

# 3D Visualization of Mechatronic Systems using OpenGL

*Jakub ČERKALA, Anna JADLOVSKÁ, Štefan SZŪRÖS*

Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Kosice, Slovak Republic

anna.jadlovska@tuke.sk, jakub.cerkala@tuke.sk, stefan.szuros@student.tuke.sk

**Abstract** — The main goal of this article is to show how to visualize dynamical processes in 3D environment on the example of nonlinear mathematical pendulum and mobile robot with differential chassis. Using Matlab or Simulink, by the solution of the model's differential equations, which describe its movement properties, it is possible to obtain data for visualization. The article presents the necessary steps to create an own C# Forms application that is based on OpenGL for visualization of dynamical system.

**Keywords** — nonlinear mathematical pendulum, mobile robot, Matlab/Simulink, visualization, 3D virtual space

## I. INTRODUCTION

The computer technology is evolving fast nowadays and it helps us to simplify our daily life. This development includes also the computer graphics technology, which is extensively used in various fields, such as entertainment or science. In entertainment, the computer graphics is used to create video games or the special effects in movies, but in fields of science, it is mainly used to visualize data. Depending on data, the visualization can be static (image, chart) or dynamic (animation) and it can be done in 2D or 3D space. This article focuses on visualization of dynamical systems presented on the example of mathematical pendulum and mobile robot with differential chassis in three dimensional space. The first step in visualization is the selection of data source. If we don't have data measured on real system, we can calculate an approximation data using mathematical model of selected dynamic system based on physical laws and mathematics. Moreover, we need 3D models that are used for the data visualization as an animation. There are several approaches to visualize data, for example by using programs like Matlab/Simulink [1], [2], which offers a complete solution for dynamical visualization. With Matlab/Simulink, we can calculate visualization data using simulation models of the chosen dynamical systems and with the *Simulink 3D Animation* toolbox [3] we can create complete visualization with 3D models. There is always an option to create a specialized application which can visualize dynamical systems that is using a graphical library in chosen programming language. In this case, the data can be obtained by using external computation in programs like Matlab/Simulink or by an internal calculation, for example using the Runge – Kutta 4<sup>th</sup> order method. This approach requires more programming but it gives the programmer more freedom.

## II. APPLICATION FOR VISUALIZING DYNAMICAL PHYSICAL PROCESSES

The presented application that enables user to visualize various mathematical pendulums at once in 3D space or a mobile robot moving in plane is a result of a bachelor thesis [4]. To program this application, the OpenTK [5], a graphical library which enables the usage of OpenGL [6] in C# programming language was used. Application can visualize data obtained by external calculation, the simulation models of mathematical pendulum and mobile robot were created in Matlab/Simulink. In case of pendulum, there is an additional option to use an internal real-time calculation during visualization programmed in a C# based on the 4<sup>th</sup> order Runge – Kutta numerical method. In this application's 3D environment it is necessary to implement a camera through which the visualization can be observed. Last but not least, the 3D models which are used to data representation have to be programmed as well or imported from an external editor. The first main task is to obtain mathematical models of systems that will be visualized.

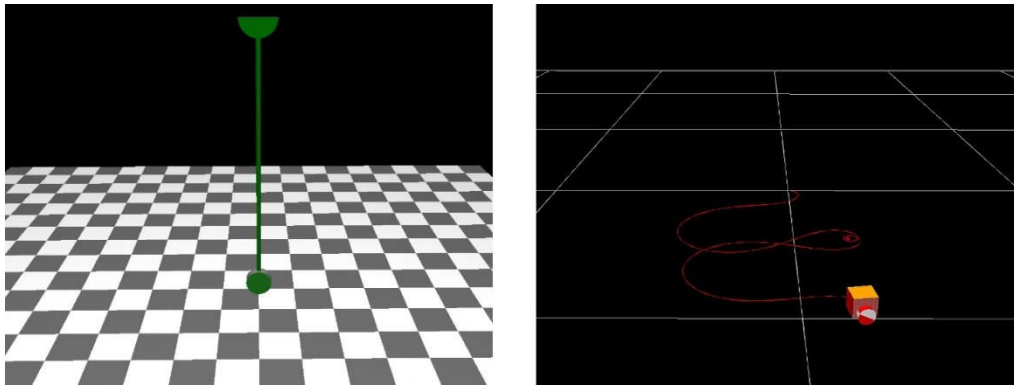


Fig. 1 Visualization of mathematical pendulum and mobile robot in created application

### III. MATHEMATICAL MODELING OF NONLINEAR MATHEMATICAL PENDULUM

Simple or Mathematical pendulum is a bob with mass  $m$  which is attached to a fixed pivot point by a rigid weightless rod of length  $l$  as shown in Fig. 2. The bob of the pendulum can swing around the pivot point along a circular arc which radius is  $l$ . Friction and air resistance are neglected. To calculate the motion of mathematical pendulum which is the angular deflection of the pendulum, we need to solve the motion equation of pendulum by using Newton's second law [7], [8].

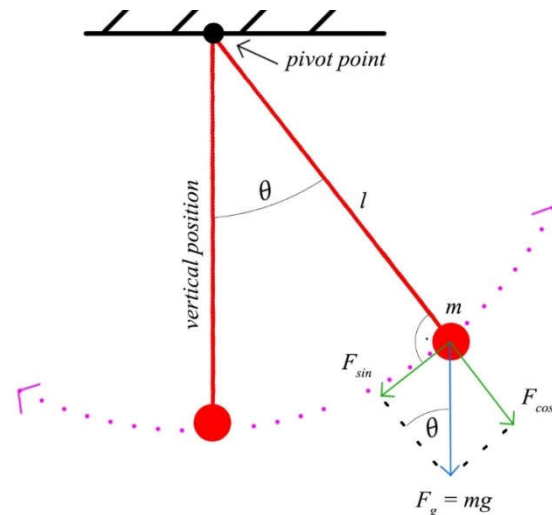


Fig. 2 The mathematical model of the mathematical pendulum

The sum of the forces  $F$  which are acting on the object is equal with the mass of the object  $m$  multiplied by the acceleration  $a$  of the object. The mathematical equation of this law can be written as

$$F = ma. \quad (1)$$

To solve the mathematical pendulum equation of motion we need to calculate the sum of forces  $F$  which are acting on the bob and the bob's acceleration  $a$ . As shown on Fig. 2, the only force that is acting on the pendulum's bob is the gravitational force  $F_g$ . Because the bob is moving along the circular arc, we can divide the gravitational force  $F_g$  into two components. The first component of the gravitational force  $F_{cos}$  is parallel to the rod of the pendulum and it doesn't affect the pendulum movement. The second component of the gravitational force  $F_{sin}$  is perpendicular to the pendulum's rod so it affects the pendulum's movement [7] and therefore it can be assumed as  $F$  in (1). To calculate the bob's acceleration  $a$ , the second derivation of the arc's length  $s$  on which the pendulum's bob moves has to be used. The arc's length  $s$  can be calculated by multiplication of the pendulum  $l$  length with an angular lean  $\theta$ , the angular acceleration of the bob is obtained as [7]

$$\frac{d^2\theta(t)}{dt^2} = \frac{a}{l}. \quad (2)$$

If we assume that  $F_{sin} = mg \sin(\theta(t))$ , then we can use the Newton's second law (1) to get the following 2<sup>nd</sup> order differential equations for motion of the mathematical pendulum [8]

$$\frac{d^2\theta(t)}{dt^2} + \frac{g}{l} \sin(\theta(t)) = 0. \quad (3)$$

The force acting on the pendulum have always the opposite direction as the angular lean of the pendulum, that's why the pendulum is always returning to its vertical position [7]. In case of damped pendulum, the damping force  $F_d$  can be included as [9]

$$F_d = -k\omega(t), \quad (4)$$

where  $k$  is the damping constant and  $\omega$  is the pendulum's angular velocity that is equal to the first time derivation of the pendulum's angular lean  $\theta$  [7]. The 2<sup>nd</sup> order motion differential equation of the mathematical pendulum with damping force can be written as two 1<sup>st</sup> order differential equation for easier numerical implementation as

$$\frac{d\theta(t)}{dt} = \omega(t), \quad (5)$$

$$\frac{d\omega(t)}{dt} = -\frac{g}{l} \sin(\theta(t)) - k\omega(t) \quad (6)$$

#### IV. MATHEMATICAL MODELING OF MOBILE ROBOT WITH DIFFERENTIAL CHASSIS

A rather different model to visualize is the mobile robot with differential chassis [10]. The mobile robots have two differentially driven wheels and it moves in plane with defined in  $\theta_{GSS}$ . The robot's position can be represented using its Centre of Gravity (**CoG**) coordinates  $x(t)$ ,  $y(t)$  and orientation angle  $\varphi(t)$ . Also, for better visualization it is required to include wheel angular positions  $\theta_R(t)$ ,  $\theta_L(t)$  to achieve the wheel rotation illusion. The position of the robot with wheel radius  $r$  and wheel to **CoG** distance  $b$  is depicted on Fig. 3.

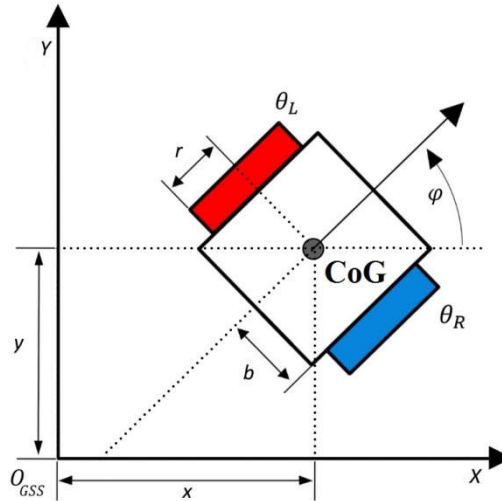


Fig. 3 The position definition of mobile robot in global plane

The mathematical model of the mobile robot usually consists of the kinematic model, that converts robot's velocities into position and the dynamic model, which represents the influences of mass, inertia or friction on resulting velocities. In this case, we use only the kinematics model defined for angular velocities  $\omega_R(t)$ ,  $\omega_L(t)$  that can be written as [10]

$$\frac{d}{dt} \begin{bmatrix} x(t) \\ y(t) \\ \varphi(t) \\ \theta_R(t) \\ \theta_L(t) \end{bmatrix} = \begin{bmatrix} r \cdot \cos \varphi(t) & r \cdot \cos \varphi(t) \\ 2 & 2 \\ r \cdot \sin \varphi(t) & r \cdot \sin \varphi(t) \\ 2 & 2 \\ \frac{r}{2b} & -\frac{r}{2b} \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \omega_R(t) \\ \omega_L(t) \end{bmatrix} \quad (7)$$

#### V. OBTAINING THE DATA FOR VISUALIZATION

For both models, the pendulum or the mobile robot, it is possible to calculate data for visualization using Matlab/Simulink using implemented simulation models. To get data in programming and simulation environment Matlab, we have to create a function that implement mathematical model of pendulum (5), (6) and using the *ode45* solver function, which uses the 4<sup>th</sup> Runge-Kutta numerical method in Matlab we can calculate the data for visualization. The

simulation result of the mathematical pendulum without damping force is shown on Fig. 4 and the Fig. 5 shows the obtained data of the mathematical pendulum with damping force.

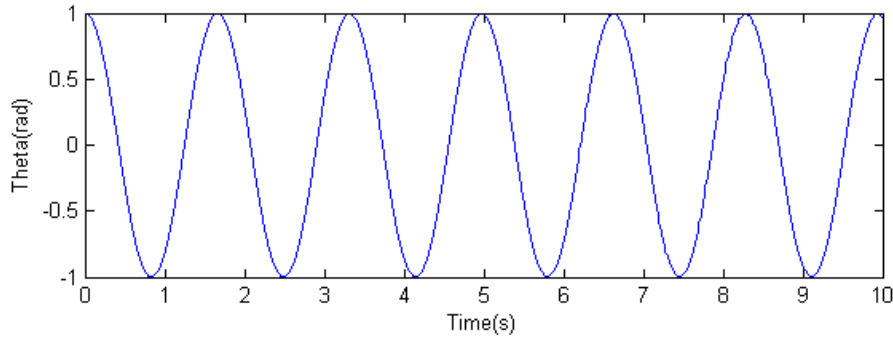


Fig. 4 The pendulum's angular deflection without damping force simulated in Matlab

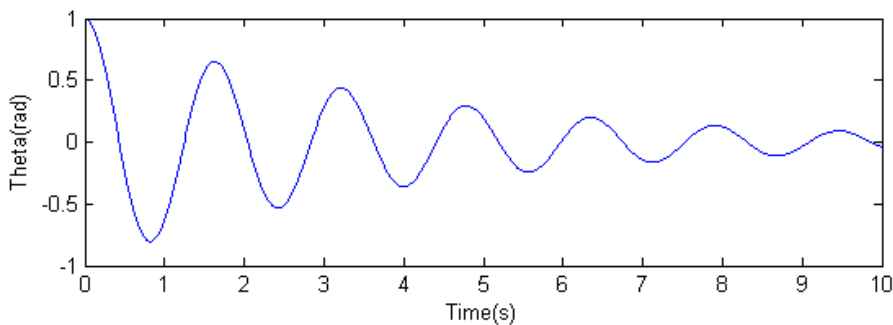


Fig. 5 The pendulum's angular deflection with damping force simulated in Matlab

The mathematical model of the pendulum (6) can be implemented also as Simulink model. The simulation results are the same as in previous approach, however the Simulink environment enables user to change the parameters or simulation properties with minimum input. The obtained plots that describe the pendulum motion are clear.

Simulation model of mobile robot with differential chassis implemented in Simulink based on its kinematics (7) can be tested for non-equal input wheel velocities  $\omega_R \neq \omega_L$  and the simulation experiment shown on Fig. 6 presents the movement results.

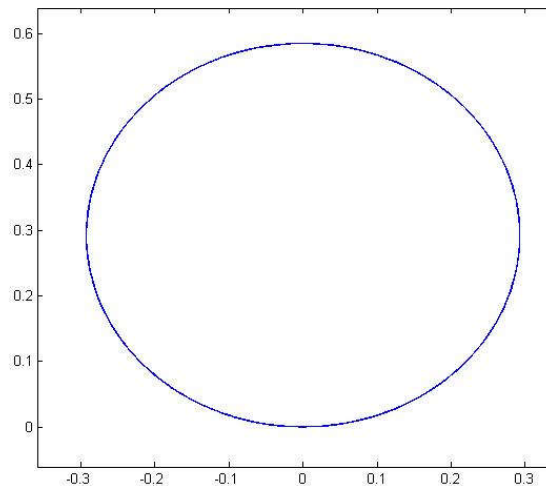


Fig. 6 Mobile robot moving on circular trajectory

The movement of the robot is usually shown in plane. But there is no clear information about the velocities of the robot, its orientation or number of preformed loops. It is unpractical to independently plot all five position variables  $[x(t), y(t), \varphi(t), \theta_R(t), \theta_L(t)]^T$  in term of time, however the visualization can show them at once in user-friendly form.

## VI. VISUALIZATION

To be able to visualize the movement of the models, it is necessary to create a 3D environment that can import and interpret the data. The key features of the visualization are the camera, model and the animation loop.

### 1) Camera

The cameras in a three-dimensional computer graphics play an important role. They are used to observe the visualization from a freely chosen position and they are used for the projection of the scene. The projection output image of from the perspective of the camera view. In presented application, a basic camera was created. This camera has unrestricted movement in 3D environment and can observe the 3D environment from every perspective.

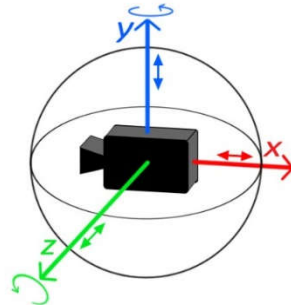


Fig. 7 The movement possibilities of the created camera

The possibilities of camera movement are depicted on Fig. 7, the flying camera have unrestricted movement in the  $x$ ,  $y$  axis and can rotate on axis  $y$  and  $x$ . When rotating around the axis  $z$ , the only restriction is applied - the camera cannot turn more than 90 degrees up and down. These restrictions of the rotation around the axis  $z$  are important to prevent the camera to flip upside down.

### 2) 3D models

3D models are used to show the data during visualization. Three-dimensional models are created by polygons which are defined by vertices. The combination of polygons will create a network of polygons, a mesh, which can contain different types of polygons, but the most common polygon is a triangle - the triangles can be used to create any other existing types of polygons. For more complex models it is always better to work with more polygons to achieve a better model accuracy.

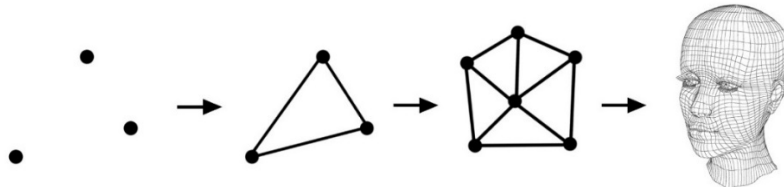


Fig. 8 The process of creating a 3D model with polygons

For visualization of the mathematical pendulum, we had to create a 3D model of the pendulum. The 3D model of a mathematical pendulum consists of a static mount, rod and weight. The rod connects the static mount and weight of the pendulum. The pendulum's rod is dynamic - its size is changed according to the length of the pendulum used in simulation. The rod and the weight are connected and they can move around a pivot point (see Fig. 9). This model was created by primitives like a cylinder and circle. The 3D models of mobile robot consist of box that represents the robot's base and two cylinders as wheels.

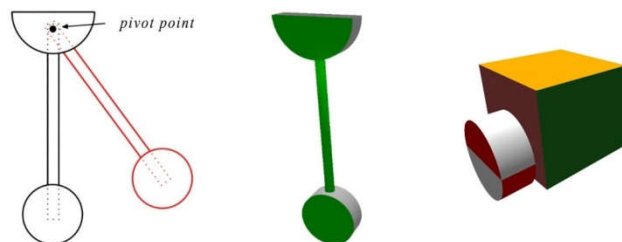


Fig. 9 3D model of the mathematical pendulum and mobile robot

Both 3D models are very simple approximations of real systems, because they are created programmatically, but they serve only as demonstration models. To improve the model quality, the best approach is to implement the model loader.

### 3) Animation of dynamic process

To create an animation of dynamic process we have to calculate projection images and display them continuously with a time delay, because this creates an illusion of movement. We can create these images easily by displaying the calculated data in the actual moments of time. Data for visualization were calculated by a fixed time step. To get a smooth visualization, the step time, used also as delay between two images has to be chosen reasonably - not too wide, but not too small. To animate the 3D models it is necessary to move or rotate them or their parts according to data.

In case of pendulum, the 3D model only moving parts are the rod and the weight. We can visualize the mathematical pendulum movement if we rotate these parts around the pendulum's pivot point (see Fig. 10). The angle of rotation depends on the calculated lean. To visualize the robot with rotating wheels, it is necessary to use rotation and translation transformations in specific order – firstly rotate wheels, than rotate the whole robot.

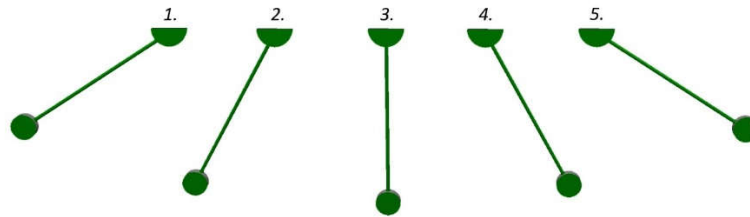


Fig. 10 Movement of the pendulum's 3D model

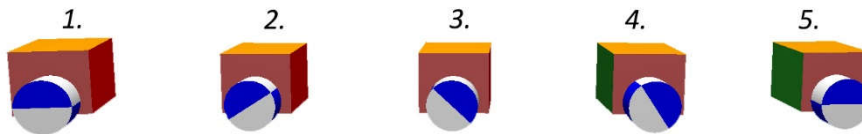


Fig. 11 Movement of the mobile robot's 3D model

## VII. CONCLUSION

In this article, we have presented the example how to create a 3D dynamical visualization of the mathematical pendulum with or without dampening and the mobile robot. With obtained mathematical models, the simulation models were programmed to get simulation data in Matlab/Simulink environment. An application with 3D environment based on OpenGL with free look flying camera was created to visualize obtained data by 3D model of the pendulum.

## ACKNOWLEDGMENT

This work has been supported by the Research and Development Operational Program for project: University Science Park Technicom for innovative applications with knowledge technology support, ITMS code 26220220182, co-financed by the ERDF (70%) and by grant KEGA - 001TUKE-4/2015 (30%).

## REFERENCES

- [1] Matlab – official webpage [online]. [cit. 2015-06-05]. <<http://www.mathworks.com/products/matlab/>>
- [2] Simulink – official webpage [online]. [cit. 2015-06-05]. <<http://www.mathworks.com/products/simulink/>>
- [3] Simulink 3D animation toolbox – official webpage [online]. [cit. 2015-06-05]. <<http://www.mathworks.com/products/3d-animation/>>
- [4] Štefan Szürös, *Návrh a programovanie 3D prostredia na báze OpenGL v jazyku C#* - Bachelor thesis (Supervisor: doc. Ing. Anna Jadlovská, PhD.). Košice: FEI, 2015. 56 s.
- [5] OpenTK - official webpage [online]. [cit. 2015-06-05]. <<http://www.opentk.com/>>
- [6] OpenGL – official webpage [online]. [cit. 2015-06-05]. <<https://www.opengl.org/>>
- [7] Edward R. Scheinerman, *Invitation to dynamical systems*, Prentice Hall College Div, 1995. ISBN 0-13-185000-8
- [8] Richard H. Enns, George C. McGuire, *Nonlinear Physics with Mathematica for Scientists and Engineers*. Boston: Birkhäuser, 2001. ISBN 0-8176-4223-4.
- [9] Douglas D. Novaes, “*Perturbed damped pendulum: finding periodic solutions via averaging method*,” *Revista Brasileira de Ensino de Física*, v. 35, n. 1, 1314 (2013), published
- [10] Čerkala, J. a Jadlovská, A. Dynamics with Friction in Mobile Robot Simulink Model. In: *Technical Computing Bratislava 2014 Proceedings*, pp. 65–81., Lap Lambert Academic Publishing, 2015. ISBN 978-3-659-40792-5.