- використання часового розподілу прильоту та вильоту;
- розроблення та впровадження нових маршрутів польоту;
- вдосконалення наявних та розроблення нових маршрутів SID та STAR;
- моніторинг сезонних збільшень кількості польотів та підтримання їх кількості в умовах максимальної допустимої пропускної спроможності.

Яскравий приклад успішної підготовки до дії в умовах високої інтенсивності продемонструвала Україна під час проведення Євро–2012.

Висновки. Для більш чіткого аналізу можливих ризиків у разі збільшення інтенсивності потрібен глибокий аналіз авіаційних подій, які траплялися за умови високої інтенсивності.

Спрогнозувати та розробити механізми дотримання ризиків на допустимому рівні в умовах високої інтенсивності досить важко.

Оскільки зріст інтенсивності може відбутися за рахунок збільшення кількості польотів, найбільш прогнозований фактор, несприятливих погодних умов, введення в дію обмеження використання повітряного простору та інше.

Однозначно, що під час збільшення хоча б на 1 % інтенсивності польотів, відбудеться збільшення ролі певної кількості факторів ризику, що, в свою чергу, потребує швидкої реакції авіаційного персоналу.

Список використаних джерел

1. *Козлов В. В.* Безопасность полетов: от обеспечения к управлению / В. В. Козлов. – М. : Аэрофлот, 2010. – 270 с.
2. *Руководство* по управлению безопасностью полетов / за ред. Р. Бенджамина. – Изд. 2-е. – Монреаль : ИСАО, 2009. – 308 с.
3. *Энциклопедия* безопасности авиации / М. С. Кулик, В. П. Харченко и др. – К. : Техника, 2008. – 1000 с.
4. *ICAO safety* report [Електронний ресурс]. – Режим доступу : www.icao.int/safety/Documents/ICAO_2014%20Safety%20Report_final_02042014_web.pdf.
5. *International* General Aviation and Corporate Aviation Risk Assessment (IGA-GARA) Project. – Department of Air Transport Cranfield University. – 2005. – 79 p.

УДК 004.94:377

Inna Khomenko, PhD in Technical Sciences
Poltava National Technical Yuri Kondratyuk University, Poltava, Ukraine

COMBINING SPACE GRAMMARS AND GEOMETRIC MODELS: SYNTHESIS, MODIFYING AND COMPARISON

І.В. Хоменко, канд. техн. наук
Полтавський національний технічний університет імені Юрія Кондратюка, м. Полтава, Україна

ПОЄДНАННЯ ПРОСТОРОВИХ ГРАМАТИК ТА ГЕОМЕТРИЧНИХ МОДЕЛЕЙ: ГЕНЕРАЦІЯ, ПЕРЕТВОРЕННЯ ТА ЗІСТАВЛЕННЯ

И.В. Хоменко, канд. техн. наук
Полтавский национальный технический университет имени Юрия Кондратюка, г. Полтава, Украина

СОЧЕТАНИЕ ПРОСТРАНСТВЕННЫХ ГРАММАТИК И ГЕОМЕТРИЧЕСКИХ МОДЕЛЕЙ: ГЕНЕРАЦИЯ, ПРЕОБРАЗОВАНИЯ И СОПОСТАВЛЕНИЕ

This proposal introduces combining space grammars and geometric models, a formalism for generation complex models of rigid solid objects. Solids are represent by their primitives with coordinate position. Labels may be associated with any of these elements. Rules match conditions of a solid or collections of solids and may modify them or create additional solids. A grammar uses an initial solid and a set of rules to produce a language of solid models.

Unary operations are introduce to ensure the validity of the representations. These operations take models that may have self-intersections, interpret the models considering the given geometry and face orientations, and produce valid models.

The proposed formalism has been implemented. Grammars that have been demonstrated that generate simple geometric forms, more extensive grammar generate engineering details.

***Key words:** space grammar, geometric models, engineering design and practice.*

Запропоновано поєднання просторових граматик і геометричних моделей, формальний опис генерації складних моделей твердотілих елементів. Тіла представлені своїми примітивами з узгодженим розташуванням. Позначення можуть бути пов'язані з будь-яким із цих елементів. Правила інтерпретують стан елементу або їх сполучення і можуть їх перетворювати або створювати додаткові елементи. Граматика використовує первісний елемент та набір правил для утворення мови твердотілого моделювання.

Унарні операції вводяться для забезпечення достовірності представлень. Ці операції приймають моделі, які можуть перетинатися, представляють моделі з урахуванням заданої геометрії й орієнтації та створюють правильні моделі. Запропонований формальний опис реалізовано. Продемонстровані граматики генерують прості геометричні форми, більш складні граматики генерують інженерні деталі.

***Ключові слова:** просторові граматики, геометричні моделі, інженерне проектування і практика.*

Предложено сочетание пространственных грамматик и геометрических моделей, формальное описание генерации сложных моделей твердотельных элементов. Тела представлены своими примитивами с согласованным расположением. Обозначения могут быть связаны с любым из этих элементов. Правила интерпретируют состояние элемента или их сочетание и могут их преобразовывать или создавать дополнительные элементы. Грамматика использует первоначальный элемент и набор правил для получения языка твердотельного моделирования.

Унарные операции вводятся для обеспечения достоверности представлений. Эти операции принимают модели, которые могут пересекаться, представляют модели с учетом заданной геометрии и ориентации и создают правильные модели. Предложенное формальное описание реализовано. Продемонстрированные грамматики генерируют простые геометрические формы, более сложные грамматики генерируют инженерные детали.

***Ключевые слова:** пространственные грамматики, геометрические модели, инженерное проектирование и практика.*

Formulation of the problem. The basic data in the training system of computer graphics are geometric models, because they help describe any engineering projects and construction problems usually solved using geometric transformation models. To date, not yet fully evaluated spatial modeling capabilities in technology training and retraining for the design and production engineering, and for their operation and maintenance.

The presentation of knowledge we understand a set of methods, formal languages and specialized tools that allow professionals subject area to determine a way to solve the problem. Variants of knowledge presentation focused on the description of the subject area and are intended to address the problem of attracting specialist subject area in the process of developing automated systems. Design procedures are created by experts specialized knowledge representation languages and using the tools included in the final software program or turn to the defined programming language.

Knowledge representation languages should be similar to the description of the subject area presented in reference books available for understanding and relevant experts. In the process the problem is solved and simplified its formalization. Formal knowledge accumulated in an accessible form for reuse. Knowledge representation allows to isolate application problems of development and structure of tasks related to programming. Subject area specialists and programmers can work on creating an automated system in parallel. Simplified support ready automated system.

Thus, one could argue that the development and implementation of tools for synthesis, modifying and comparison of geometric models, including computerization system for engineering education is relevant and promising research due to the trend towards individualization of learning and efficiency of education.

Recent research and publications. Recently, much attention should spatial grammar, languages which are set spatial figures, and not one-dimensional strings. In particular, Mr. Wang and W. Hrosky [4] examined the relationship between the classical structural models, parallel spatial grammars and programmable parallel grammars that have sufficient capacity both in theoretical and applied aspects. They were investigated general features like the generation of space using spatial grammars defined structural model of parallel spatial grammars and programmable parallel grammars. But many issues in this area remain open.

The purpose of the article. Given the advantages and disadvantages of traditional languages and languages of artificial intelligence is proposed to implement specialized language that uses the principles of artificial intelligence declarative languages and sold in object-oriented languages. Designed we should take into account the specific representation of geometric models.

About Constructive Solid Geometry (CSG). A CSG model is based on the topological notion that a physical object can be divided into a set of primitives (basic elements or shapes) that can be combined in a certain order following a set of rules (Boolean operations) to form the object. Each primitive is bounded by a set of surfaces; usually closed and orientable. A CSG model is fundamentally and topologically different from a B-rep model in that the former does not store explicitly the faces, edges, and vertices. Instead, it evaluates them whenever applications' algorithms need them, e.g., generation of line drawings. The concept of primitives offers a different conceptual way of thinking that may be extended to model engineering processes such as design and manufacturing. It also appears that CSG representations might be of considerable importance for manufacturing automation as in the study of process planning and rough machining operations.

The modeling domain of a CSG scheme depends on the half-spaces that underlie its bounded solid primitives, on the available rigid motion and on the available set operators. For example, if two schemes have the same rigid motion and set operations but one has just a block and a cylinder primitive and the other has these two plus a tetrahedron, the two schemes are considered to have the same domain. Each has only planar and cylindrical half-spaces, and the tetrahedron primitive the other system offers is just a convenience to the user and does not extend its modeling domain. Extending the solid modeling domain to cover sculptured surfaces requires representing a "sculptured" half-space and it is supporting utilities.

Primitives themselves are considered valid "off-the-shelf" solids. In addition, some packages, especially those that support sweeping operations, permit users to utilize wireframe entities to create faces that are swept later to create solids.

There is a wide variety of primitives available commercially to users. However, the four most commonly used are the block, cylinder, cone and sphere. These are based on the four natural quadrics: planes, cylinders, cones, and spheres. These quadrics are considered natural because they represent the most commonly occurring surfaces in mechanical design which can be produced by rolling, turning, milling, cutting drilling, and other machining operations used in industry.

From a user-input point of view and regardless of a specific system syntax, a primitive requires a set of location data, a set of geometric data, and a set of orientation data to define it completely. Primitives are usually translated and or rotated to position and orient them properly before applying boolean operations.

For interactive process geometric 3D-modeling bodies convenient to use mathematical logic operations (Fig. 1):

– Union $A \cup B$; this operation brings together two solids into one and removes areas that intersect or overlap;

– Intersection $A \cap B$; this operation creates a solid that contains only the intersection of two solids;

– Subtraction $A \setminus B$; this operation allows you to remove one of the solids and any solids intersection.

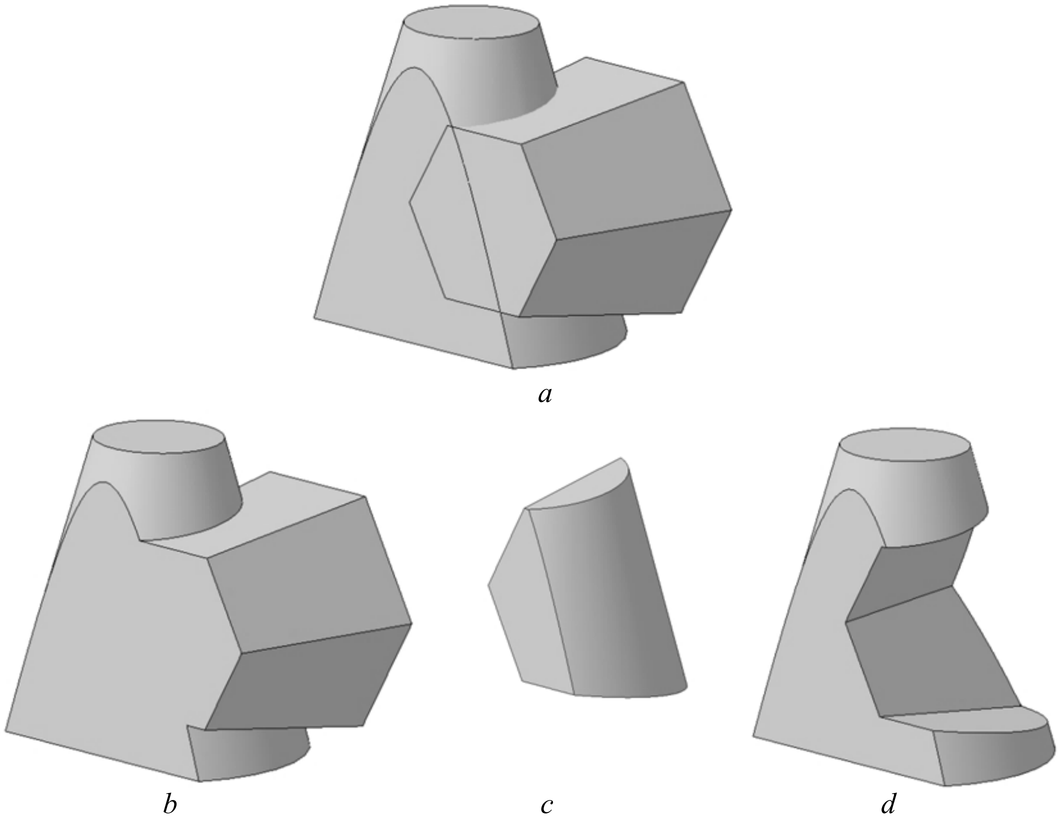

Fig. 1. Operations with solids:
a –position of solids A and B; b – union $A \cup B$ *;c – intersection* $A \cap B$ *; d – subtraction* $A \setminus B$

Grammar modeling. Proposed language representation of geometric parametric models described grammar G_T = (E, A, x :: = y, S), where E – finite set of elements; A – finite set of operations; x :: = y – finite set of inference rules; S – geometric patterns (S∈E).

In the simplest case, the transactions volume bodies withdrawal rules are presented as follows [1]:

<model> :: = <item> | <model> <operation> <item>

<operation> :: = $\cup$ | $\cap$ | \

Further expansion resulted grammar is adding special operations modeling, including the establishment of membranes, chamfers, fillets, ribs, arrays and other elements. In general, modeling grammar ambiguous or non-determined.

Tree Grammar output simulation model defines a tree or wood constructions of the product. The root object tree – model part or product. Components assembly unit – parts and assembly units – are independent model. Each leaf in this tree corresponds operand, and each parent node – operation (Fig. 2).

The nodes of the tree is the real body and the relevant parameters.

Presentation models as binary trees (degree not more than two) can effectively perform the most common operation – rebuilding model.

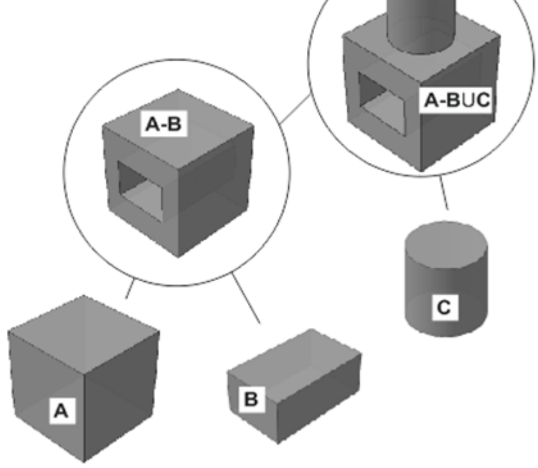

Fig. 2. The tree of output modelling grammar

Figure 3 shows a typical solid and its building primitives. This solid can be built following the steps below:

$$\left.\begin{array}{l} B_1 = block \quad positioned \quad properly \\ B_2 = block \quad positioned \quad properly \\ B_3 = block \\ B_4 = B_3 \quad moved \quad properly \quad in \quad the \quad X \quad direction \\ C_1 = cylinder \quad positioned \quad properly \\ C_2 = C_1 \quad moved \quad properly \quad in \quad the \quad X \quad direction \\ C_3 = cylinder \quad positioned \quad properly \\ C_4 = C_3 \quad moved \quad properly \quad in \quad the \quad X \quad direction \end{array}\right\} Primitives' \quad definitions$$

$$\left.\begin{array}{l} S_1 = B_1 \quad \cup * \quad B_3 \\ S_2 = S_1 \quad \cup * \quad C_1 \\ S_3 = S_2 \quad \cup * \quad C_3 \end{array}\right\} Construct \quad left \quad half$$

$$\left.\begin{array}{l} S_4 = B_2 \quad \cup * \quad B_4 \\ S_5 = C_2 \quad \cup * \quad S_4 \\ S_6 = C_4 \quad \cup * \quad S_5 \end{array}\right\} Construct \quad right \quad half$$

$$S = S_3 \quad \cup * \quad S_6 \} Model$$

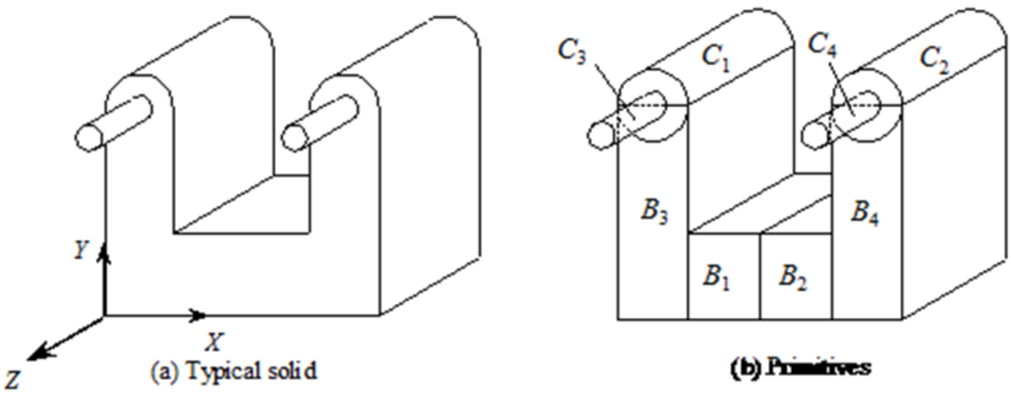

Fig. 3. A typical solid and its building primitives

A CSG tree is defined as an inverted ordered binary tree whose leaf nodes are primitives and interior nodes are regularized set operations. The creation of a balanced, unbalanced, or a perfect CSG tree depends solely on the user and how he/she decomposes a solid into its primitives. The general rule to create balanced trees is to start to build the model from an almost central position and branch out in two opposite directions or vice versa. Another useful rule is that symmetric objects can lead to perfect trees if they are decomposed properly. Figure 4 shows a perfect CSG tree (a) and an unbalance CSG tree (b).

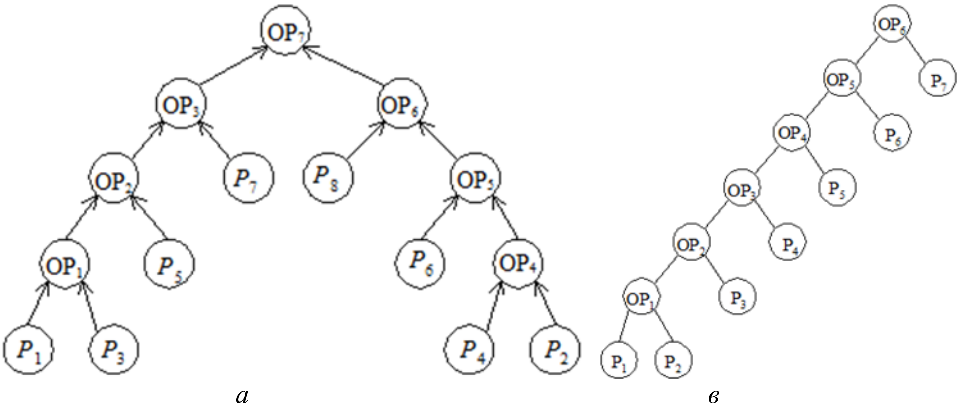

Fig. 4. CSG trees: a – of a typical solid; b – unbalanced CSG tree

Language of modelling. Language L, based on grammar GT = (E, A, x :: = y, S) is designed to create knowledge bases saturated engineering graphics. The author realized modeling language means creating models interpreted in the form of expansion by means of Pascal among Delphi Borland. The basic operation of the language is equivalent modify models.

In the unformalized form equivalent transformation takes place, for example, in the recording when combining commutativity law in the form $a+b=b+a$. Equality of variables is called a model and reflects the infinite set of equations, if the variables in place to substitute an item (Fig. 5).

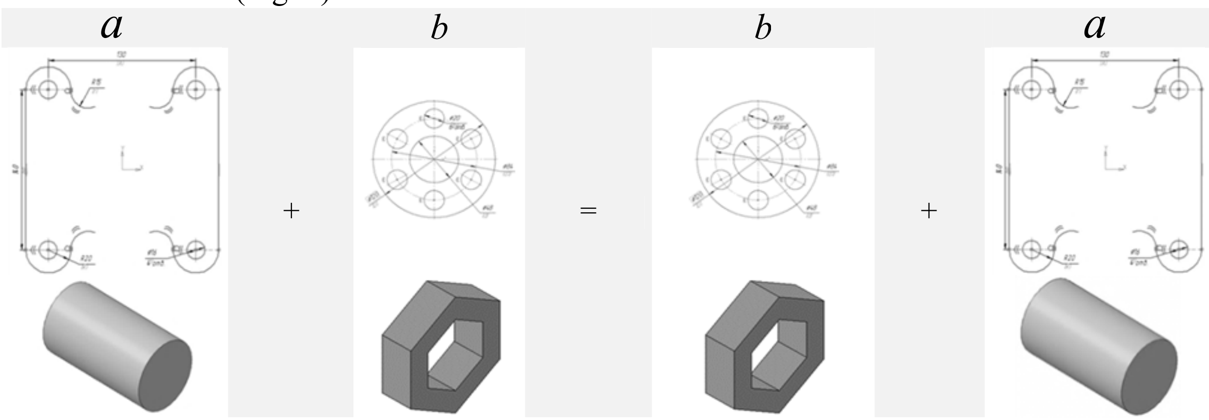

Fig. 5. The set of equations when use the Union

Conclusions and suggestions. The developed formal models comparison processes of structural transformation and geometric models, which enables create dynamic knowledge base. Created structure of the learning process engineering disciplines and developed tools for creating graphics and control tasks.

The practical significance of the results is in create an effective information resource formation technology computer aided teaching engineering disciplines based on the developed methods and software. Information technology implemented in universities of Ukraine and used in the study of engineering disciplines.

References

1. *Варсанофьев Д. В.* Основы компиляции [Электронный ресурс] / Д. В. Варсанофьев, А. Г. Дымченко. – Режим доступа : http://www.codenet.ru/progr/alg/cmp/intro.php.

2. *Пальчевський Б. О.* Інформаційні технології проектування технологічного устаткування : монографія / Б. О. Пальчевський. – Луцьк : Луцький НТУ, 2012. – 572 с.

3. *Хоменко І. В.* Еквівалентні перетворення геометричних моделей / Інна Володимирівна Хоменко // Східно-Європейський журнал передових технологій. – 2013. – № 1/4 (61). – С. 34–42.

4. *Book* of Abstracts // The 6th Intern. Conf. on Mathem. Modelling, Sant-Louis, 1987.

5. *Sosnovsky S.* Web-based Parameterized Questions as a Tool for Learning / S. Sosnovsky, O. Shcherbinina, P. Brusilovsky // Proceedings of E-Learn 2003, Arizona USA. – 2003.