Modelling the monopedal robot

M Niechciał¹, D Rybarczyk¹ and J Buśkiewicz²
¹ Poznań University of Technology, Institute of Mechanical Technology, Piotrowo 3, Poznań, Poland
² Poznań University of Technology, Institute of Applied Mechanics, Piotrowo 3, Poznań, Poland
dominik.rybarczyk@put.poznan.pl

Abstract. The article describes modelling of the monopedal robot with the application of the numerical method for calculating kinematics. The first part presents a short review of the literature on monopedal robots and the theory of such designs. Due to the complexity of the natural solutions of the bipedal locomotion, the monopedal alternative simplifies the problem, by limiting its scope to the pulse movement in one axis– as required by jumping. In the subsequent part, basic mathematical and kinematic relations connected with a chosen design. The model was based on a rigid quadrangle driven by a four-bar linkage. The effect was a set of vector equations, required to calculate its kinematics. Due to the complexity of the design, it could not be calculated using traditional analytical methods. The prepared kinematic structure was simulated using the numerical method to give the basis for the future experimental tests. The mathematical model was solved in the Wolphram Mathematica environment. Its main purpose was to analyse the movement of the final link that rests on the ground. The results from the simulation were compared to the movement parameters derived from a model designed in CAD software to verify its correctness. All values of links length and starting angular position, required to calculate the model, were taken from the said CAD project.

1. Introduction
One of the unresolved issues in modern robotics that limits the practical application of robots is a method of effective and smooth locomotion. The domination of mobile robots based on wheeled vehicle base is the outcome of simplicity in design and control. Due to its nature, however, the wheeled design is ineffective in most of the non-laboratory environments [1]. A possible progress in the area of locomotion problem is located in the bio mimic solution, reproducing animalistic limbs generated by the evolution. The most intuitive design human-wise is legged locomotion [2].

The most important advantages of the legged solution are [3]:
- adaptive to uneven terrain
- use of isolated footholds
- provide active suspension
- the environmental effects of legged vehicles are less than wheeled or tracked vehicles

Due to difficulties in control and supervision of bipedal designs [4] various projects have been started to devise monopedal jumping robots. The latter design requires dynamic balance and a special structure. The first model of a jumping robot was presented by Matsuoka in 1979, with the first solutions to the mentioned problems. The actuator was constructed of a pneumatic cylinder, restoring
energy on impact; the balance in flight was controlled by rotational mass with the mass moment of inertia.

To perform a jump motion it needs a mechanical resonator that stores or modulates the power in any way. Most solutions to this problem apply a series of elastic actuators, in between the drive link and the motor, or another source of power. Its basic benefit is isolating and filtering all impacts delivered by dynamic jumping or falling, storing the energy of the motor for a period of time, until the launch, and the change in the definition of the control system. Due to the actual input delivered to the mechanism not being directly connected with the motor shaft, the real information about the force and speed is delivered by analysing relative rotary position between the elastic element and its elastic constant.

To ensure a correct motion, the jump itself must be repetitive or controlled in flight [5]. Because of a robust movement, most constructions also include the solution dealing with flight control [6].

1.1. Projects overview

One of the most important properties of all jumping robots is the ability to generate impulse of force that highly surpasses the dynamic possibilities of the drive. Owing to the selected method of storing and transferring energy, produced by the input link, the entire mechanism works similarly to a catapult. In the very basic example, given by nature, it is represented by the biological structure of frogs’ legs, which are unable to maintain a process of walking, but due to modulating force, accumulated in crouching, enable jumping [7].

The catapult mechanism was represented in a simple flea-inspired mechanism size of 20 mm, acquiring extraordinary performance [8]. The combination of three spring actuators made of shape memory alloy (SMA) generates and stores energy during mechanical locking of the links. Due to its nature, further electrical heating of the springs enables an even greater force to be locked inside links. After the release, the mechanism jumps to the distance of up to 30 times its body size.

A design that also employs springs to accumulate energy is the Michigan robot [9]. The whole design is an enclosed kinematic chain consisting of 6 links creating a hexagon with four springs that force two opposed sides away from each other. Springs tension is generated by a rotor on one side connected with it, and with the link on the other. Its simplistic design with minimal influence of electronics in the drive causes that the mass could be minimised to achieve great effects in performance: with 25 grams of mass and almost 1 meter of jumping height [10]. Due to its robust flight phase, it was required to control its position with a tail [11], which is an offset mass connected with the body by a rotating link. It is a common solution based on animalistic tail [12].

A successful solution for jumping and control was presented by the Penn Jerboa robot [13]. The design is very much similar to the Michigan one, but for the purpose of maximum control and repetitive motion, its height was greatly reduced. The robot is driven by an enclosed quadrangle that is extended and compressed by an actuator with a spring on it. To bring more tension to the spring the whole robot is pressed to the ground by the momentum of the bouncing tail. Even though the height of the jump is only 0.16 meter, it is consistent, quick and predictable.

One of the designs that solves the problem of accumulating energy and controlling the flight trajectory is developed by the Berkeley team in a robot called Salto [14]. Its design is based on a rigid quadrangle driven by a four-bar linkage that moves a lever connected to the ground. Its greatest benefit is the modulation of the mechanical advantage that is the change of momentum to force ratio during the rotation of the drive. Thanks to low advantage at the starting position, the input energy of the motor is stored in a synthetic spring, which mediates in torque transmission, and moves the four-bar linkage in a slow manner. After reaching further angle the advantage raises allowing releasing momentum accumulated in the spring to acquire an impulse of the jump. Thanks to the construction of the robot, the final point of the mechanism moves in a linear manner aligned with the rotation axis of the motor perpendicularly to the ground. With this trait, the process of jumping can be in a straight line upwards without any oscillations in the air. The objective of this paper is to recreate the kinematic
model of the construction using numerical methods, analyse its correctness with CAD project, composed for the calculation purpose, and perform prototype testing of its movement in work.

2. The kinematic model
A numerical simulation is required to perform geometric and kinematic analyses including determination of path, velocity and acceleration of mechanism points. The vector closed-loop method is utilised. The considered mechanism is of fourth class according to structural Assur classification [15].

![Figure 1. Models used in mathematical analyses. a) from the original Berkeley idea [14], b) the kinematic scheme on which further calculation was based on, c) model created in CAD environment.](image)

The first step is to analyse the structure of the mechanism and compute its mobility [16]. The links GH and HJ are driven (passive group according to Assur classification) and do not change the mobility value, therefore, they are disregarded until the numerical method has been presented. The number of links (n) in the mechanism equals 7 and the number of the kinematic pairs (p) equals 10. Then, the mobility (w) of the mechanism equals to 1.

\[
\begin{align*}
    w &= 3 * n - 2 * p \\
    1 &= 3 * 7 - 2 * 10
\end{align*}
\]

All the kinematic equations have to include fixed revolute pivots K or O, which connect the mechanism to the frame. The first closed loop, which can be derived, is a modified four-link mechanism composed of joints OABCD.

The closed loop vector equation can be rewritten as two algebraic equations by equalling X and Y components [17]. The equations contain four variables (the indices at the angles correspond to the link lengths). Every angle is measured between familiar link and the horizontal line led from the joint connecting it with a previous link [18, 19].

\[
\begin{align*}
    \text{Eq1: } & L_1 \cos(\phi_1(t) + \pi) + L_3 \cos(\phi_3(t)) \\
    & = L_1 \cos(\phi_1(t)) + L_4 \cos(\phi_4(t)) + L_5 \cos(\phi_5(t)) \\
    \text{Eq2: } & L_1 \sin(\phi_1(t) + \pi) + L_3 \sin(\phi_3(t)) \\
    & = L_1 \sin(\phi_1(t)) + L_4 \sin(\phi_4(t)) + L_5 \sin(\phi_5(t))
\end{align*}
\]
Figure 2. Visual presentation of the vector calculus providing the first set of equations.

The active link and joint K are connected through rigid quadrangle DBEF with defined angles. Rotations of all of these are dependent on a single one according to given geometric relationships.

Figure 3. Geometric structure of the rigid quadrangle with its angles assigned.

\[
\phi_6 = \phi_5 + \beta - \pi \quad (5)
\]

\[
\phi_8 = \phi_5 - \alpha \quad (6)
\]

\[
\phi_7 = \phi_5 - \alpha - \gamma \quad (7)
\]

Providing formulas expressed by (Eq. 5-7) a kinematic equation can be given. The second loop consists of chains KEB and OCB. Lengths of links KE and EB are denoted as \( L_9 \) and \( L_6 \), respectively.
Eq3: \[ L_1 \cos(\phi_1(t)) + L_4 \cos(\phi_4(t)) = L_9 \cos(\phi_9(t)) + L_6 \cos(\phi_5(t) + \beta - \pi) + X \]  

Eq4: \[ L_1 \sin(\phi_1(t)) + L_4 \sin(\phi_4(t)) = L_9 \sin(\phi_9(t)) + L_6 \sin(\phi_5(t) + \beta - \pi) + Y \]  

The equations 8-9 are sufficient to describe the complete active kinematic chain of the mechanism. To numerically analyse the entire mechanism, in particular, the motion of point J, the equations 3, 4, 8 and 9 with another closed loop must be solved. This one is built on chains KGHF and OADF. Point F is built into the link HJ, therefore, the rotation angle of link HF provides information on the position of point J. Length \( L_91 \) is corresponding to the link KG, \( L_{10} \) to GH, \( L_{11} \) to HF and \( L_{12} \) to DF.

Eq5: \[ L_{91} \cos(\phi_9(t)) + L_{10} \cos(\phi_{10}(t)) + L_{11} \cos(\phi_{11}(t)) = L_1 \cos(\phi_1(t) + \pi) + L_3 \cos(\phi_3(t)) + L_{12} \cos(\phi_5(t) + \pi - \alpha) \]  

Eq6: \[ L_{91} \sin(\phi_9(t)) + L_{10} \sin(\phi_{10}(t)) + L_{11} \sin(\phi_{11}(t)) = L_1 \sin(\phi_1(t) + \pi) + L_3 \sin(\phi_3(t)) + L_{12} \sin(\phi_5(t) + \pi - \alpha) \]  

This set of 6 equations 3, 4, 8, 9, 10, 11 was used to create the numerical simulation.

2.1. The numerical programme

To determine the mechanism angular positions starting from the known, prescribed first position, first and second derivatives of the closed-loop equations must be computed [20].

The matrix equation for angular velocities is then:

\[ \mathbf{v} = \mathbf{b} \cdot \mathbf{A}^{-1} \]  

where:

\( \mathbf{v} \) – angular velocities matrix,
\( \mathbf{b} \) – column vector of prescribed data,
\( \mathbf{A} \) – matrix of coefficients at unknown angular velocities.

The angular acceleration is computed from (where \( \mathbf{b}p \) and \( \mathbf{A}p \) are the derivatives of \( \mathbf{b} \) and \( \mathbf{A} \)):

\[ \mathbf{a} = (\mathbf{b}p - \mathbf{A}p \cdot \mathbf{v}) \cdot \mathbf{A}^{-1} \]  

Having given sets of derivatives and the initial position, an iterative method can be applied for every next instance.

\[ p(t + \delta t) = p(t) + \mathbf{v} \cdot \delta t + \frac{1}{2} \mathbf{a} \cdot \delta t^2 \]  

The programme using the numerical method for kinematic analysis was implemented in the environment of Wolfram Mathematica. The first step was to input all written equations into the script. Thanks to the character of the platform, which resembles a text editor, it was intuitional and self-explanatory.
The derivatives were computed and all the equations were transformed to the matrix form to change the format of selected equation arrays. For this purpose, the Mathematica function CoefficientArrays [ , ] was used with the given parameter of \( v \) array. The method segregates all variables in a selected array focusing on a given parameter. The output is a set of different objects (A, b and \( v \) matrices) that compose into the primary formula in equation 12. Because of the fact that all the next matrices are derivatives of the generated one, these were calculated using the D [ , ] function.

\[
\begin{align*}
eq & = L_1 \text{Cos}[\phi_1[t] + \pi] + L_3 \text{Cos}[\phi_3[t]] = \\
& = L_1 \text{Cos}[\phi_1[t]] + L_4 \text{Cos}[\phi_4[t]] + L_5 \text{Cos}[\phi_5[t]]; \\
eq & = L_1 \text{Sin}[\phi_1[t]] + L_3 \text{Sin}[\phi_3[t]] = \\
& = L_1 \text{Sin}[\phi_1[t]] + L_4 \text{Sin}[\phi_4[t]] + L_5 \text{Sin}[\phi_5[t]]; \\
eq & = L_1 \text{Cos}[\phi_1[t]] + L_4 \text{Cos}[\phi_4[t]] = = L_9 \text{Cos}[\phi_9[t]] + L_6 \text{Cos}[\phi_5[t] + \beta - \pi]; \\
eq & = L_1 \text{Sin}[\phi_1[t]] + L_4 \text{Sin}[\phi_4[t]] = = L_9 \text{Sin}[\phi_9[t]] + L_6 \text{Sin}[\phi_5[t] + \beta - \pi]; \\
eq & = L_9 \text{Cos}[\phi_9[t]] + L_1 \text{Cos}[\phi_1[t]] = = L_1 \text{Cos}[\phi_1[t] + \pi] + L_3 \text{Cos}[\phi_3[t]] + L_1 \text{Cos}[\phi_5[t] + \pi - \alpha]; \\
eq & = L_9 \text{Sin}[\phi_9[t]] + L_1 \text{Sin}[\phi_1[t]] = = L_1 \text{Sin}[\phi_1[t] + \pi] + L_3 \text{Sin}[\phi_3[t]] + L_1 \text{Sin}[\phi_5[t] + \pi - \alpha]; \\
& = \{\phi_3[t], \phi_4[t], \phi_5[t], \phi_9[t], \phi_10[t], \phi_11[t]\} \\
& = \{\phi_3'[t], \phi_4'[t], \phi_5'[t], \phi_9'[t], \phi_10'[t], \phi_11'[t]\} \\
& = \{\phi_3''[t], \phi_4''[t], \phi_5''[t], \phi_9''[t], \phi_10''[t], \phi_11''[t]\}
\end{align*}
\]

**Code 1. Definition of the equations.**

After arranging all 18 equations from the mathematical model into matrix form, the input values are presented as follows:

\[
\begin{align*}
\phi_3'[t] \\
\phi_4'[t] \\
\phi_5'[t] \\
\phi_9'[t] \\
\phi_{10}[t] \\
\phi_{11'}[t]
\end{align*}
\]

\[
A_{11} = \begin{bmatrix}
L_1 \text{Sin}[\phi_1[t]] & L_4 \text{Sin}[\phi_4[t]] & L_5 \text{Sin}[\phi_5[t]] & 0 & 0 & 0 \\
L_3 \text{Cos}[\phi_3[t]] & -L_4 \text{Cos}[\phi_4[t]] & -L_5 \text{Cos}[\phi_5[t]] & 0 & 0 & 0 \\
0 & -L_4 \text{Sin}[\phi_4[t]] & -L_5 \text{Sin}[\phi_5[t]] & 0 & 0 & 0 \\
L_3 \text{Sin}[\phi_3[t]] & 0 & L_1 \text{Sin}[\phi_{10}[t]] & L_9 \text{Sin}[\phi_{11}[t]] & -L_1 \text{Sin}[\phi_{10}[t]] & L_1 \text{Sin}[\phi_{11}[t]] \\
L_1 \text{Cos}[\phi_1[t]] & 0 & L_2 \text{Cos}[\phi_{10}[t]] & L_9 \text{Cos}[\phi_{11}[t]] & L_1 \text{Cos}[\phi_{11}[t]] & L_1 \text{Cos}[\phi_{10}[t]] \\
L_3 \text{Cos}[\phi_3[t]] & 0 & L_1 \text{Cos}[\phi_{10}[t]] & -L_2 \text{Cos}[\phi_{11}[t]] & L_9 \text{Cos}[\phi_{11}[t]] & L_1 \text{Cos}[\phi_{11}[t]] \\
\end{bmatrix}
\]

\[
b = \begin{bmatrix}
2L_1 \text{Sin}[\phi_1[t]] \phi_1'[t] \\
-2L_1 \text{Cos}[\phi_1[t]] \phi_1'[t] \\
-2L_1 \text{Sin}[\phi_1[t]] \phi_1'[t] \\
L_1 \text{Cos}[\phi_1[t]] \phi_1'[t] \\
L_1 \text{Cos}[\phi_1[t]] \phi_1'[t] \\
L_1 \text{Cos}[\phi_1[t]] \phi_1'[t] \\
\end{bmatrix}
\]
\[ a: \begin{bmatrix} \phi^3''[t] \\ \phi^4''[t] \\ \phi^5''[t] \\ \phi^9''[t] \\ \phi^{10}''[t] \\ \phi^{11}''[t] \end{bmatrix} \]  

\[(18)\]

\[ A_p: \begin{bmatrix} -L_{10}\cos(\phi_1(t))&0&0&0\cr L_{14}\cos(\phi_1(t))&-L_{10}\sin(\phi_1(t))&0&0\cr -L_{14}\sin(\phi_1(t))&L_{10}\cos(\phi_1(t))&0&0\cr 0&0&0&0\cr L_{16}\cos(\phi_1(t))&L_{10}\sin(\phi_1(t))&-L_{14}\cos(\phi_1(t))&-L_{14}\sin(\phi_1(t))\cr 1.25\sin(\phi_1(t))&-L_{14}\cos(\phi_1(t))&L_{10}\sin(\phi_1(t))&-L_{14}\sin(\phi_1(t)) \end{bmatrix} \]  

\[(19)\]

\[ B_p: \begin{bmatrix} 2L_1\cos(\phi_1(t))\phi_1'[t]^2 + 2L_1\sin(\phi_1(t))\phi_1''[t] \\ 2L_1\sin(\phi_1(t))\phi_1'[t]^2 - 2L_1\cos(\phi_1(t))\phi_1''[t] \\ -L_1\cos(\phi_1(t))\phi_1'[t]^2 - L_1\sin(\phi_1(t))\phi_1''[t] \\ -L_1\sin(\phi_1(t))\phi_1'[t]^2 + L_1\cos(\phi_1(t))\phi_1''[t] \\ -L_1\cos(\phi_1(t))\phi_1'[t]^2 - L_1\sin(\phi_1(t))\phi_1''[t] \\ -L_1\sin(\phi_1(t))\phi_1'[t]^2 + L_1\cos(\phi_1(t))\phi_1''[t] \end{bmatrix} \]  

\[(20)\]

All values of length and angular position in time \( t = 0 \) were filled with those from the CAD model. The algorithm for numerical calculation is based on filling given matrices with values, calculated in the previous step to determine the next position.

After all the angular positions were computed for one cycle of the active link motion, the position in XY plane of point J was determined using vector formula:

\[ X_j = X_f + L_{91}\cos(\phi_9) + L_{10}\cos(\phi_{10}) + L_{110}\cos(\phi_{11}) \]  

\[(21)\]

\[ Y_j = Y_f + L_{91}\sin(\phi_9) + L_{10}\sin(\phi_{10}) + L_{110}\sin(\phi_{11}) \]  

\[(22)\]

2.2. Simulation

The maximum and minimum distance between the drive and J point was 219.96 mm and 171.51 mm. All length values were provided from the CAD model. The simulation was performed with a set of given parameters:

- Number of iterations \( lp = 100 \)
- Rotary speed of the motor \( n = 140 \text{ rpm} \)
- Angular speed of the rotor \( \omega = \pi/30 \times n \)
- Time of the single time step \( dt = (\pi/2/\omega)/lp \)
- Time of the full simulation \( tfull = \pi/2/\omega = 0.107 \text{ s} \)
- Starting angular position of the drive \( \alpha = 0^\circ \)
- Final angular position of the alpha drive \( \Phi = 90^\circ \)

The numerical solution gave a smooth, continuous movement without visible unpredictable spikes. The verification was performed by comparison of the final angular position of selected links with the one from the CAD model. The difference between the two readings amounted to less than 0.2°, thus classifying the method and the calculations as correct.
Figure 4. Algorithm of the numerical loop.

Figure 5. Movement of the final point on XY plane, in millimetres.
After calculating the movement of the final link on the XY-plane and placing it on the model in and adequate scale, it was visible that the movement was not entirely linear. The position of the final point was shifting to the back of the leg in its terminal phase. That should have a negative impact on the quality of the jump, because of the changing the orientation of the robot in the air. That gives an open field for study and testing with a view to further optimisation.

2.3. Experimental testing
To further analyse the given data, test the possibilities of the model and give the basis for further optimisation, the physical prototype was constructed.

![Figure 6. Theoretical movement of the final point, in scale.](image)

The links of the robot were created using a 3D printer. All links were connected with 3 mm brass rods locked in place with glued rings and placed in plain bearings, made by Igus company. The rotary force was generated by a torque spring and was transferred via rubber hose to the drive link.

![Figure 7. Method of transferring momentum onto drive link.](image)

The spring was primarily stretched to generate greater momentum. The mechanism was released and measured by the BNO055 accelerometer and gyroscope placed on top of the body, to check the
performance. The model jumped to a height of approximately 150 mm that is roughly the size of the robot in its crouching position.

![Figure 8. Process of jumping (time steps in milliseconds).](image)

There was a visible alteration of the angular position of the body of the robot, as a consequence of the nonlinear character of the trajectory of the foot. Analysing the process of the jump by a gyroscope, there is a visible change in rotation, reaching the value of 97° per second.

![Figure 9. Chart of the angular position in the perpendicular axis. In the final phase, the change is inconsistent due to multiaxis rotation and impact on landing.](image)

![Figure 10. Chart of the acceleration in the Z-axis, shows a rise in acceleration just before the jump, shown in the red circle (positive values towards the ground). First high value is due to the method of stabilisation before the jump.](image)
Data from the Z-axis resulted in a significant gain in acceleration shortly before the moment of the jump. The model is, therefore, correct in its basic principle: storing energy inside the series elastic element (hose) until the rise of efficiency.

3. References

[1] Fankhauser P, Bjelonic M, Bellicoso D, Miki T and Hutter M 2018 Robust Rough-Terrain Locomotion with a Quadrupedal Robot 2018 IEEE International Conference on Robotics and Automation (ICRA) 1-8

[2] Burdick J and Fiorini P 2003 Minimalist jumping robots for celestial exploration Int. J. Robot. Res 22(7) pp. 653–674

[3] Hardarson F 1997 Locomotion for difficult terrain

[4] Sayyad A, Seth B and Seshu P 2007 Single-legged hopping robotics research—A review Robotica 25(5) pp. 587–613

[5] Schlegel M, Kovac M, Zufferey J C and Floreano F 2010 Steerable miniature jumping robot Auton Robot 28 295–306

[6] Johnson A, Libby T, Chang-Siu E, Tomizuka M, Full R and Koditschek D 2012 Tail assisted dynamic self righting University of Pennsylvania

[7] Astley H C and Roberts T J 2011 Evidence for a vertebrate catapult: elastic energy storage in the plantaris tendon during frog jumping Biology Letters 8(3) pp. 386–389

[8] Minkyun N, Seung-Won K, Sungmin A, Je-Sung K and Kyu-Jin C 2012 Flea-inspired catapult mechanism for miniature jumping robots IEEE Transactions on Robotics 8(5) pp. 1007–1018

[9] Zhao J 2013 MSU Jumper: A single-motor-actuated miniature steerable jumping robot IEEE Transactions on Robotics 29(3) pp. 602–614

[10] Zhao J, Xi N, Cintron F J, Mutka M W and Xiao L 2012 A single motor actuated miniature steerable jumping robot IEEE/RSJ Int. Conf. Intell. Robots Syst. pp. 4274–4275.

[11] Zhao J, Zhao T, Xi N, Cintrón F, Mutka M and Xiao L 2015 Controlling Aerial Maneuvering of a Miniature Jumping Robot Using Its Tail arXiv:1502.05347

[12] Jusufi A, Kawano D, Libby T and Full R 2010 Righting and turning in mid-air using appendage inertia: Reptile tails, analytical models and bio-inspired robots Bioinspiration & biomimetics. 5. 045001. 10.1088/1748-3182/5/4/045001

[13] De A and Koditschek D 2016 The Penn Jerboa: A Platform for Exploring Parallel Composition of Templates Technical Report to Accompany

[14] Haldane D, Plecnik M, Yim J and Fearing R 2016 A power modulating leg mechanism for monopedal hopping 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 4757–4764

[15] Parszewski Z 1978 Teoria maszyn i mechanizmów (Warszawa: Wydawnictwa Naukowo-Techniczne)

[16] Gronowicz A 2003 Podstawy analizy układów kinematycznych (Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej)

[17] Frączek J and Wojtyra M 2008 Kinematyka układów wieloczłonowych : metody obliczeniowe (Warszawa: Wydawnictwa Naukowo-Techniczne)

[18] Krzyżanowski P 2012 Obliczenia inżynierskie i naukowe : szybkie, skuteczne, efektywne (Warszawa: Wydawnictwo Naukowe PWN)

[19] Kurnik W 2012 Wykłady z mechaniki ogólnej (Warszawa: Oficyna Wydawnicza Politechniki Warszawskiej)

[20] Buśkiewicz J 2008 On the application of the mathematica environment for solving the problems of dynamics of mechanisms of third class Vibrations in Physical Systems 23 pp. 71–78