Indonesian Journal of Physics
Vol. 35. No.1, 23 August 2024
Journal homepage: http://ijphysics.com

Object's Movement Simulation with Air Drag: Aerodynamics Wall and Knuckle's Effect

Daniel Tohari[1*], Fiki Taufik Akbar Sobar[1]
[1]*The Faculty of Mathematics and Natural Science, Bandung Institute of Technology*
Jalan Ganesha 10, Bandung 40132, Indonesia

(Received: 17 May 2024, Revised: 7 June 2024, Accepted: 25 June 2024)

Abstract

The drag force of air and objects can be analyzed using the Stokes or Quadratic, also known as the Newtonian method. In this research, a Newtonian model was created numerically in Python using a 4th order Runge-Kutta integrator. The integrator will solve the acceleration function experienced by the object when given air resistance into a position function. The object's movement influenced by variation of drag's coefficient will provide variations in the location of the aerodynamic wall, the condition when the object moves vertically downwards. The movement of an object that is influenced by air resistance under certain conditions will also experience a lifting force, due to the rotational orientation angle of the object. The movement of objects influenced by these two forces can be modeled as knuckle movement. The results obtained from the numerical solution of the two object conditions are then made into a simple application in the form of a GUI so that users can easily operate the object simulation. This research will be a solution for students to understand the motion of objects influenced by the drag forces that occur when taking classical mechanics courses.

Keywords: Simulation, drag force, aerodynamic wall, knuckle's effect, the 4th order of Runge-Kutta

INTRODUCTION

Phenomena that occur in the surrounding environment can be modeled with physics. Numerical computing methods are often used to model the phenomena that occur. Remember that with this system, researchers can make some predictions of natural phenomena easily and efficiently [1]. Research was carried out on moving objects in the air that experience air resistance. The effects that arise from this air resistance force are aerodynamic walls and effects knuckles or the effect of shaking objects. In addition to being affected by air resistance, the knuckle effect is also affected by lifting force.

The simulation uses the Python language program and the Qt Designer application version 5.0. Python is a programming language that is easy to apply to all physical models. There are several advantages that Python has in modeling systems, namely, interface the resulting applications and program outputs are interactive and have the ability to carry out high-level programs [2]. The integrator in this research uses the 4th order Runge-Kutta, which has relatively high accuracy with small precision settings for ODE (ordinary differential equation) [3,4]. The method for solving problems of the motion of objects in the air by considering the quadratic drag force model is carried out numerically to be a solution for solving the same problem using the analytical method, considering that the quadratic drag force equation has a solution that is not simple.

The research carried out has quite a significant urgency, considering that there still needs to be more visualization of the movement of objects in the air with the phenomenon of aerodynamic walls and the effects of knuckles. The drag force factor of the Newtonian or quadratic model in Newtonian mechanics courses also makes students need help understanding the physical character or movement of these objects. This research will motivate students to understand better the patterns and behavior of the motion of objects in the air that take into account air resistance and attract students' interest in studying physics using computational methods.

THEORY/CALCULATION

Aerodynamics Wall

The air drag force with the Newtonian model is proportional to the negative value of the square of

[1*] Daniel Tohari. E-mail address: daniel.catatanpersonal@gmail.com

the velocity of an object moving through the air. Equation (1) shows the drag force of the Newtonian model:

$$\vec{F} = -bv^2\ \hat{v} \qquad (1)$$

b in Equation (1) is the air drag coefficient whose value depends on the density of the air (ρ), a constant value of around 0.4 to 0.5 based on experiments carried out by Eiffel, and the radius of the object (r) [5]. The drag coefficient b can be written in Equation (2). The value v^2 is the speed of an object moving in the air for two-dimensional object motion. It is a unit vector that also explains the direction of the drag force acting on the object in the opposite direction to its motion. The drag force between the object and the air produces the acceleration of the object, which is slowed down by the acceleration function in the x-axis and y-axis directions according to Equations (3) and (4):

$$b = \frac{1}{2}\rho C_D \pi r^2 \qquad (2)$$

$$a_x = -\frac{b}{m} v\, v_x \qquad (3)$$

$$a_y = -g - \frac{b}{m} v\, v_y \qquad (4)$$

Objects moving with air drag will experience a change in the direction of motion only in the vertical direction at a certain time, due to a reduction in the velocity value in the direction of the horizontal axis. Objects that have experienced pure vertical downward motion seem to be blocked by a virtual wall called an aerodynamic wall. The location of this aerodynamic wall can be determined numerically using the function ‘for’ in Python. This function is executed when the horizontal velocity value approaches zero and when this condition is reached, the aerodynamic wall will be plotted.

Knuckle Effect

Objects that move in the air, apart from experiencing air drag, will also experience lifting force, which results in a swaying effect in the side direction or what is often known as a knuckle effect. The knuckle effect on objects is influenced by lateral force, angular velocity, and rotational orientation angle [6]. Determining the angle will produce an object trajectory with a knuckle effect different. Figure 1 shows several angular directions on the ball that experience the knuckle effect in the air.

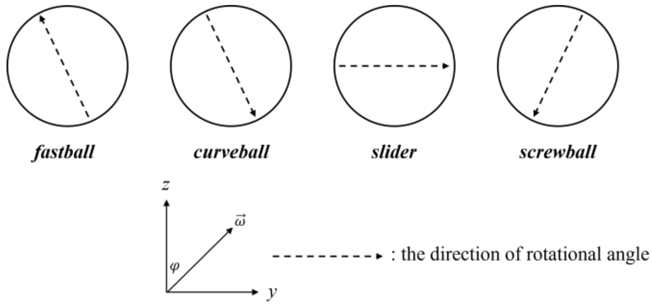

Figure 1.The direction of rotational angle

Knuckle effect on this research was carried out using a baseball with a numerical function that can be written in Equations (5) to (12):

$$\frac{dx}{dt} = v_x \qquad (5)$$

$$\frac{dy}{dt} = v_y \qquad (6)$$

$$\frac{dz}{dt} = v_z \qquad (7)$$

$$\frac{dv_x}{dt} = -F(v)\ v\ v_x \qquad (8)$$

$$\frac{dv_y}{dt} = -g - F(v)\ v\ v_y \qquad (9)$$

$$\frac{dv_z}{dt} = -F(v)\ v\ v_z + g\ G(\varphi) \qquad (10)$$

$$\frac{d\varphi}{dt} = \omega \qquad (11)$$

$$\frac{d\omega}{dt} = 0 \qquad (12)$$

the angular speed in Equation (11) physically is a result of the ball’s rotation by the ‘pitcher’, so that apart from swaying laterally (experiencing a knuckle effect), the ball will rotate. This angular speed occurs when the ‘pitcher’ throws a baseball at the orientation angle, φ. The drag force and lift force experience in

Equations (8), (9), and (10) can be written in Equations (13) and (14) [6]:

$$\frac{f_D}{m} = -F(v)\, v\, \hat{v} \tag{13}$$

$$G(\varphi) = 0{,}5[\,\sin(4\varphi) - 0{,}25\sin(8\varphi) + 0{,}08\sin(12\varphi) - 0{,}025\sin(16\varphi)\,] \tag{14}$$

The function in Equation (14) tells about the lateral force (a sideways direction force on the object). It depends on orientation angle based on the Watts and Sawyer's experiment for a baseball and mathematically constructed by Giordano [6]. This lateral force will produce the 'wobble' of object sideways when an object is moving in the air. It can be hypothesized for another types of ball will have different lateral force form.

The function $F(v)$ in Equation (13) is expressed as in Equation (15) and take value for v_d = 35 m/s and Δ = 5 m/s for this experiment [6]. The variations value of v_d and Δ can be used for the further research. The function $F(v)$ is a drag force per unit mass constant and it depends on object's speed on the air.

$$F(v) = 0{,}0039 + \frac{0{,}0058}{1 + \exp[(v - v_d)/\Delta]} \tag{15}$$

SIMULATION AND RESULTS

The flow chart for this numeric research can be seen in Figure 2:

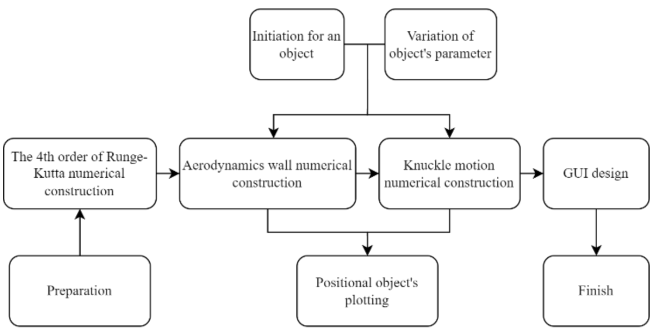

Figure 2. Flow chart for numerical experiment

the first step is preparing the equipment, a personal computer with the specifications in Table 1:

Table 1. Personal computer specifications

Components	Types
Operating system	Windows 11 Pro 64 bit
Motherboard	B450M
Processor	AMD Ryzen 5 3600-6 Core
RAM	16 GB DDR4
Storage	SSD PNY CS2241 x4 M.2
VGA	Nvidia GeForce GTX 1650

The 4th order RK integrator is defined as a function in the Python language before performing numerical programming to determine the motion of an object with an aerodynamics wall and knuckle effect. For the further research, comparing the integrator method for modelling these phenomena is necessary to see the accuracy differences between the several integrators used.

Aerodynamics Wall

The acceleration function in Equations (3) and (4) is integrated into a position function in the horizontal and vertical directions with a 4th order RK integrator. The initiation objects for the aerodynamics wall program can be seen in Table 2.

Table 2. Object's initiation for aerodynamics wall program

Parameters	Values
Gravity acceleration	9.8 m/s^2
Object's mass	0.8 kg
Object's velocity	$v_x = 20$ m/s ; $v_y = 20$ m/s
Initial position	$x_0 = 0$ m ; $y_0 = 15$ m
Drag coefficient	0.10

The variations of drag coefficient used in this research are shown in Table 3. It chosen randomly. There is no limitation or restriction to determine the value of drag coefficient, also the drag coefficient can be determined with the Equation (2).

Table 3. The variations of drag coefficient

Drag coefficients	Values
b_1	0.10
b_2	0.25
b_3	0.40
b_4	0.55

the solution for acceleration function of the object using the 4th order of RK integrator in Python is constructed in Visual Studio Code with the results of the object position function shown in Equation (16):

$$y(t+\Delta t) = y(t) + \frac{h\left(k_{1y}+2k_{2y}+k_{3y}+k_{4y}\right)}{6} \quad (16)$$

the results of the horizontal and vertical position function in Equation (16) are then plotted, and the location of aerodynamics wall with a numerical code is arranged logically when the horizontal velocity of an object approaches zero.

Aerodynamics Wall Position

Based on the object initiation given in Table 2, Equation (3), Equation (4), and variations in the drag coefficient values in Table 3, the results of plotting the position of the object with the location of the aerodynamics wall are obtained. It can be analyzed as shown in Figure 3. The trajectory pattern shown by an object subjected to air resistance cannot be analyzed using classical parabolic motion or simple kinematics, which was first discovered by Galileo [1]. The drag factor depends on the value of the drag coefficient, b will make the equations of motion of objects easy to model numerically. The greater the drag coefficient value will indicate the presence of an aerodynamics wall that becomes near the initial position of the object. The location of the aerodynamics wall for each object with a value of drag 1 up to drag 4 sequentially are 14.38 meters, 7.02 meters, 4.73 meters, and 3.61 meters.

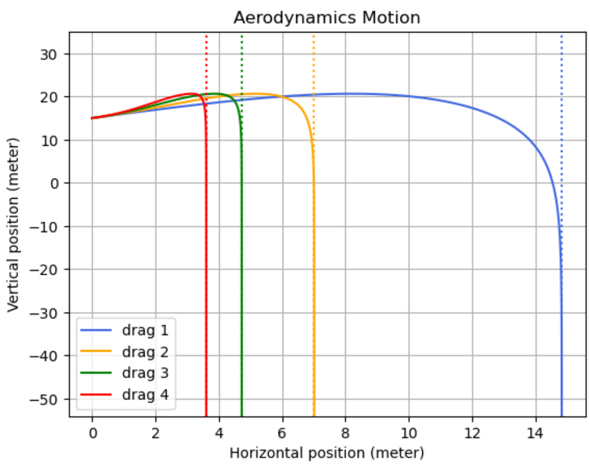

Figure 3. Aerodynamics wall position

The objects, when entering the aerodynamic wall area, will move only in the vertical direction. An interesting condition occurs when the aerodynamic wall only appears when the object is already at a negative height (under the ground), so the aerodynamic wall cannot actually be observed in this condition. The solution that can be used is to change the initial velocity value of the y-axis, which is greater, and reduce the initial velocity in the x-axis direction so that it will make the object move with a greater elevation angle calculated from the x-axis according to Equation (17) [8].

$$\tan\theta = \frac{v_y}{v_x} \quad (17)$$

Knuckle Effect

Initiation of an object for knuckle effect can be seen in Equations (18) to (25) and Table 4:

$$x(t=0) = 0 \quad (18)$$

$$y\,(t=0) = 0 \quad (19)$$

$$z\,(t=0) = 0 \quad (20)$$

$$v_x\,(t=0) = v_0\cos\theta \quad (21)$$

$$v_y\,(t=0) = v_0\sin\theta \quad (22)$$

$$v_z\,(t=0) = 0 \quad (23)$$

$$\varphi\,(t=0) = \varphi_0 \quad (24)$$

$$\omega\,(t=0) = \omega_0 \quad (25)$$

Table 4. Object's initiation for knuckle effect

Parameters	Values
Gravity acceleration	9.8 m/s^2
Initial speed	29.06 m/s
Elevation angle θ	4°
Orientation angle φ	0°
Angular speed ω	2.09 rad/s

variations for the initial rotation orientation angle of the object given in Table 5:

Table 5. Orientation angle variation

Orientation angle variation	Values (°)
φ_1	0.0
φ_2	22.5
φ_3	45
φ_4	67.5

Knuckle Effect Motion for Baseball

Based on the initiation's object given in Table 4, Equations (18) to (25), and variations in the object rotation orientation angle values in Table 5, the results of the sway pattern plot are obtained. Knuckle effect of the object observed from above (y-axis) as shown in Figure 4.

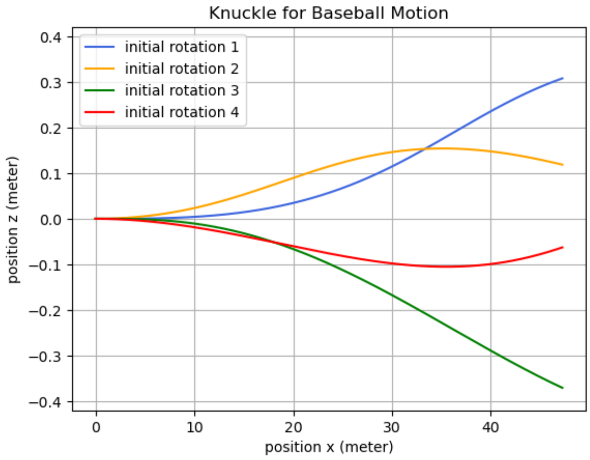

Figure 4. Variation of knuckle effect motion

The results obtained are in the form of movement pattern of knuckle effect on baseball with the sway leading from the positive z-axis towards the negative z-axis when the object's initial rotation orientation angle becomes larger, the movement pattern for rotation orientation angles of 0 and 22.5 degrees deviates upwards, while for rotation orientation angles of 45 and 67.5 degrees deviated downwards (relative on the graph). The four balls move in a curved path to the side (y-axis) with large deviation trajectory when the orientation angle difference is small enough. This trajectory occurs when the 'pitcher' combines rotational speed (ω) at angle φ, this is good also for the 'pitcher' who will learn/are learning to throw the baseball so that it creates a knuckle effect and as a result the player's opponent will have difficulty guessing the direction of the ball's movement and also based on this phenomena, knuckle's effect is quite difficult to determine numerically and moreover with analitic method. For the further experiment, using the real experiment in laboratory is needed to make knuckle's effect motion more easily observed directly.

GUI Design

The simulation was constructed in Qt Designer 5.0 to make it easier for users to run it later. The simulation design process with the Qt Designer 5.0 application can be seen in the flow chart in Figure 5:

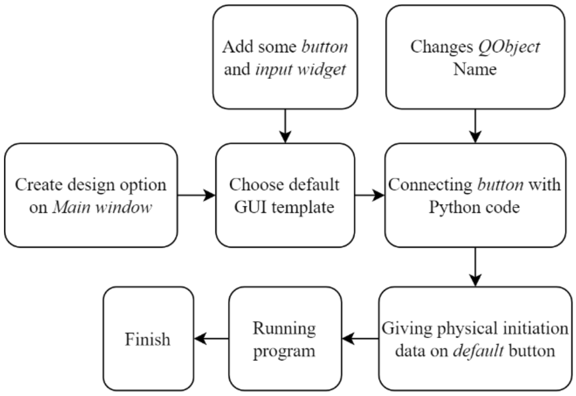

Figure 5. GUI design flow chart

The first step is selecting one of the features, 'new form' provided by Qt Designer 5.0. Option 'main window' is chosen as the general template design on GUI as seen in Figure 6:

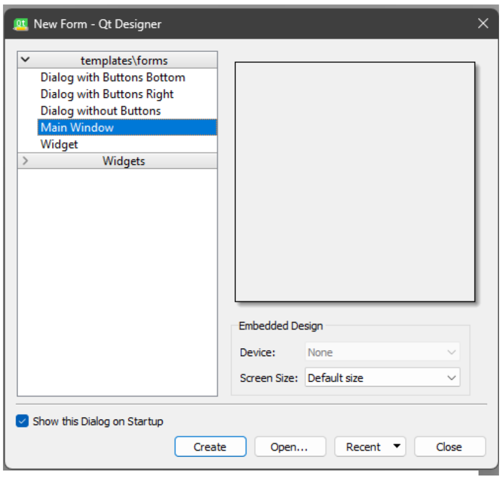

Figure 6. The first interface of Qt Designer 5.0

The next step is to set the display that will be displayed on the user when the simulation is run. Some features are used to add some buttons on the GUI using a ribbon that appears on the left in the Qt Designer 5.0 application. The display of initial design setting for users using the simulation can be seen in Figure 7:

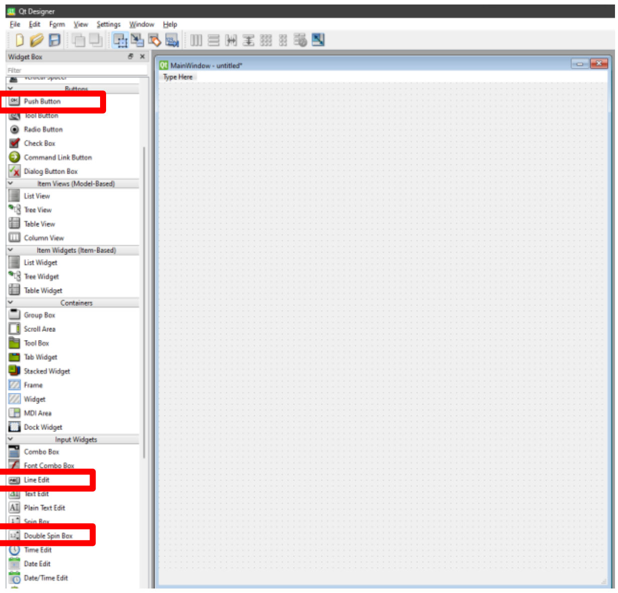

Figure 7. Some button options for designing simulation

initiation of objects needed for make it works on the simulation GUI in this research, namely 'double spin box', 'line edit', and 'push button' types. Input or initial information such as object mass, object speed, drag coefficient, and initial position of the object uses label display widget. The 'double spin box' feature can only read input in the form of numbers, whereas 'line edit' is used specifically for input that will be varied in this research. Every button feature for this simulation is set for its size, label, initial value, and its interface in section property editor shown in Figure 8, each input is given a different label to be connected to the Python code in the Visual Studio Code application so that the simulation can run automatically when the run button is executed.

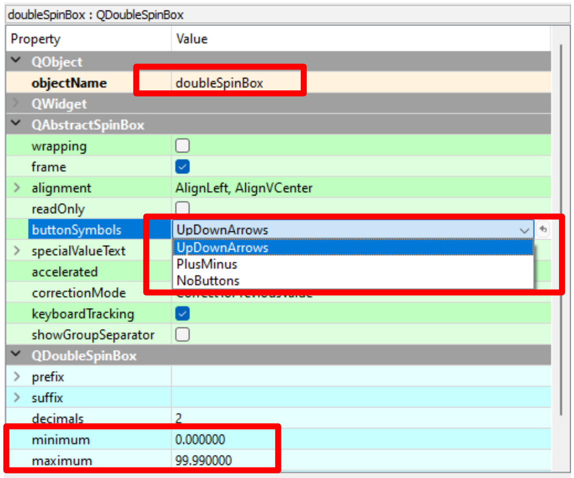

Figure 8. Properties button features

GUI Output

Two simulations ready to be processed into a GUI are then converted to Python in the Terminal on Visual Studio Code. Several numeric codes are changed to be "called" when each button in the simulation application is executed. The views of two simulations can be combined into the same GUI. Figure 9 shows the interface of the simulation. The variation of the air resistance coefficient with the rotation orientation angle can be directly entered by separating the initiations with a semicolon (;).

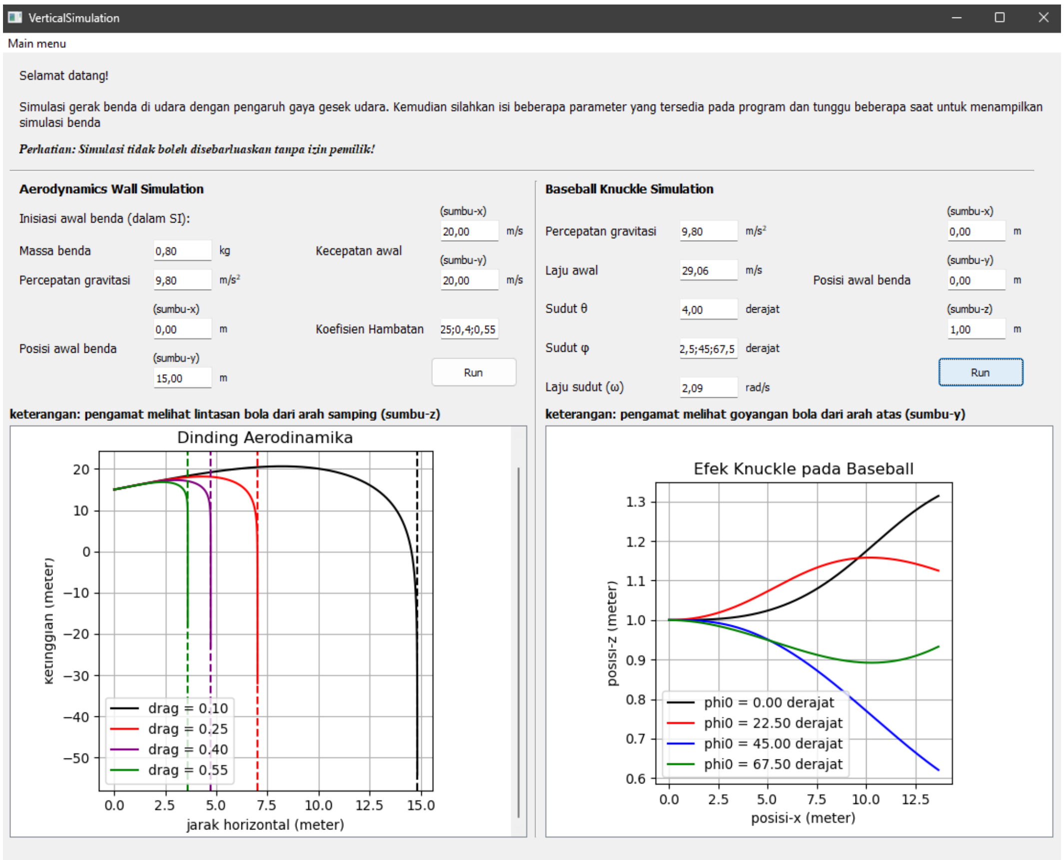

Figure 9. GUI interface

CONCLUSION

Research on an object that moves under the influence of air resistance produces motion aerodynamics wall and knuckle effect. These two objects are simulated in the Python language program with a 4th order Runge-Kutta integrator with different object condition initiations. The greater the variation in the drag coefficient, the distance of the aerodynamics wall will be closer to the initial position of the object and the greater the orientation angle of rotation of the ballbaseball will make a knuckle motion in a random one. The simulation results in the Visual Studio Code were then simulated with the Qt Designer 5 application so that users could easily vary and analyze drag coefficient values and rotation orientation angles.

REFERENCES

[1] D. V. Schroeder, Physics Simulations in Python A Lab Manual 4th edition, Creative Common License, Utah, IV, 2021.

[2] K. Parganiha, Linear Programming with Python and Pulp, *International Journal of Industrial Engineering Research and Development*, **9**, 1, 2018.

[3] Z. Gao, Research on Numerical Integration Algorithm in Molecular Dynamics Simulation, *Advances in Engineering Research*, **138**, 226, 2017.

[4] J. Jung, C. Kobayashi, and Y. Sugita, Kinetic Energy Definition in Velocity Verlet Integration for Accurate Pressure Evaluation, *Journal of Chemical Physics*, **148**, 1, 2018.

[5] C. Cohen *et al.*, The aerodynamic wall, *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **470**, 2161, 2013.

[6] R. Fitzpatrick, Introduction to Computational Physics, Flooved, London, 126, 2013.

[7] S. Wulansari, R. Oktamaypasha, and B. Setiaji, Slingshot Simulation: Application to Improve Parabolic Motion Concept Comprehension for High School Students, *Jurnal Pendidikan Fisika*, **8**, 58, 2023.