GMDH-Based Learning System for Mobile Robot Navigation in Heterogeneous Environment

Anatoliy Andrakhanov(✉) and Alexander Belyaev

Department of Control Systems and Mechatronics, National Research Tomsk Polytechnic University, Lenin Avenue, 30, Tomsk 634050, Russian Federation
riml282a@gmail.com, belyaewas@mail.ru

Abstract. One of the key tasks of mobile robotics is navigation, which for Outdoor-type robots is exacerbated by the functioning in an environment with a priori of unknown characteristics of underlying surfaces. In this paper, for the first time, the learning navigation system for mobile robot based on the group method of data handling (GMDH) is presented. The paper presents the results of training of models both for evaluating the robot's pose (coordinates and angular orientation) in heterogeneous environment and classification of the type of underlying surfaces. In addition to the direct readings of the on-board sensors, additional parameters (reflecting how the robot perceives the surface terramechanics) were introduced to train the models. The results of testing of the obtained models demonstrate their performance in an essentially heterogeneous environment, when areas of the underlying surfaces are comparable with the robot's dimensions. This testifies the operability of developed GMDH-based learning system for mobile robot navigation.

Keywords: Mobile robot · Heterogeneous environment · Underlying surface Testing ground · Navigation · Coordinates evaluation · Machine learning Inductive modeling · GMDH · Active neuron · Festo Robotino

1 Introduction

Despite the rapid development of mobile robotics, the development of an intelligent control system remains one of the main challenges in the creation of autonomous robotic systems.

One of the key tasks is the navigation, which can be divided into 2 parts: the estimation of the current position (coordinates and angular orientation) of the robot in the working space and the development of control actions on the actuators to sequentially achieve all the intermediate robot positions along the planned trajectory. In this case, the solution of the second part of the problem is impossible without solving of the first. In some cases, the evaluation of the robot position in the environment can be carried out only by means of on-board inertial system, because global positioning systems (GPS) may be not exist (planetary rovers), may be inaccessible (fire fighting robots, autonomous mining vehicles) or suppressed by electronic countermeasures equipment (combat robots).

N. Shakhovska and V. Stepashko (eds.), *Advances in Intelligent Systems and Computing II*, Advances in Intelligent Systems and Computing 689,
https://doi.org/10.1007/978-3-319-70581-1_1

For outdoor-type mobile robots, this problem is exacerbated by the natural conditions of the environment which is characterized by the a priori unknown of an environment model, the heterogeneous characteristics of surfaces to be traversed, and the difficulty of determining the features of the robot-terrain interaction based on on-board sensor readings.

There are a great number of papers on this subject [1–9]. In a first approximation the four main approaches can be identified for solving navigation problem in a heterogeneous environment (Table 1).

Table 1. The man ways to solve the problem of a mobile robot's navigation in a heterogeneous environment

Approach to models construction	Classification of the underlying surface type	
	Used	Not used
Detection and analysis of a physical patterns of robot-terrain interaction	S. Khaleghian and S. Taheri [9] Sensors: 3-axis and 1-exis accelerometers, encoders Method: Fuzzy logic	L. Ojeda et al. [7] Iagnemma K. et al. [8] Sensors: encoders, gyroscopes, accelerometers [8] + current sensors [7] Method: Wheel slip analysis
Construction of non-physical models by means of machine learning	DuPont E.M. et al. [6] Sensors: 3-axis gyroscope, 3-axis accelerometer Method: Probabilistic neural network	A.A. Andrakhanov [10, 11] Sensors: encoders, motor current sensors Method: Twice Multilayered Modified Polynomial Neural Network with active neurons

There is currently no generally accepted dominant methodology at present, and each research group is trying to solve the problem in its own way. The authors of this work believe that one of the most promising paths is to construct non-physical models using the advantages of the inductive modeling approach.

The basic method of the inductive modeling approach is Group Method of Data Handling (GMDH). To date, the most complete overview of the use of GMDH in robotics is shown in the work [12]. This method was already used by the authors to solve the problem of evaluating the robot's pose in homogeneous and heterogeneous environments and demonstrated an acceptable result [10, 11].

2 The Advantages of GMDH for Finding Dependencies Based on the Analysis of Sensor Readings

A number of field experiments were carried out in order to determine the interaction between the robot's propulsion system and the underlying surface of various types based on on-board sensor readings.

A serial-produced mobile platform Festo Robotino was used as the mobile robot. Experiments were carried out at a specially designed testing ground consisting of 28 modules with different terramechanical characteristics (Fig. 1a). The testing ground was designed in such a way that the terramechanical interaction of its areas with Festo Robotino's wheel system was correspond in terms of quality with the level of terramechanical interaction of the outdoor robot with the areas of the natural environment. Figure 1b shows the robot's test motions (along a square and a triangle path) with the same motor speed setpoints under conditions of a homogeneous (an ideal flat surface) and heterogeneous (Testing Ground) environment. Holonomic character of robot movements is provided due to 3 wheels of omnidirectional type, located at an angle of 120° with respect to each other (Fig. 1c).

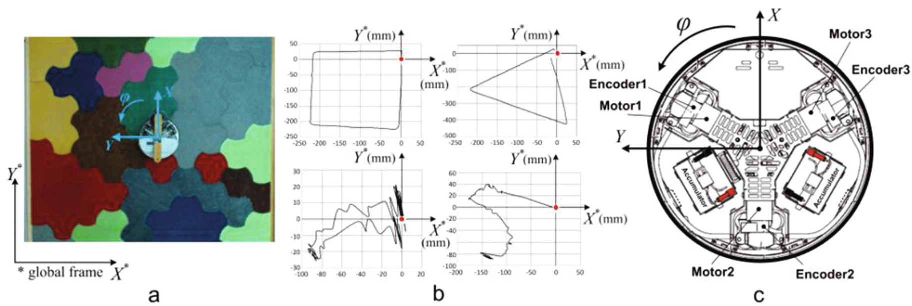

Fig. 1. (a) Testing Ground appearance and the robot; (b) Trajectories for test motions (along a square and a triangle path) in homogeneous (upper) and heterogeneous (lower) environments; (c) arrangement of wheels in Robotino's omni-drive system [13] (X, Y, φ – Robotino's local coordinate system)

It is clear from Fig. 1b that the areas of the testing ground have a significant impact on the nature of the robot's movement. In order to maximize the transparency between sensor readings and the nature of the robot's motion, we decided to assign only the simplest robot motions and only the homogeneous areas of the testing ground consisting of modules of the same type. Using the simplest driving setpoints makes it possible to eliminate the features of the robot's kinematics and propulsion system, which in turn allows it to make complex motions (curvilinear motion with rotational motion) without being affected by the features of the environment. Moving in a homogeneous local area eliminates the comlex influence of different local areas on the robot behavior because on the testing ground the each wheel interacts with its local area (Fig. 1a).

Five types of local surfaces, which were different in terms of terramechanics, were selected by expertise, as well as four of the simplest movements setpoints: three translational motion, without rotational component (along the X-axis, along the Y-axis, in the XY plane at the same X- and Y-axis speeds) and rotational motion without translational component, all motions mentioned in relation to the robot's local coordinate system (Fig. 1c). The speed of the translational motion was set to be 100 mm/s (for both the X- and Y-axes), and that of rotational motion was 24, 48 and 96 (deg/s).

Sensor data subject to analysis included: $\{N\} = \{N_1, N_2, N_3\}$ – a set of incremental encoders values, $\{\omega\} = \{\omega_1, \omega_2, \omega_3\}$ – values of the speeds of motors, $\{I\} = \{I_1, I_2, I_3\}$ – values of the motors consumption currents, $\{g\} = \{g_x, g_y, g_z\}$ – angular velocity values in relation to the X, Y and Z axes (Z-axis is perpendicular to XY-plane), $\{a\} = \{a_x, a_y, a_z\}$ – a set of acceleration values along the X, Y and Z axes.

Figure 2 shows the readings of three sensors when the robot traverses over five different underlying surfaces.

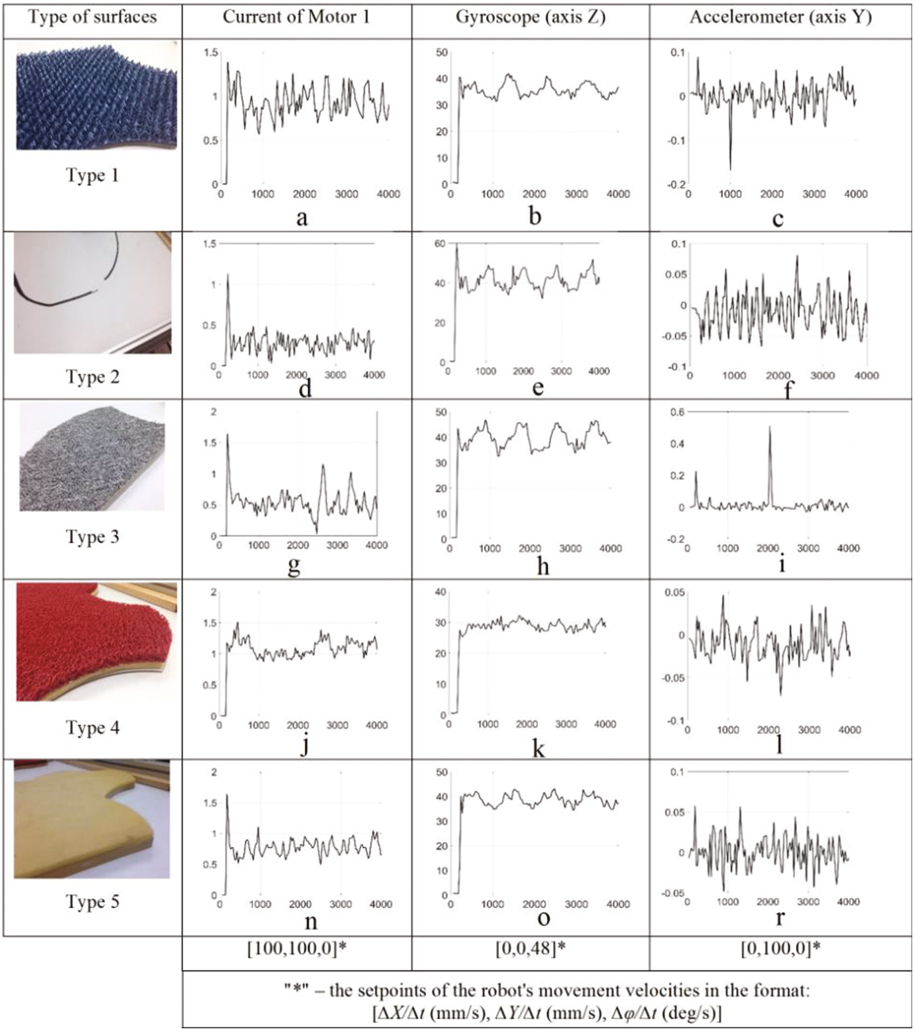

Fig. 2. Sensor readings during the robot's movement over different surfaces (horizontal axis for all graphics is the time axis in milliseconds)

The analysis shows that, on the one hand, the sensor data are correlated to the type of underlying surface and the nature of the robot movement on this surface. On the other hand, the correlation is ambiguous.

For example, the average values of currents in the first motor (I_1) demonstrate the laboriousness of overcoming the area (Fig. 2a and g), on the one hand, while on the other hand, these values may be close despite the fact that the surfaces may have different terramechanical characteristics (Fig. 2a and j). In addition, there are other features in the sensor data, which reflect the interaction of the robot with a surface of a particular type. In particular, the gyroscope data along the Z-axis (g_z), shown in Fig. 2b and h, clearly show the periodic behavior and the curve shape when moving over a particular surface. However, these may not be shown in the sensor data (Fig. 2k). A similar trend is observed in accelerometer readings (a_y): the features may manifest themselves (Fig. 2i), though not always (Fig. 2l and f).

Inasmuch as there are unique data properties for different subsets of sensors, it is necessary to derive models on their basis by using machine learning and data mining techniques. In our opinion, one promising tool for evaluating the surface type and the coordinates using sensor readings is the inductive approach.

The advantages of GMDH (as a method of inductive modeling) for the aforementioned research topics and developments are shown in Table 2. This method provides maximal flexibility at the stage of model construction both in handling the parameters of data sample (the method of dividing the sample into the training and the test parts, sorting by the dispersion of the output variable, etc.) and the training algorithm parameters (for neural algorithms, it means selecting the maximum degree of a partial description of a neuron, the number of selectable neurons in the layer, the maximum number of layers in the network, etc.).

Table 2. The benefits of GMDH in addressing the navigation problem

The problem of navigation in a heterogeneous environment	The GMDH advantages
A variety of tools is necessary to generate methods for evaluating coordinates and determining the underlying surface type. For instance, in the well-known paper [6], the technical solution contained operations of extracting the most significant features (principal component analysis), interpolation (fundamental splines), clustering (Eigen-transformation), and classification (probabilistic neural network)	The method includes a wide range of algorithms for predicting, classifying, clustering, identifying and data mining The unified methodological basis for the aforementioned spectrum of algorithms contributes to the standardization of the system's program modules

(*continued*)

Table 2. (*continued*)

The problem of navigation in a heterogeneous environment	The GMDH advantages
There are no simple and obvious correlations between sensor readings and the terramechanical properties of the underlying surfaces	The most effective input variables (with respect to some quality criterion) are selected automatically from the set of variables available to the system, and relationships in data are interpreted The resulting dependencies have an analytical form (it is also typical for the GMDH-type neural networks), which enhances the capabilities for analysis and makes it possible to interpret the results
The number of local areas that affect the robot motion in different ways can be arbitrary large. Therefore, it is necessary that the functional dependences derived by the onboard computing system should be generalizable for other areas that have not yet been traversed	The resulting dependencies have a generalizing ability because an external criterion of model quality is used (evaluation of model parameters and selection of model structure are performed using independent data subsamples)
Considering that the size of local areas may be relatively small, it is important that the methods used to derive models be able to work with short data samples In some cases (for instance, the time limit for making a decision, limited learning time, as well as energy costs, and so on), it makes sense to collect a relatively small number of samples even if the size of areas is significant	For short, inaccurate, or noisy data, an optimal nonphysical model can be found, whose accuracy is higher and structure is simpler than the structure of a complete physical model [14] Finding a solution within a limited training time is guaranteed. The system can calculate the training time before training algorithm run

In addition, GMDH provides great opportunities in analyzing dependencies found during the training phase: which sensor readings are used for the model's output as the input variables; with what degrees and/or coefficients these variables are included in the model; how often these variables are chosen by the neural network algorithm when constructing a neural network structure from layer to layer sequentially, etc.

In order to ensure maximum access to these features and the advantages of GMDH, the authors have decided to develop a navigational training system on its basis.

3 Navigation Learning System

3.1 Description of the System

The system architecture is represented in Fig. 3 (only the basic connections between the modules are shown).

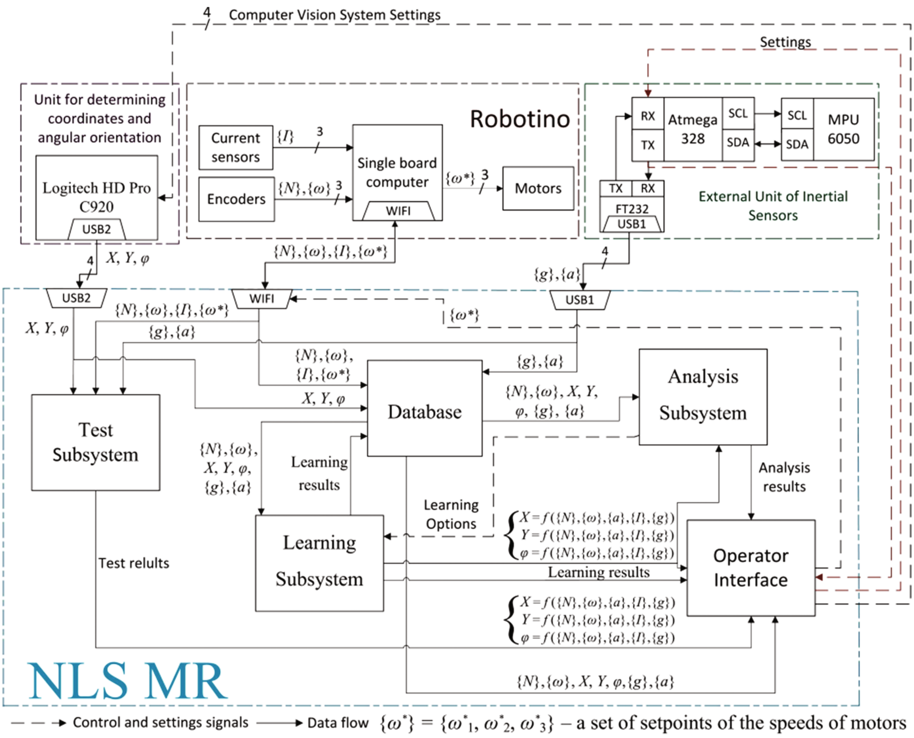

Fig. 3. The navigation learning system architecture

The navigation learning system (NLS) consists of the following modules:

- The Learning Subsystem that implements computational intelligence algorithms within a framework of the inductive modeling approach.
- The Database which is necessary for collecting and storing sensor data as well as learning results.
- The Test subsystem which is required to test the hardware of the system (sensors and actuators of the robot, unit for determining coordinates and angular orientation) and the learning subsystem. The hardware test is based on comparing the stored data (sensor readings, the coordinates and angular orientation values) and the data obtained as a result of the robot's test movements. Testing of the learning subsystem is done by models training on test data samples and comparing the obtained results with the stored results.
- The Analysis subsystem which analyzes the previous results of models training in order to determine the influence of different subsets of input variables, the variants of splitting a data sample, the quality criteria and the algorithm parameters on the quality of obtained models.
- The Operator interface that provides the operator's access to data samples, as well as to the results of testing, training and analysis.

The hardware which is external in relation to NLS includes the on-board sensors and actuators of the Festo Robotino platform, the additional unit of inertial sensors (three-axis gyroscope and three-axis accelerometer) and the Full HD camera to obtain the coordinates and angular orientation of the robot.

This system was used for training and testing models in all the experiments mentioned below.

3.2 The Models Training Algorithm

Twice-Multilayered Modified Polynomial Neural Network with Active Neurons (TMMPNN) algorithm makes it possible to find the optimal network structure (from the view of the external criterion) and partial descriptions of neurons automatically (in the self-organization mode). The concept of twice-multilayered polynomial neural network algorithm was first proposed by A.G. Ivakhnenko and J.A. Muller [15].

The modification is that the generation of partial descriptions on each layer (starting from the 2nd) involves not only the outputs of neurons of the previous layer, but also the input variables. Thus, such modification provides an opportunity to avoid losing important input variables on the first and subsequent layers of the network. The structure of the modified polynomial neural network is shown in Fig. 4 [12]:

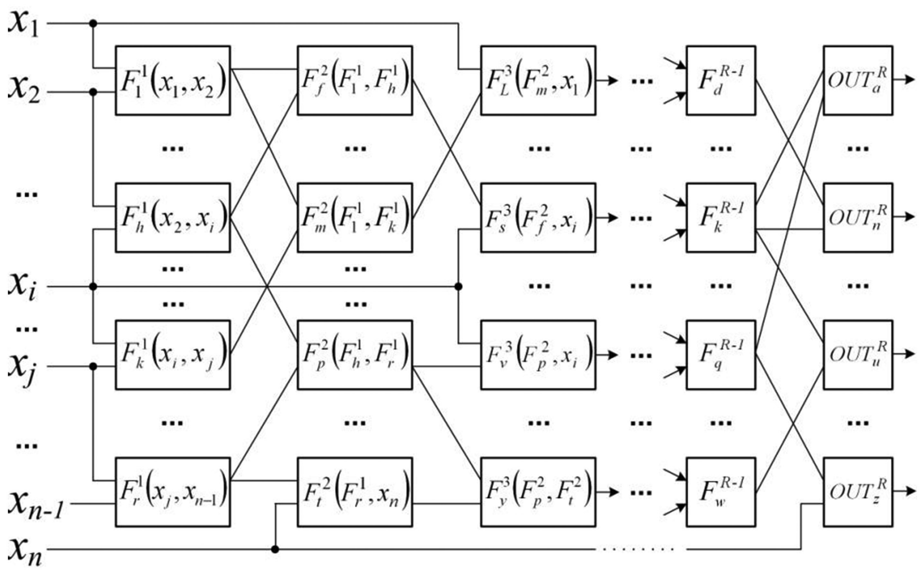

Fig. 4. Twice-multilayered modified GMDH-type polynomial neural network ($x_1, x_2, x_i, x_j, x_{n-1}, x_n$ are input variables; $F_k^1(x_i, x_j)$ are partial descriptions of k-th selected neuron of the 1st layer; $F_d^{R-1}, F_k^{R-1}, F_q^{R-1}, F_w^{R-1}$ are partial descriptions of neurons of the layer (R–1); $OUT_a^R, OUT_n^R, OUT_u^R, OUT_z^R$ are partial descriptions of selected neurons of the output layer)

Since this algorithm was described in detail in earlier papers [10, 12] and its software implementation was published on the CD to the book [12], we will discuss only the main points related to the settings of this algorithm in the following experiments.

The classic combinatorial algorithm of GMDH [16] is used to search for partial descriptions of neurons. In this case, two-input neurons were used limited by the maximum polynomial degree 2 of the partial description:

$$F_k^l(x_i, x_j) = a_0 + a_1 \cdot x_i + a_2 \cdot x_j + a_3 \cdot x_i^2 + a_4 \cdot x_j^2 + a_5 \cdot x_i \cdot x_j \quad (1)$$

The regularity criterion was used as an external criterion for the selection of partial descriptions of neurons:

$$CR = \frac{1}{N_B} \sum_{i=1}^{N_B} (f_i - y_i)^2 \quad (2)$$

where N_B is the number of rows of the testing data sample; f_i is the output of the model for row i; y_i is the output value for row i of the data sample.

In case of network structure construction the regularity criterion was also used as an external criterion for the selection of the best neurons of a layer. The external layer criterion (the arithmetic mean value of the regularity criteria of the best neurons in a given layer), the limit of maximal network capacity (maximal number of layers $\times$ number of selected best neurons in a layer) and the additional stopping criterion (an improvement in the value for the external layer criterion should be more than ε from layer to layer) were used as stopping criteria of the expansion of network layers.

The algorithm, criteria and settings described were also used for the training of a surface type classifier. When using a trained classifier, the threshold condition is applied to the network output: «1» if the network output is greater than or equal to 0.5, and «0» otherwise.

3.3 Forming Sets of Input Variables for Models Training

In Sect. 2, it was shown that it was difficult to estimate the coordinates of the robot's position and the type of the underlying surfaces directly from the sensor readings. In the paper [17], parameters reflecting how the robot sense the terramechanics of a surface based on sensor readings were introduced. In this work, we also introduced additional parameters of such type to increase the number of relevant variables on the training stage.

The three parameters that characterize the displacement in a given local area are the robot's velocities in its local coordinate system:

$$\begin{pmatrix} V_x \\ V_y \\ \Omega \end{pmatrix} = R \cdot \begin{pmatrix} -\frac{2}{3}\cos(\alpha - \theta) & \frac{2}{3}\sin(\alpha) & \frac{2}{3}\cos(\alpha + \theta) \\ -\frac{2}{3}\sin(\alpha - \theta) & -\frac{2}{3}\cos(\alpha) & \frac{2}{3}\sin(\alpha + \theta) \\ \frac{1}{3 \cdot L} & \frac{1}{3 \cdot L} & \frac{1}{3 \cdot L} \end{pmatrix} \cdot \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} \quad (3)$$

where V_x, V_y is the velocity along the X and Y axes of the robot's local coordinate system; Ω is the angular rotation velocity of the robot in the local coordinate system; ω_1, ω_2, ω_3 are the angular speeds of wheels (associated with the speeds of the motors through the 1:16 gear ratio); L is the distance from the center of the robot to the wheel (125 mm); R is the wheel radius (40 mm); α is the robot orientation angle; θ is the wheel orientation angle (30°).

The kinematics Eq. (3) was also used by the authors to obtain the parameters of the laboriousness of translational and rotational motion of the robot:

$$\begin{pmatrix} I_x \\ I_y \\ I_\varphi \end{pmatrix} = R \cdot \begin{pmatrix} -\frac{2}{3}\cos(\alpha-\theta) & \frac{2}{3}\sin(\alpha) & \frac{2}{3}\cos(\alpha+\theta) \\ -\frac{2}{3}\sin(\alpha-\theta) & -\frac{2}{3}\cos(\alpha) & \frac{2}{3}\sin(\alpha+\theta) \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \cdot \begin{pmatrix} I_1 \\ I_2 \\ I_3 \end{pmatrix} \tag{4}$$

where I_x, I_y are the values of the currents that characterize the laboriousness of the robot's movement along the X and Y axes of the local coordinate system; I_φ is the value of the current which characterizes the laboriousness of the robot's angular rotation in the local coordinate system; I_1, I_2, I_3 are the consumption currents of motors.

In (4), the consumption currents of motors have the same sign, which is determined by the direction of the wheel rotation, as for the speeds of motors in (3). Unlike (3), R and L values are not used in (4) because, firstly, they are not related to a geometric transformation of the current vectors. Secondly, the dimension of the output quantities and their physical meaning will be inconsistent with each other, which is unacceptable. Thirdly, these values influence only the amplitude of the output values, which is not important for GMDH, since the TMMPNN algorithm independently selects necessary weighting coefficients. The preliminary analysis showed an appropriate separability for all five types of surfaces used in the experiments in case of use of parameters – I_x, I_y, I_φ (Fig. 5b, c and d).

Another parameter used, which characterizes the interaction of the robot with the surface, is I_Σ – the total consumption current of motors:

$$I_\Sigma = |I_1| + |I_2| + |I_3| \tag{5}$$

As can be seen in Fig. 5a, the mean value of this parameter varies for different types of surfaces, which makes it a useful variable both for classifying the surface type and for estimating the coordinates and angular orientation. Since the robot arrives to different coordinates on different surfaces with the same motor velocities setpoints, the coordinate estimation can be related to this parameter.

In addition to the aforementioned absolute parameters, the following relative parameters were also introduced:

$$\begin{gathered} T_x = \frac{V_x}{I_x}; T_y = \frac{V_y}{I_y}; T_\varphi = \frac{\Omega}{I_\varphi}; T_z = \frac{g_z}{I_\varphi}; T_x^* = \frac{V_x^* - V_x}{I_x}; T_y^* = \frac{V_y^* - V_y}{I_y}; \\ T_\varphi^* = \frac{\Omega^* - \Omega}{I_\varphi}; T_z^* = \frac{\Omega^* - g_z}{I_\varphi} \end{gathered} \tag{6}$$

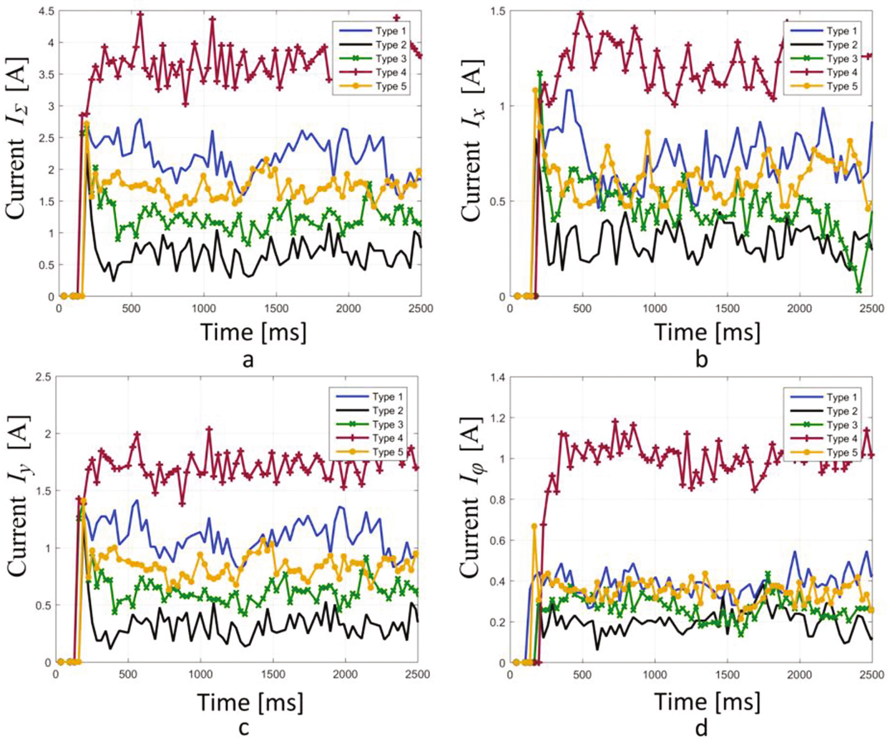

Fig. 5. Analysis of the relevance of the parameters (the setpoints of the robot's movement velocities in the format – [V_x^* (mm/s), V_y^* (mm/s), Ω^* (deg/s)]: (a) [0; 100; 0]; (b) [100; 100; 0]; (c) [0; 100; 0]; (d) [0; 0; 24])

The values V_x, V_y, and Ω are calculated using actual wheel speeds values, based on (3), while I_x, I_y, and I_φ are calculated using the current sensors values based on (4). The values V_x^*, V_y^*, and Ω^* are the setpoints of the robot's movement velocities in the local coordinate system.

It should be noted that the dimension of relative parameters has the physical interpretation as a unit of the translational/rotational movement on a particular surface for the expended current impulse, which is normalized to the same type and direction of movement. In the case of a difference in the numerator between the setpoint and the real velocity, the physical interpretation changes into: by how many millimeters/degrees the actual displacement/rotation of the robot on the surface will differ from the setpoint value after one current impulse.

Thus, while implementing inductive modeling, three sets of input variables were used:

- $\{V_1\} = \{\{N\},\{\omega\},\{I\},\{g\},\{a\}\}$ are values obtained directly from the robot's sensors;
- $\{V_2\} = \{V_x, V_y, \Omega, I_x, I_y, I_\varphi, I_\Sigma\}$ are absolute parameters obtained by means of mathematical transformations of values measured by sensors;

- $\{V_3\} = \left\{T_x, T_y, T_\varphi, T_z, T_x^*, T_y^*, T_\varphi^*, T_z^*\right\}$ are relative parameters obtained by means of algebraic relations between values of the second and the first sets.

The purpose of the experiments series was:

- Determination of the obtained models accuracy for robot pose evaluation taking into account three sets of parameters;
- Estimation of the relevance of each set of parameters for the constructing of robot pose estimation models;
- Determination of the accuracy of the underlying surface type classification taking into account three sets of parameters;
- Estimation of the relevance of each set of parameters for constructing classifiers of the surface type;
- Testing the obtained models during robot movement in an essentially heterogeneous environment, when dimensions of both the surfaces and the robot are comparable.

4 Results of Experiments

4.1 Results of Models Training

All experiments were carried out on five selected types (see Fig. 2) of surfaces using the Festo Robotino mobile platform. There were 30 robot launches lasting 4 s with the following combinations of robot movement setpoints (in the format [$\Delta X/\Delta t$ (mm/s), $\Delta Y/\Delta t$ (mm/s), $\Delta\varphi/\Delta t$ (deg/s)]): [100,0,0], [0,100,0], [100,100,0], [0,0,24], [0,0,48], [0,0,96]. Data sample was formed by dividing of the sensor readings into half-second intervals, sensor values was averaged for these intervals (except values of $\{N\}$, the increment of values per half-second intervals were calculated for this case). Thus, for each robot launch four examples were included into the training data sample, four – into the testing data sample.

All models were obtained with help on software that was published on the CD to the book «GMDH-Methodology and Implementation in C» [12].

In the all experiments, the constraints were used to both the maximum power of neuron (power – 2) and the network capacity (10 layers x 10 neurons per layer). The choice of these parameters is due to the experience of our previous experiments, including in [10, 11]. In particular, it was found that such a limitation on the network capacity makes it possible to obtain the most stable (in terms of accuracy and bias) models.

Since the absolute error in the determination of the coordinates used by the computer vision unit is 1 mm, an additional criterion for stopping the network construction $\varepsilon = 0.1$ was given. Data sample was divided into the two equal parts (training and testing).

The results of the experiments are shown in Table 3 ("[Avr]" is the arithmetic mean error, "Max" is the maximum error, "GM" are denoted ("General Model") the training results of model on the combined data sample for all types of the surfaces) and Table 4.

The best trained models with minimum error are highlighted in bold in Tables 3 and 4. The average values ("[Avr]") of the coordinates and angular orientation in Table 3 are less 1 (mm or deg) on the all types of underlying surfaces for the all subsets of input variables.

Table 3. Results of models training for robot pose estimation

Value	Type	Input variable set						
		$\{V_1\}$	$\{V_2\}$	$\{V_3\}$	$\{V_1\}$, $\{V_2\}$	$\{V_1\}$, $\{V_3\}$	$\{V_2\}$, $\{V_3\}$	All
		Max [Avr]	Max [Avr]	Max [Avr]	Max [Avr]	Max [Avr]	Max [Avr]	Max [Avr]
X, mm	1	**3.6**	6.0	6.8	4.6	3.6	4.9	4.6
	2	10.5	**9.3**	9.4	10.9	10.5	9.7	10.9
	3	**5.9**	8.1	6.4	6.3	5.9	6.6	6.1
	4	2.2	2.1	2.8	2.0	2.3	**1.8**	2.0
	5	7.2	7.2	6.8	**6.1**	7.2	6.3	6.4
	GM	**8.7 [2.02]**	9.9 [1.7]	12.6 [2.16]	10.5 [1.7]	9.3 [1.8]	10.0 [1.8]	10.3 [1.7]
Y, mm	1	5.1	**4.8**	6.6	4.8	7.3	4.8	4.8
	2	9.0	8.9	8.1	7.7	8.1	8.5	**6.5**
	3	**6.1**	9.7	29.3	12.2	6.2	8.6	12.2
	4	**1.5**	2.2	2.0	1.9	1.7	2.8	2.0
	5	**5.5**	8.1	9.2	5.8	5.8	9.2	9.2
	GM	9.9 [1.7]	15.1 [1.5]	243 [6.6]	**9.0 [1.4]**	9.9 [1.7]	15.1 [1.5]	9.0 [1.4]
φ, deg	1	**5.3**	6.8	65.2	5.5	5.3	6.8	5.5
	2	4.3	5.8	9.0	4.9	**4.0**	5.8	4.9
	3	6.4	5.7	7.6	6.0	6.1	**4.9**	6.0
	4	4.8	9.2	**3.3**	4.8	4.8	9.2	4.8
	5	8.2	7.7	**3.6**	4.5	5.4	6.5	5.4
	GM	14.6 [1.4]	**10.3 [1.74]**	305 [13.8]	10.6 [1.6]	14.6 [1.4]	11.1 [1.6]	10.6 [1.6]

Table 4. Percentage of correct classification for trained classifiers

Type	Input variable set						
	$\{V_1\}$	$\{V_2\}$	$\{V_3\}$	$\{V_1\}$, $\{V_2\}$	$\{V_1\}$, $\{V_3\}$	$\{V_2\}$, $\{V_3\}$	All
1	81.2	85.0	81.2	86.3	82.5	**88.0**	82.1
2	96.6	95.7	83.3	**97.0**	96.2	95.7	97.0
3	86.3	85.0	81.6	84.6	85.0	81.6	**86.8**
4	97.9	**98.7**	79.5	97.8	97.9	98.7	98.3
5	78.6	**82.1**	79.9	79.5	78.2	79.5	80.8

Insomuch as this neural network is based on the inductive principles of self-organization of models, the very process of the self-organization of its structure serves not only as a means of obtaining the final model but also as a tool for analysis. Thus, based on the selection of appropriate input variables on each layer of the network by active neurons, we can estimate the contribution of the sensor data to the overall dependency. The received structures of the GMDH-type neural networks for the best models of robot pose estimation are shown in Fig. 6.

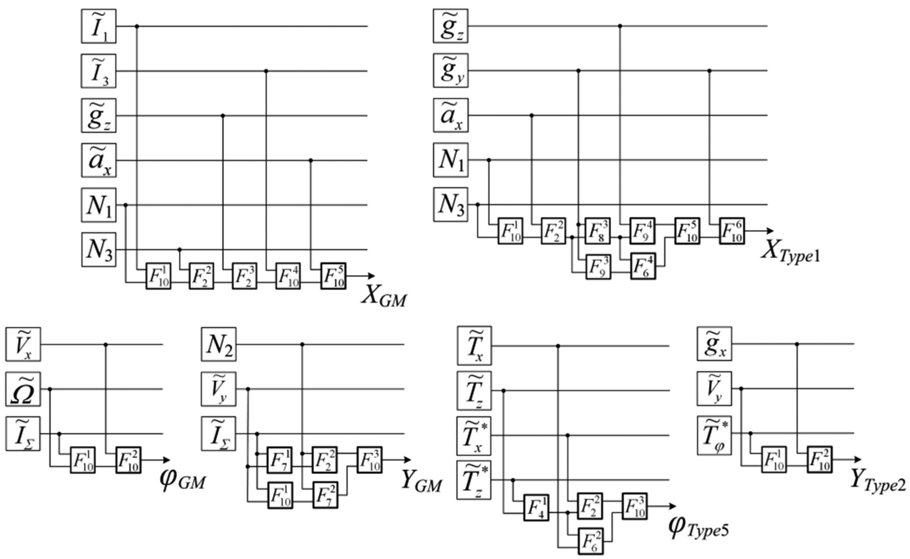

Fig. 6. Structures of twice-multilayered modified polynomial neural networks for the some best trained models ("∼" – average value per half-second interval, "Type" – the type of underlying surface (see Table 3), the networks evaluates X, Y and φ in 0.5 s of robot's movement)

Active neurons first of all choose input parameters taking into account their direct physical correspondence with the output variable, although the GMDH algorithm constructs non-physical models. For example:

- In case X_{GM} in Fig. 6 the neurons chosen the values of I_1, I_3 and N_1, N_3, which are directly related to X-coordinate (see Fig. 1c). The a_x is also directly related to the output variable.
- In case Y_{GM} the neurons chosen the V_y and N_2, which are directly related to Y-coordinate (see Fig. 1c). The choice of I_Σ is also associated with the Y-variable evaluation, since, as mentioned above, for each type of surface this parameter has different values (see Fig. 5a).

At the same time, the choice of some input parameters is not obvious, because the features of robot-terrain interaction for specific types of surfaces are taken into account (for example, g_y and g_z for X_{Type1}, g_x and T^{*}_{φ} for Y_{Type2}).

Analysis of classifiers shows that in order to construct better models, active neurons in all cases use the variables of $\{V_2\}$.

4.2 Results of Testing the Trained Models During the Robot Movement in a Heterogeneous Environment

This section shows some results of testing of the trained models in an essentially heterogeneous environment, when the areas of the surfaces are comparable with the robot's dimensions. Thus, the test conditions for navigation are much more difficult than at the training stage. First, at the training stage the robot moved along separate homogeneous surfaces of several types. But in this test, the effects of not only the influence but also the mutual influence of the properties of different surfaces on the robot's movement appear. Secondly, such transition zones between surfaces (when different wheels are simultaneously located at different surfaces) appear often during the movement, which accelerates the accumulation of navigation errors.

The task was to movement along the triangle (lengths of the sides are assigned by operator) through the surfaces of different types, using only the on-board sensor readings (without the signal from the global positioning system) and the trained models. Trained models were used by the robot to determine the achievement of the vertexes of triangle (with the aim of changing the movement direction) and to correct the trajectory during the movement. To determine the deviations from the desired trajectory the outputs of these models were used as a feedback signals instead of GPS signals. With these deviations, the control signals for the robot motions are generated by the method of proportional regulation well-known in the automatic control theory.

The purpose of the series of experiments was to determine the performance of the best trained models (see Table 3) for two cases:

- movement for mentioned above conditions by means of models general to all types of surfaces (denoted "GM" in Table 3);
- movement under the same conditions using coordinates evaluation models specialized for a specific type of surface (we denote this models as "MT" ("Model for Type")). These models are selected by signal from the corresponding classifier. If signals from all classifiers are absent or there are signals from several classifiers, then the coordinates and orientation angle are evaluated by the "GM"-models.

In Fig. 7 shows the final trajectory of the robot along the specified sides of triangle.

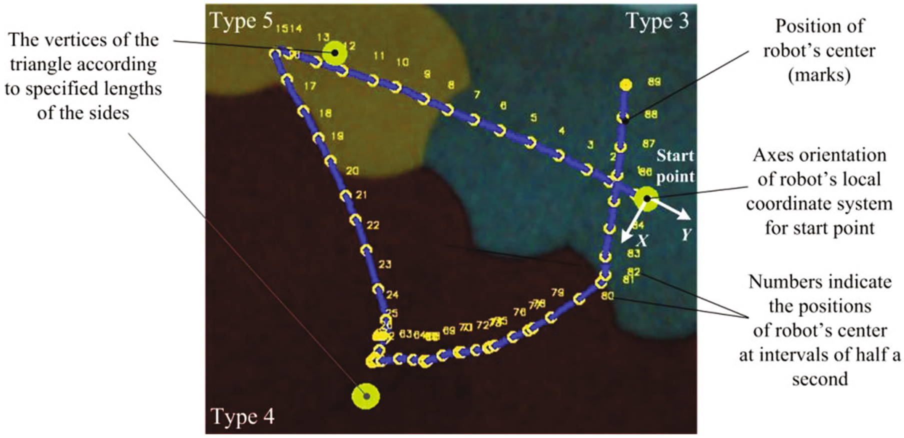

Fig. 7. Trajectory of robot movement obtained by means of "GM"-models

The achievement of the coordinates of the vertices of the triangle was estimated by means of the best "GM" models for X, Y and φ. Also these models were used for trajectory correction during the robot movement. In this experiment, the length of the side of triangle was set at 0.5 m. In combination with three different types of surfaces and the robot's diameter of 37 cm provides specified conditions for testing the models, because about half of path the robot moves through the transition zones. For example, already at points 7 and 12 the robot's wheels are simultaneously located on two surfaces: for the first point – both on 3 and 5 types of surfaces, for the second point – both on 4 and 5 types of surfaces.

In Fig. 8b and d shows the evaluation of the X and Y coordinates by means of both the specified models (X_{Type3}, X_{Type4}, X_{Type5}, Y_{Type3}, Y_{Type4}, Y_{Type5}) for three types of surfaces (see Fig. 7) and best general models for all types of surfaces (X_{GM}, Y_{GM}). As X_{CSV} and Y_{CSV} are denoted real coordinates of robot movement detected by computer vision system (see Fig. 3). As can be seen from the curves of X_{CSV} and Y_{CSV}, the movement of the robot is complex (for example, deviations of coordinates for a period of 15-23 robot's steps and pause in the movement for a period of 30–57 steps) and is not rectilinear, which indicates a significant influence of the surfaces on it. It can also be seen that on separate time intervals, different "MT"-models are more accurate than general models. At the same time, general models (X_{GM}, Y_{GM}) show an acceptable (in average) result during the overall time of the movement.

Figure 8a and c show the errors in determining both X and Y coordinates by the best "GM"-models (X_Error_{GM}, Y_Error_{GM}) and the set of specialized models (X_Error_{MT}, Y_Error_{MT}) that are selected at the moment the classifier of surface type is triggered (output is equal "1") in accordance with the above rule. In general, the trained classification models demonstrate their operability. However, in addition to its own classification errors (for example, at the interval 55–75 steps in Fig. 8f), there are errors

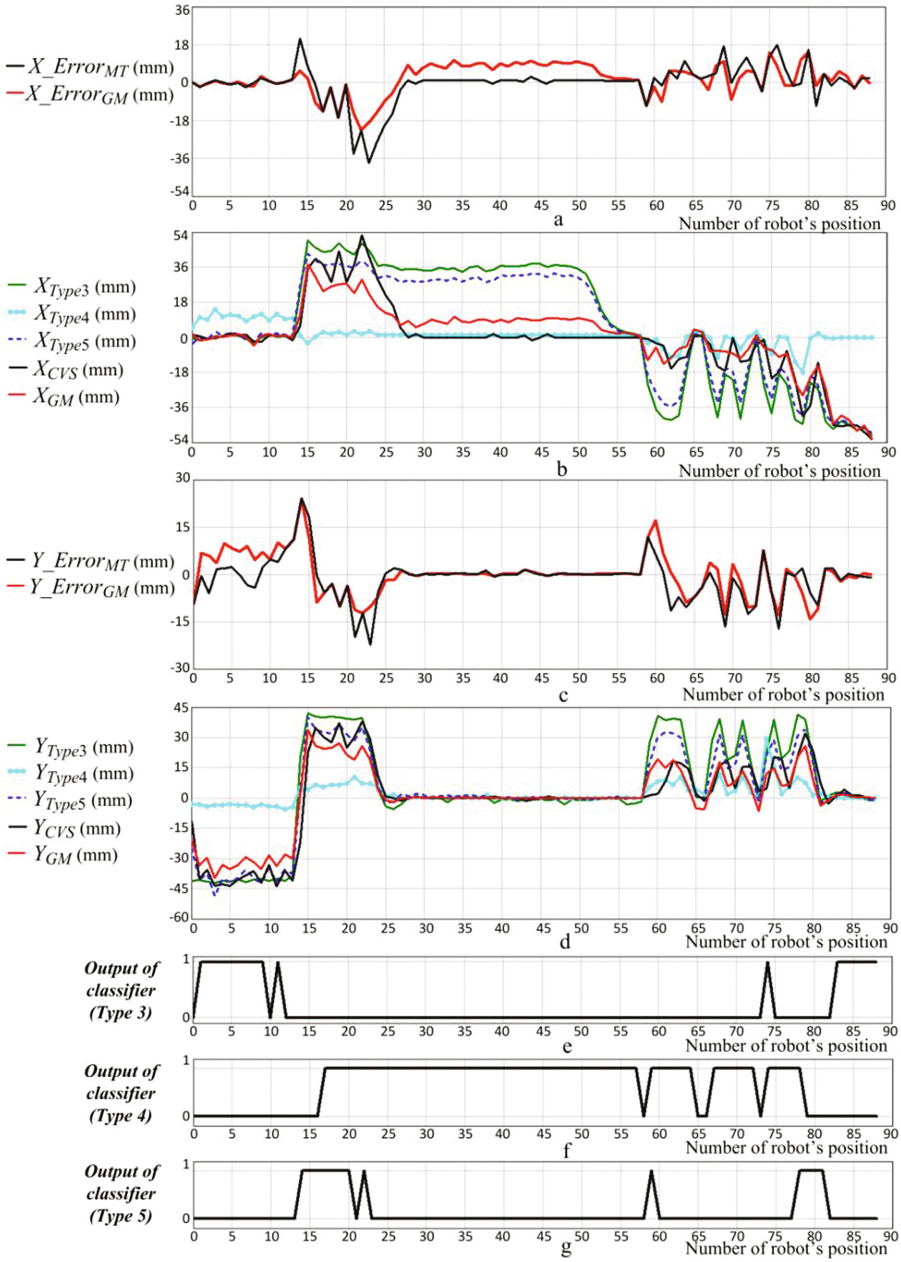

Fig. 8. Graphs of both outputs and errors for models of evaluating coordinates and for classification models during the robot movement

caused by the transition of the robot from one surface to another (for example, at the interval of 15–20 steps in Fig. 8f and g, two classifiers are triggered simultaneously).

In Fig. 8a, b , c and d all values of output variables for each robot's step are shown in robot's local coordinate system (Fig. 1c).

Based on the results of testing the models, we can conclude the following:

- With the simultaneous contact of the robot wheels with surfaces of different types, there are classification errors (i.e. errors of selection of "MT"-models) and errors of "MT"-models. This leads to the fact that the use of general (i.e., averaged for all types of surfaces) and specified models gives a comparable result.
- In general, all models (both pose estimation and classifications) demonstrate their performance by an example much more complex than the conditions for their training. This indicates the operability of developed learning navigation system (in point of view quality of obtained models) and practical applicability of GMDH to solving the problem of local navigation of a robot.

It should be noted that the purpose of this section was to test the obtained models, and not to solve complex questions of developing a system for local navigation. The focus of this study is the synthesis of models, but to increase the accuracy of the local navigation system, it is necessary to consider a wider range of issues related more to the stage of using the obtained models, rather than to the stage of their training. In order to improve the quality of the obtained trajectory and the accuracy of movement to the given coordinates, it is necessary to solve the following tasks: synthesis of a better regulator for robot motion along the desired trajectory; the development of methods to use the classifiers and "MT"-models according to considered conditions and etc.

In Fig. 8b and d the accuracy both "GM"- and "MT"-models during test motion correlates with its accuracy on training stage (see Table 3).

5 Conclusion

We have obtained higher accuracy (arithmetic mean error is less) of models for evaluating the coordinates and angular orientation than in our previous research [10, 11] due to the extension of the input parameters set (only $\{N\}$, $\{\omega\}$ and $\{I\}$ were used in past).

In general case, it is not sufficient to use a certain subset of input parameters to obtain better models for different outputs (X, Y and φ) and underlying surface types. We recommend using parameters derived from sensor readings (for example, $\{V_2\}$ and $\{V_3\}$) to improve the quality of the models. For example, it is interesting to note that the variables of $\{V_2\}$ were selected by active neurons to train classifiers for each of the five types of surfaces (also see best training results (highlighted in bold) in Table 4). Also it should be noted that the physical meaning of the set of parameters $\{V_2\}$ and $\{V_3\}$ is not associated to this testing ground and this robot, which makes it possible to use them in other projects on the same subject.

The results of testing of the trained models (both pose estimation and classification) demonstrate their performance in an essentially heterogeneous environment (the areas of the surfaces are comparable with the robot's dimensions). It is important to note that the conditions for testing of models were much more complicated than the conditions for their training. First, during the training there were no transition zones, when different wheels are simultaneously located at different surfaces. Secondly, the movement in these zones was half of the path. Thirdly, the control actions on the motors were changed dynamically (during the robot movement in Fig. 7) due to correction of both the trajectory and the orientation angle (at the training stage, the control actions were statically assigned from a specified set of motor velocities). As result, the parameters of robot movement and, as a consequence, the readings of the sensors were significantly different from those observed at the training stage. Fourthly, the models (both for pose estimation and classification) were used as a feedback signal for motors control, which could disturb stability of robot movement along the trajectory. Thus, testing was a very serious test for bias of models. From this point of view, the results shown in Fig. 8 testify the practical possibility of models training based on GMDH for task of local navigation in heterogeneous environment.

Future work is the development of methods and algorithms for applying both the trained models and learning navigation system to construct the local navigation system based on the readings of on-board sensors (in the absence of a GPS-signals) in heterogeneous environment.

References

1. Rogers-Marcovitz, F., George, M., Seegmiller, N., Kelly, A.: Aiding off-road inertial navigation with high performance models of wheel slip. In: Proceedings of the IEEE International Conference on Intelligent Robots and Systems, pp. 215–222. IEEE, Vilamoura, Portugal (2012)
2. Madhavan, R., Nettleton, E., Nebot, E., Dissanayake, G., Cunningham, J., Durrant-Whyte, H., Corke, P., Roberts, J.: Evaluation of internal navigation sensor suites for underground mining vehicle navigation. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 999–1004. IEEE, Detroit, MI, USA (1999)
3. Koch, J., Hillenbrand, C., Bems, K.: Inertial navigation for wheeled robots in outdoor terrain. In: Fifth International Workshop on Robot Motion and Control, pp. 169–174. IEEE, Dymaczewo, Poland (2005)
4. Bingbing, L., Adams, M., Ibañez-Guzmán, J.: Multi-aided inertial navigation for ground vehicles in outdoor uneven environments. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 4703–4708. IEEE, Barcelona, Spain (2005)
5. Liu, Y., Xiong, R., Wang, Y., Huang, H., Xie, X., Liu, X., Zhang, G.: Stereo visual-inertial odometry with multiple kalman filters ensemble. IEEE Trans. Indus. Electron. **63**(10), 6205–6216 (2016)
6. Dupont, E., Collins, E., Coyle, E., Roberts, R.: Terrain classification using vibration sensors: theory and methods. In: New Research on Mobile Robotics, pp. 1–41 (2010)
7. Ojeda, L., Cruz, D., Reina, G., Borenstein, J.: Current-based slippage detection and odometry correction for mobile robots and planetary rovers. IEEE Trans. Robot. **22**(2), 366–378 (2006)
8. Iagnemma, K., Ward, C.: Classification-based wheel slip detection and detector fusion for mobile robots on outdoor terrain. Auton. Robots **26**(1), 33–46 (2009)
9. Khaleghian, S., Taheri, S.: Terrain classification using intelligent tire. J. Terramech. **71**, 15–24 (2017)

10. Andrakhanov, A.: Technology of autonomous mobile robot control based on the inductive method of self-organization of models. In: Proceedings of the 7th International Symposium “Robotics for Risky Environment – Extreme Robotics”, pp. 361–368. Saint-Petersburg, Russia (2013)
11. Andrakhanov, A.: Navigation of autonomous mobile robot in homogeneous and heterogeneous environments on basis of GMDH neural networks. In: Proceedings of the 4th International Conference on Inductive Modelling, pp. 133–138. Kiev, Ukraine (2013)
12. Tyryshkin, A., Andrakhanov, A., Orlov, A.: GMDH-based modified polynomial neural network algorithm.In: GMDH-methodology and Implementation in C (With CD-ROM), Imperial College Press, World Scientific, London (2015). ISBN: 978-1-84816-610-3
13. Robotino Manual (Order No. 544305). http://www.festo-didactic.com/ov3/media/customers/1100/544305_robotino_deen2.pdf. Accessed 05 June 2017
14. Ivakhnenko, A., Ivakhnenko, G.: The review of problems solvable by algorithms of the group method of data handling. Pattern Recogn. Image Anal. Adv. Math. Theory Appl. **5**(4), 527–535 (1994)
15. Ivakhnenko, A., Ivakhnenko, G., Mueller, J.: Self-organization of neuronets with active neurons. Int. J. Pattern Recogn. Image Anal. Adv. Math. Theory Appl. **4**(4), 177–188 (1994)
16. Madala, H.R., Ivakhnenko, A.G.: Inductive Learning Algorithms for Complex System Modeling. CRC Press, Boca Raton (1994). ISBN 0-8493-4438-7
17. Martin, S., Murphy, L., Corke, P.: Building large scale traversability maps using vehicle experience. In: Desai, J.P., Dudek, G., Khatib, O., Kumar, V. (eds.) 13th International Symposium on Experimental Robotics 2012, STAR, vol. 88, pp. 891–905. Springer, Heidelberg (2013)