Comparative analysis of the virtual and physical experiments with a snake robot’s rectilinear movement

V V Varlashin\textsuperscript{1} and O A Shmakov\textsuperscript{2}

\textsuperscript{1} Dept. of Mechatronics and Robotics, Peter the Great St. Petersburg Polytechnic University, Saint Petersburg, 195251, Russia
\textsuperscript{2} Russian State Scientific Center for Robotics and Technical Cybernetics, Saint Petersburg, 194064, Russia
E-mail: v.varlashin@rtc.ru

Abstract. The paper describes virtual and physical experimental studies of the “rectilinear movement” locomotion of a snake robot. Software for controlling the robot movements and registering the modules movements was developed in MATLAB and Simulink. A parameterized virtual model of a snake robot was implemented in the MSC Adams. Authors propose a method for processing experimental data and demonstrate the comparison results of the virtual and physical experiments.

1. Introduction

Virtual and physical experimental studies of snake-like robots (SR) have been conducted for several decades. The first research about serpentine locomotion of biological objects is presented in [1]. Further works [2] allowed identifying common types of locomotion: a serpentine crawling (or a lateral undulation), a sidewinding, a concertine, a rectilinear crawling (or a caterpillar crawling), swimming, and moving on rods. Such artificial locomotion modes, as a rectilinear movement (an analog of the serpentine mode), a rolling (or a spin), a screw movement, and a tank turn, are proposed specifically for the implementation of an SR motion [3, 4]. In [4, 5] a generalization of the serpenoid curve is presented for the description of an SR movement.

This article provides experimental confirmation of the correctness of the mathematical description of kinematics, proposed in [5]. The “rectilinear movement” locomotion is used as an example. The methods of a virtual and physical experiment are used to obtain information about the displacement of the modules in space. The article also proposes a method for processing experimental data to conduct a comparative analysis of the results of the physical and virtual experiments. This method takes into account the oscillations of the SR motion.

2. Control principles for the snake robot

In [5] authors propose a general view of the dependences of the absolute angles of the deviation of the skeletal line ort in the horizontal $\varphi_{\text{hor}}$ and vertical $\varphi_{\text{ver}}$ planes on time $t$ and the arc coordinate $s$:

$$
\varphi_i(s, t) = \frac{1}{2} \varphi_{i0+} \sin \left( s + \alpha_i + (v_i t + \beta_i) \right) + \frac{1}{2} \varphi_{i0-} \sin \left( s + \alpha_i - (v_i t + \beta_i) \right), \quad i = h \lor v. \quad (1)
$$
Here $\varphi_{i0+}$ and $\varphi_{i0-}$ — parameters that define a wave’s type (a traveling or a standing wave), $\alpha_i$ — parameters that define a wave’s form, $\beta_i$ — parameters that define an oscillation phase ratio in orthogonal axes, and $v_i$ — frequencies of the oscillations.

For the movement of SR, consisting of the mechatronic modules (MM) connected by universal orthogonal joints, dependences (1) take the following form:

$$\varphi_i(j,t) = \frac{1}{2} \varphi_{i0+} \sin \left(\frac{2\pi}{N} (jK_i + \frac{t}{T} + (\alpha_i + \beta_i))\right) + \frac{1}{2} \varphi_{i0-} \sin \left(\frac{2\pi}{N} (jK_i + \frac{t}{T} + (\alpha_i - \beta_i))\right).$$

Here $i = h \lor v$, $j = 1, N$, $T$ — an oscillation period in a plane, $K_i$ — a half number of antinodes in a wave, and $N$ — a number of modules. We also assume each MM as a link in this context.

Chosen the “rectilinear movement” locomotion is formed by adding a horizontal wave with a large amplitude (more than 40$^\circ$) and a vertical wave with a small amplitude (less than 25$^\circ$) with double frequency. This ensures the contact of the SR body elements with the planar surface at the side regions of the body. Detailed locomotion scheme can be found in [4], p. 478.

The velocity of the SR is proportional to the velocity of the wave propagation and the amplitude of the horizontal wave. The direction of motion depends on the combination of the direction of the orthogonal waves propagation, their lengths and the relative phase shift. In this article, the frequency parameters and the vertical wave’s amplitude are assumed to be fixed.

3. Physical experiment

A snake-like robot “Zmeelok-3M” [6] developed in RTC have been chosen for a physical experiment. The robot is shown in figure 1. Its specifications are presented in table 1.

![Figure 1. A snake-like robot “Zmeelok-3M”.

Table 1. Specifications of the snake-like robot “Zmeelok-3M”.

| Parameter                  | Value          | Parameter                  | Value          |
|----------------------------|----------------|----------------------------|----------------|
| Number of links            | 10             | Degrees of freedom of each module | 2              |
| Link weight                | 0.4 kg         | Max. joint angle along yaw axis | $\pm 45^\circ$ |
| Robot weight               | 4.3 kg         | Max. angular velocity along yaw axis | 1000$^\circ$/s |
| Link diameter              | 80 mm          | Stall torque along yaw axis | 3.2 N·m        |
| Link length                | 80 mm          | Max. joint angle along pitch axis | $\pm 45^\circ$ |
| Voltage                    | 36.0 V         | Max. angular velocity along pitch axis | 1000$^\circ$/s |
| Interface                  | RS-485         | Stall torque along pitch axis | 3.2 N·m        |

A testing area (see figure 2) includes Vicon motion capture system, consisting of eight Vicon Vero cameras with infrared illumination, and laboratory programmable power supply TDK-Lambda GEN 40-38 for recording power consumption of the robot during the movement. The
interaction of the elements of the testing area is provided by software implemented in the MATLAB and Simulink. Passive pearl reflective markers are used for tracking the movement of the SR modules. The markers are attached to the modules and combined into separate groups by the Vicon Tracker software. Each group corresponds to the head or the tail module (see figure 1). It is necessary to use at least three markers to determine the position of a group.

![Figure 2. (a) Testing area, (b) Vicon Tracker workspace.](image)

4. Virtual experiment

A 3D model of the SR “Zmeelok-3M” was created via SOLIDWORKS for virtual experimental study. The model was parameterized according to the design of the SR body elements and their inertial characteristics. Other parameters of the model match the specification (see table 1). Figure 3 shows an SR mechatronic module and the robot with 10 such modules as links.

![Figure 3. 3D models of (a) the SR mechatronic module, and (b) the robot.](image)

A computer model also created for virtual experiment includes: mathematical model of the SR dynamics, motor imitation model, mathematical model of the control system.

A computer model was developed using MATLAB and Simulink and MSC ADAMS. The 3D model was imported from SOLIDWORKS Motion into MSC ADAMS as an assembly of solid bodies. Then it was parameterized and supplemented with a testing area model. Simulation of the dynamics of a mechanical system with unilateral contact was implemented in the MSC ADAMS. The control system was developed in MATLAB and Simulink.

Each mechatronic module is a universal joint with two degrees of freedom and orthogonal axes lying in the same plane. The joint axes are connected with cylindrical bodies, which act as anchor points. The motion of the bodies can be defined in MSC ADAMS as the dependencies of the positions (orientations), velocities, accelerations, or the forces and torques applied to the bodies. It is recommended to use the forces or torques for modeling the complex systems with contacts. The use of the position (velocity, acceleration) dependencies can lead to discontinuities
in the forces and torques during the forward dynamic problem. In addition to the corruption of the results, the simulation speed is reduced. In computer model, the bodies movements are determined by the torques, and so the forward dynamic problem is solved correctly.

A sensor emulation is implemented in MSC ADAMS using the Measure tool. The measured values of the angles and angular velocities are sent to the control system model, connected as a dynamic linked library. The control system model is connected to MSC ADAMS via Control tool. The interaction of the SR bodies with the planar surface is carried out by the contacts built using standard tools of MSC ADAMS.

Normal contact forces between the bodies is calculated using the Impact function model:

\[
F_c = \begin{cases} 
-c d^a - b d, & d > 0, \\
0, & d \leq 0.
\end{cases}
\]

Here \(c\), \(a\) — contact stiffness and exponent, \(d\) — a penetration depth, \(b\) — a damping coefficient.

Coulomb friction forces are specified using static and dynamic coefficients and stiction and friction translational velocities between the objects [7, 8, 9]. The coefficients are determined experimentally and set in a model with static \(\mu_s = 0.7\) and dynamic \(\mu_d = 0.63\) coefficients, translational velocities are chosen according to the integration step. MSC ADAMS does not support a stiction translational velocity \(V_s \neq 0\), so it is impossible to model perfect stiction.

Motors are represented by an imitative model as a PID controller and first-order transfer function with a time constant \(T = 10^{-4}\) s and a maximum torque 3.2 N·m according to the specification of the SR “Zmeelok-3M”. Computer model does not take into account damping, a backlash of the gearboxes, and the deformations of the MM body elements [10]. Figure 4 presents a scheme of the motor’s imitative model in closed-loop control of the MM actuator.

![Figure 4. A motor imitative model in closed-loop control.](image)

5. Software

The software was implemented via MATLAB and Simulink. It has a modular structure — software elements are grouped according to their functional purpose.¹ This allows to debug individual elements separately and then use the debugged elements as a part of the full system.

Control system uses a Real-Time Sync block of the Simulink Desktop Real-Time toolbox. The maximum frequency of the model is 1.0 kHz.

A TDK-Lambda group of elements is used to work with a laboratory programmable power supply. The source voltage is constant, and the current is measured. A virtual COM-port is used for communication with power supply. The maximum frequency is 3.0 Hz and it is limited by the internal power supply filter. The position of the modules is registered by the Vicon group of elements, which receives UDP packets from the Vicon Tracker with a frequency of 1.0 kHz.

Control System block is operating at a frequency of 100 Hz and generates control signals for the robot. Different frequencies are used for different blocks to obtain the correct MM position

¹ Here we do not provide a detailed Simulink model due to its size, but it certainly is available on request.
measures and send a control signal to the robot with minimal time mismatches. The Control System block is used as part of a DLL-library connected to the computer model of the robot in MSC ADAMS. This ensures the unification of the developed software.

6. Data processing
The velocity of the head and tail modules can be measured as the average object velocity:

\[ V_{\text{avg}} = \frac{\sqrt{(X_i - X_f)^2 + (Y_i - Y_f)^2}}{t_i - t_f}. \]

Here \( V_{\text{avg}} \) — the average velocity, \( X_i, Y_i \) — the initial coordinates of the head or tail module, \( X_f, Y_f \) — the terminal coordinates, \( t_i, t_f \) — the initial and the terminal time.

Taking into account the oscillations of the SR motion, the coordinates \( X \) and \( Y \) should be determined at the instants of time corresponding to the same phase of the motion of the module. To eliminate the influence of oscillations, the following variant of data processing is proposed:

(i) A numerical integration of the movements of the head or tail module over time along the coordinates \( X \) and \( Y \) (a plane of the SR motion). Left Riemann sum used for the integration:

\[ X_j = \sum_{i=0}^{j-1} x_i (t_{i+1} - t_i), \quad Y_j = \sum_{i=0}^{j-1} y_i (t_{i+1} - t_i). \]

Here \( j \in (1, n), \) \( x_i \) and \( y_i \) — coordinates of the head or tail module at the time moment \( t_i, \) \( X_j \) and \( Y_j \) — the integrated coordinates of the head or tail module.

(ii) An approximation of the integrated values by a parabola using the least squares method:

\[ z(t) = at^2 + bt + c, \quad \sum_{i=1}^{n} (Z_i - z(t_i))^2 \rightarrow \min. \]

Here \( t \) and \( (t_i) \) — time moments, \( Z_i \) — the integrated coordinates \( X_i \) or \( Y_i. \)

(iii) A derivative of the approximated function: \( \dot{z}(t) = 2at + b, \) where \( 2a \) is an estimated velocity of the movement of the head or tail module along one of the coordinates.

(iv) An obtaining the resulting average movement velocity:

\[ V_{\text{avg}} = \sqrt{V_{ax}^2 + V_{ay}^2} = \sqrt{(2a)^2 + (2a)^2}. \]

The robot is controlled by a PC running under Windows 7 OS, which is not real-time. Thus the issuance of the control actions at strictly defined intervals, in the order of milliseconds, is difficult or impossible. The discrete Fourier transform is used to obtain the amplitude spectrum of a discrete periodic signal and so to estimate the temporal characteristics of the motion:

\[ X_k = \sum_{n=0}^{N-1} x_n e^{-j2\pi kn/N}. \]

Energy consumption is estimated based on the measured current and voltage values during the movement. Taking into account the constant voltage on the laboratory programmable power supply, the average value of the power consumption can be estimated as the product of the average current value and the voltage:

\[ P_{\text{avg}} = I_{\text{avg}} U = \frac{U}{N} \sum_{i=1}^{N} I(t_i). \]
The results of a physical experiment processing are shown in table 2. The measurement error is estimated as a random value with the number of measurements $N = 10$ and the confidence level $\alpha = 99\%$ of the Student’s t-distribution.

The results of a virtual experiment processing are shown in table 3. For comparison, the simulation results are presented with static coefficients $\mu_s = 0.7, \mu_s = 0.5$ and dynamic coefficients $\mu_d = 0.65, \mu_d = 0.45$ respectively.

### Table 2. Results of a physical experiment processing.

| $\varphi_{hor}$ | $V_{avg}$ | $\Delta V$ | $f$ | $\Delta f$ | $P_{avg}$ |
|-----------------|-----------|------------|-----|------------|-----------|
| 40°             | 37.4 mm/s | 1.9 mm/s   | 0.9668 Hz | 0.002 Hz  | 81.9 W    |
| 50°             | 40.8 mm/s | 1.8 mm/s   | 0.9837 Hz | 0.022 Hz  | 84.2 W    |
| 60°             | 45.9 mm/s | 1.2 mm/s   | 0.9937 Hz | 0.002 Hz  | 88.1 W    |
| 70°             | 52.3 mm/s | 2.3 mm/s   | 0.9935 Hz | 0.002 Hz  | 91.1 W    |
| 80°             | 59.6 mm/s | 1.4 mm/s   | 0.9932 Hz | 0.004 Hz  | 92.8 W    |
| 90°             | 68.1 mm/s | 2.3 mm/s   | 0.9936 Hz | 0.005 Hz  | 93.8 W    |

### Table 3. Results of a virtual experiment processing.

| $\varphi_{hor}$ | $V_{avg}$ ($\mu_s = 0.7, \mu_d = 0.65$) | $V_{avg}$ ($\mu_s = 0.5, \mu_d = 0.45$) |
|-----------------|-----------------------------------------|-----------------------------------------|
| 40°             | 33.4 mm/s                                | 34.2 mm/s                                |
| 50°             | 38.9 mm/s                                | 39.9 mm/s                                |
| 60°             | 44.4 mm/s                                | 45.7 mm/s                                |
| 70°             | 50.4 mm/s                                | 52.4 mm/s                                |
| 80°             | 57.8 mm/s                                | 60.2 mm/s                                |
| 90°             | 66.1 mm/s                                | 68.5 mm/s                                |

The results showed the correctness of the developed computer model. The discrepancy with the physical experiment did not exceed 10\% for small values of the amplitude angle and 5\% for large values. With the values $\mu_s = 0.5$ and $\mu_d = 0.45$, the discrepancy was significantly smaller.

### Acknowledgments

This research is supported by the Ministry of Science and Higher Education of the Russian Federation. State assignment at the Russian State Scientific Center for Robotics and Technical Cybernetics № 075-00924-19-00.

### References

[1] Gans C 1974 *How snakes move, in Vertebrate Structures and Functions: Readings from “Scientific American”* (San Francisco, CA: W.H. Freeman and Company) pp 38–47
[2] Dowling K J 1997 *Limbless Locomotion: Learning to Crawl with a Snake Robot* (Pittsburgh: CMU) p 144
[3] Hirose S and Umetani Y 1976 *Trans. of the Society of Instrument and Control Engineers* **12** pp 543–547
[4] Ivanov A A 2004 *Reptilians Motion Simulation* ed A Zinkovskiy (St. Petersburg: SPbPU) pp 471–494
[5] Ivanov A A, Demidov D and Shmakov O 2008 *Proc. of XI Russian Scientific and Practical Conf. on Current Problems of Defence and Safety* pp 297–303
[6] Ivanov A A, Demidov D and Shmakov O 2013 *St. Petersburg State Polytechnical University J.: Computer Science, Telecommunication, and Control Systems* **164** pp 132–138
[7] Semenov Y A and Semenova N S 2019 *Lecture Notes in Mechanical Engineering* pp 169–177
[8] Evgrafov A N, Karazin V I and Petrov G N 2019 *Lecture Notes in Mechanical Engineering* pp 119–127
[9] Nosonovsky M and Breki A D 2019 *Entropy* **21** 620
[10] Rostova E, Rostov N and Sokolov B 2018 *Lecture Notes in Computer Science* **11097** pp 222–232