ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/315005604

# Autonomous Underwater Vehicle Mathematical Model and Simulator

**Conference Paper** · December 2016

DOI: 10.1145/3029610.3029639

CITATIONS
6

READS
1,412

**2 authors:**

Roman Fedorenko
Innopolis University
**36** PUBLICATIONS **150** CITATIONS

SEE PROFILE

B.V. Gurenko
Southern Federal University
**38** PUBLICATIONS **154** CITATIONS

SEE PROFILE

**Some of the authors of this publication are also working on these related projects:**

Project Airships automatic control View project

Project Tensegrity Robotics. View project

All content following this page was uploaded by B.V. Gurenko on 17 October 2017.

The user has requested enhancement of the downloaded file.

# Autonomous Underwater Vehicle Mathematical Model and Simulator

Roman Fedorenko
Southern Federal University
347922 Taganrog, Russian Federation
44 Nekrasovskiy
+7 (8634) 37-16-94
frontwise@gmail.com

Boris Gurenko
Southern Federal University
347922 Taganrog, Russian Federation
44 Nekrasovskiy
+7 (8634) 37-16-94
boris.gurenko@gmail.com

## ABSTRACT
Testing and debugging of real equipment is a time consuming task. In particular, in the case of marine robots, it is necessary each time to carry out the transportation and deployment of a robot on the water. Experiments with not yet fully functional prototype of marine robot equipped with expensive hardware is in the meantime very risky. Therefore, the use of simulators is affordable way to accelerate the development of robotic systems from the viewpoint of labor effort and cost of experiments. This paper presents a simulator specifically designed for autonomous unmanned underwater vehicles.

## CCS Concepts
• **Computer systems organization→Embedded software** • **Computer systems organization→Robotic control** • **Computer systems organization→Robotic autonomy**

## Keywords
Use case; control system; autonomous underwater vehicle; mathematical model; simulator; architecture; software.

## 1. INTRODUCTION
Marine robotics is rapidly developing [1-3]. Modern underwater vehicles are capable of performing a wide range of tasks, which includes environmental and climate monitoring, oceanographic research, deep-water systems and devices service, underwater fields search, protection of water areas, etc. Autonomous underwater vehicles (AUV) can most effectively address all these tasks. Use of AUVs reduces operating costs and probability of error by decreasing the degree of involvement of the human operator, increasing uptime of mobile systems, reduction in weight and size of underwater vehicles and others. However, to design the control system of such a complex object, it is necessary to construct a mathematical model that would fully take into account all of the parameters unsteadiness, the nonlinearity and multiple-connection of the dynamics of AUV as an object of control, the forces of interaction of the body with a viscous environment.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
*ICCMA '16*, December 07-11, 2016, Barcelona, Spain
© 2016 ACM. ISBN 978-1-4503-5213-0/16/12…$15.00
DOI: http://dx.doi.org/10.1145/3029610.3029639

Despite the practical importance of the development of autonomous underwater vehicles, remoteness and aggressiveness of the environment in which they operate, makes the development process quite complicated. In addition to the necessity of the delivery and deployment of the robot in the operational environment, there is also a risk of damage to expensive equipment throughout the process of debugging.

Using a simulation environment is a key tool that reducing the relevance of the problems mentioned above. In addition, the use of the simulator allows simulating specific conditions that are difficult or dangerously reproducible in real tests, including equipment, power supply failure, external disturbances and obstacles [4].

Another advantage of simulation is the possibility of simultaneous usage of simulator by several members of the team, which is impossible when dealing with real prototype, which is usually produced in a single copy.

Autonomous underwater vehicle (AUV) simulation tool requires the development of a mathematical model and three-dimensional modeling and visualization of the environment, the operating mode in real time, flexibility, and ability to work on different computer configurations and high integration with the robot control system.

The last requirement means the necessity for equal and uniform operation of control system with both simulated and real physical devices (such as sensors, actuators).

This paper presents the development process of AUV simulator that satisfies the requirements listed above. This simulator is designed to simulate the process of automatic control, using of control systems, navigation, communication and information analysis of autonomous underwater vehicles with a high level of information autonomy that using synergistic regulators and neural network motion planners for the intellectual organization of behavioral strategies and trajectory planning in uncertain environments and nonlinear observers to estimate the immeasurable external and parametric disturbances acting on underwater vehicles [5-7].

Section I of the article presents AUV mathematical model used in simulator core, Section II describes the architecture of the simulator. Some examples of the simulation are presented in section III.

## 2. MATHEMATICAL MODEL OF AUV
To derive a mathematical model of AUV two Cartesian coordinate systems shown in Figure 1 were used.

Mathematical model of AUV can be presented in the following vector-matrix form on the basis of the known equations of a rigid body kinematics and dynamics:

$$\dot{Y} = \Sigma(\theta, X) = \Sigma\begin{pmatrix} \Sigma_P(\theta, X) \\ \Sigma_\theta(\theta, X) \end{pmatrix} \tag{1}$$

$$M\dot{X} = \left[F_d(P, V, \omega) + F_u(\delta) + F_v(G, A_{\Pi}, R_{\Gamma})\right] \tag{2}$$

$$T_{uy}\frac{d\delta}{dt} + \bar{\delta} = \Psi_{uy}(\delta, U) \tag{3}$$

where $T_{uy}$ is a diagonal matrix of time constants of actuators and $\Psi$ is a vector of nonlinear functions of right sides of the actuators dynamics;

$\bar{\delta}$ - AUV's actuators inputs vector;

$\bar{U}$ - controls vector formed by the control system of AUV;

$X = [V^T \omega^T]^T$– vector of internal coordinates. Its components are the projections of vectors of the linear $V$ and angular $\omega$ speed of AUV on the axes $OX$, $OY$ and $OZ$ of body-fixed coordinate system $K$;

$M$ - (m × m)-matrix of mass and inertial parameters, whose elements are the mass, moments of inertia, added masses of AUV;

$F_u(\delta)$ - m-vector of control forces and moments;

$F_d(P, V, \omega)$ - m-vector of non-linear elements of the AUV's dynamics;

$F_v(G, A_{\Pi}, R_{\Gamma})$ - m-vector of measured and unmeasured external forces and moments of gravity $G$, buoyancy $A_{\Pi}$ and hydrodynamics $R_{\Gamma}$;

$Y$ - n-vector of position $P$ and orientation $\Theta$ (output coordinates) of body coordinate system relative to the base coordinate system;

$\sum(\Theta, X)$ - n-vector of kinematic constraints, consisting of a vector of linear velocity $\Sigma_P(\theta, X) = A^T V$ and angular velocity $\Sigma_P(\theta, X) = A_\omega \omega$ related to the coordinate system relative to the base;

$\sum_P(\Theta, X)$ - vector of linear velocities of the body coordinate system relative to the base coordinate system;

$\sum_\Theta(\Theta, X)$ - the angular velocity of the body frame relative to the base coordinate system.

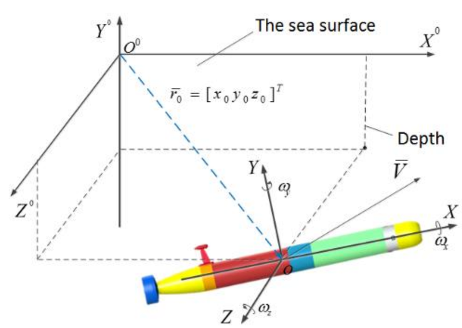



**Figure 1. $K^0(OX^0Y^0Z^0)$ and K(OX Y Z) coordinate frames**

# Hydrodynamic coefficients and added masses

Expressions for the calculation of hydrodynamic coefficients and added masses are of the form:

$$F_x = k\cdot\left(cx_0 + cx_V\cdot V_0 + cx_{a_2}\cdot\alpha_0^2 + cx_{b_2}\cdot\beta_0^2\right),$$

$F_y = k\cdot(V_0{}^2\cdot(cy_0 + cy_a\cdot\alpha_0 + cy_{a_3}\cdot\alpha_0{}^3 + cy_{a_2}\cdot\alpha_0\cdot|\alpha_0| + cy_b\cdot|\beta_0| + cy_{b_2}\cdot\beta_0{}^2) + V_0\cdot 1{,}02\cdot\omega_z),$

$$F_z = k\cdot\left(V_0{}^2\cdot\left(cz_b\cdot\beta_0 + cz_{b_2}\cdot\beta_0\cdot|\beta_0| + cz_{b_3}\cdot\beta_0\cdot\beta_0{}^2 + cz_{ab_2}\cdot\beta_0\cdot|\alpha_0|\right) + V_0\cdot\left(-1{,}19\cdot\omega_y\right)\right),$$

$M_x = k_m\cdot(V_0{}^2\cdot(mx_b\cdot\beta_0 + mx_{ab}\cdot\beta_0\cdot\alpha_0 + 0\cdot\omega_x + mx_{ab_2}\cdot\alpha_0\cdot\beta_0\cdot|\beta_0|) + V_0\cdot(-0{,}090\cdot 1\cdot\omega_x)),$

$M_y = k_m\cdot(V_0{}^2\cdot(my_b\cdot\beta_0 + my_{b_2}\cdot\beta_0\cdot|\beta_0|) + V_0\cdot(-1{,}23\cdot 1\cdot\omega_y))$ ,

$M_z = k_m\cdot(V_0{}^2\cdot(mz_0 + mz_a\cdot\alpha_0 + mz_{a_2}\cdot\alpha_0\cdot|\alpha_0| + mz_{ab}\cdot\alpha_0\cdot\beta_0) + V_0(-1{,}181\cdot 1\cdot\omega_z)),$

$$\begin{cases} \lambda_{11} = k_{11}\rho U; \quad \lambda_{22} = k_{22}\rho U; \\ \lambda_{33} = k_{33}\rho U; \quad \lambda_{26} = k_{26}\rho UL; \\ \lambda_{35} = k_{35}\rho UL; \lambda_{66} = k_{66}\rho UL^2 \\ \lambda_{44} = k_{44}\rho UL^2; \quad \lambda_{55} = k_{55}\rho UL^2; \end{cases} \tag{4}$$

where $V_0=(V_x{}^2+V_y{}^2+V_z{}^2)^{0,5}$ , $\alpha_0=\arctan(-V_y/V_x)$ , $\beta_0=\arcsin(V_z/V_0)$, $\rho$ – density of water, U – the volume of AUV (U=0.2793), $S=U^{2/3}$, $k=\rho\cdot S\cdot V_0{}^{2/3}$;

$cx_0=0.06805$, $cx_V=-3.57\cdot10^{-3}$, $cx_{a_2}=-0.204$, $cx_{b_2}=-0.069$;

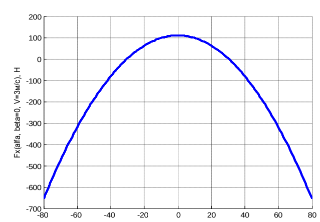
$cy_0=-4{,}011\cdot10^{-4}$ , $cy_a=1{,}25$ , $cy_{a_2}=0{,}312$ , $cy_{a_3}=-0{,}224$ , $cy_b=-0{,}00701$, $cy_{b_2}=-0{,}420$;

$cz_b=-1{,}174$, $cz_{b_2}=-0{,}449$, $cz_{b_3}=0{,}34$, $cz_{ab_2}=-0{,}095$;

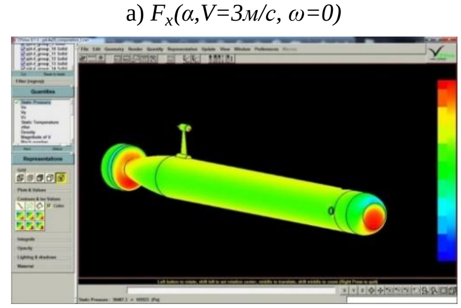
$mx_b=9{,}30\cdot10^{-3}$, $mx_{ab}=0{,}131$, $mx_{ab_2}=-0{,}064$;

$my_b=0{,}0680$, $my_{b_2}=0{,}0389$;

$mz_0=5{,}47\cdot10^{-4}$, $mz_a=0{,}0289$, $mz_{a_2}=0{,}0855$, $mz_{ab}=3{,}02\cdot10^{-5}$;




a) $F_x(\alpha, V=3м/с, \omega=0)$



b) The distribution of the hydrodynamic pressure on the surface of the vehicle

**Figure 2. Dependence of the hydrodynamic force $F_x$, on the angle of attack α at a speed V = 3 m/s**

Pressure distribution on AUV's enclosure with fins (see Figure 2b) and dependence of the coefficient of drag (see Figure 2a) on the angle of attack were investigated by means of a numerical simulation and hydrodynamic characteristics study package by NUMECA International.

# 3. THE ARCHITECTURE OF AUV SIMULATOR

The universal environments of mathematical modeling, which are used in the development of robotic systems (such as, for example, MATLAB and Simulink), allow performing numerical simulations and visualizing of results with plots. Despite the fact that these tools are very useful for the development of individual modules and algorithms, their use for the simulation of the robot system in general, including the interaction with the hardware, three-dimensional visualization, is constrained. Thus, the use of the universal environments of mathematical modeling is not enough and specialized solution is required.

Existing specialized simulators are intended only for water surface ships, or are proprietary and have a closed architecture [8-10]. To our knowledge, none of the simulators at the same time satisfy all the requirements listed above. In this regard, it was decided to develop own specialized simulator.

Developed simulator is a complex of three applications:

– application of AUV and the marine environment simulation (including blocks of modeling of the marine environment, the hydrodynamic characteristics, characteristics of the power plant, actuators, AUV motion in common);
– application of three-dimensional visualization and simulation of the obstacles avoidance system of AUV (including the unit three-dimensional visualization of motion of AUV and the environment);
– application of shipborne control station and information display (operator's interface).

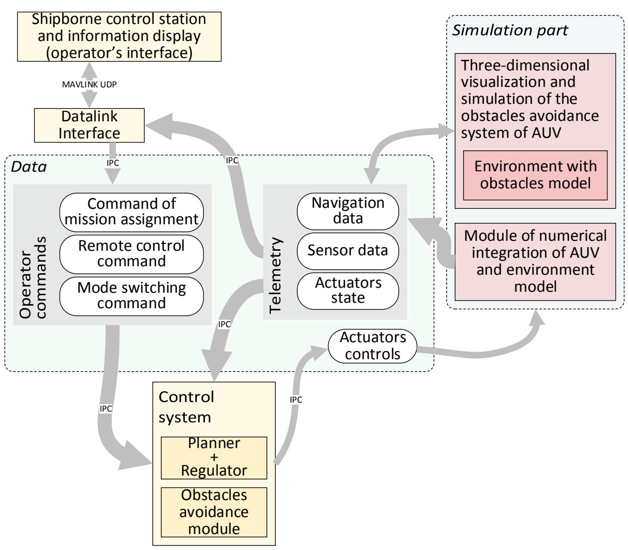



**Figure 3. Functional scheme of the simulator**

Applications of simulation complex are integrated with each other; their interaction is carried out automatically by the TCP/IP and UDP protocols.

The simulation program of AUV and marine environment MarineSimulation is the main application of the complex and can be run separately, but the program of three-dimensional visualization and simulation of the AUV obstacles avoidance system – VTModelingSystem and shipborne control station and information display – YellowShark extend the functionality of the complex.

Functional scheme of the simulator is shown in

Figure 3.

Note that the simulator has a modular architecture and includes modules of on-board control system and ship control station and information display as well as the communication channel interface, marked with orange filling in the scheme, which operate equal in simulation and at real experiments.

The data exchange between modules is carried out through universal data interfaces – "Named Pipes", highlighted in gray in the diagram. When running in simulation mode telemetry data is published and actuators controls are read by the simulator software modules. At real tests telemetry data published by relevant software interfaces of sensors and navigation system, and actuators controls are transmitted to the electronic control units of the actuators through appropriate software interfaces. Thus, the equal operation of control system units in simulation and in real tests is realized. Commands from the shipborne control station and information display (operator's interface) are transmitted to the control system, and telemetry data to the shipborne control station and information display is the same way in simulation and in real tests. Thus, there is no difference in operating task for the operator in the simulation mode from the real operation of the control system.

For the data exchange between the control system and shipborne control station and information display (operator's interface) MAVLink protocol is used.

MAVLink format is designed for packing and unpacking message structures (in the languages C / C ++, C # or Python) to send them over the communication channel. Message format is defined in file XML and then converted to code in languages C / C ++, C # or Python.

Simulation of AUV environment with obstacles as well as three-dimensional visualization is a separate simulator's module. This module receives the telemetry data from the numerical integration module to set the position of AUV and states of its actuators. The output of the module is simulated sonar data by which the module of obstacle avoidance of control system performs obstacle avoidance.

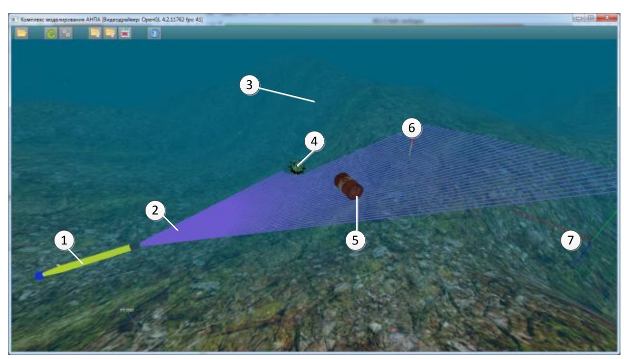


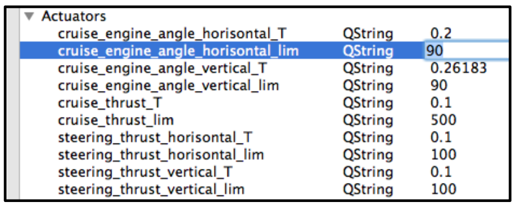

**Figure 4. Appearance of the program after loading project scene's test**

The module also provides a three-dimensional visualization of the operational environment of AUV (scene). The modeling scene consists of the following elements, shown in Figure 4:

– model of AUV (item 1);
– model sensor subsystem of AUV - Sonar (item 2);
– underwater landscape (item 3);
– obstacles (items 4, 5);
– the current target object (item 6);
– the coordinate's origin of the scene (item 7).

Configuring of the simulator is performed by a file in INI format and GUI that helps modify this file in a hierarchical structure as shown in Figure 5. Settings of the simulator are:

– characteristics of AUV;
– AUV's actuators characteristics ;
– setting of AUV's control system;
– power consumption characteristics of the payload;
– setting the AUV obstacles avoidance system;
– settings of flow simulation.



| Actuators | | |
|---|---|---|
| cruise_engine_angle_horisontal_T | QString | 0.2 |
| cruise_engine_angle_horisontal_lim | QString | 90 |
| cruise_engine_angle_vertical_T | QString | 0.26183 |
| cruise_engine_angle_vertical_lim | QString | 90 |
| cruise_thrust_T | QString | 0.1 |
| cruise_thrust_lim | QString | 500 |
| steering_thrust_horisontal_T | QString | 0.1 |
| steering_thrust_horisontal_lim | QString | 100 |
| steering_thrust_vertical_T | QString | 0.1 |
| steering_thrust_vertical_lim | QString | 100 |

**Figure 5. Settings editing of simulation complex**

# 4. Simulator Use Cases

Developed simulator allows to test AUV in various conditions, estimate energy consumption, check fault detection, hydro acousticsonar and so on. Simulation of different variants of AUV movement

Ability to simulate different variants of AUV's movement is shown in Figure 6 (various options of the motion trajectory are set).

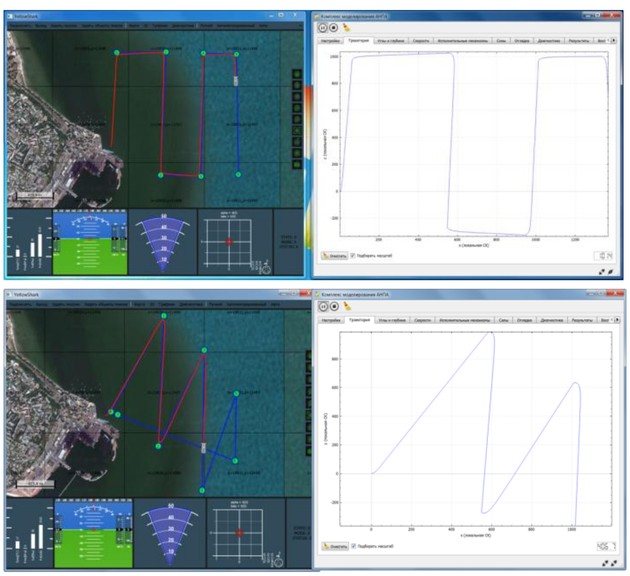


**Figure 6. Simulation of different variants of AUV's movement: left - defined (blue) and simulated (red) trajectory of AUV on the map; right - the result trajectory in the form of a coordinates plot**

## 4.1 Estimating the energy consumption

The dependence of the energy consumption from the payloads' mode usage is shown in Figure 7. In the section 0-1 payload 1 is on, in the section 1-2 payloads 1 and 2 are on, in the section 3 no payload is operating.

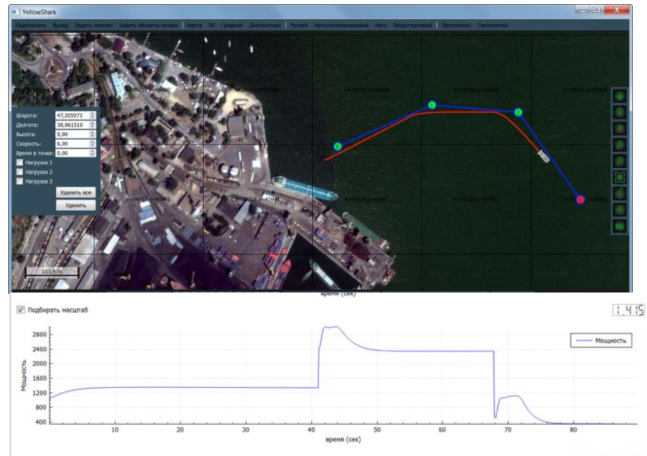


**Figure 7. The dependence of the energy consumption from the payloads' mode usage**

## 4.2 Fault detection simulation

Simulation and fault diagnostics is carried out through the sections of AUV, as shown in Figure 8. Block status can be one of the following: normal (without light or green when hovering), operating with errors (yellow) and emergency (red).

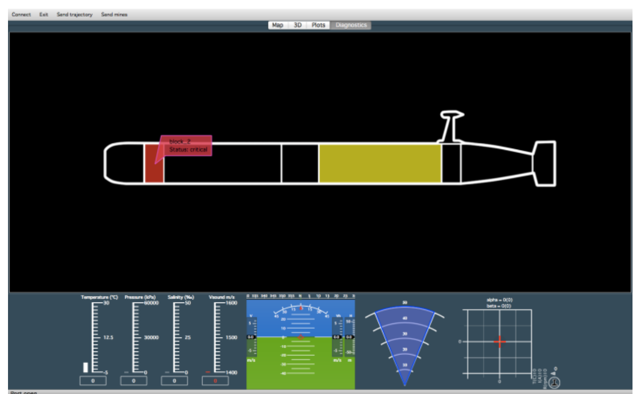


**Figure 8. Diagnostic view**

## 4.3 Simulation of the hydroacoustic sonar

A window that displays an adapted information from model of the hydroacoustic sonar is shown in Figure 9.

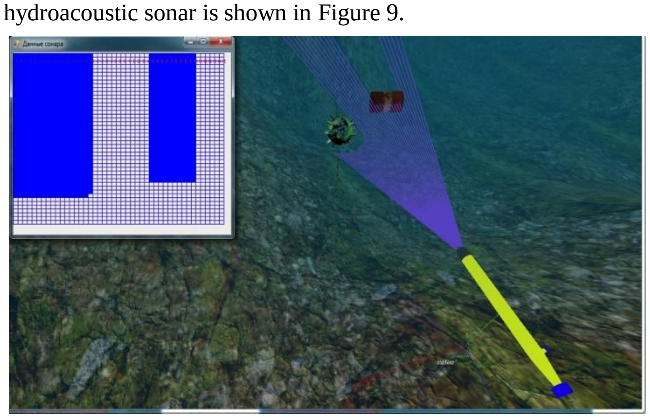


**Figure 9. Window displaying adapted for neural network model information from the hydroacoustic sonar**

### 4.4 The search of underwater objects process simulation

In the search process simulation indication of object recognition is implemented. The accuracy of detection of objects is determined by the search accuracy of AUV navigation system, which is set in the input parameters of simulation. Coordinates data of detected objects, their number and the detection time are displayed in the simulation results and presented as plots (Figure 11).

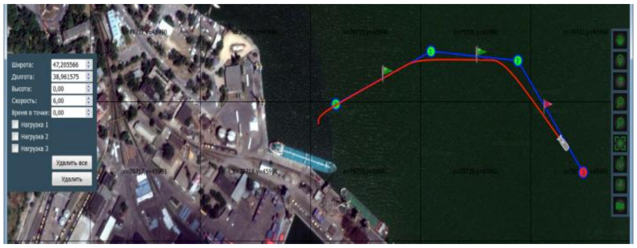


**Figure 10. Indication of search objects**

In Figure 9, the section 0-1 payload 1 is on, in the section 1-2 payloads 1 and 2 are on, in the section 3 – no payload is on. Green flag mark detected object, the red means that object was not found.

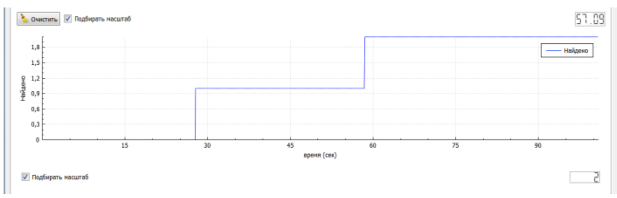


**Figure 11. The number of detected objects**

## 5. CONCLUSIONS

Development of underwater vehicles is a complex process where many resources are consumed by testing stage. Simulator, presented in this paper allows to simplify this procedure moving preliminary checks and trials to virtual environment. This significantly speeds up the development and saves resources. Besides, the proposed simulator can be noted for ability to simulate special conditions that can be hardly reproduced or even dangerous on real life, including equipment failures, power systems, external disturbances, obstacles and so on. The described architecture precisely copies the real environment, providing equal operation of control system modules in simulation and in the water. Thus the operator has no differences when working in simulator or performing the real mission.

## 6. ACKNOWLEDGEMENTS

The authors are deeply grateful to their supervisor professor Pshikhopov V.Kh. Research supported by Russian Fond of Basic Research, research work #114041540005 for state institutions of higher education and job research organizations in the field of scientific activities, grant from the President of the Russian Federation SSc-3437.2014.10, YSc-1098.2103.10 and grant of the president of Russian Federation for state support of young Russian scientists with p.h.d. MK-3089.2017.8.

## 7. REFERENCES

[1] von Alt, Christopher. "Autonomous underwater vehicles." *Autonomous Underwater Lagrangian Platforms and Sensors Workshop*. Vol. 3. 2003.

[2] Yuh, Junku, and Michael West. "Underwater robotics." *Advanced Robotics* 15.5 (2001): 609-639.

[3] Antonelli, Gianluca, Thor I. Fossen, and Dana R. Yoerger. "Underwater robotics." *Springer handbook of robotics*. Springer Berlin Heidelberg, 2008. 987-1008.

[4] Lewandowski, Edward M. "The dynamics of marine craft." *Training 2015* (2004): 08-17.

[5] Pshikhopov, V.Kh., Medvedev, M.Yu., Gaiduk, A.R., Gurenko, B.V., Control system design for autonomous underwater vehicle, 2013, *Proceedings - 2013 IEEE Latin American Robotics Symposium, LARS 2013*, pp. 77-82, doi:10.1109/LARS.2013.61.

[6] V.Kh. Pshikhopov, M.Yu. Medvedev, “Block design of robust control systems by direct Lyapunov method,” *IFAC World Congress*, Volume # 18, Part# 1, pp. 10875-10880, 2011, doi: 10.3182/20110828-6-IT-1002.00006.

[7] Pshikhopov, V.Kh., Ali, A.S., "Hybrid motion control of a mobile robot in dynamic environments," 2011, *Proceedings of IEEE International Conference on Mechatronics*, ICM 2011, pp. 540 – 545, doi: 10.1109/ICMECH.2011.5971345.

[8] Uriarte, Fabian M., and Robert Hebner. "Development of a multicore power system simulator for ship systems." *Electric Ship Technologies Symposium (ESTS)*, 2011 IEEE. IEEE, 2011.

[9] Xie, Yanhui, et al. "A PC-Cluster Based Real-Time Simulator for All-Electric Ship Integrated Power Systems Analysis and Optimization." *Electric Ship Technologies Symposium, 2007*. ESTS'07. IEEE. IEEE, 2007.

[10] Racine, Benjamin J., and Eric G. Paterson. "CFD-based method for simulation of marine-vehicle maneuvering." *35th AIAA Fluid Dynamics Conference and Exhibit, Toronto, Ontario Canada. 2005*.

View publication stats