Identification of the Control Object Dynamics by Symbolic Regression

Askhat Diveev[1] and Igor Prokopiev[2]

Abstract— **The problem of building a more accurate mathematical model of the control object in order to solve the problem of navigation over a certain time interval in the absence of signals from global positioning systems is considered. To solve the problem, it is assumed that there are additional results of an experiment in which control actions on an object were measured in order to ensure its movement along a complex trajectory and the position of the control object as its reaction to these control actions. To clarify the model of the control object, the identification problem is solved, in which it is necessary to find the right parts of the differential equations describing the dynamics of the movement of the object, i.e. its acceleration along all three geometric axes. The peculiarity of the work lies in the fact that to approximate the right parts of differential equations, a symbolic regression method is used, which by machine learning finds mathematical expressions in encoded form. A description of the method of symbolic regression of universal code is presented. The description describes in detail the small variations of the universal code that are used in finding the optimal solution. An example of identification of a quadcopter model is given from experimental data obtained when an object moves along complex closed trajectory.**

I. INTRODUCTION

The identification problem of a mathematical model of a control object, because with the increasing complexity of robot control objects is today relevant and requires the development of special solution methods. Typically, physical mathematical models of objects or mathematical models of objects recorded up to parameter values [1] are used to solve the identification problem. The most difficult is non-parametric identification [2], when it is necessary to find not only the parameters of the model, but also its structure. An accurate model is necessary for solving various tasks, including the optimal control problem [3].

In this work, symbolic regression is used to structurally identify and find the mathematical model of the control object. The problem of autonomous robotization is the most demanded and most urgent problem today, the solution of which should obviously greatly advance the development of robotics [4] and bring long-awaited breakthrough results of its application. Without solving the problems of autonomous robotization, all today's achievements in the field of robotics represent minor improvements in the results obtained in the seventies of the last century mainly by Soviet scientists when creating a great lunar rover, which for a year plowed the expanses of an uninhabited planet, albeit under human control, but located at a distance of 400.000 kilometers from the control object. The historical fact of the existence of the Soviet lunar rover is indisputable, unlike the flight to the moon of American astronauts, since the work of the lunar rover was tightly controlled by American colleagues, who even shared information they knew about the lunar surface and took part in the control of the lunar rover.

One of the main problems of autonomous robotization is the robot's determination of its location in the absence of global positioning systems. The movement of the robot in the presence of high-precision positioning can hardly be called autonomous because it is movement according to a pre-written program with position correction based on the results of processing signals received from positioning systems. It should be noted here that global positioning systems so far make it possible to receive signals depending on the terrain with an accuracy of 5 meters or worse, which is often not enough for the robot to complete the task. In this case, additional navigation is required.

This work is devoted to solving the navigation problem based on the use of a high-precision internal mathematical model. This model can be built on the basis of modern computational methods and installed on the on-board processor of the robot at the stage of its full-scale tests. It should be noted that the presence of an internal model of the robot does not interfere with its work and does not in any way deny the use of other navigation systems, inertial, odometrical sensors, mapping systems, terrain orientation, the use of video cameras with trained neural networks that determine the range of movement and rotation angles by video signal.

The internal model, in general, if it allows obtaining the necessary information with satisfactory accuracy, can be the central unit of the navigation system, in which all signals coming from various devices used for navigation are concentrated and processed. On the basis of the internal model, on the one hand, it is possible to assess the accuracy of the incoming information, in particular, the substitution of information by the enemy when using global positioning systems, on the other hand, to clarify the model itself on the basis of reliable signals from sensors. Refining the model during the functioning of the robot is the process of learning it, i.e. an obvious element of artificial intelligence. As a result, based on the processing of signals from the model, the robot can build further paths of movement to perform the task assigned to it.

In this paper, a symbolic regression method [5] is used to build an internal model. This computational tool, like

[1]Askhat Diveev is chief researcher of Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, 119333, Vavilov Str., 44/2 FRC CS RAS, and RUDN University, Russia aidiveev@mail.ru

[2]Igor Prokopiev is leading researcher of Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, 119333, Vavilov Str., 44/2 FRC CS RAS, Russia fvi2014@list.ru

artificial neural networks, allows you to approximate any complex function, but unlike neural networks, symbolic regression presents the researcher with a mathematical expression of the desired function. The difficulty in using symbolic regression is that. that the search for an optimal mathematical expression is carried out on the space of function codes, the metric in which is not directly related to the proximity of the codes. Arithmetic operations cannot be performed on the code space, in particular, taking a gradient, calculating the Euclidean distance between the codes. Here it is more efficient to search in the vicinity of one mathematical expression, the basic one, and change the basis in to a better solution in the search process.

II. PROBLEM STATEMENT

Let there be data from experiments conducted with a robotic device. In particular, there are many control programs are defined in the form of piecewise linear control dependencies on time

$$\mathrm{U}^* = \{\mathbf{u}^{*,1}(\cdot), \dots, \mathbf{u}^{*,M}(\cdot)\}, \tag{1}$$

where $\mathbf{u}^{*,i}(\cdot) = [u_1^{*,i}(\cdot) \dots u_m^{*,i}(\cdot)]^{\mathrm{T}}$ is a vector function of program control,

$$\mathbf{u}^{*,i}(\cdot) = \mathbf{u}^{*,i}(t),\ 0 \leq t \leq t_{f,i},$$

where $t_{f,i}$ is a time of program control i, $i = 1, \dots, M$.

Program trajectories of the control object are given, which are obtained as a result of application of the control program to the robotic device

$$\mathrm{X}^* = \{\mathbf{x}^{*,1}(\cdot), \dots, \mathbf{x}^{*,M}(\cdot)\}. \tag{2}$$

where

$$\mathbf{x}^{*,i}(\cdot) = [x_1^{*,1_1}(\cdot) \dots x_n^{*,i}(\cdot)]^{\mathrm{T}}$$

is a vector function of motion paths of all components of the vector of states of the control object.

It is necessary to find a mathematical model of the control object in the form of a system of ordinary differential equations

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}), \tag{3}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$.

The system (4) should ensure obtaining minimum value of the following criterion

$$J = \sum_{i=1}^{M} \int_0^{t_{f,i}} \|\mathbf{x}^{*,i} - \mathbf{x}(t, \mathbf{x}^{0,i})\| dt \to \min_{\mathbf{f}(\mathbf{x},\mathbf{u})}. \tag{4}$$

where $\mathbf{x}(t, \mathbf{x}^{0,i})$ is a particular solution of the system (3) from the initial state $\mathbf{x}^{0,i} = \mathbf{x}^{*,i}(0)$.

The set of the program controls (5) and the program trajectories (2) is called a training set. We believe that the training was successful if for any program control not from in the training set

$$\tilde{\mathrm{U}}^* = \{\mathbf{u}^{*,M+1}(\cdot), \dots, \mathbf{u}^{*,M+K}(\cdot)\}, \tag{5}$$

the following condition is performing

$$J = \int_0^{t_{f,M+k}} \|\mathbf{x}^{*,M+k}(t) - \mathbf{x}(t, \mathbf{x}^{0,M+k})\| dt \leq \varepsilon. \tag{6}$$

where $\mathbf{x}^{*,M+k}(t)$ is a program trajectory of the control object, obtained by the program control $\mathbf{u}^{*,M+k}(t)$, $k \in \{1, \dots, K\}$. T+o constr+uct of the mathematical madel of control object in the form (3) the+ symbolic regression is used.

III. SYMBOLIC REGRESSION

Symbolic regression allows you to find mathematical expressions of formulas using evolutionary calculation. All symbolic regression methods encode mathematical expressions with a special code, and then search for the desired formula using a special genetic algorithm in which crossover and mutation operations are performed taking into account, the form of encoding. To encode a mathematical expression, an alphabet of elementary functions predetermined by the researcher is used. In this work, we use a relatively new method of symbolic regression, universal code [6], [7].

As an alphabet of elementary functions, we use functions without arguments, or arguments and parameters of the desired mathematical expression, functions with one argument and functions with two arguments.

Here is an example of encoding. Let a mathematical expression be given

$$y = \exp\left(-\sqrt{x_1^2 + x_2^2}\right) \cos(q_1 x_1 + q_2). \tag{7}$$

To code this the mathematical expression the following functions are enough.

Functions without arguments or arguments and parameters of the desired mathematical expression

$$\mathrm{F}_0 = \{f_{0,1} = x_1, f_{0,2} = x_2, f_{0,3} = q_1, f_{0,4} = q_2\}.$$

Functions with one argument

$$\begin{aligned} \mathrm{F}_1 \quad = \quad & \{f_{1,1}(z) = z, f_{1,2}(z) = z^2, f_{1,3}(z) = -z, \\ & f_{1,4}(z) = \exp(z), f_{1,5}(z) = \cos(z), \\ & f_{1,6}(z) = \sqrt{z}\}. \end{aligned}$$

Functions with two arguments

$$\mathrm{F}_2 = f_{2,1}(z_1, z_2) = z_1 + z_2, f_{2,2}(z_1, z_2) = z_1 \cdot z_2.$$

All base functions have twins lower index. The first number is a number of arguments, the second number is the function number in that set functions with the same number of arguments.

To code each basis function the integer vector with two components is used. Values components coincide with indexes of the function.

To code the mathematical expression (7) by base functions from the alphabet it is necessary to present it in codes of base functions. Let's rewrite the mathematical expression (7) with consider elements of the alphabet.

$$\begin{gathered}y=\exp\left(-\sqrt{x_1^2+x_2^2}\right)\cos(q_1x_1+q_2)=\\=f_{1,4}(-f_{1,6}(x_1^2+x_2^2))f_{1,5}(q_1x_1+q_2)=\\=f_{1,4}(f_{1,3}(f_{1,6}(f_{1,2}(x_1)+f_{1,2}(x_2))))f_{1,5}(q_1x_1+q_2)=\\=f_{2,2}(f_{1,4}(f_{1,3}(f_{1,6}(f_{2,1}(f_{1,2}(x_1),f_{1,2}(x_2))))),\\f_{1,5}(f_{2,1}(f_{2,2}(q_1,x_1),q_2))=\\=f_{2,2}(f_{1,4}(f_{1,3}(f_{1,6}(f_{2,1}(f_{1,2}(f_{0,1}),f_{1,2}(f_{0,2}))))),\\f_{1,5}(f_{2,1}(f_{2,2}(f_{0,3},f_{0,1}),f_{0,4})).\end{gathered}$$

Now we write all indexes functions sequentially as integer vectors with two components

$$\begin{aligned}\tilde{y} \;=\; &\left(\begin{bmatrix}2\\1\end{bmatrix},\begin{bmatrix}1\\4\end{bmatrix},\begin{bmatrix}1\\3\end{bmatrix},\begin{bmatrix}1\\6\end{bmatrix},\begin{bmatrix}2\\1\end{bmatrix},\begin{bmatrix}1\\2\end{bmatrix},\begin{bmatrix}0\\1\end{bmatrix},\begin{bmatrix}1\\2\end{bmatrix},\begin{bmatrix}0\\2\end{bmatrix},\right.\\&\left.\begin{bmatrix}1\\5\end{bmatrix},\begin{bmatrix}2\\1\end{bmatrix},\begin{bmatrix}2\\2\end{bmatrix},\begin{bmatrix}0\\3\end{bmatrix},\begin{bmatrix}0\\1\end{bmatrix},\begin{bmatrix}0\\4\end{bmatrix}\right).\end{aligned} \tag{8}$$

In common case a code of mathematical expression has N two components integer vectors

$$\tilde{y}=\left(\begin{bmatrix}a_1(1)\\a_2(1)\end{bmatrix}\begin{bmatrix}a_1(2)\\a_2(2)\end{bmatrix}\cdots\begin{bmatrix}a_1(k)\\a_2(k)\end{bmatrix}\cdots\begin{bmatrix}a_1(N)\\a_2(N)\end{bmatrix}\right). \tag{9}$$

To define correct of code the index of code symbol is used. The index of code symbol shows, how many codes must stay to right of the current code. The index of code symbol is calculated by equation

$$T(k)=T(k-1)+a_1(k)-1. \tag{10}$$

Correct code has the following values of the index code symbol

$$\begin{array}{rcl}T(0)&=&1,\\T(k)&\geq&0,k=1,\dots,N-1,\\T(N)&=&0.\end{array} \tag{11}$$

If the conditions (11) is fulfill, then the code is correct.

To calculate a mathematical expression by its code the following equation is used

$$\begin{gathered}j=0,\ k=N,\dots,1,\\\text{if } a_1(k)=0, \text{ then}\\j\leftarrow j+1,\ r_j\leftarrow s_{0,a_2(k)}\\\text{else, if } a_1(k)=1, \text{ then}\\r_j\leftarrow f_{1,a_2(k)}(r_j),\\\text{else, if } a_1(k)=2, \text{ then } j\leftarrow j-1,\\r_j\leftarrow f_{2,a_2(k)}(r_j,r_{j+1}).\end{gathered} \tag{12}$$

Result of calculations is stored in the variable r_1 at $k=1$.

To increase the complexity of the desired mathematical expression, the universal code uses vector notation. Each component of the vector of the universal code is the code of a separate mathematical expression. All component code lengths are limited and can contain no more than N symbols.

$$\mathrm{Y}=(\mathbf{y}_1,\dots,\mathbf{y}_L), \tag{13}$$

where

$$\mathbf{y}_i=\left(\begin{bmatrix}a_{i,1}(1)\\a_{i,2}(1)\end{bmatrix}\cdots.\begin{bmatrix}a_{i,1}(N)\\a_{i,2}(N)\end{bmatrix}\right), \tag{14}$$

$i=1,\dots,N$.

Real length of each component i is $N_i\leq N$. All codes of components are calculated sequence. After calculation of the component, its value is adding to the set argument F_0. Values of codes of components are calculated sequence. After calculation of the code component, its value is adding to the set of arguments F_0 (**??**) of the alphabet of elementary function as in the Cartesian genetic programming, This value can be used in next code components.

A vector of small variation for universal code has dimension 5

$$\mathbf{w}=[w_1\ w_2\ w_3\ w_4\ w_5]^T, \tag{15}$$

where w_1 is a type of variation, w_2 is the component number, w_3 is the symbol number, w_4, w_5 are addition parameters for small variation.

A vector of small variation has the following types.

$w_1=0$ is a change of the second component of symbol code

$$a_{w_2,2}(w_3)=w_4. \tag{16}$$

$w_1=1$ is an elimination of a code element.

$$\begin{gathered}\text{If } a_{w_2,1}(w_3)=w_4=1 \text{ , then}\\a_{w_2,1}(w_3+i-1)\leftarrow a_{w_2,1}(w_3+i),\\a_{w_2,2}(w_3+i-1)\leftarrow a_{w_2,2}(w_3+i),\\i=1,\dots,N_{w_2}-w_3-1,\\N_{w_2}\leftarrow N_{w_2}-1.\end{gathered} \tag{17}$$

$$\begin{gathered}\text{If } a_{w_2,1}(w_3)=w_4=2 \text{ , then}\\j=1,\ T=1,\\\text{while } T>0 \text{ do}\\T\leftarrow T+a_{w_2,1}(w_3+j)-1,\ j\leftarrow j+1,\\a_{w_2,1}(w_3+i-1)\leftarrow a_{w_2,1}(w_3+i+j),\\a_{w_2,2}(w_3+i-1)\leftarrow a_{w_2,2}(w_3+i+j),\\i=1,\dots,N_{w_2}-j,\\N_{w_2}\leftarrow N_{w_2}-j.\end{gathered} \tag{18}$$

$w_1=2$ is an insert of function with one argument.

$$\begin{gathered}\text{If } N_{w_1}<N \text{ , then}\\a_{w_2,1}(i)\leftarrow a_{w_2,1}(i-1),\\a_{w_2,2}(i)\leftarrow a_{w_2,2}(i-1),\\i=N_{w_2}+1,\dots,w_3+1,\\a_{w_2,1}(w_3)=1,\\a_{w_2,1}(w_3)=w_4,N_{w_2}\leftarrow N_{w_2}+1.\end{gathered} \tag{19}$$

$w_1=3$ is an insert of function with two arguments.

$$\begin{gathered}\text{If } N_{w_2}<N-1 \text{ , then}\\a_{w_2,1}(i)\leftarrow a_{w_2,1}(i-2),\\a_{w_2,2}(i)\leftarrow a_{w_2,2}(i-2),\\i=N_{w_2}+2,\dots,w_3+2,\\a_{w_2,1}(w_3)=2,\\a_{w_2,2}(w_3)=w_4,\\a_{w_2,1}(w_3+1)=0,\\a_{w_2,2}(w_3+1)=w_5,\\N_{w_2}\leftarrow N_{w_2}+2\end{gathered} \tag{20}$$

According to the rules of correct code in a universal code, you can always remove or insert function with one argument. Deleting a function with two arguments removes the function itself and the subexpression that defines the first argument of that function. Thus, this variation removes at least one argument of the entire argument of the mathematical expression. When inserting a function with two arguments, if the length of the code record allows, then the new function itself with two arguments and the argument of the mathematical expression are inserted. Thus, when this small variation is performed, the argument of the entire mathematical expression is inserted.

The minimum number of small code variations on that another code is differented from changed code is called the distance between the codes of possible solutions. To increase the search area for a possible solution in the code space in the vicinity of the basic solution, we enter the parameter - the depth of variation.

The depth of variation is performed maximum number of small variations of the basis solution. The initial population of possible solutions is the set of ordered sets of small variations of the basis solution

$$\mathrm{W} = (\mathrm{W}_0, \mathrm{W}_1, \dots, \mathrm{W}_H), \tag{21}$$

where $\mathrm{W}_i = (\mathbf{w}^{i,1}, \dots, \mathbf{w}^{i,d})$, d is a depth of variation, $i = 1, \dots, H$,

The initial population (21) includes the the set of variation vectors of the basic solution

$$\mathrm{W}_0 = (\mathbf{w}^{0,1}, \dots, \mathbf{w}^{0,d}),$$

where all variation vectors of the basic solution have zero value of its components

$$\mathbf{w}^{0,i} = [0\ 0\ 0\ 0\ 0]^{\mathrm{T}},\ i = 1, \dots, d. \tag{22}$$

Zero variations vectors don't make variation of basic solution, but they can be used in operation crossover with search of the optimal solution.

According to principle of small variation all possible solutions except a basic solution are set in the form as an order set of small variation vectors. All genetic operations are performed over these sets. For crossover operation two possible solutions or parents are selected.

$$\begin{array}{l} \mathrm{W}_\alpha = (\mathbf{w}^{\alpha,1}, \dots, \mathbf{w}^{\alpha,d}), \\ \mathrm{W}_\beta = (\mathbf{w}^{\beta,1}, \dots, \mathbf{w}^{\beta,d}). \end{array} \tag{23}$$

A crossover point is determined randomly, $c \in \{1, \dots, d\}$/ After that two new possible solutions are received by replacement components after the crossover point

$$\begin{array}{l} \mathrm{W}_{H+1} = (\mathbf{w}^{\alpha,1}, \dots, \mathbf{w}^{\alpha,c}, \mathbf{w}^{\beta,c+1}, \dots, \mathbf{w}^{\beta,d}), \\ \mathrm{W}_{H+2} = (\mathbf{w}^{\beta,1}, \dots, \mathbf{w}^{\beta,c}, \mathbf{w}^{\alpha,c+1}, \dots, \mathbf{w}^{\alpha,d}), \end{array} \tag{24}$$

where H is a number possible solutions in an initial population, d - is a depth of variation.

IV. COMPUTATIONAL EXPERIMENT

Consider the problem of identifying an internal model for the spatial motion of a quadcopter.

The mathematical model of the spatial motion of a quadcopter as a solid is as follows

$$\begin{array}{lcl} \dot{x}_1 & = & x_4, \\ \dot{x}_2 & = & x_5, \\ \dot{x}_3 & = & x_6, \\ \dot{x}_4 & = & u_4(\sin(u_3)\cos(u_2)\cos(u_1)+ \\ & & \sin(u_1)\sin(u_2)), \\ \dot{x}_5 & = & u_4\cos(u_3)\cos(u_1) - \mathrm{g}_c, \\ \dot{x}_6 & = & u_4(\cos(u_3)\sin(u_1)- \\ & & \cos(u_1)\sin(u_2)\sin(u_3)), \end{array} \tag{25}$$

where $\mathbf{x} = [x_1\ x_2\ x_3\ x_4\ x_5\ x_6]^{\mathrm{T}}$ is a vector of the state space, x_1 is a longitudinal coordinate, x_2 is a flight altitude, x_3 is a literal coordinate, x_4 is a speed in longitudinal direction, x_5 is a speed of flight altitude change, x_6 is a flight speed in lateral direction, $\mathbf{u} = [u_1\ u_2\ u_3\ u_4]^{\mathrm{T}}$, is a control vector, u_1 is the angle roll of rotation about the axis $\{0, x_1\}$, u_2 is the angle jaw of rotation about the axis $\{0, x_2\}$, u_3 is the angle pitch of rotation about the axis $\{0, x_3\}$, u_4 is the total thrust of all quadcopter screws, g_c is the acceleration of gravity, $\mathrm{g}_c = 9.80665$.

The control vector has restrictions

$$\mathbf{u}^- \leq \mathbf{u} \leq \mathbf{u}^+, \tag{26}$$

where $\mathbf{u}^- = [-\pi/6\ -\pi\ -\pi/6\ 0]^{\mathrm{T}}$, $\mathbf{u}^+ = [\pi/6\ \pi\ \pi/6\ 12]^{\mathrm{T}}$.

Real quadcopters differ from the model by the presence of an angular stabilization system. They can have different quality and different transition times. The real mathematical model of the quadcopter, taking into account the dynamics of the angular stabilization system, has the following form

$$\begin{array}{lcl} \dot{x}_1 & = & x_4, \\ \dot{x}_2 & = & x_5, \\ \dot{x}_3 & = & x_6, \\ \dot{x}_4 & = & u_4(\sin(u_3)\cos(u_2)\cos(u_1)+ \\ & & \sin(u_1)\sin(u_2)), \\ \dot{x}_5 & = & u_4\cos(u_3)\cos(u_1) - \mathrm{g}_c, \\ \dot{x}_6 & = & u_4(\cos(u_3)\sin(u_1)- \\ & & \cos(u_1)\sin(u_2)\sin(u_3)), \\ \dot{x}_7 & = & f_7(x_7, u_1), \\ \dot{x}_8 & = & f_8(x_8, u_2), \\ \dot{x}_9 & = & f_9(x_9, u_3), \end{array} \tag{27}$$

where where $f_7(x_7, u_1)$, $f_8(x_8, u_2)$, $f_9(x_9, u_3)$ are unknown equations describing the dynamics of roll u_1, yaw u_2 and pitch u_3 stabilization systems.

Therefore we need to find the more accuracy last three equations in the model (25). For this purpose symbolic regression is used. To define the basic quadcopter model the following expression is used. The first three equations of the basic model coincide with the first three equations of the model (25), the rest have the following form

$$\begin{array}{lcl} \dot{x}_4 & = & u_4(\sin(u_3)\cos(u_2)\cos(u_1)+ \\ & & \sin(u_1)\sin(q_1 u_2)), \\ \dot{x}_5 & = & u_4\cos(u_3)\cos(q_2 u_1) - \mathrm{g}_c, \\ \dot{x}_6 & = & u_4(\cos(u_3)\sin(u_1)- \\ & & \cos(u_1)\sin(u_2)\sin(q_3 u_3)), \end{array} \tag{28}$$

where q_1, q_2, q_3 are the desired parameters that are searched together with the structure of mathematical expressions by the method of symbolic regression To obtain a training sample, the movement of the quadcopter along closed trajectories in the forms of a circle and a figure of eight was determined. The total Euclidean error of deviation of the calculated trajectories from the experimental ones and the sum of the maximum time deviations of the calculated trajectories from the experimental ones were determined as search criteria.

As a result, the symbolic regression method, the universal code, found the following model of the spatial motion of the quadcopter

$$\begin{aligned} \dot{x}_4 &= \sqrt[3]{q_1} q_2 (z_0 + (z_2 + u_4) \times \\ &\quad \sin(\sin(u_1) \sin(q_1 u_2))), \\ \dot{x}_5 &= q_2^2 (z_1 - q_4), \\ \dot{x}_6 &= q_2 (z_2 - (u_3 + u_4) \cos(q_3 u_1) \times \\ &\quad \sin(u_2) q_3 u_3), \end{aligned} \tag{29}$$

where

$$\begin{aligned} z_0 &= (\arctan(\rho_{18}(\rho_{17}(u_1) + 2u_1 + u_4))) \times \\ &\quad \sin(sin(u_3)) \cos(u_2^2) \cos(u_1), \end{aligned}$$

$$z_1 = u_4 \cos(-\sin(\sin(q_1 u_3))) \cos(\sin(q_1 q_2 = -q_1^3 q_2^3)),$$

$$z_2 = u_4 \cos(u_2) \arctan(u_1),$$

$$\rho_{17}(\alpha) = \operatorname{sgn}(\alpha) \ln(|\alpha| + 1),$$

$$\rho_{18}(\alpha) = \operatorname{sgn}(\alpha)(\exp(|\alpha|) - 1),$$

$q_1 = 0.93799$, $q_2 = 0.94946$, $q_3 = 1.23730$, $q_4 = 9.80665$.

The figure 1 shows projections on the horizontal plane of the trajectories of the quadcopter movement in a circle a. In the figure, blue lines indicate projections of real trajectories, and black lines are the trajectories obtained using the identified model

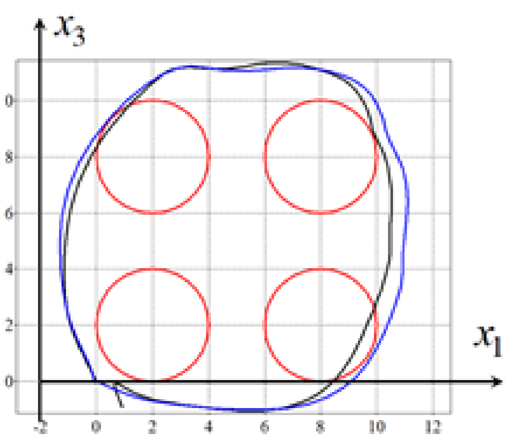

Fig. 1. Inductance of oscillation winding on amorphous magnetic core versus DC bias magnetic field

In the figures 2,...,6 graphics of changings of the model variables and real object in time at movement on the circle are presented.

How we can see from graphics a mathematical model obtained by symbolic regression enough accuracy describes

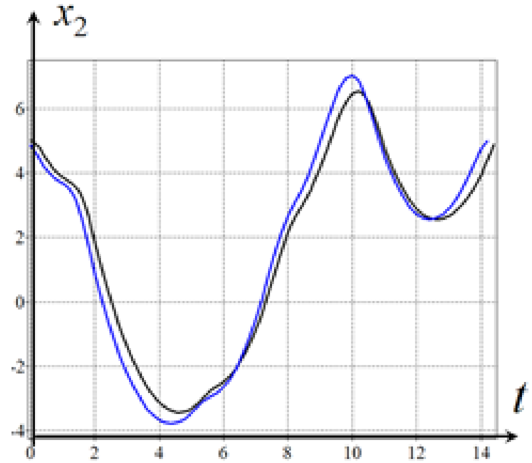

Fig. 2. Graphic of variable x_2 at movement on the circle. Blue line is real object, black line is identification model

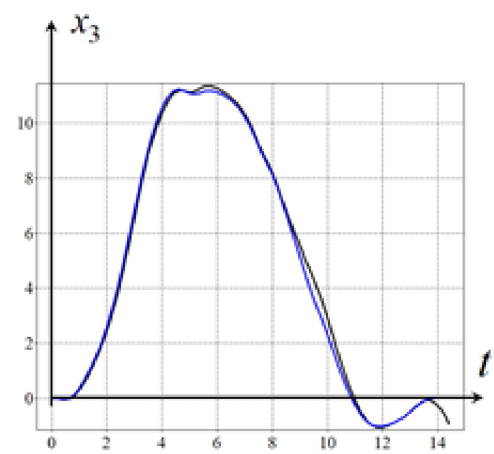

Fig. 3. Graphic of variable x_3 at movement on the circle. Blue line is real object, black line is identification model

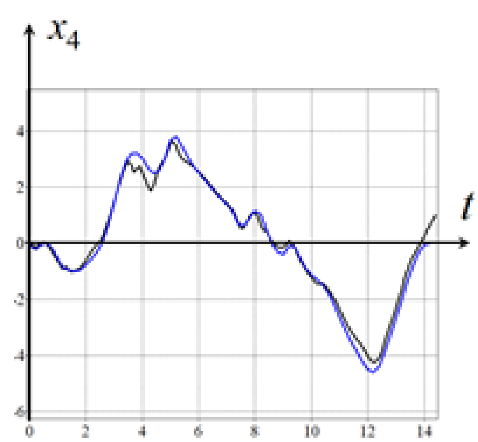

Fig. 4. Graphic of variable x_4 at movement on the circle. Blue line is real object, black line is identification model

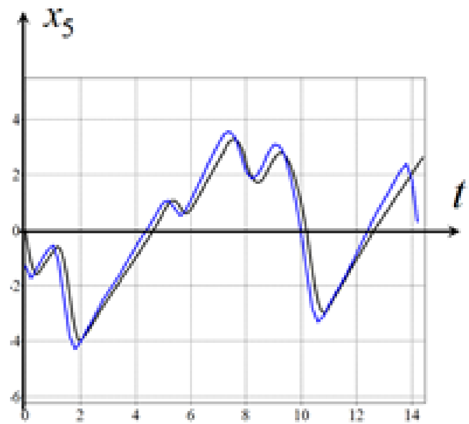

Fig. 5. Graphic of variable x_5 at movement on the circle. Blue line is real object, black line is identification model

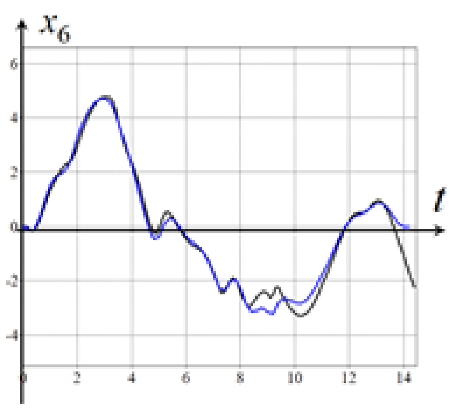

Fig. 6. Graphic of variable x_6 at movement on the circle. Blue line is real object, black line is identification model

movement of real object on total variables. The same graphics were obtained at movemenet of the control object on the curve of the form of eight. Accuracy mathematical model of the object with consider dynamics of control system allows to obtain the vector of the state space for object at absence global position systems. The navigation problem without global position systems is very important for autonomous robotics. Here for solving this problem high accuracy mathematical model is proposed. Such model located on the board of the robot can return the state of the object on control signals only on results of simulation.

During the period of existence of signals of global positioning systems, the model receives control signals and returns the state of the quadcopter, which is then compared with information coming from global positioning systems. If for some modes of operation of the quadcopter the difference between the assessment of the state of the object based on information received from global positioning systems and the model is significantly different, then this is the basis for additional training of symbolic regression to refine the internal model.

Such a scheme of constant refinement of the object model for various modes of its functioning should lead to a model that for this quadcopter will allow assessing its state with sufficiently high accuracy. A high-precision model of a real instance of a quadcopter will provide the ability to assess the state of a control object in a stream for a significantly long period, i.e. solve the problem of navigation in the absence of signals from global positioning systems.

V. CONCLUSIONS

The identification problem is considered. It is necessary to contruct high accuracy mathematical model of the control object to solve the navigation problem at condition of global position sysytems abcence. For this purpouse symbolic regression is used. We refine the mathematical model with consider dynamics of the control system. This dynamics for each object can be itself. Therefore the mathematical model for two objects the same types can be different. As example in the work the quadcopter is cosidered. For it the mathematical model is refined by the symbiolic regression. How computing experiments showed the symbolic regression allows to identificate high accuracy mathematical model of control object.

To obtain a training set, a mathematical model of the control object was taken and the places of occurrence of an error in assessing the state of the object were determined. As a rule, the mathematical model of the control object does not take into account the transition process between the control signal and the generation of control action directly on the object. To take into account the parameters of this transient process, an updated mathematical model of the control object was developed. The parameters themselves of the transient process are determined experimentally.

REFERENCES

[1] Moeller, D.P.F. Parameter Identification of Dynamic Systems. In: Mathematical and Computational Modeling and Simulation. Springer, Berlin, Heidelberg.2004, pp.257-310, https://doi.org/10.1007/978-3-642-18709-4_5.

[2] Matzkin, R.L. Nonparametric Identification in Structural Economic Models. Annual Review of Economics. 2013, No 5 (1),pp. 457–486.

[3] Ashchepkov, L.T., Dolgy, D.V., Kim, T., and Agarwal, R.P. Identification Problem. Chapter in Optimal Control. Springer, Cham. 2022, pp 101–105 https://doi.org/10.1007/978-3-030-91029-7_8.

[4] Zielinska, T. (2013). Design Issues and Robots Autonomy. In: Viadero, F., Ceccarelli, M. (eds) New Trends in Mechanism and Machine Science. Mechanisms and Machine Science, vol 7. Springer, Dordrecht. https://doi.org/10.1007/978-94-007-4902-3_72.

[5] Diveev, A., Shmalko, E. Machine Learning Control by Symbolic Regression, 2021, Springer, 158 p.

[6] Diveev, A.,Sofronova, E. Machine Search Problem for Mathematical Expression//Doklady, Mathematics, 2024, Volume 110, pages S25–S34. DOI:10.1134/S1064562424602087.

[7] Diveev, A. Combined Symbolic Regression Approach and its Application for Synthesized Optimal Control//2025 11th International Conference on Control, Decision and Information Technologies (CoDIT), 15-18 July 2025, Split, Croatia, IEEE. DOI: 10.1109/CoDIT66093.2025.11321521.

[8] Diveev, A. Small Variations of Basic Solution Method for Non-numerical Optimization. IFAC-PapersOnLine, 2015, 48(25), pp. 28–33.//doi.org:10.1016/j.ifacol.2015.11.