



A GUI for Computing Hybrid Nanofluid Boundary Layer Flow using Bvp4c Solver in Matlab: Educational Purposes for University Students

Farizza Haniem Sohut¹, Siti Khuzaimah Soid^{2,*}, Anuar Ishak¹

¹ Department of Mathematical Sciences, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia.

² School of Mathematical Sciences, College of Computing, Informatics, and Mathematics, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia.

ARTICLE INFO

Article history:

Received 29 November 2022

Received in revised form 3 April 2023

Accepted 14 April 2023

Available online 25 April 2023

Keywords:

Fluid dynamics; boundary layer;
graphical user interface; education

ABSTRACT

This work offers illustrations of a graphical user interface (GUI) in MATLAB software. The governing partial differential equations are extremely useful in many industries, especially in analyzing the problem regarding this application through the analysis of the thermal and acceleration characteristics in a system. It is significant for determining whether the system is in the best momentum and whether the surface or ambient flow is good for the system to run. In addition, fluid mechanics is a subject required in certain universities (i.e., Engineering Faculty), thus it is important to have GUI as their learning tool. This GUI is a feature in MATLAB that is used to illustrate the velocity and temperature profiles and to display the numerical analysis on skin friction and local Nusselt numbers attractively and interactively. This interface can be designed by dragging and dropping the visual components onto the layout design. The GUI can be very helpful to undergraduate students in increasing their fundamental understanding of fluid flow phenomena.

1. Introduction

Fluid mechanics is one of the important courses available at several universities. For undergraduate students majoring in mechanical, energy, civil, construction or chemical engineering, this course is a bit tough since the fundamental fluid concepts and principles are covered, plus they need to have a very strong mathematical background to solve fluid-related engineering problems. Normally, problems related to fluid mechanics, such as fluid dynamics require students to transform the partial differential equations (PDEs) into nonlinear ordinary differential equations (ODEs) using the similarity transformation [1-3]. According to Minichiello *et al.*, [4], most problems in fluid mechanics require students to apply calculus to derive partial differential equations. Also, the early fluid mechanics courses are more mathematically focused and less illustrative of the visually appealing flows seen in everyday life. Unsurprisingly, undergraduates frequently describe these courses as mathematically burdensome, cognitively challenging and aesthetically unappealing.

* Corresponding author.

E-mail address: khuzaimah@tmsk.uitm.edu.my

<https://doi.org/10.37934/ard.103.1.111>

Students are enticed to learn fluid mechanics when learning through visualisation [5]. Visual experience, in contrast to mathematical problem solving, has been shown to develop a deeper conceptual understanding and a better perception of fluid flow. This deeper understanding often makes fluid mechanics research interesting as well as motivating. Computational fluid dynamics (CFD) is a tool that allows the numerical solution of fluid flow problems through the use of computers. Traditionally, the only way for an engineer to optimise their design is to conduct physical tests on product prototypes. With the rise of computers and ever-growing computational power, the field of CFD has become a commonly applied tool for predicting real-world physics. In a CFD software analysis, fluid flow and its associated physical properties, such as velocity, pressure, viscosity, density and temperature, are calculated based on defined operating conditions. Besides that, Xiao [6] mentioned that the use of CFD software is beneficial in assisting students in understanding the abstract flow phenomenon. Also, to inspire students with flow visualization and CFD solutions can help students better understand basic equations and their applications.

2. Literature Review

2.1 Fluid

A fluid is defined as a substance that deforms continuously under the action of shear stress. An important consequence of this definition is that there are no shear stresses when a fluid is at rest. However, the motion of fluid occurs when some forces are applied to the fluid, giving rise to fluid dynamics [7]. Both liquids and gases are fluids. Some substances, such as glass are technically classified as fluids. However, the rate of deformation in glass at normal temperatures is so minor as to make its consideration as a fluid impractical. Burmeister [8] mentioned that hydrodynamics is one of the principles of fluid dynamics, where fluid movements cause the existence of forces on a solid surface. There is a conservation law in fluid motion in which the conservation of mass, the conservation of momentum and the conservation of energy are the identifications of the phenomena of fluid behaviour. Some fluids particularly liquids, have densities that remain almost constant across vast pressure and temperature ranges. Fluids with this property are typically considered incompressible. From a static point of view, air is a compressible fluid and water is incompressible [9].

2.2 MATLAB

The MATLAB boundary value problems (BVPs) solver, also called *bvp4c* solver introduced by Shampine *et al.*, [10], for solving BVPs for ordinary differential equations in MATLAB. They implemented the 3-stage Lobatto IIIa finite difference method, which provides fourth-order accuracy uniformly in the interval where the function is integrated, known as the collocation formula. To initiate the *bvp4c* solver, the nonlinear ODEs should be first reduced to a first-order system before it can be solved. In this solving approach, the appropriate guess for the initial values and the boundary layer thickness η_∞ is important to obtain the correct numerical solutions. Numerous problems in science and engineering arise in multiple ordinary differential equations with multiple boundary conditions that are indicated by the BVPs. The BVPs may have more than one solution or only one solution or probably the solution does not exist. As there might be many solutions, *bvp4c* requires guessing values for the solution of interest [11]. Besides that, this *bvp4c* solver is attractive due to its capability of controlling the size of residuals even for crude mesh points. It is believed that the results obtained are reliable and robust since many researchers implement *bvp4c* to generate results and it demonstrates a consistent result with the experimental analysis [12-15]. According to Shampine *et*

al., [11], *bvp4c* is an effective solver, but its fundamental method and the computational settings are not appropriate for high accuracies or problems with extremely sudden changes in their solutions.

2.3 Graphical User Interface (GUI)

Human-computer interaction is sometimes assumed to be completely dependent on the design and use of computer technology. However, underneath such technologies there is a type of user interface known as GUI, which allows people to interact with electronic devices *via* graphical icons and audio indicators. Designing the look and behaviour of a GUI is vital in software application programming because GUI displays elements that transmit messages and represent actions that the user may do. A mouse for example, may be used to move a cursor and click on a programme icon to launch it. In the meantime, there is a lot that can be designed with a command line interface where one needs to know how and where to browse to the program's directory, list the files and then run the file. Levy [16] stated that GUI is a computer application that allows people to interact with computers using symbols, visual metaphors and pointing devices. Since it was proven by Levy [16] that GUI has made use of computers become easier to master, more pleasurable and natural, a lot of applications were conducted using GUI, based on the previous studies.

Similarly, another project by Tahri *et al.*, [17] created MATLAB-based GUI application but with a different aim, which is to help decision-makers prioritize choices and criteria goals. The researchers employed two separate example applications to improve the decision-support system process in a variety of domains. The project was also a success based on the results shown. In addition, a project on reprinting systems with a graphical user interface was also completed. The goal of this project is to use constraint satisfaction to aid the human process of interpreting Japanese historical texts. In contrast to the projects mentioned in the previous paragraph, the constructed GUI is intuitively useful, however there are several areas in the GUI that need to be enhanced according to a survey conducted by the researchers [18]. This is due to the fact that the system is still being worked on. On the other hand, the GUI is still viable and useful since future GUI advancements will undoubtedly be made, resulting in better solutions being provided [18]. Furthermore, Cetin *et al.*, [19] explore a GUI prepared by MATLAB to demonstrate the results of the aforementioned fundamental fluid mechanics problems (viscous flow), consisting of external and internal flow problems. Hence to summarize, the usage of a GUI is global because it is adaptable and suited for a wide range of applications in diverse industries.

In order to see the illustrations of the results, students need to go through such a long process. Such as transformations of the PDEs to ODEs and then numerically solve using a *bvp4c* solver in MATLAB software. Students have to deal with the coding and so on. With the aid of a GUI, the colourful interface and interactive arrangement of the interface will make the illustrations of the simulation more interesting. Not only students, lecturers and engineers will be interested to know how the final output compares to the long processes. Hence, this study presents a GUI prepared by MATLAB to assist students and instructors in handling the mathematical derivation and visualizing the results of the fundamental fluid flow problems. We believe that this kind of GUI can be very helpful to undergraduate students in increasing their fundamental understanding of fluid flow phenomena. In this introductory text, the option is taken to explain the basic ingredients of the finite element method on a very simple, purely mathematical, problem and to give fluid dynamics illustrations in detail only for simple problems. Thus, the current study aims to illustrate the final results by developing a GUI with GUIDE in MATLAB for educational purposes. The reasons were to assist students, lecturers and engineers in further understanding this fluid dynamics, especially the results of drag force and heat transfer, as well as the effects of the parameters on the profiles.

3. Methodology

3.1 Model Analysis

The problem of the unsteady 3D rotating flow of a hybrid nanofluid on a stretching sheet was considered in a study by Mohd Sohut *et al.*, [14] and is given as follows,

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{1}$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} = 2\omega v + \frac{\mu_{hnf}}{\rho_{hnf}} \frac{\partial^2 u}{\partial z^2} \tag{2}$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} = -2\omega u + \frac{\mu_{hnf}}{\rho_{hnf}} \frac{\partial^2 v}{\partial z^2} \tag{3}$$

$$\frac{\partial w}{\partial t} + u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z} = \frac{\mu_{hnf}}{\rho_{hnf}} \frac{\partial^2 w}{\partial z^2} \tag{4}$$

$$\frac{\partial T}{\partial t} + u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} + w \frac{\partial T}{\partial z} = \frac{1}{(\rho C_p)_{hnf}} \left(k_{hnf} + \frac{16\sigma^* T_\infty^3}{3k^*} \right) \frac{\partial^2 T}{\partial z^2} \tag{5}$$

Subject to boundary conditions,

$$u = u_w(x, t) = \frac{cx}{1 - \alpha t}, \quad v = 0, \quad w = 0, \quad T = T_w \quad \text{at } z = 0 \tag{6}$$

$$u \rightarrow 0, \quad v \rightarrow 0, \quad T \rightarrow T_\infty \quad \text{as } z \rightarrow \infty$$

The bvp was solved numerically using a finite difference method, which is known as the bvp4c. This package is a built-in code available in MATLAB software which is basically based on the three-stage Lobatto IIIA formula and is fabulously described by Shampine *et al.*, [10]. Initially, the system of similarity equations from the higher order is converted into the first order set of ODEs *via* incorporating the new variables,

$$\frac{\mu_{hnf}/\mu_f}{\rho_{hnf}/\rho_f} f''' + ff'' - f'^2 + 2\Omega h - \beta \left(f' + \frac{\eta}{2} f'' \right) = 0 \tag{7}$$

$$\frac{\mu_{hnf}/\mu_f}{\rho_{hnf}/\rho_f} h'' + fh' - f'h - 2\Omega f' - \beta \left(h + \frac{\eta}{2} h' \right) = 0 \tag{8}$$

$$\frac{1}{Pr} \frac{1}{(\rho C_p)_{hnf}/(\rho C_p)_f} \left(\frac{K_{hnf}}{K_f} + \frac{4}{3} Rd \right) \theta'' + f\theta' - \beta \frac{\eta}{2} \theta' = 0 \tag{9}$$

Subject to the boundary conditions,

$$\begin{aligned}
 f(0) &= 0, \quad f'(0) = \lambda, \quad h(0) = 0, \quad \theta(0) = 1 \\
 f'(\eta) &\rightarrow 0, \quad h(\eta) \rightarrow 0, \quad \theta(\eta) \rightarrow 0 \quad \text{as } \eta \rightarrow \infty
 \end{aligned}
 \tag{10}$$

The step or mesh size and the position of the edge of the boundary layer must be adjusted for distinct values of the controlling parameters to maintain the certain goal of accuracy and precision. We here exploited the deval function in the available MATLAB software to compute the outcome at particular points. Figure 1 shows the numerical method using bvp4c solver in MATLAB to obtain numerical solution of the boundary layer problems. In order to obtain the numerical solutions, there is a long processes that we have to go through. Such as reducing the PDEs to ODEs before the system of ODEs being solved numerically using bvp4c. These numerical solutions are very important for engineering and industrial processes where the analysis of drag force and heat transfer is needed so that final products will have good performance. However, engineers, industrial people, engineering students, etc. are more interested in the results compared to the computational ones on the black box.

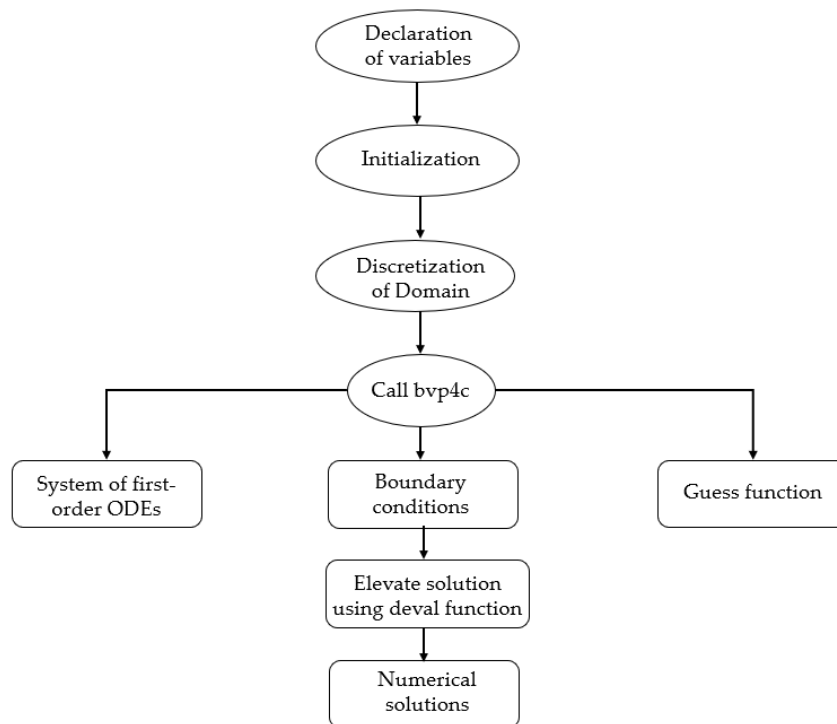


Fig. 1. Numerical approach using bvp4c solver in MATLAB [20]

Figure 2 presents the GUI flowchart or known as pictorial interface that provide one click control in the MATLAB software applications. Providing simple controls helps to achieve the specific outcome. GUI is an event-driven program due to the input that acquires at any given time and the use of the call-back functions to execute the program and provides results. Moreover, this product enables to illustrate and share comprehensive predictions about the influence of parameters on the mathematical governing equation.

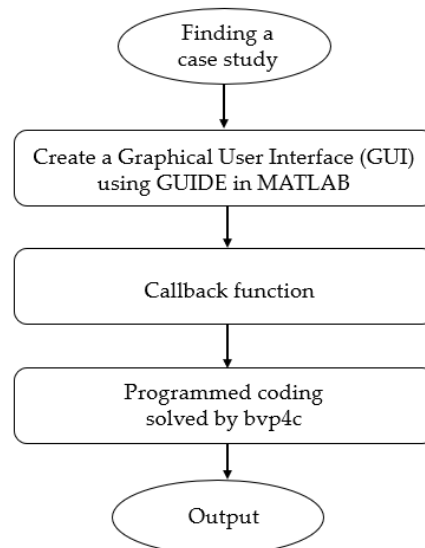


Fig. 2. GUI process flowchart

The GUI was able to make an analysis of the impacts of relevant parameters through graphical illustration and numerical data. The GUI of this project was developed using the GUIDE feature in MATLAB R2021a. The method of calculation used to solve the boundary layer problem was the `bvp4c` solver, which was employed in previous research [14]. This user interface was designed for two purposes. One of them was that the user can obtain boundary layer profiles based on the effects of various physical parameters. Then, the user can also obtain the results of the skin friction coefficient and the local Nusselt number.

In addition, GUI provide point-and-click control of your software applications, eliminating the need for others to learn a language or type commands in order to run the application. These interface can be shared both for use within MATLAB and also as standalone desktop or web applications. Figure 3 shows the visual components of the present study. This interface can be designed by dragging and dropping the visual components onto the layout design. The components of this interface as shown in Figure 3 and located on the left side where the component is dragged. There are many types of components that we can use to design the interface. However for this study, the components used are pushbutton, edit text, static text, panel and axes. For the parameters, edit text was used where we can type the desired values of the parameter and edit it if we want to vary the values. These relevant parameters follow the requirements of the model problem. While for the counter, calculate, reset and exit buttons respectively, the pushbutton component was applied. Each button was given a function and coded using the Call-back function, as mentioned in the flowchart in Figure 2. Call-back function is a function that executes when a user interacts with a uicontrol function. A component can have only one Call-back. However, components such as labels do not have Call-back because those components only display information. Also, the function behind each interface was coded and used in the calculation to display the output of the profiles and the numerical data of the skin friction coefficient and the local Nusselt number. Therefore, students can directly focus on the final output for their analysis on the fluid dynamics. It is zero cost and saves time. Furthermore, the choice of colours and arrangements to set up this interface plays an important role in capturing the attention of the viewers. Great visualisation can attract people's attention, thus providing a better understanding of the analysis. Additionally, a background picture can also be included to make the interface more interactive and attractive.

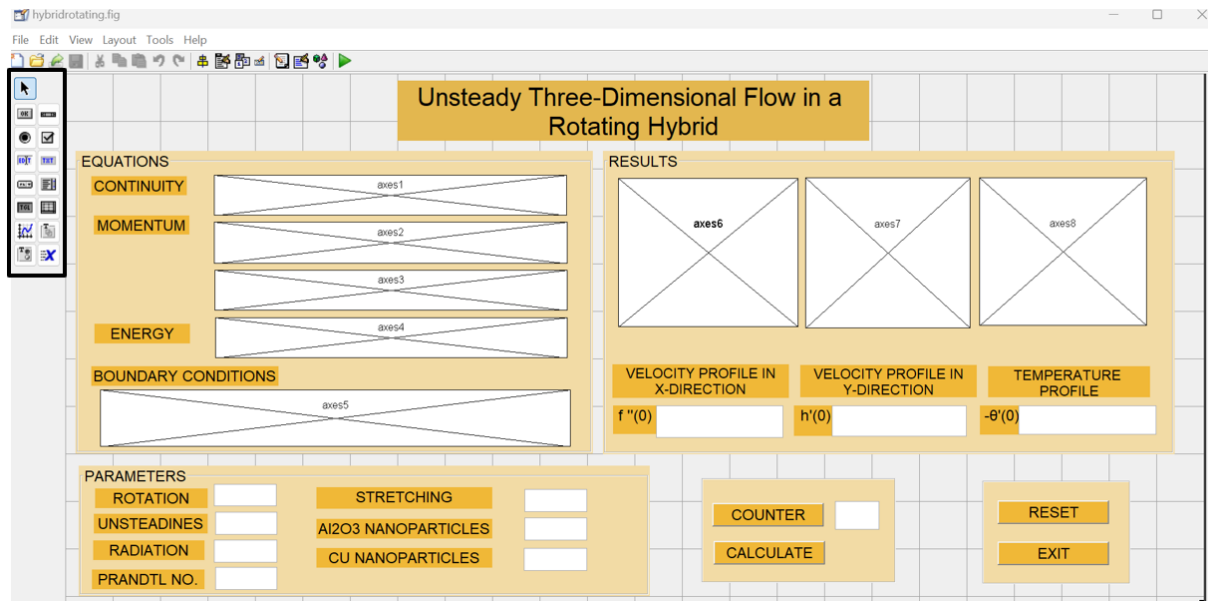


Fig. 3. Screenshot of the layout of the GUI

4. Results and Discussion

A GUI, often known as a pictorial surface allows for simple control to illustrate and provide thorough predictions about the influence of physical parameters on the governing equations. Screenshot of the steps using GUI are displayed in Figure 4 to 9, respectively. Figure 4 shows the interface of the GUI, which includes the "Equations" panel, the "Parameters" panel, the "Results" panel, the "Counter" button, the "Calculate" button, the "Reset" button and the "Exit" button, respectively. Here, all the governing equations and parameters follow the research on unsteady 3D flow in a rotating hybrid nanofluid over a stretching sheet. Next, to begin the analysis users need to hit the "Counter" button in order to display the governing equations and boundary conditions as depicted in Figure 5. The box beside the "Counter" button will appear with the number zero.

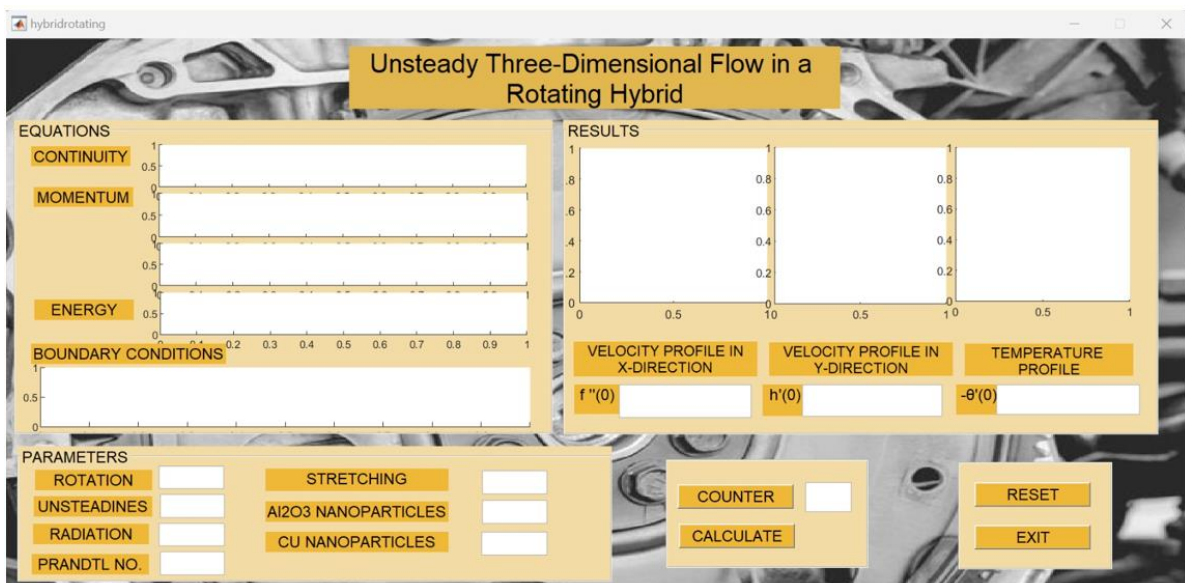


Fig. 4. Screenshot of the interface of the GUI

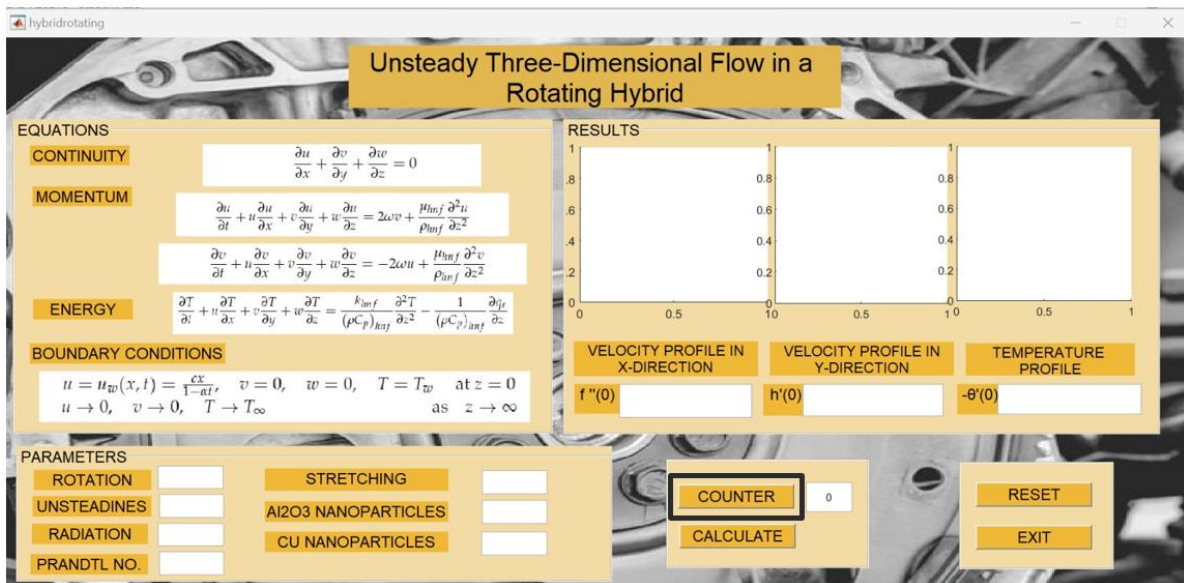


Fig. 5. Screenshot of the counter button

The next step is to define the values of the parameters, as shown in Figure 6. In the selection of the parameter values, users should follow the previously conducted research on the related problem. In this case, the unsteadiness parameter varies. The values used were -2, -1.5 and -1. Since the unsteadiness parameter varies, the other parameters should be fixed. The values for the other parameters, such as the rotation parameter was 0.01, the radiation parameter, 2.3, the Prandtl number, 6.2, the stretching parameter, 1, the alumina (Al₂O₃) nanoparticle volume fraction, 1 %, and the copper (Cu) nanoparticle volume fraction, 1 %, respectively. Once done defining the values of the parameters, users need to hit the "Calculate" button to generate the result, as illustrated in Figure 7.

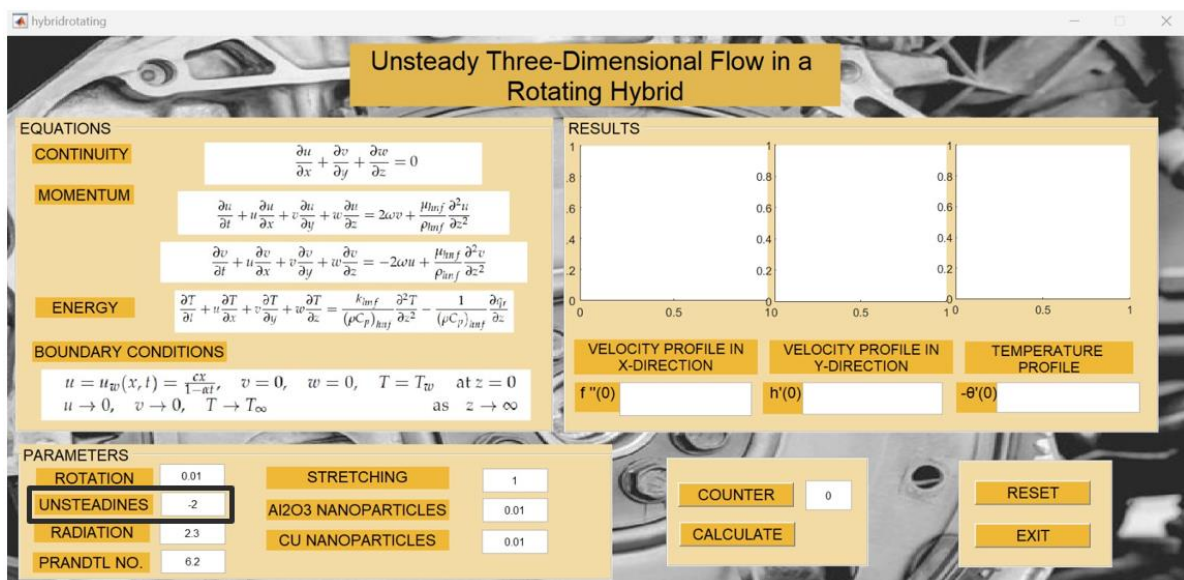


Fig. 6. Screenshot of the variation of the physical parameters

The results were displayed both graphically and numerically. The graph part includes the velocity profile on the x-axis, the velocity profile on the y-axis and the temperature profile, respectively. The numerical results display the skin friction coefficient in the x direction, the skin friction coefficient in the y direction and the local Nusselt number, respectively. It is worth mentioning that the number

displayed beside the "Counter" button will change to 1, which means it is done generating for the first time. Users will repeat the step of inserting the next value of the unsteadiness parameter and hitting the "Calculate" button until the desired results were obtained.

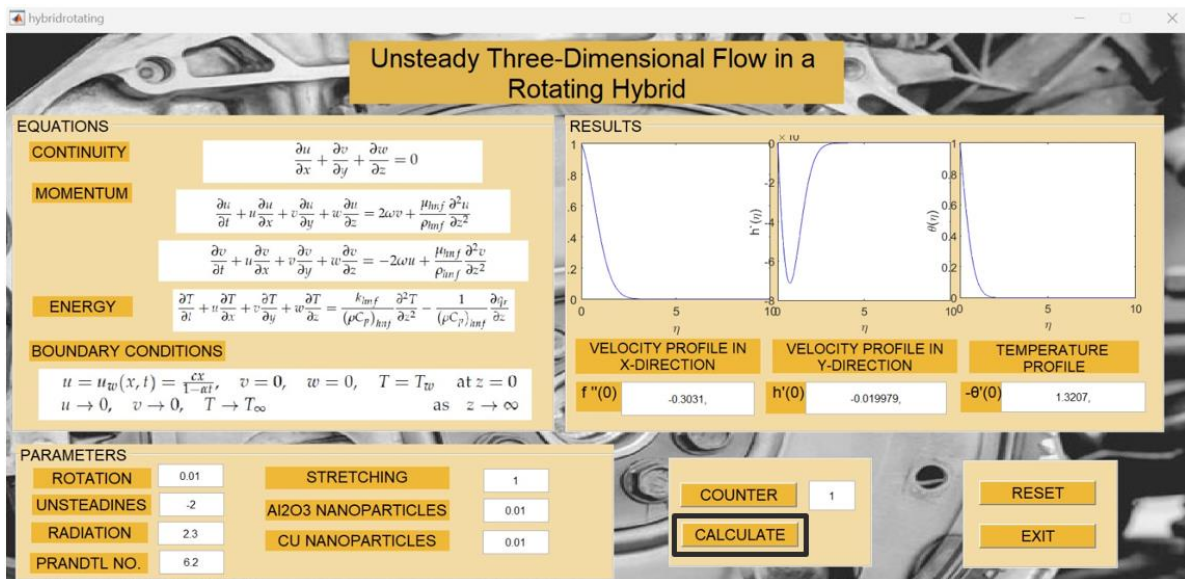


Fig. 7. Screenshot of the calculate button

Furthermore, Figure 8 shows the results obtained from the analysis. The number at the counter shows 3, which means there were 3 values generated. The profiles are illustrated with different colours in order to differentiate the effects of the unsteadiness. The results illustrated by GUI was in line with the result obtained by Mohd Sohut *et al.*, [14]. Hence, it proved that this interface can be used for the analysis of the fluid dynamics problems. In fact, it is very attractive and easier to learn fluid dynamics with the GUI. The effects on the profiles, skin friction coefficient and local Nusselt number can be seen clearly by the users. With the help of these visualizations, the users can understand the influences of the physical parameters on the fluid flows.

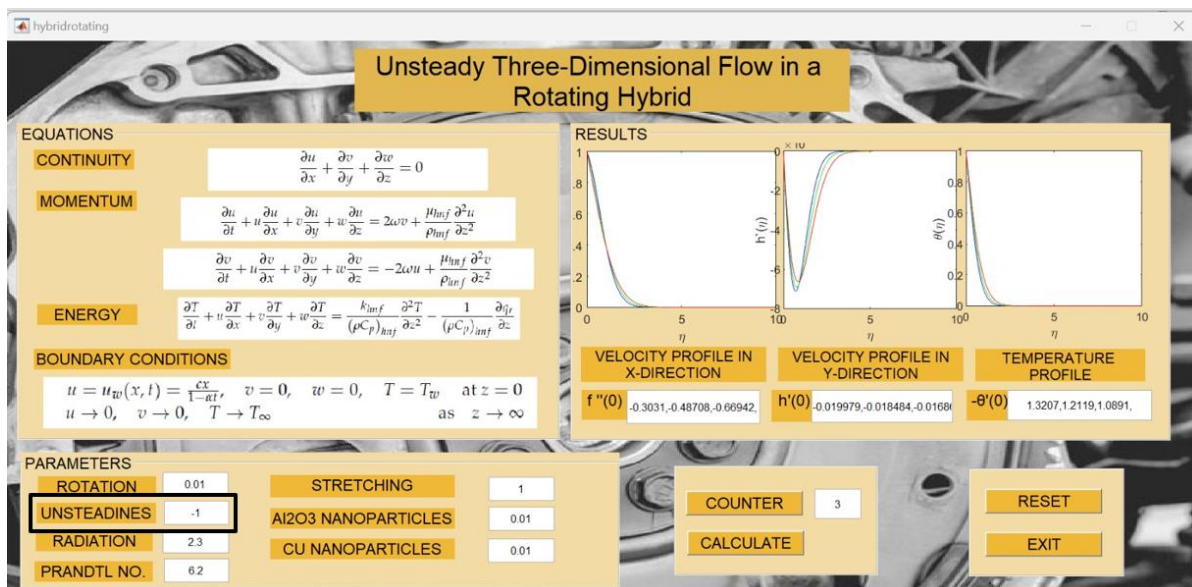


Fig. 8. Screenshot of the interface after the calculations

After the analysis was executed and the users want to analyze the effect of other parameters, they can reset the GUI. Users can hit “Reset” button as shown in Figure 9. All the parameters value will be erased as well as the results. If the users don’t want to continue the analysis, they can just hit the “Exit” but and the interface will be closed automatically.

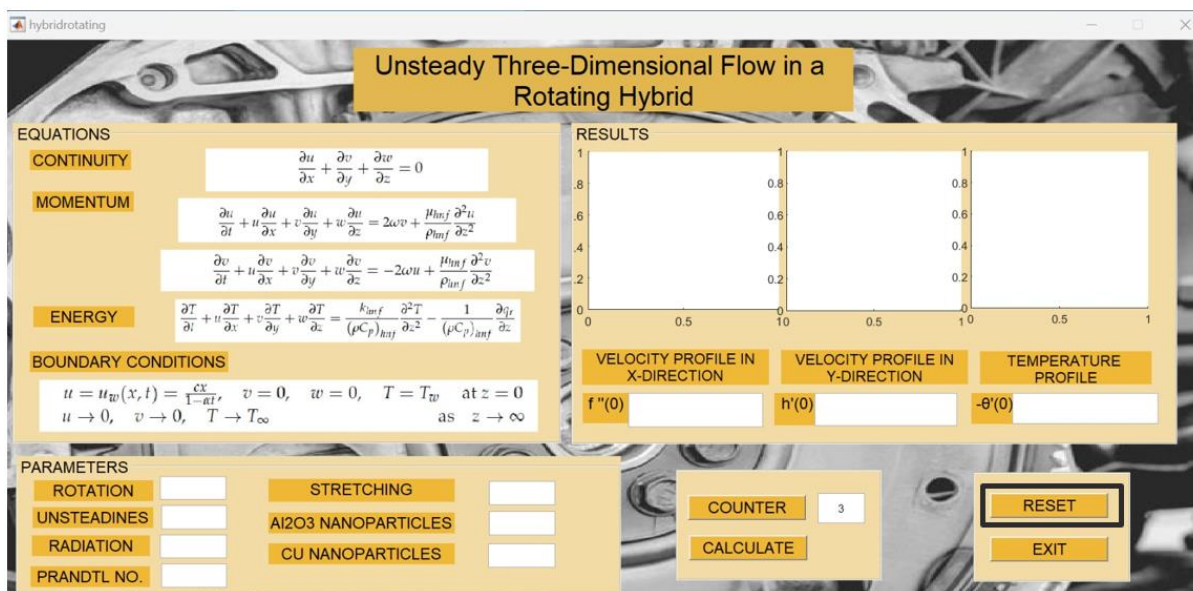


Fig. 9. Screenshot of the reset button

5. Conclusions

The simulator for computation of the boundary value problem was successfully developed and had met the aims and objectives of the current study. The objectives of this study were to develop a system that can numerically solve and illustrate the governing equations. Besides that, it illustrates the results attractively and interactively. From this study, we have gained a lot of new knowledge about building applications in MATLAB. The programming environment provided by MATLAB was also doable. This study was beneficial in fluid dynamic analysis for engineering students as well as mathematics students. This GUI can also be the fluid dynamics simulator application, especially for beginners, such as engineering student who is required to learn fluid mechanics as one of the university requirements.

Acknowledgement

This research was funded by a grant from the Ministry of Higher Education of Malaysia (FRGS/1/2021/STG06/UITM/02/11) and Universiti Teknologi MARA Shah Alam is gratefully acknowledged.

References

- [1] Ishak, Anuar, Roslinda Nazar, Norfifah Bachok, and Ioan Pop. "MHD mixed convection flow near the stagnation-point on a vertical permeable surface." *Physica A: Statistical Mechanics and its Applications* 389, no. 1 (2010): 40-46. <https://doi.org/10.1016/j.physa.2009.09.008>
- [2] Khashi'ie, Najiyah Safwa, Norihan Md Arifin, Roslinda Nazar, Ezad Hafidz Hafidzuddin, Nadiyah Wah, and Ioan Pop. "Mixed convective flow and heat transfer of a dual stratified micropolar fluid induced by a permeable stretching/shrinking sheet." *Entropy* 21, no. 12 (2019): 1162. <https://doi.org/10.3390/e21121162>
- [3] Sohut, Farizza Haniem, Anuar Ishak, and Siti Khuzaimah Soid. "MHD Stagnation Point of Blasius Flow for Micropolar Hybrid Nanofluid toward a Vertical Surface with Stability Analysis." *Symmetry* 15, no. 4 (2023): 920. <https://doi.org/10.3390/sym15040920>

- [4] Minichiello, Angela, David Armijo, Sarbajit Mukherjee, Lori Caldwell, Vladimir Kulyukin, Tadd Truscott, Jack Elliott, and Aditya Bhouraskar. "Developing a mobile application-based particle image velocimetry tool for enhanced teaching and learning in fluid mechanics: A design-based research approach." *Computer applications in engineering education* 29, no. 3 (2021): 517-537. <https://doi.org/10.1002/cae.22290>
- [5] Hertzberg, Jean, Bailey Renee Leppek, and Kara E. Gray. "Art for the sake of improving attitudes toward engineering." In *2012 ASEE Annual Conference & Exposition*, pp. 25-206. 2012.
- [6] Xiao, Guoquan. "A Study on the bridging strategy between the theory of fluid mechanics and engineering applications." In *2019 International Conference on Modelling, Simulation and Big Data Analysis (MSBDA 2019)*, pp. 230-235. Atlantis Press, 2019. <https://doi.org/10.2991/msbda-19.2019.35>
- [7] Batchelor, George Keith. *An introduction to fluid dynamics*. Cambridge university press, 1967.
- [8] Burmeister, Louis C. *Convective heat transfer*. John Wiley & Sons, 1993.
- [9] Welty, James, Gregory L. Rorrer, and David G. Foster. *Fundamentals of momentum, heat, and mass transfer*. John Wiley & Sons, 2020.
- [10] Shampine, Lawrence F., Ian Gladwell, and Skip Thompson. *Solving ODEs with matlab*. Cambridge university press, 2003. <https://doi.org/10.1017/CBO9780511615542>
- [11] Shampine, Lawrence F., Jacek Kierzenka, and Mark W. Reichelt. "Solving boundary value problems for ordinary differential equations in MATLAB with bvp4c." *Tutorial notes* 2000 (2000): 1-27.
- [12] Anuar, Nur Syazana, and Norfifah Bachok. "Double solutions and stability analysis of micropolar hybrid nanofluid with thermal radiation impact on unsteady stagnation point flow." *Mathematics* 9, no. 3 (2021): 276. <https://doi.org/10.3390/math9030276>
- [13] Khashi'ie, Najiyah Safwa, Ezad Hafidz Hafidzuddin, Norihan Md Arifin, and Nadiyah Wahi. "Stagnation point flow of hybrid nanofluid over a permeable vertical stretching/shrinking cylinder with thermal stratification effect." *CFD Letters* 12, no. 2 (2020): 80-94.
- [14] Mohd Sohut, Noor Farizza Haniem, Siti Khuzaimah Soid, Sakhinah Abu Bakar, and Anuar Ishak. "Unsteady Three-Dimensional Flow in a Rotating Hybrid Nanofluid over a Stretching Sheet." *Mathematics* 10, no. 3 (2022): 348. <https://doi.org/10.3390/math10030348>
- [15] Waini, Iskandar, Anuar Ishak, and Ioan Pop. "Hybrid nanofluid flow towards a stagnation point on a stretching/shrinking cylinder." *Scientific Reports* 10, no. 1 (2020): 9296. <https://doi.org/10.1038/s41598-020-66126-2>
- [16] Levy, Steven. "Graphical user interface." *Encyclopaedia Britannica* (2018).
- [17] Tahri, Meryem, Mohamed Maanan, Haytham Tahri, Jan Kašpar, Ratna Christmiari Purwestri, Zohreh Mohammadi, and Róbert Marušák. "New Fuzzy-AHP Matlab based graphical user interface (GUI) for a broad range of users: Sample applications in the environmental field." *Computers & Geosciences* 158 (2022): 104951. <https://doi.org/10.1016/j.cageo.2021.104951>
- [18] Yamazaki, Atsushi, Kazuki Sando, Tetsuya Suzuki, and Akira Aiba. "A handwritten Japanese historical kana reprint support system: Development of a graphical user interface." In *Proceedings of the ACM Symposium on Document Engineering 2018*, pp. 1-4. 2018. <https://doi.org/10.1145/3209280.3229117>
- [19] ÇETİN, Barbaros, Yakuphan ÖZTÜRK, and Serdar TAZE. "Viscous flow calculations for undergraduate fluid mechanics education using MATLAB."
- [20] Khan, Umair, Aurang Zaib, Ioan Pop, Sakhinah Abu Bakar, and Anuar Ishak. "Stagnation point flow of a micropolar fluid filled with hybrid nanoparticles by considering various base fluids and nanoparticle shape factors." *International Journal of Numerical Methods for Heat & Fluid Flow* 32, no. 7 (2022): 2320-2344. <https://doi.org/10.1108/HFF-07-2021-0445>