ATLANTIS PRESS

Advances in Intelligent Systems Research, volume 166

7th Scientific Conference on Information Technologies for Intelligent Decision Making Support (ITIDS 2019)

Intelligent Method of the Automated Design of Technological Processes of Machining

Alexey Lutov
Institute of aviation technologies and materials
Ufa State Aviation Technical University
Ufa, Russia
lutov1@mail.ru

Yuri Ryabov
Institute of aviation technologies and materials
Ufa State Aviation Technical University
Ufa, Russia
ryabov_yuri_atp@mail.ru

Rinat Shaydullin
Institute of aviation technologies and materials
Ufa State Aviation Technical University
Ufa, Russia
sh-rinat1996@yandex.ru

***Abstract*—The paper presents an intelligent method of computer-aided design of technological processes of mechanical processing, based on the principles of an intelligent approach to information processing and the use of knowledge systems in solving technological design problems. A unified approach to the design of technological processes is proposed, implemented through a graph model representing the relationships between data. When developing operations, structural synthesis is used based on successive chains of knowledge of transitions. For this purpose, knowledge elements have been created and are described in the XSD language. Elements of knowledge of technological transitions determine the technological actions associated with the processing of the elementary surfaces of the part. The practical implementation of chains of executive transitions is proposed to be implemented using functional programming. To implement the digital transformation of the technological preparation of mechanical processing using the intelligent approach, a method of forming a prototype of the technological process was developed.**

Keywords—Intelligent method, graph model, technological process, knowledge element, functional design, functional programming, synthesis, XSD language.

I. Introduction

Currently, an important scientific task is the study of methods and models of digital transformation of high-tech processes based on the principles of an intelligent approach to information processing and the application of the knowledge system. All existing systems for the development and operation of technological processes are aimed only at automating the receipt of technological documents and control programs and do not meet the requirements of digital production due to the lack of intelligent technologies.

In the present paper, the method of computer-aided design of technological processes of the intelligent level is considered, allowing the technologist to find the right solution and apply modern tools to create an effective process. A unified approach to the design of technological processes is proposed, implemented through a graph model representing the relationships between data.

The work was carried out within the framework of the RFBR grant No. 15-07-04811a “Methodology for creating an integrated intelligent environment of technological training and monitoring of controlled processes of high-tech industries”.

II. Relevance of the Work

The relevance of the work is characterized by the achievement of the intelligent level in the digital transformation of high-tech production processes, based on the modeling of data on machining processes and specific knowledge of technological preparation processes. Such industries include machining, carried out on multi-functional CNC machines using active control and monitoring equipment and tools.

The scientific significance of the work is to create a unified strategy for the digital transformation of production and technological processes of mechanical processing and the formation of a knowledge base about these processes. The possibility of applying the knowledge system when performing technological preparation of production was considered in various papers [1–4]. Knowledge is proposed to be used to describe the product and to describe the rules of production in the design of technological processes. To represent knowledge of the structure of operations of technological processes, it is proposed to use the knowledge representation language to enter the values of variables and various conditions and determine the end of action of conditions [3]. In [5], questions were considered in the field of artificial intelligence and, in particular, questions of knowledge representation and reasoning processes, which connect knowledge with reality based on the use of intelligent agents.

The use of deductive machine learning based on the use of shells of expert systems and knowledge acquisition models is described in the work of the authors E. C. Ogu, Y.A. Adekunle [6]. Using the proposed knowledge acquisition model, human resource specialists can reuse expert system shells and program their knowledge expertise directly in the knowledge base.

III. Formulation of the Problem

The intelligent method of computer-aided design of technological processes of mechanical processing will allow to create an enterprise with an automated controlled technological process [7]. The implementation of the method is carried out through the integration of modern IT technologies, which can significantly increase the productivity of design.

An intelligent approach to the the digital transformation of production and technological processes is proposed implemented through a graph model representing the relationships between data. In the graphs, objects are represented by nodes, and the ways in which these objects are interconnected. As a result, interconnections are data, description and evaluation of which we need to understand the relationships between the constituent elements [8]. Relationships between nodes have a name and direction, because the start and end nodes are always defined for them. The graph approach allows you to create the required patterns among the objects related to a specific area. Organizing a graph allows you to write data once and then interpret it in different ways according to the relationship. This universal and expressive structure allows you to visually model the scenarios of machining processes. For data of any size and value, the graph data model is the best way to present and retrieve data associated with complex relationships in the machining industry.

One of the mechanisms to improve the efficiency of the technological process (TP) is the use of knowledge in its formation, defining various states of TP (metadata TP). Knowledge contains the skills and experience of technologists and workers involved in the machining production of manufacturing products. Knowledge is proposed to be presented in the form of knowledge elements that describe both technological objects and parts, and for technological transitions [9]. The latter determine the relationship between objects.

IV. GRAPH MODEL OF TECHNOLOGICAL PROCESS

An alternative to relational databases, data are graph databases. They offer a more natural presentation of information based on the same logic that we encounter in real life. They are initially focused on relationships between objects, and these relationships may have different characteristics [10]. Graph data models reflect one of the trends in modern business: the complex and dynamic relationships of leverage in highly-related data for obtaining competitive advantages [11].

In this case, the data model graph (Fig. 1) reflects the transformation of the part from workpiece Z at the workplace WP1, which includes a multifunctional CNC machine (machining center - MC), technological equipment (TE), a set of cutting tools (cutting tool - CTj) and operator O. Processing is carried out from the control program (control program - CP), which is set by the operator. The operation performed is the relationship between the workplace and the new state of the part after it has been processed - Di. Figure 1 shows two operations performed and the corresponding state of part D1 and D2.

The presented graph describes and connects two domains, one of which defines the area of workplaces (WP) for machining parts with used equipment, tools, accessories, control programs and operators. Another domenon determines the status of the part (D1 and D2) as a result of operations performed. The operations "005 and 010" are the relationship between the workplace and the corresponding state of the part. All the features of the operation are reflected in the properties of the "Run-operation" relationship. The data on all the objects and relationships under consideration are in a "key-value" format.

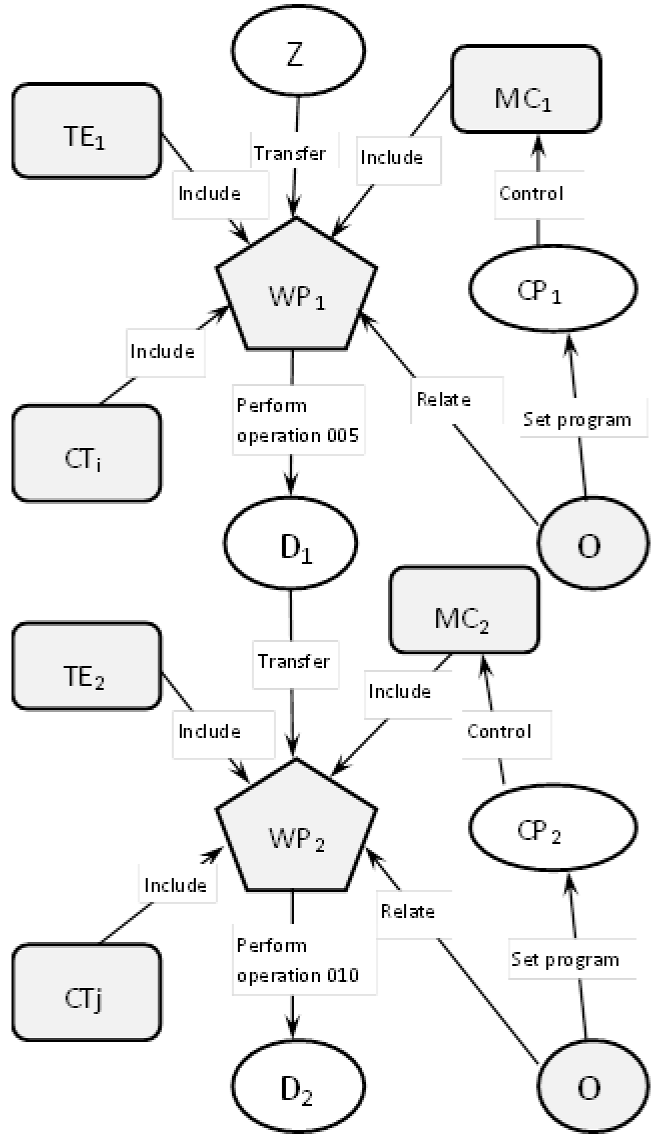

Fig. 1. Graph data model of mechanical machining

The paper uses the Titan DB in which Hadoop is used for analyzing graphs and for batch processing of graphs [12, 13]. Scaling graph processing for bypass in real-time and analytic queries is a fundamental advantage of titanium. An example of the listing of the program for describing the machining data graph in the Titan database environment is given below, where the graph description consists of edge labels, property keys and vertex labels used in them.

cd $TITAN_HOME/bin/gremlin.sh
Creating graph
g=TitanFactory.open
Enter vertexes and edges
v_d = g.addVertex ('type', 'detal' 'name', 'Z')
v_wp1 = g.addVertex ('type', 'workplace', 'name', 'WP1')
e_tr1= v_d.addEdge ('transfer1', v_wp1, 'type', 'transfer')
v_te1=g.addVertex ('type', 'technological_equipment', name', 'TE1')
e_in1= v_te1.g.addEdge ('include1', v_wp1, 'type','includes')
v_ct1 = g.addVertex ('type', 'cutting_tool', 'name', 'CTi')
e_in2= v_ct1.g.addEdge ('include2', v_wp1, 'type','includes')
v_mc1=g.addVertex ('type','machining_center','name',MC1')
e_in3=v_mc1.g.addEdge ('include3',v_wp1, 'type','includes')
v_cp1=g.addVertex ('type','control_programm', 'name', 'CP1')
e_cont1=v_cp1.g.addEdge ('conrol1',v_mc1,'type','controls')
v_o1 = g.addVertex ('type', 'operator', 'name', 'O1')
e_rel1 = v_o1.g.addEdge ('relate1', v_wp1, 'type', 'relates')
e_set11.g.addEdge ('control2',v_cp1, 'type','set_programms')
v_d1 = g.addVertex ('type', 'detal' 'name', 'D1')
e_perf1 = v_wp1.g.addEdge ('perform_operation_005', v_d1, 'type', 'performs')
v_wp2 = g.addVertex ('type', 'workplace', 'name', 'WP2')
e_tr2= v_d1.addEdge ('transfer2', v_wp2, 'type', 'transfer')
v_te2 = g.addVertex ('type', 'technological_equipment', 'name', 'TE2')
e_in2 = v_te2.g.addEdge ('include4', v_wp2, 'type', 'includes')
v_ct2 = g.addVertex ('type', 'cutting_tool', 'name', 'CTj')
e_in5 = v_ct2.g.addEdge ('include5',v_wp2, 'type','includes')
v_mc2 = g.addVertex ('type', 'machining_center', 'name', 'MC2')
e_in6 =v_mc2.g.addEdge ('include6',v_wp2,'type','includes')
v_cp2 = g.addVertex ('type', 'control_programm', 'name', 'CP2')
e_cont2 = v_cp2.g.addEdge ('control3', v_mc2,'type', 'controls')
v_o2 = g.addVertex ('type', 'operator', 'name', 'O2')
e_rel2 = v_o2.g.addEdge ('relate2', v_wp2, 'type', 'relates')
e_set2 = v_o2.g.addEdge ('control4', v_cp2, 'type', 'set_programms')
v_d2 = g.addVertex ('type', 'detal' 'name', 'D2')
e_perf2 = v_wp2.g.addEdge ('perform_operation_010', v_d2, 'type', 'performs')

Thus, the use of graph database allows you to visually form a technological process in the form of a graph and store information about operations attached to graph edges and information about objects attached to nodes (equipment, control programs, operators, terminals, etc.). Based on the data of the graph model, it is possible to form a route technology for processing a part. The mechanism for calculating graphs includes a process data recording system with OLTP properties, which serves requests and responds to requests from users. Requests to extract, transform, and load data transfer data from the database recording system to the graph calculation engine for performing offline queries and analysis.

The graph database includes not only the language of requests, but also algorithms that support complex reasoning: path analysis, clustering and ranking of vertices, identification subgraph, and much more. The capabilities of applied graph computing offer a flexible, intuitive structure of technological data along with algorithms for the efficient use of this structure [14]. As a result, a graph database can perform data mining.

V. Synthesis of Operations

After the graph of the data model should proceed to the development of operations consisting of individual transitions. A method of structural synthesis of technological operations based on sequential knowledge chains of transitions is proposed. For what should be formed the elements of knowledge of technological transitions describing the processing of elementary surfaces of the parts, as well as associated with the implementation of auxiliary transitions.

To form a knowledge base in the field of technological preparation of production, it is necessary to create elements of knowledge (EK) of objects and EK of technological transitions. The description of EK is proposed to be implemented in the XSD language used in the description of the metadata of the XML document. Knowledge is described by an element consisting of several characterizing elements for using a generalized transition. Attributes are used to describe the properties of elements. In EK, data types for elements and attributes are indicated.

Elements of knowledge of technological transitions describe technological actions associated with the processing of elementary surfaces of the part or associated with the implementation of auxiliary transitions. Working transitions can be (examples):

- trim the end, maintaining a long size;
- sharpen the contour, maintaining long and diametral dimensions;
- process the groove, maintaining its size and position;
- drill a hole of a given diameter and length;
- to mill a surface, maintaining certain sizes.

In an EK technological transition, information should be indicated on the dimensions of the machined part to be maintained, the cutting tool used and the type of processing used. Auxiliary transitions determine the installation and removal of parts, alignment and fixing parts, reinstall parts. Figure 2 shows an example of an EK for the "sharpen contour" transition.

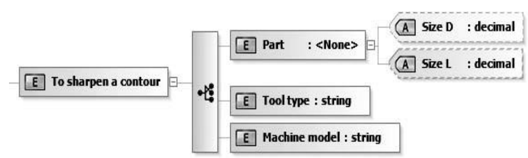

Fig. 2. Knowledge element "sharpen countour"

***Text Description** of an EK*

```
<?xml version="1.0" encoding="utf-8" ?>
<!--Created with Liquid XML Studio Developer Edition 9.0.11.3078
(http://www.liquid-technologies.com)-->
<xs:schema elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Sharpen_countour">
    <xs:complexType>
      <xs:all>
        <xs:element name="Detal">
          <xs:complexType>
            <xs:attribute name="Dimension Di" type="xs:decimal" />
            <xs:attribute name="Dimension Li" type="xs:decimal" />
          </xs:complexType>
        </xs:element>
        <xs:element name="Cutting_tool_type" type="xs:string" />
        <xs:element name=" Processing_type" type="xs:string" />
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

The description of EK in the XSD language is made in the program Liquid XML Studio [15, 16], which is a graphical environment for developing XML and their schemes in the XSD language.

The process of formation of the technological process is a set of procedures for constructing a graph model of the technological process and structural synthesis of operations from EK transitions. The synthesis of performance transitions for processing and assembling a product is a sequence of transitions necessary to achieve the required 3D data — the product model and attributes defined by the product drawing [17]. These atri-butami are:

- geometric type of surface;
- established dimensional tolerances;
- Installed surface roughness;
- established tolerances of maximum deviations of the shape and location of surfaces;
- type of heat treatment, etc.

As a result, an i - operation is formed with the help of a chain of executive transitions (CIC) i composed of elements of knowledge Ek, Ekk, Ekn, Ekm. The synthesis of executable transitions is carried out not only due to the influence of the attributes of the product drawing and 3D visualization - the model, but also conditions such as [17]:

- the maximum value of tolerance, which determines the type of transition (rough, semi-finishing or finishing);
- grade of the material being processed;
- tool material grade;
- accuracy class of the machine;
- the necessary use of the number of simultaneously controlled coordinates of the movement of the tool.

The following represent examples of forming operations with a body treatment rotation:

```
(EK(rastochit_syrye_kulachki)
EK(ustanovit_detal)
EK(podrezat_torets)
EK(tochit_poverhnosti)
EK(sverlit_otverstie)
EK(tochit_kanavku)
EK(proverit_ramery_detali) EK(snyat_detal)).
```

VI. PRACTICAL IMPORTANCE OF WORK

Practical implementation of chains executive transitions is more efficiently implemented using functional programming. In this case, the programs are built from logically articulated definitions of functions. Definitions consists of organizing computations, control structures, and nested, often self-invoking (recurcive) function calls [18]. To do this, select Clojure, which is the dynamic programming language for the Java Virtual Machine (JVM). Clojure is a Lisp dialect that possesses the power of Lisp [19]. Clojure programs can use any Java library. Clojure runtime is also one of the most efficient and reliable Java runtimes in the world.

Writing programs for synthesizing operations based on linking knowledge elements to transitions into a single chain is possible only after converting XSD files of knowledge elements to the Java Class using the XJC compiler [20] in the Intellij IDEA environment. Next, create a separate file to import all classes. When building a chain of executive transitions from an EK, it is necessary to create functions for each element. The following are examples of EKs:

1. For an auxiliary transition “install and align part”

```
(def ustanovit_i_vyverit_detal [UstanovitIVyveritDetal
 stanovitIVyveritDetal $ SposobUstanovki
 UstanovitIVyveritDetal $ VremyaUstanovki
 UstanovitIVyveritDetal $ KodPrisposobleniyaZazhimnoe
 UstanovitIVyveritDetal $ KodPrisposobleniyaIzmeritelnogo
 UstanovitIVyveritDetal $ PrecisionVyverit
 UstanovitIVyveritDetal $ ModelStanka
 UstanovitIVyveritDetal $ TipDetal]).
```

2. For the working transition "Sharpen contour"

```
(def tochit_kontur [TochitKontur
 TochitKontur $ ModelStanka
 TochitKontur $ OborotyShpindelya
 TochitKontur $ TopPlastiny
 TochitKontur $ MarkaSplavaPlastiny
 TochitKontur $ VidPoverhnosti
 TochitKontur $ Podacha
 TochitKontur $ GlubinaRezaniya
 TochitKontur $ MarkaMateriala
 TochitKontur $ TermicheskayaObrabotka
 TochitKontur $ CoordinatesKontura
 TochitKontur $ TipDergavkiTochitKontur])
```

To implement the digital transformation of the technological preparation of mechanical processing with the application of the intelligent approach, a methodology for the formation of a prototype of the technological process has been developed. The technique includes the following steps:

1. To form a library of elements of knowledge about technological transitions used equipment, cutting and measuring tools and devices used for the group of parts being processed.
2. Highlight the defining attributes of the drawing of the part and the drawing of the workpiece from the point of view of selecting the necessary elements of knowledge about technological transitions and typical functions for processing a given part.
3. Build a graph of the route data for processing a given part with the definition of the vertices of the workplaces and states of the workpiece and arcs of forming operations.
4. Specify the data at the vertices of the data graph of the processing route.
5. Select the typical functions of processing a group of parts from the database, built from elements of knowledge about technological transitions.
6. Synthesize part processing operations using typical part processing functions and necessary elements of technological transition knowledge, considering the satisfaction of quantitative and qualitative indicators of part drawing attributes.

For details such as "body rotation", examples of the formation of typical functions for performing operations are shown:

Et <turning> ⇒ (E) (Esetting) (Esurface) (Eblunt sharp edges);

Ebor <boring> ⇒ (Esetting) (Ebutt end) (Esurface) (Echeck dimensions) (Epart removal);

Efturn <finishing turning parts> ⇒ (Esetting) (Ebutt end) (Esurface) (Eslot) (Epart removal);

Ecfturn <countour finishing turning parts> ⇒ (Esetting) (Ebutt end) (Econtour) (Eslot) (Epart removal);

E <post-processing> ⇒ (Esetting) (Ecent) (Edrill) (Esink) (Eexpand) (Epart removal).

As a result, the operation is formed from an EZ of technological transitions in the Clojure language, which has the following form:

```
(ns main.clojure.operations.oper030
   (: require [main.clojure.elements.tokarnayaobrabotka: as tok]
 [main.clojure.elements.vspomogatelnyeperekhody: as vs]))
(def oper_030 [vs / ustanovit_detal install part
              tok / tochit_kontur sharpen contour
              vs / Perezakrepit_detal refill part
              tok / podrezat_torec trim the butt
              tok / rastochit_poverhnost bore surface
              tok / pritupit_ostrye_kromki to blunt sharp edges
              vs / proverit_razmery_detali check dimensions
              vs / snyat_detal]) remove detail
(seq oper_030) call the chain.
```

VII. Conclusion

1. Digital transformation of the machining process leads to the formation of a controlled technological process and will allow to create an information environment of the intelligent level of the mechanical machining process control.

2. The use of a graph data model of a technological process will allow achieving higher performance when working with interrelated data, compared to relational databases.

3. The ability to expand graphs means that you can add new types of relationships, new nodes and new subgraphs to the existing structure, without breaking existing queries. This has a positive effect on the productivity of the development of mechanical processing technology and reduces the risks to the project.

4. For the effective use of knowledge of technological transitions proposed the use of functional programming, which allows to increase the efficiency of the synthesis of technological operations.

References

[1] V.V. Bogdanov, “Formirovanie bazy znanij dlya SAPR TP na osnove baz dannyh tehnologicheskoho naznacheniya” [Formation of a knowledge base for CAD TP on the basis of technological purpose databases], Izv. vuzov. Priborostroenie, vol. 53, no 6, pp. 47-50, 2010.

[2] E.I.Yablochnikov, V.I. Molochnik and A.A. Solomatina, “Kompleksnoe ispolzovanie baz znanij v avtomatizirovannyh sistemah tehnologicheskoy podgotovki proizvodstva” [Complex use of knowledge bases in automated systems for technological preparation of production], Izv. vuzov. Priborostroenie, vol. 53, no 6, pp. 51-54, 2010.

[3] G.B. Evgenev, B.V. Kuzmin and V.I. Rubahina, “Metody i sredstva upravleniya zhiznennym ciklom izdelij mashinostroeniya” [Methods and tools for managing the lifecycle of engineering products], Sistemy upravleniya, svyazi i bezopasnosti, no 4, pp. 198-216, 2015.

[4] Y.M. Berliner, O.V. Taratynov, “SAPR tekhnologa mashinostroitelya: uchebnik” [CAD technologist machine-builder tutorial], FORUM: INFRA-M, 336 p, 2015.

[5] Artificial Intelligence. A Modern Approach. Stuart J. Russel and Peter Norvig. Pearson Education, Inc., Pages 1152, 2003.

[6] C.O. Emmanuel, Y.A. Adekunle, “Basic Concepts of Expert System Shells and an Efficient Model for Knowledge Acquisition,” International Journal of Science and Research. (electronic journal), vol. 2 Issue 4, April 2013, pp 554-559. URL: www.ijsr.net.

[7] A.G. Lyutov, Y.V. Ryabov, “Model kompleksnogo avtomati-zirovannogo upravleniya processami v mashinostroenii” [Model of integrated automated process control in engineering], STIN, no 5. pp. 2-8, 2016.

[8] I. Robinson, J. Webber, and E. Eifrem, Graph Databases. O’Reilly Media, Incorporated, 2013.

[9] A.G. Lyutov, Y.V. Ryabov, R.I.Shaydullin, I.I. Shambazov, “Intellektual'noe upravlenie processami tekhnologicheskoj podgotovki mashinostroitelnogo proizvodstva” [Intelligent management of technological preparation of machine-building production], Vestnik YUrGU. Seriya «Komp'yuternye tekhnologii, upravlenie, radioehlektronika», vol.17, no 3, pp.117 – 124, 2017.

[10] https://tinkerpop.apache.org (accessed 04 October 2017).

[11] Martin Fauler, Pramodkumar Dzh. Sadaladzh. “NoSQL: novaya metodologiya razrabotki nerelyacionnyh baz dannyh” [NoSQL: new methodology for developing non-relational databases], Trans. to rus. : OOO “I.D. Vilyams”, 192 p, 2017.

[12] http://s3.thinkaurelius.com/docs/titan/1.0.0/ archoverview.html Titan:db (accessed 12 January 2018).

[13] https://www.revulytics.com/blog/setting-up-titan-1-0-apache-hbase Setting up Titan 1.0 with Apache Hbase (accessed 12 January 2018).

[14] https://www.datastax.com/products/datastax-enterprise-graph DataStax Enterprise Graph (accessed 22 January 2018).

[15] https://www.liquid-technologies.com/xml-studio Liquid Studio (accessed 15 June 2018).

[16] http://windows.en.all-softwares.com/Liquid-XML-Studio-2014/4-654-362_527.php Liquid XML Studio 2014. (accessed 17 June 2018).

[17] A.G. Lyutov, Y.V. Ryabov, “Sintez tekhnologicheskogo proces-sa dlya robotizirovannogo kompleksa na osnove znanij” [Synthesis of technological process for a robotic complex based on knowledge], Mekhatronika, avtomatizaciya, upravlenie., vol. 18, no 10, pp. 660-664, 2017.

[18] E.H. Hyuvyonen, J. Seppyanen “Mir Lispa. Tom 1. Vvedenie v yazyk Lisp i funkcional'noe programmirovanie” [World Lisp. Volume 1. Introduction to Lisp and Functional Programming], Trans. To rus. Mir, 1990.

[19] C. Ehmerik, B. Karper, K. Grand “Programmirovanie na Clojure” [Clojure Programming] Trans. A. N. Kiseleva, DMK Press, 816 p, 2013.

[20] https://docs.oracle.com/javase/8/docs/technotes/tools/unix/xjc.html Java platform, Standart Edition Tool Reference. (accessed 06 June 2018).