Journal of Engineering and Applied Sciences 12 (Special Issue 6): 7859-7864, 2017
ISSN: 1816-949X

The Language of the Description of the Functional Objects

Alexander G. Zhikharev, Sergey I. Matorin, Irina A. Kuprieva,
Esen B. Kulumbaev and Stanislava B. Smirnova
Belgorod State National Research University, Pobedy St., 85, 308015 Belgorod, Russia

Abstract: The study deals with the conceptual basis of the calculus of functional objects as a formal theory of systems. The basic concepts and definitions of the calculus of functional objects are presented within their framework the functional object is considered as a system described in terms of the system-object approach "unit-function-object". Formal bases for describing the methods of nodal objects within the framework of the supposed calculus are considered, basic constructions of the language for describing functional nodes are proposed.

Key words: System-object modeling, system, system approach, representation of semantic dependencies, simulation modeling, system functioning, system-object method of knowledge representation

INTRODUCTION

Let's consider the basic conceptual positions of the system approach which will later take into account the formal constructions of the calculus underlying the developed formal system theory. First, the system is considered as a functional object, the function of which is due to the function of an object of a higher layer (i.e., the supersystem) (Melnikov, 1978). Second, any system is necessarily connected with other systems and these links represent flows of deep-layer elements of related systems (Melnikov, 1978). At the same time, the interconnections these systems are functional, the connections between the subsystems of this system are supportive. Last, the consequence of the above given definition of the system and understanding of the relationship between systems is the representation of the system in the form of a triune construct "Unit-Function Object" (UFO element) (Matorin *et al*., 2005) where:

- Unit-a structural element of the supersystem in the form of a cross-connection of this system with other systems
- Function-the dynamic (functional) element of the supersystem which performs a certain role from the point of view of maintaining the supersystem by balancing the links of this node
- Object is a substantial element of a supersystem that realizes a given function in the form of some material formation possessing constructive, operational characteristics, etc.

Literature review: Previously, the authors carried out studies on the formalization of the system-object approach using the theory of the Grenander patterns (Milner *et al*., 1989) and the calculation of Milner processes (Grenander, 1979). However, it was not possible to get a full description of the systems as unit-function-object elements with their help. At the moment, the most promising for the formalization of the UFO approach are the ideas embedded in the calculation of the objects by Martin and Cardelli (1996). The understanding and formulation of an abstract object in this calculus allowed us to propose a formal description of the UFO element as a special unit object as well as a formal description of the connection as a special streaming object (Zhikharev *et al*., 2013, 2015a, b; Matorin *et al*., 2017), when developing the System-Object Knowledge Representation Method (SOKRM). These formalisms are used by us further in constructing the calculus of functional objects, i.e., calculation of systems as UV-elements.

MATERIALS AND METHODS

In order to establish the above mentioned calculus a set of stream objects L corresponding to the set of constraints of the system in introduced into consideration:

$$L = \{l_1, l_2, ..., l_i, ..., l_n\} \quad (1)$$

where, n is the number of stream objects (links of the system). Each nth element of the set L is a special stream

Corresponding Author: Alexander G. Zhikharev, Belgorod State National Research University, Pobedy St., 85, 308015 Belgorod, Russia

object (corresponding to the specific connection of the system) which in accordance with the theory of the Abbadi-Cardelli objects, consists of fields does not include methods and has the following form:

$$l_n = \left(r^1,\ r^2,\ \dots,\ r^k\right) \quad (2)$$

where, $l_n \in L$; k is the number of fields of the streaming object l_n; r^1, r^2, ..., r^k are fields of the streaming object which are the "identifier-meaning" couple. In further formulations when referring to the stream and node objects fields we use the standard entry of the form $l_n.r^1$. The set L goes as follows:

$$L = \left\{ \begin{matrix} l_1 = \left(r_1^1,\ r_1^2,\ \dots,\ r_1^{k1}\right),\ l_2 = \left(r_2^1,\ r_2^2,\ \dots,\ r_2^{k2}\right),\ \dots, \\ l_n = \left(r_n^1,\ r_n^2, \dots,\ r_n^{kn}\right) \end{matrix} \right\} \quad (3)$$

where, the lower indices of the fields r represent the number of the streaming object that is the parent and the upper indices of the fields r is the ordinal number of the field within the parent stream object and k_n is the number of fields of the streaming object l_n. We denote the set of fields of the streaming object l_n of the variable R_n, then:

$$R_n = \left\{ r_n^{kn} \mid r_n^{kn} = \left(\text{identificator, meaning}\right)\right\} \quad (4)$$

So, "L" is set flow objects (system relationships) can be defined like:

$$L = \left\{l_n \middle| l_n = \left(R_n\right)\right\} \quad (5)$$

According to main UFO methodology statements, basic relationships hierarchy in the case-flow object hierarchy, occurs within any graphoanalitical model. Besides, basic hierarchy has set of such kinds of predefined relationships:

- Relationships class V, through which material resources "flow"
- Relationships class E, through which energy resources "flow"
- Relationships class D, through which data is transported
- Relationships class C, through which controlling information is transported

In appliance with abovementioned let's add, according to graphoanalitical UFO-Model basic hierarchy such predefined flow objects into major part of L-system flow objects as:

- l^v; parent flow object, representing material objects class
- l^e; parent flow object, representing energetical objects class
- l^d; parent flow object, representing data objects class
- l^c; parent flow object, representing controlling data objects class

The above basic flow objects have no fields consequently they are represented in such way:

$$l = () \quad (6)$$

Other L-set elements that aren't basic are their descendants besides every descendant flow object inherits parent's flow objects field. Then, let's denominate other L-set elements belonging to some class like l^k where the upper index k-represents flow object class and respectively can have such values as: v, e, d, c. Suchwise, for particular system model in functional object calculus terminology a set of "L" flow objects can be viewed in hierarchical way. Next let's pay attention to the set of node "S" objects that corresponds to most UFO elements system, according to main statements of SOMP3 (Zhikharev *et al*., 2013):

$$S = \{s_1, s_2, \dots, s_j, \dots, s_j\} \quad (7)$$

where, j is amount of node objects (systems) every j-e element from the "S"-set is a special node object (corresponding a particular system/UFO element) that according to Abadi-Cardelli's theory of objects consists of fields and methods and has such view:

$$s_j = \left(U,\ F,\ O\right)$$

where, U is a set of fields for interface flow objects of node objects description s_j that correspond to most functional relationships of a particular system.

Set $U = L_? \cup L_!$, where $L_?$ is a set of ingoing interface flow objects, corresponding to outgoing relationships of a system. Flow objects indexes "?" and "!" in work are used like ingoing definition "?" and outgoing "!" flow object in relations to node object (Fig. 1):

$$\text{With: } L_? \subset L;\ \subset L! \subset L$$

f is is node object method s_j, describing function, transforming ingoing interface flow objects (ingoing system relationships) $L_?$ to outgoing $L_!$. Then, let's represent method of node object like:

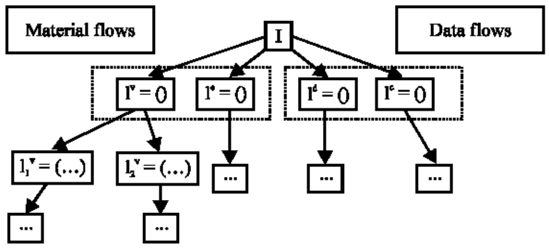

Fig. 1: Set of system flow objects in hierarchical view

$$f(L_?)L_! \quad (9)$$

where, f is method node object (system function) with the area for defining L and the area for $L_!$ values, respectively. O-set of fields for description of node object (system) s_j characteristics which elements have the following format:

$$O = \{o_i \mid o_i = (\text{identificator, meaning})\} \quad (10)$$

where, i is an amount of node object fields s_j. Set of fields for description of object system characteristics consist of 3 sub-sets:

$$O = O_? \cup O_! \cup O_f \quad (11)$$

$O_?$ field set contains interface ingoing characteristics of node object. For each field of each flow object there is a corresponding type. Suchwise of set of ingoing flow objects consist of 1 element (flow object) and set of fields of incoming object consists of 2 elements represented in such way:

$$L_? = \{l_1 = (r_1, r_2)\} \quad (12)$$

In this way-corresponding set $O_?$ will be represented as:

$$O_? = \{o_1, o_2\} \quad (13)$$

$O_?$ set output will depend on the amount of ingoing interface flow objects and and their fields. If the set-output:

$$|L_?| = n \quad (14)$$

and output of ingoing flow objects:

$$|L_?^1| = m_1, |L_?^2| = m_2, |L_?^n| = m_n \quad (15)$$

Then, output of corresponding set of object $O_?$ interface characteristics is:

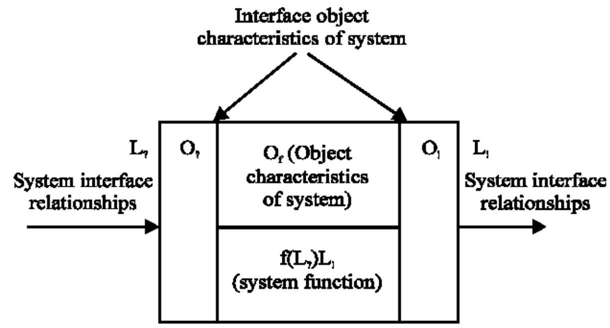

Fig. 2: Graphical formalism of system as UFO-element

$$|O_?| = \sum_1^n \left|{}^n_?\right| \quad (16)$$

Output of set $O_!$ (corresponds to outgoing interface flow objects), similarly with Eq. 15 is calculated by the following in Eq. 16:

$$|O_!| = \sum_1^n \left|{}^n_?\right| \quad (17)$$

Set O_f contains object characteristics of system, inherent to the object that implements function and their amount depends on particular system: Suchwise, within system calculating system described in Eq. 7 is represented like:

$$s_j = (L_?,\ L_!;\ f(L_?)L_!;\ O_?,\ O_!,\ O_f \quad (18)$$

Graphical representation of this Eq. 17 is displayed on Fig. 2. We'll consider such view to be graphical formalism, similarly to graphical formalism, it is a forming element in Grenanders pattern theory. This non-derivative object will be an elementary data carrier in our calculation. Let, pay attention to formal basis of modeling system functioning.

As it was said before, system functional features nenoaiu are represented in formula 18 by function of node object $f(L_?)L_!$ that describes process of transforming ingoing flow objects to outgoing. Let's look at the general case of system function description. As it was said before, any modeling system represents or will represent some part of a real world with considering it's interactions with other systems, targets of it's existence, etc. and a respectively exists in time. That being the case, let's imagine system model, represented in formula 8 in the following way:

$$s_j(t) = (U(t),\ f(t),\ O(t)$$

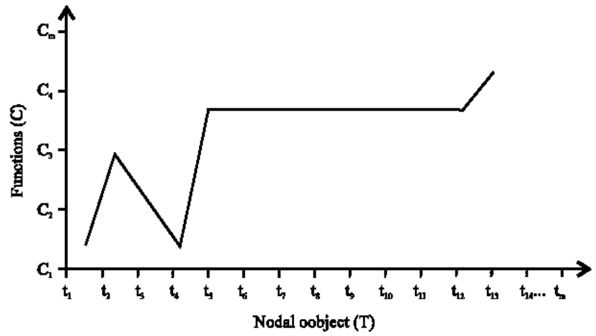

Fig. 3: The law of nodal object functioning diagram

System as flow object represents a function of time t in the presented expression. We denoted such components of system as a node-intersection of inlinks and outlinks denoted as U(t); system function as f(t) and an object that implements the function as O(t). The meaning of these notations is that the system can change taking into account changes of all its components with time. Front end flow objects of system (its bundle and structure); function of system depending on supersystem constraint and the object realizing system function can change.

RESULTS AND DISCUSSION

Introduce in the explore calculus the term "state" of system. In case of model explore introduce the term state of nodal object. The state of nodal object s_j is a bundle of fixed values of front-end flow objects fields $L_?$ and $L_!$ at any given point in time t. In this case, the process of system functioning s_j can be represented as a process of changing system state at different points in time t_1, t_2, ... Denote all possible states set of the system as C and get the law:

$$C = f(t) \tag{20}$$

where, f is a function of explored nodal object. The expression (Eq. 20) we will understand as the law of nodal object functioning, positing at any given point in time t_n, the system which model represents a nodal object, accepts the state c_n as a result of its functioning. Graphically, the law of nodal object functioning can be represented as a diagram as shown in Fig. 3.

Time segments are on X-line, all possible system conditions are on Y-line. By so doing, a diagram of system conditions change is the diagram of system functioning. And the main aim which allow to construct a valid models of functional system is a problem of expression modeling (Eq. 20), videlicet function f.

If the bundle of input stream objects $L_? = (l_?1, l_?2, \ldots, l_?n)$ and the bundle of output stream objects $L_! = (l!1, l!2, \ldots, l!m)$ then we write nodal object function with the following expression:

$$f(L_?, t)L_! = \begin{cases} l_{!1}.r^* : l_{!1}.r^* = \varphi_1(L_?);\ delay = t_1; \\ l_{!2}.r^* : l_{!2}.r^* = \varphi_2(L_?);\ delay = t_2; \\ l_{!m}.r^* : l_{!m}.r^* = \varphi_2(L_?);\ delay = t_m \end{cases} \tag{21}$$

where, the delay operator simulates a time delay for converting an input to an output of duration t_m and the function φ_m is a single-valued mathematical dependence of the field of the output streaming object $l_{!1}.r^*$ of the set of input values $L_?$. Consider for example an abstract system that has a streaming object with one field $l_? = (x)$ and an output stream object with one field $l_! = (y)$ as the input streaming object at the same time we take for the condition that the field of the outgoing streaming object depends on the input according to the following law:

$$l_!.y = a + l_?.x^b \tag{22}$$

and the time t = 2*a a is spent for one iteration procedure where the constant parameters a and b represent the object characteristics of the system, then the method of the nodal object can be written in the form:

$$f(l_?)l_! = \{l_1 = 600, r_2\, 2.5, r_3 = 10^5); l_2 = (r_1)) \tag{23}$$

Moreover, the delay time, as it can be seen from expressions Eq. 22 and 23, can also have a certain functional dependence, for example, on the object characteristics of the node. If consider an example of modeling an existing system. As a system, we will consider the thermoforming machine HSC-660A, intended for the production of plastic containers. The characteristics of this technical device are as follows: width of the plastic liner is 600 mm; thickness is 2.5 mm; productivity is 34 cycles per minute; length of the consumed cloth for one cycle is 250 mm, the number of plastic cups in one cycle is 12. These characteristics indicate that for one minute the thermoforming machine of the model under consideration squeezes 34×12 = 408 plastic cups and consumes 250×34 = 8500 mm of the plastic cloth. For this case we should impose to consideration two material flow objects:

$$L = \left(l_1 = \left(r_1 = 600,\ r_2 = 2.5,\ r_3 = 10^5\right);\ l_2 = \left(r_1 = 0\right)\right) \tag{24}$$

where, l_1^v is flow object which is the foil roll $l_1^v.r_1$ = 600 mm wide, $l_1^v.r_2$ = 2.5 mm thick $l_1^v.r_3$ = 100 m length; l_2^v flow object which is end product-plastic cups with one field $l_2^v.r_1$ is plastic cups quantity, initial value equals zero. Further, we should impose in the model the following knot object:

$$S = \left(s_1 = \left(L_?,\ L_!;\ f\left(L_?\right)L_!;\ O_?,\ O_!,\ O_f\right)\right) \tag{25}$$

where, $L_? = (l_1)$; $L_! = (l_2)$; $O_? = \varnothing$; $O_! = \varnothing$; $O_f = (O_{f1} = 250, O_{f2} = 12, O_{f3} = 34)$. The set of object characteristics O_f includes aggregate technical merits the length of plastic bed for one cycle, the quantity of made cups per one work cycle and productivity (quantity per minute), respectively. Let's see into knot object method in detail. In accordance with value expression (Eq. 21), the knot object method will be represented the following way:

$$f(L_?, t)L_! = \{l_2.r_1 : l_2.r_1 = \varphi_1(l_1); \text{delay} = t \tag{26}$$

In accordance with technical characteristics of under discussion aggregate, you can see that there is the following correspondence between entry and output:

$$L_2.r_1 = \frac{O_{f2} \times l_1.r_3}{O_{f1}} \tag{27}$$

This correspondence is valid under the conditions: $l_1.r_1$ = 600 and $l_1.r_2$ = 2.5. The conditions estimate the foil roll of initial stock to be corresponding to technical characteristics of the reforming machine. The time in minutes spent for stamping $l_2.r_1$ cups will be equal:

$$l_2.r_1 = \frac{O_{f1} \times l_1.r_3}{O_{f1}} \tag{28}$$

As a result, knot object method takes the following form:

$$\begin{cases} l_2 : r_1.l_2\, r_1 = \dfrac{f(L_?, t)L_! O_{f2} \times l_1.r_3}{O_{f1}} = \text{delay} = \dfrac{l_2.r_1}{O_{f2} \times O_{f3}} \\ \text{upon condition: } l_1.r_1 = 600;\ l_1.r_2 = 2.5 \end{cases} \tag{29}$$

Considered example demonstrate the use of functional object calculation for imitation of technical systems functionality, certainly in the capacity of knot object there could be systems and processes of another type, for example, organizational-business processes, concepts semantic nets, neural networks, physical processes, etc.

The language possibilities of functional nodes description and functional relations of inputs from outputs determine classes of possible objects in analyzed system. In accordance with the key points of system-object approach "unit-function-object" can be determined four classes of system functional relations of inputs from outputs by the number of parameters.

System s_j is the object which is taking a unit with one input and one output and realizing transform function of one variable. Unit object method in this case is given by:

$$f(L_?, t)L_! = \{l_!.r = \varphi(l_?); \text{delay} = t \tag{30}$$

where, $|L_?| = 1$, $|L_!| = 1$. System s_j is the object which is taking a unit with several inputs and one outputs and realizing transform function of several variable. Unit object method in this case is given by:

$$f(L_?, t)\ L_! = \{l_!.r = \varphi(l_{?1}, ..., l_?n); \text{delay} = t \tag{31}$$

where, $|L_?| = n$, $|L_!| = 1$. System s_j is the object which is taking a node with one input and several outputs. Typically, the function which realizes such object, consists of a set of subfunctions. Unit object method in this case is given by:

$$f(L_?, t)L_! = \begin{cases} l_{!1}.r: l_{!1}.r = \varphi_1(l_?); \text{delay} = t_1; \\ l_{!n}.r: l_{!n}.r = \varphi_n(l_?); \text{delay} = t_n \end{cases} \tag{32}$$

where, $|L_?|=1$, $|L_!| = n$. System s_j is the object which is taking a unit with several inputs and several outputs. Unit object method in this case is given by:

$$f(L_?, t)L_! = \begin{cases} l_{!1}.r: l_{!1}.r = \varphi_1(l_{?1}, ..., l_{?m}); \text{delay} = t_1; \\ l_{!n}.r: l_{!n}.r = \varphi_n(l_{?1}, ..., l_{?m}); \text{delay} = t_n \end{cases} \tag{33}$$

where, $|L_?| = m$, $|L_!| = n$. In the description of unit object methods (Eq. 9), corresponding to the second and the third type, generally it is necessary to conduct additional unit decomposing till then multiplicity of the set of output flow objects will not be equal to one.

At the same time, in some cases this rule can be ignored. For example, in case the dependency of the output aspects from the input is not complex, it can be described through language descriptions of functional units (Zhikharev *et al*., 2016).

Language of functional units description should have a set of features for defined description of functional relationships of unit objects methods and for simulation of systems functioning:

- Opportunity to read and write some properties of interface flow objects of unit object
- Opportunity to read and write object characteristics of nodal object

- Opportunity to organize time delay of function execute
- Opportunity to organize mathematical calculation
- Opportunity to visualize intermediate values of used variable

CONCLUSION

The possibility of formalizing the system-object approach "unit-function-object" and the system-object knowledge representation method based on it is investigated. The expediency of using for this calculation the objects by Martin and Cardelli (1996) and some ideas of the theory of Grenander's patterns is shown. In terms of the above calculus, a special object is formulated that represents the system as an UFO-element and the corresponding graphical representation.

RECOMMENDATIONS

With the help of the presented elementary operations, it is possible to build more complex operators for processing streaming and node objects within the framework of this calculus. In the future such operators will allow to build algorithms for automatic design of system models in the form of combinations of unit and stream objects. In addition, the obtained results show the expediency of constructing a formalized system theory by expanding and improving the calculation of special objects as systems within the framework of the UFO-approach. One of the key areas of further development of the calculus of functional objects is the development of semantic means of the language for describing functional objects, since, it is the method of the nodal object that determines the logic of the transformation of input streaming objects into the output. The developed formal device will allow to uniquely determine the functioning of the system and as a result, it will be possible to build system models reflecting not only the system's static indicators but also dynamic, determining the states of the system, changing states in time and so on.

ACKNOWLEDGEMENTS

Researchers are supported by grants of the Russian Foundation for Basic Research: 16-07-00460-a and 16-07-00193-a and the President's Grant for the state support of young Russian researchers of PhD of the Russian Federation MD-2687.2017.6.

REFERENCES

Grenander, U., 1979. Lectures on the Theory of Images. Springer, Berlin, Germany, Pages: 384.

Martin, A. and L. Cardelli, 1996. A Theory of Objects. Springer, Berlin, Germany, Pages: 397.

Matorin, S.I., A.S. Popov and V.S. Matorin, 2005. Organization simulation technology in the light of a new unit-function-object approach. Autom. Doc. Math. Ling., 39: 1-8.

Matorin, S.I., O.A. Zimovets and A.G. Zhikharev, 2017. System-wide principles in terms of the system-object approach node-function-object. Works ISA. RAS., 66: 10-17.

Melnikov, G.P., 1978. Systemology and Language Aspects of Cybernetics. Sov. Radio Publisher, Moscow, Russia, Pages: 368.

Milner, R., J. Parrow and D.A. Walker, 1989. Calculus of mobile processes-part I: LFCS report 89-85. Master Thesis, University of Edinburgh, Edinburgh, Scotland.

Zhikharev, A.G., S.I. Matorin and N.O. Zaitseva, 2015a. System-object toolkit for simulation of technological processes and transport streams. Artif. Intell. Decis. Making, 4: 95-103.

Zhikharev, A.G., S.I. Matorin and N.O. Zaitseva, 2015b. About perspectives of simulation technological processes functioning with using system-object approach node-function-object. Intl. J. Appl. Eng. Res., 10: 31363-31370.

Zhikharev, A.G., S.I. Matorin, E.M. Mamatov and N.N. Smorodina, 2013. About the system-object method of knowledge representation. Sci. Statements BelSU. Ser. Comput. Sci., 8: 137-146.

Zhikharev, A.G., S.I. Matorin, O.A. Zimovets, M.S. Zhikhareva and V.I. Rakov, 2016. The simulation modeling of systems taking into account their internal parameters change. Intl. J. Pharm. Technol., 8: 26933-26945.