points,
2. Find the point of intersection of two straight lines, etc.

Section Graph -Formula

1. Find reproduction of the coordinates of the constructed point,
2. Find the equation of a constructed straight line, etc.

Section Converting Function Graphs

1. Parallel movement in the direction of the abscissa axis $x \Rightarrow x - a$,
2. Stretching (compression) from the abscissa axis $x \Rightarrow kx$, etc.

The use of the "*Graphs*" for solving a system of linear equations graphically and an algorithm for solving it using references of the “Guide” are illustrated in Fig. 8.

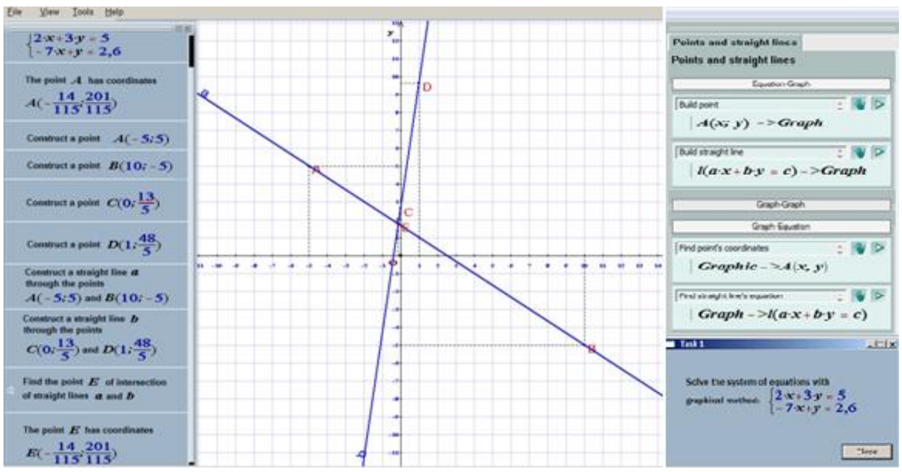

Fig. 7. Graphical method of solving the system of linear equations

We note that often the course of the solution of the *LGT* can be described as the language of algebra, as well as in the language of geometry ("find the intersection of the *GO*" or "solve the system of equations").

The proposed methodology was verified experimentally during the introduction into the educational process of “Integrated environment for the study of the course "Analytical geometry" in higher educational institutions of Ukraine. The experiment showed that the study of the course "Analytical geometry" provides a higher level of mathematical competence in the students of technical specialties [18, 19].

4 Conclusions

Practical mathematical activity of the student is the main form of educational activity in the study of mathematics. It consists in getting the course of the *LMT* solution.

In the work:

- the mathematical model of the LGT is defined within the framework of mathematical model of the training module and construction of the corresponding algorithms of computer algebra.

- the problem of forming a complete, non-contradictory and methodically correct list of transformations, by which one can carry out logical derivation, as a step-by-step solution to *LGT*, is solved.

- mathematical models of methods for supporting the solution of *LGT* in SCMEP are constructed.

Object-oriented analysis of the problem revealed the main classes of transformations of geometric objects (constructors, selectors and converters (elementary tasks)).

Obviously, in addition to algebra and geometry, this approach can be applied to such subject areas, where the content of educational tasks is the formal properties of interacting objects (physics, engineering, etc.).

References

1. Podkolzin A.S. Komp'juternoe modelirovanie logicheskih processov. Tom 1. Arhitektura i jazyki reshatelja zadach. /A.Podkolzin. –M: Fizmatlit. –2008, 1024 s.
2. Lvov M.S. Metodi proektuvannya system komp'juternoi pidtrimky matematichnoi osvity / M.S. Lvov, O.V. Spivakovsky // Matematychni modeli i suchasni informacijni tehnologii:zb. nauk. prac NAN Ukraini. Kiev. 1998, S. 101-111.
3. Lvov M. Osnovnie princypy postroenija pedagogicheskih programmnih sredstv poddergki prakticheskih zanjatiy / Lvov M. // Upravljajushhie sistemy i mashiny. – 2006. – № 6. – S. 70-75.
4. Lvov M.S. Shkilna systema kompjuternoji algebry TerM 7-9. Pryntsypy pobudovy ta osoblyvosti vykorystannia. / M.S. Lvov // Naukovyi chasopys NPU im. Drahomanova, seriia №2. Kompiuterno-oriientovani systemy navchannia: zb.nauk. prats/ redkol. – K.: NPU im. Drahomanova. –#3(10) -2005. S. 160–168.
5. Lvov M., Shishko L., Chernenko I., Kozlovsky E. Systems of computer mathematics for educational purposes as the means of reception of procedural knowledge. CEUR Workshop Proceedings of the 12th International Conference on ICT in Education, Research and

Industrial Applications. Integration, Knowledge and Harmonization Transfer, -K.:, Ukraine, June 21-24, -2016, S. 215-130.

6. Lvov M., Shishko L., Chernenko I., Kozlovsky E. Mathematical Models of Supporting the Solution of the Algebra Tasks in Systems of Computer Mathematics for Educational Purposes CEUR Workshop Proceedings of the 14th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer, - Kyiv, Ukraine, May 14-17, - 2018, P. 472-487
7. A.s. 12440 MONU. Derzhavnyi departament intelektualnoi vlasnosti. Zbirka kompiuternykh prohram Prohramno-metodychnyi kompleks «TerM VII» pidtrymky praktychnoi navchalnoi matematychnoi diialnosti. Versiia 2.0, reliz 03.» osvitnoi haluzi «Matematyka» dlia zahalnoosvitnikh navchalnykh zakladiv / Lvov M., Spivakovsky O., vs'ogo 9 osib; zaiavl. 10.03.2005. Biul. №7. – 2005.
8. A.s. 32724 MONU. Derzhavnyi departament intelektualnoi vlasnosti. Komputerna prohrama «Intehrovane seredovische vivchennia kursu «Analitichna geometria» dlia vischih navchalnih zakladiv» / Peschanenko V., Lvov M., Spivakovsky O.; vs'ogo 8 osib; zaiavl. 06.04.10. Biul. №21. – 2010.
9. Lvov M.S. Koncepcija proghramnoji systemy pidtrymky matematychnoji dijaljnosti./ M.S.Lvov // Komp'juterno-orijentovani systemy navchannja: Zb. nauk. praci/ K.:NPU im. M.P.Draghomanova.– Vyp. 7. – 2003. – S. 36–48.
10. Pol A. Ob'ektno-orientirovannoe programmirovanie na S / Pol A. – [2-e izd.].–Spb.; M.«Nevskiy dialekt»–izd-vo «BINOM», 1999.– 462 s.:il
11. Bugrov Ya.S. Elementyi lineynoy algebryi i analiticheskoy geometri / Bugrov Ya.S. – M.: Nauka, 1988. – 222 s.
12. Vyigodskiy M.Ya. Spravochnik po vyisshey matematike / Vyigodskiy M.Ya. - [8-e izd.]. – M.: «Nauka», 1966. – 872 s.: il.
13. Lvov M.S. Term VII – shkilna systema kompiuternoi alhebry. / M.S. Lvov // Kompiuter u shkoli ta simi. – 2004. – №7. – S. 27–30.
14. Lvov M.S. Proektirovanie logicheskogo vyvoda kak poshagovogo reshenija zadach v matematicheskih sistemah uchebnogo naznachenija / M. S. Lvov // Upravljajushhie sistemy i mashiny. – 2008. – №1. – S.25–32.
15. Lvov M.S., Shishko L.S., Chernenko I.E. Pro orhanizatsiiu kontroliu znan yak zvorotnoho zviazku v systemakh kompiuternoi matematyky navchalnogo pryznachennia / L.S. Shishko, I.E. Chernenko // Informatsiini tekhnolohii v osviti: Zb. nauk. pr. Vyp 16. - Kherson: Vyd. KhDU, 2013. – S. 29-42.
16. Windham Wyatt K. Geometry Activities for Middle School Students with The Geometer's Sketchpad /K. Windham Wyatt, A. Lawrence, G. M. Foletta// Key Curriculum Press.– 1998.– P. 221.
17. Lvov M. Software Module Representing a Geometry Tasks in Mathematical Systems of Educational Appointment / M. Lvov, D. Melnyk //CEUR Workshop Proceedings of the 14th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer, Volume II: Workshops Kyiv, Ukraine, May 14-17, –2018, – P. 73-82.
18. Grigor'eva V.B. Komp'yuterno-orientovana metodika navchannya analitichnoyi geometriyi maybutnih programistiv: dis. kand. … ped. nauk: 13.00.02 / Grigor'eva Valentina Borisivna. – Herson: HDU, 2017 – 270 s.
19. Zaytseva T.: The Introduction of the Competence-based Approach in Educational Process of Training of Skippers. In: 12th International Conference on ICT in Education, Research and Industrial Applications: Integration, Harmonization and Knowledge Transfer, pp. 687-699. Kyiv, Ukraine, CEUR-WS, 1614, (2016).