Design and Debug Technology Complex Dynamic Systems

V. M. Nikiforov[1]
Academician Pilyugin Center
Moscow, Russia
[1]V.M.Nikiforov@gmail.ru

K. A. Neusypin, M. S. Selezneva
Bauman Moscow State Technical University
Moscow, Russia
neysipin@mail.ru

***Abstract*— The problem of "through" design of UGS with the help of a combination of 3-D programs and Matlab is considered. A mathematical model of the UGS is formed and its adequacy is proved.**

Keywords*— *CAD-technology; "Through" design; uniaxial gyrostabilizer

INTRODUCTION

Modern development of software related to the class of CAD-programs (computer-aided design) allows qualitatively changing the process of synthesis of dynamic systems (DS). Modern CAD-programs [1, 2] allow:

- create 3-D models of DS on the physical level;
- take into account the design features of the objects under consideration, which are difficult to describe mathematically;
- verify the correctness of constructive conjugation of various nodes at the stage of creating DS;
- produce design documentation that meets the requirements of a certain National Standards, etc.

Received 3-D models of DS are science and information-intensive objects. One of the most common 3-D modeling packages is Autodesk 3-d max, AutoCad, SolidWorks, Compass, etc. In particular, the SolidWorks was first created in 1993 and has found wide application because of the user-friendly interface [3, 4].

On the other hand, programs aimed at synthesis and analysis of DS are actively developing, such as MatLab, MathCad, Mathamatica, etc., which is working effectively with models represented by mathematical ones [5, 6, 7].

Today, the ability to integrate various programs, in particular, the SolidWorks and MatLab programs, allows obtaining effective methods of combined DS design by combining physical and mathematical modeling. The joint application of 3-D programs with MatLab makes it possible to build "Through" design technologies that significantly reduce the development cycle of DS by increasing it to a qualitatively new level. The ability to quickly create a graphical interface using the MatLab GUI (Graphical User Interfaces) and Windows application, gives the development process the flexibility and completeness.

The meaning of "Through" technology is the effective transmission of data and results of a particular current design stage to all subsequent stages. The basic of these technologies is the modular construction of CAD, the using of common databases and knowledge bases at all stages of project implementation and the characteristics of technologies are wide possibilities of modeling and control at all stages of design.

These technologies are of particular relevance in the design of complex DS, in particular, measuring complexes, including gyroscopic navigation systems [8, 9].

In the present article, using the example of a simplified uniaxial gyrostabilizer (UGS) model on a mobile base in the "coarse reduction (CR)" mode, we solved the task of terminal management and considered the main stages of the 3-D "through" design process.

UGS in CR mode is considered as an absolute rigid body rotating with respect to the stabilization axis in the form of an elementary solid platform having one degree of freedom around the axis of stabilization.

I. FIRST STAGE. CREATING A 3-D MODEL OF UGS ON A MOBILE BASE IN SOLIDWORKS

The physical model of UGS, built in SolidWorks, includes a gyroscope located in a certain way on a platform rotating relative to the stabilization axis. The body of the UGS rotates relative to the stationary base, Fig. 1, Fig. 2.

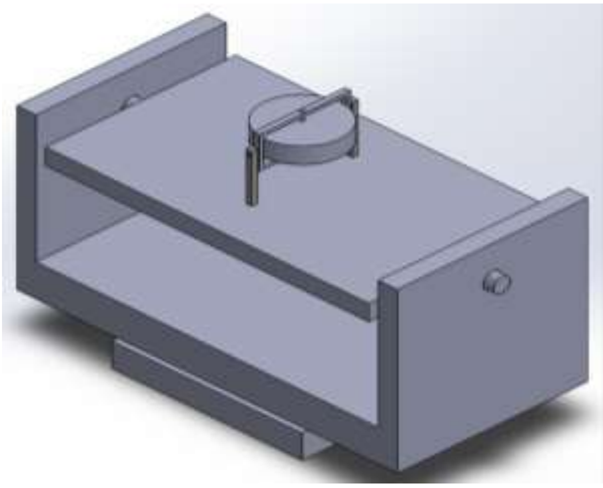

Fig. 1. (File cborka_platforma_ZEMLIA_f_4.slx)

978-1-5386-5824-6/18/$31.00 ©2018 IEEE

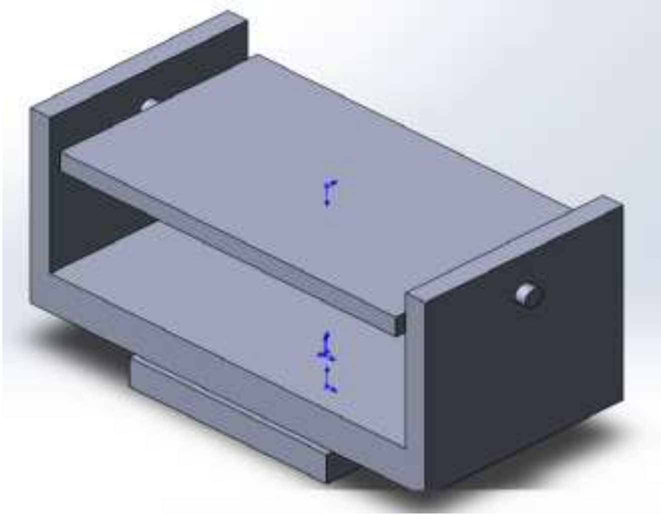

Fig. 2. (File cborka_platforma_ZEMLIA_f_4.slx)

Simplified 3-D model UGS in the "CR" mode, is a platform that rotates around the axis of stabilization of the body located on the moving base without taking into account the movement of the gyroscope, because gyroscope at precession reaches "technological stops" and does not take any further participation in the movement. All elements of the UGS construction are considered as absolutely rigid bodies. At this stage, a DS model is constructed at the physical level, which takes into account the materials of the parts and all their couplings, backlashes in the rotation nodes, etc., files of all DS parts with the extension *.sldprt and a file of the general assembly with the extension *.sldasm are created. At this stage, you can verify the assembly correctness of DS, its performance and eventually generate the necessary design and technical documentation. However, at this stage, it is impossible to analyze the dynamical properties of DS, control quality parameters and synthesize of DS regulators, etc.

II. Second stage. Constructing a mathematical model of the UGS

As it was noted in the Introduction, the possibility of integrating programs that build 3-D models at the physical level with programs that form mathematical models (MM), in particular, SolidWorks and MatLab programs, allows qualitative improvement of the development level and significantly shorten the manufacturing cycle time DS.

After converting the file received at the first stage in the *.sldasm format, a file is created in the *.xml format and then a file of the Simulink Matlab *.slx format. The file converted from the *.sldasm to *.slx format for the UGS in question, which is shown in Fig. 2.

Adding the Simulink model shown in Fig. 2 corresponding entry and exit points, we obtain a model for obtaining a linearized model in the state space (see Fig. 3).

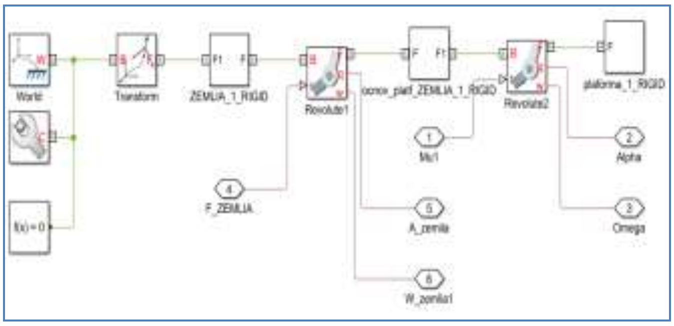

Fig. 3.

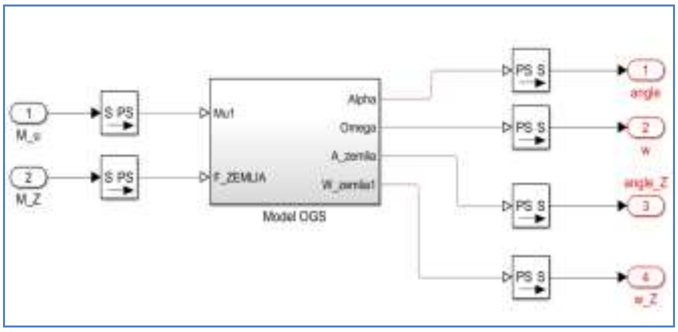

Fig. 4.

After applying the command linmod and ss, we get the four companion matrices of the state space model in the following form:

sys =

A =	**x1**	**x2**	**x3**	**x4**	**B =**	**u1**	**u2**	**C =**	**x1**	**x2**	**x3**	**x4**	**D =**	**u1**	**u2**
x1	**0**	**1**	**0**	**0**	**x1**	**0**	**0**	**y1**	**0**	**0**	**1**	**0**	**y1**	**0**	**0**
x2	**0**	**0**	**0**	**0**	**x2**	**0**	**33.69**	**y2**	**0**	**0**	**0**	**1**	**y2**	**0**	**0**
x3	**0**	**0**	**0**	**1**	**x3**	**0**	**0**	**y3**	**1**	**0**	**0**	**0**	**y3**	**0**	**0**
x4	**0**	**0**	**0**	**0**	**x4**	**895.9**	**0**	**y4**	**0**	**1**	**0**	**0**	**y4**	**0**	**0**

Continuous-time state-space model.

The model in the form of transfer functions is presented below:

From input 1 to output...

$$W11(s) = \frac{895.9}{s^2 + 0.08959s}, \quad W12(s) = \frac{895.9}{s + 0.08959}, \quad W13(s) = 0, \; W14(s) = 0.$$

From input 2 to output...

$$W21(s) = 0, \; W22(s) = 0, \; W23(s) = \frac{33.69}{s^2 + 0.003369s}, \quad W24(s) = \frac{33.69}{s + 0.003369}.$$

When converting a file with *.xml to Simulink *.xls, the *DataFile.m m-file is created, in which all the overall mass characteristics, couplings of parts and orientation of the DS are specified.

The result of the second stage of the design is the transition from the physical model to the linearized mathematical model in the state space (transfer function).

III. THIRD STAGE. THE RESULTS OF MATHEMATICAL MODELING OF THE TERMINAL MANAGEMENT OF THE UGS IN THE REGIME OF "COARSE REDUCTION"

At this stage it is necessary to carry out an analysis of the adequacy of the mathematical model obtained at the second stage by real model obtained at the first stage. For this purpose, we simulate the terminal control in the mode of UGS CR, assuming that the sensor of the gyro-block is on the technological stop and does not participate in the movement.

In order to verify the adequacy of the mathematical models obtained, let us consider the solution of the thermal problem for controlling the UGS. We use previously obtained terminal control laws for various degrees of "softness" for UGS on a fixed base [10]:

if $k_m = 0$

$$M_u = 6\cdot\frac{\alpha_k - \alpha(t)}{(t_k - t)^2} - \frac{2\cdot\omega_k + 4\cdot\omega(t)}{t_k - t};$$

if $k_m = 1$

$$M_u = 12\cdot\frac{\alpha_k - \alpha(t)}{(t_k - t)^2} - 6\cdot\frac{\omega_k + \omega(t)}{t_k - t};$$

if $k_m = 2$

$$M_u = 20\cdot\frac{\alpha_k - \alpha(t)}{(t_k - t)^2} - \frac{12\cdot\omega_k + 8\cdot\omega(t)}{t_k - t};$$

if $k_m = 3$

$$M_u = 30\cdot\frac{\alpha_k - \alpha(t)}{(t_k - t)^2} - 10\cdot\frac{2\cdot\omega_k + \omega(t)}{t_k - t}.$$

Note. We understand the indicator of the degree of control "softness" k as the maximum order of the derivative of the control action equal to zero at the final moment of time in the vector of the final state. For example, for $k_m = 3$, the final state vector has the following form:

$$Q = (q_1\ q_2\ q_3\ q_4\ q_5\ q_6)^T =$$

$$\left(\alpha(t_k)=\alpha_k\ \ \omega(t_k)=\omega_k\ \ M_u^{(0)}(t_k)=M_{u0k}=0\ \ M_u^{(1)}(t_k)=M_{u1k}=0\ \ M_u^{(2)}(t_k)=M_{u2k}=0\ \ M_u^{(3)}(t_k)=M_{u3k}=0\right)^T.$$

The implementation of mathematical modeling of the UGS terminal control in the CR mode based on the obtained models (1) ... (3) is shown in Fig. 4 and 5. The physical and mathematical models are shown in Fig.4 respectively MODEL UGS and LTI MODEL UGS.

As a result of the simulation, animation of the UGS terminal control appears in the CR mode. Comparison of the output characteristics (deviation angles and angular velocities of the platform) of the UGS terminal control in the CR mode with a fixed base for the physical and mathematical models showed the same dependencies, which indicates the adequacy of the models in question (Fig. 7).

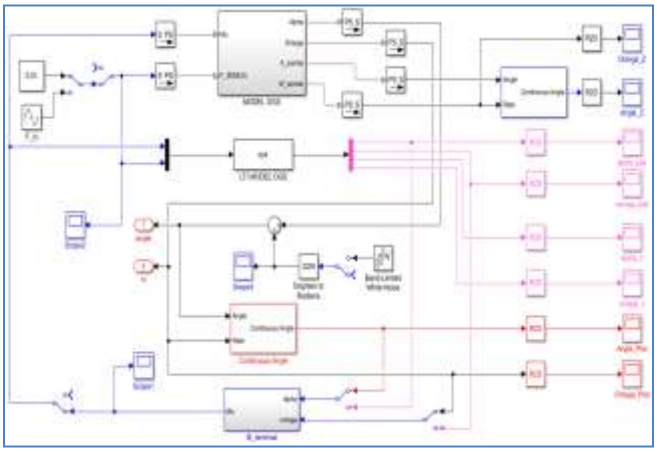

Fig. 5.

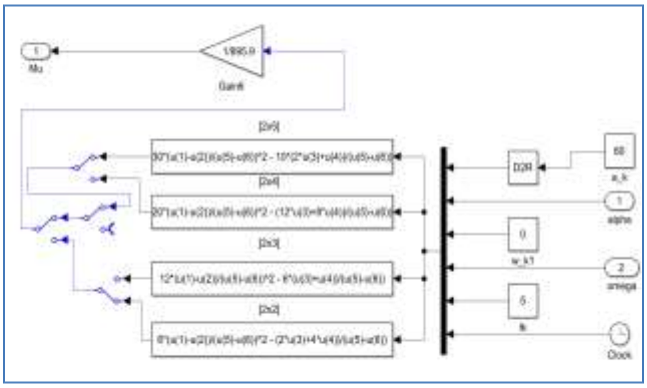

Fig. 6.

At this stage, the mathematical model is refined.

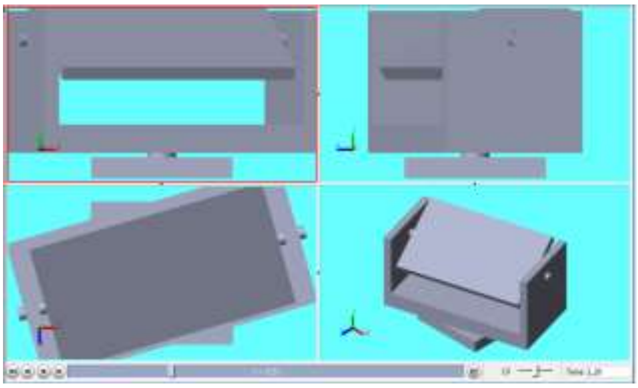

Fig. 7.

IV. FOURTH STAGE. CONSTRUCTING THE GRAPHICAL INTERFACE OF CAD- "THROUGH" DESIGN TECHNOLOGY OF THE UGS TERMINAL CONTROL IN CR MODE

For the flexibility and convenience of solving specific problems, it is advisable to create an interface in the Matlab environment using the GUI application. The necessary options on the interface screen depend on the class of the task being solved.

V. Fifth stage. Creating a Windows-based application CAD - "Through" design technology of dynamic systems

To automate the "through" design process, it is advisable to use the Matlab compiler, which allows to convert Matlab programs to Windows applications and libraries that can work independently of the Matlab system itself. You can compile m-files, MEX-files and other Matlab codes.

The **deploytool** command calls a development environment that allows to create a project, add the necessary files to the project, create an application or library and make an installation package for distribution to the user.

The created application can be installed on another computer that has the same operating system as the computer on which the application was created. In addition to the software, the MCRIn-staller.exe file must be included in the installation package. This file contains an archive containing all the Matlab libraries required for Matlab operation.

Activating the developed application will cause a graphical interface. The loaded assembly file from SolidWorks is transformed into the corresponding Simulink file. After determining the necessary entry and exit points, it is possible to form a mathematical model. After setting the necessary perturbing and control actions, the required task is solved and the required output parameters are determined.

If the specified output parameters are unreachable, corrections must be made to the design of the product and repeat the process.

References

[1] Mashkov S.V. The program Autodesk AutoCAD. Training manual on auto-matyzirovannom design. Moscow. Publ. Alliance-Press. 2007. 448 p.

[2] Guznenkov V.N., Zhurbenko P.A., Bondarev T.P. SolidWorks 2016. Three-dimensional modeling of parts and execution of electronic drawings. 2017. 128 p.

[3] Sufliaeva N.E. AutoCAD in engineering graphics. Russian version. A short course of 2D drafting. 2011. 32 p.

[4] Tugolukov E.N., Tkachev A.G., Rukhov A.V. Designing of complex systems. Tambov. Publishing house of Tamb. state. tech. university. 2008. 32 p.

[5] Bonch-Bruevich A.M. Analysis of the circuit modeling results in the packages Multisim 10 and MATLAB. 2013. 28p.

[6] Gander V., Grzhebichek I. Solution of problems in scientific calculations using Maple and MATLAB. Publ. Vassamedina. 520 p.

[7] Dyakonov V.P. MATLAB 6.0 / 6.1 / 6.5 / 6.5 + SP1 + Simulink 4/5. Signal and image processing. Moscow. Publ. SOLON-Press. 2004. 592 p.

[8] Seleznev M.S. et al. Algorithms for information processing of navigation systems and aircraft systems. Ministry of Education and Science of the Russian Federation. BMSTU – Moscow. Publishing house BMSTU. 2018. 234 [4] p.

[9] Pupkov K.A., Neusypin K.A. The theory and implementation of control systems. Moscow. Bioinform. 1997.

[10] Nikiforov V.M., Trunov Yu.V., Shiryaev A.S. Vektornoe “sverkhmyagkoe” terminal'noe upravlenie odnoosnym girostabilizatorom v rezhime grubogo privedeniya. TRUDY FGUP “NPTsAP” “SISTEMY I PRIBORY UPRAVLENIYa”. Moscow. Publ. Izdat – Print. 2017. No.3