Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

Conjugate Gradient MATLAB GUI using AHP-Based Usability Model

Nur Idalisa[1,*], Nurul Hafawati Fadhilah[1], Norhaslinda Zullpakkal[1], Mohd Rivaie[1], Wan Rosanisah Wan Mohd[2], Imza Fakhri[1], Ibrahim Mohammed Sulaiman[3,4]

[1] College of Computing, Informatics and Mathematics, Universiti Teknologi MARA (UiTM), Kuala Terengganu Campus, 21080 Kuala Terengganu, Terengganu, Malaysia
[2] Universiti Malaysia Terengganu, 21300 Kuala Terengganu, Terengganu, Malaysia
[3] Department of Mathematics and Statistics, School of Quantitative Sciences, Universiti Utara Malaysia, Sintok, 06010 Bukit Kayu Hitam, Kedah, Malaysia
[4] Faculty of Art and Education, Sohar University, Sohar, Sohar 311, Oman

ABSTRACT

The Conjugate Gradient (CG) method is a proficient numerical technique for solving Unconstrained Optimisation (UO) problems. The greatest challenge of the CG algorithm is the complexity and time-consuming nature of data collection, especially when dealing with complex or extensive problems. Hence, a suitable interface is required to overcome this drawback. One implementation of such an interface is a graphical user interface (GUI), which can offer a user-friendly means of inputting parameters and displaying results. The GUI simplifies the data collection process, enhancing efficiency and speeding up its application in CG method research. This paper utilised MATLAB application designer to construct a GUI using an Analytic Hierarchy Process (AHP)-based evaluation method as a guideline. The integration of AHP helped to optimise the GUI design in terms of usability and effectiveness, thereby ensuring that the final product meets essential criteria. The AHP analysis revealed that accuracy, task completion time, response time, consistency, completeness, and ease of use are the five most important criteria for assessing GUI usability. The new CG-MATLAB GUI has been noted to meet the essential usability criteria.

Keywords:
Graphical user interfaces; evaluation tools; usability; conjugate gradient; AHP

1. Introduction

The CG algorithm, or conjugate gradient, is closely related to the Krylov subspace method. The subspace method has been recognized as one of the ten most influential algorithms that have significantly contributed to the advancement and application of science and engineering during the 20th century [1]. The CG method effectively addresses extensive Unconstrained Optimization (UO) functions. This class of functions diversified tremendously over the decades, with numerous varieties and developments. In previous studies, most researchers who used CG approaches provided successful evidence showing CG's effectiveness in solving UO functions [2,3].

The standard UO function can be written as:

* *Corresponding author.*
E-mail address: nuridalisa@uitm.edu.my

https://doi.org/10.37934/araset.58.1.262273

$$\min_{x \in R^n} f(x). \tag{1}$$

The function defined in Eq. (1) is often minimized using an iterative formula of the form:

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 1, 2, 3, \ldots \tag{2}$$

where α_k and d_k refer to the step-size and search direction, respectively.

This study employed an exact or inexact line search to establish the step-size, whereas the CG method determines the search direction. The CG method is classified into different categories: classical, hybrid, spectral, modified, and three-term methods. An important component of the CG formula is the search direction defined as:

$$d_k = -g_k + \beta_k d_{k-1} \tag{3}$$

where g_k is the gradient of function and β_k is the CG coefficient.

Hestenes and Steifel (HS) developed the first linear CG coefficient in 1952. The HS formula was used to solve linear systems with positive, definite coefficient matrices [6]. Fletcher and Reeves (FR) presented a non-linear version of the HS method for the CG coefficient in 1964 [7]. Meanwhile, in 1969, Polak, Ribiere, and Polyak presented a new CG coefficient with better convergence properties than the HS method, Polak-Ribiere-Polyak (PRP) [8]. Afterwards, other CG methods were also introduced, such as Conjugate Descent (CD) [9], Liu-Storey (LS) [10] and Dai-Yuan (DY) [11]. Table 1 summarizes the formulations utilized by previous researchers, which have served as benchmarks and are widely used among present researchers [4,5].

Table 1
CG coefficients

CG coefficients	Formulation	Ref
HS	$\beta_k^{HS} = \dfrac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T(g_k - g_{k-1})}$	[6]
FR	$\beta_k^{FR} = \dfrac{g_k^T g_k}{g_{k-1}^T(g_{k-1})}$	[7]
PRP	$\beta_k^{PRP} = \dfrac{g_k^T(g_k - g_{k-1})}{g_{k-1}^T(g_{k-1})}$	[8]
CD	$\beta_k^{CD} = -\dfrac{g_k^T g_k}{d_{k-1}^T(g_{k-1})}$	[9]
LS	$\beta_k^{LS} = -\dfrac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T(g_{k-1})}$	[10]
DY	$\beta_k^{DY} = \dfrac{g_k^T g_k}{d_{k-1}^T(g_k - g_{k-1})}$	[11]

Algorithm 1 (The general Conjugate Gradient Algorithm)
Step 1: Initialization: Set $k = 0$. Select the initial point $x_0 \in \mathbb{R}_n$.
Test a criterion for stopping the iterations.
If criterion is satisfied, stop; otherwise continue with Step 2.
Step 2: Compute the step size, α_k.
Step 3: Update the iteration using Eq. (2).
Step 4: Compute β_k using any preferred CG coefficient.
Step 5: Generate the search direction using Eq. (3).
Step 6: Set $k = k + 1$ and return to Step 1.

MATLAB is one of the most widely employed software applications for mathematical analysis, simulation, and image processing. It was utilised in this research to compute the CG method for solving UO problems. In the numerical phase of CG research, data collection is a crucial stage that enables researchers to evaluate the performance and accuracy of the proposed CG method and compare it to existing methods. During the data collection, suitable input data sets are selected or generated. The quality and quantity of the employed data sets might influence the validity and dependability of numerical experiments. Subsequently, the CG method is utilised to assess the obtained outcomes.

Figure 1 displays the coding for the MATLAB script file that we use to obtain the output from the CG algorithm (Algorithm 1). Depending on the number of test functions, initial points, and CG methods involved, modifying and executing the script multiple times is necessary. For example, the complete sequence may consist of 912 [12] and 2304 [13].

```
OP1=                %input: beta CG
OP2=                %input: test function
n =                 %input: size of test function

x0(1:n)= %input: initial point

[fi,ngi, time, NOI]=CG(x0,OP1,OP2);            %CG algorithm

T=table(fi,ngi,time,NOI)     % ouput: efficiency criteria in table
```

Fig. 1. The MATLAB script file for Algorithm 1

The challenging circumstances and the need to prevent recurrent process implementation require reconsidering the existing concept. One potential solution is to convert the programme into a Graphical User Interface (GUI). This conversion would enhance the accessibility of theoretical practices and algorithms, improve the data collection process, and minimise the risk of human errors.

GUI is a user-friendly interface that enables users to interact with digital devices or software applications. It employs graphical elements such as icons, menus, and buttons instead of inputting commands. The interface was designed to exhibit a significant level of user-friendliness, allowing users to operate it with minimal programming knowledge [14]. This utilization may contribute to increased productivity and accessibility, as well as enhanced user experiences. Designing the GUI is a challenging and complex task. It requires thorough analysis, iterative design, and usability testing [15].

According to the International Organization (ISO) for usability's definition, usability refers to "the degree to which specified users can utilize a product to achieve specific goals with effectiveness, efficiency, and satisfaction in a specified context of use" [16]. This definition outlines the three critical factors to consider when evaluating usability: users, goals, and use context. By optimizing the three factors and prioritizing usability in the product design process, developers can create simpler, more efficient, and more satisfying products for users. Due to this, adoption and higher levels of loyalty towards the products could be achieved. Therefore, it is possible to achieve increased adoption rates and heightened brand loyalty among users.

The literature presents a range of usability models. The initial usability models were introduced from 1991 to 1999 [17]. Nevertheless, ISO standards and Nielsen models [18] are the most prevalent and frequently cited usability models in the literature. Nielsen's 1992 model comprises five fundamental usability attributes: efficiency, satisfaction, learnability, memorability, and errors. These attributes are still extensively employed and considered essential for evaluating the usability of digital products.

Ajibola *et al.,* [19] proposed the Mobile Shopper Application Development (MOSAD) model for evaluating the usability of mobile commerce (M-commerce) applications. This model comprises the following five attributes: efficiency, effectiveness, satisfaction, error rate, and learnability. These attributes were explicitly designed to meet the requirements of M-commerce applications, which present unique usability challenges compared to other digital products. The evaluated product's requirements and characteristics commonly determine the usability model selection. However, all usability models have the same objective: to determine how effectively and satisfactorily a product's intended consumers can use it to accomplish their objectives.

Saaty *et al.,* [20] proposed the Analytical Hierarchy Process (AHP) to aid researchers with diverse expert backgrounds in identifying priorities for their multi-criteria problems and ensuring consistency ratios. This decision-making method entails decomposing a complex decision problem into a hierarchical structure of criteria and sub-criteria to address multiple criteria. As proposed by Roy *et al.,* [21], usability evaluation criteria frequently involve attractiveness, controllability, efficiency, usefulness, and learnability. The sub-criteria correspond to specific features or aspects of the product that relate to each attribute. This study employs the AHP method to designate priority rankings for evaluating the usability score of a GUI designed for applying the CG method to solving UO test functions.

2. Analytical Hierarchy Process

The Analytical Hierarchy Process (AHP) is an analytical method that provides a hierarchical structure of selected data that can be executed by a diverse range of individuals, including general users, clients, and specialists. Input data comprises subjective and objective variables, including evaluation standards and environmental characteristics. This decision-making method is extensively used in usability evaluation and has been applied in various studies to determine the usability rating of websites [22-25] and other digital products [26]. It often comprises three hierarchical levels: main goal, criteria, and alternative. However, achieving this within the first two levels may be feasible given certain conditions. This model's primary objective is to rank the variables in order of importance based on criteria and sub-criteria.

In this research, a set of five criteria and 12 sub-criteria are used to assess the GUI's usability. Figure 2 illustrates the proposed model for usability testing. The design was focused on five key GUI features that improve usability. The three-level hierarchical model starts with the goal of CG GUI usability. The second level of the model consists of five distinct attributes: attractiveness (C1), controllability (C2), efficiency (C3), learnability (C4), and efficacy (C5). Meanwhile, the last level depicts the sub-criteria for each criterion. Two sub-criteria were established for the attractiveness criterion: pleasant (C11) and organised (C12). Link (C21) and consistency (C22) are two sub-criteria influencing the controllability evaluation. The number of clicks (C31), time to complete a task (C32), and response time (C33) were utilised to evaluate the effectiveness. The sub-attributes that affect the learnability of a GUI are learning time (C41), explanation (C42), and ease of use (C43). Meanwhile, accuracy (C51) and completeness (C52) are the sub-criteria affecting the effectiveness of the GUI.

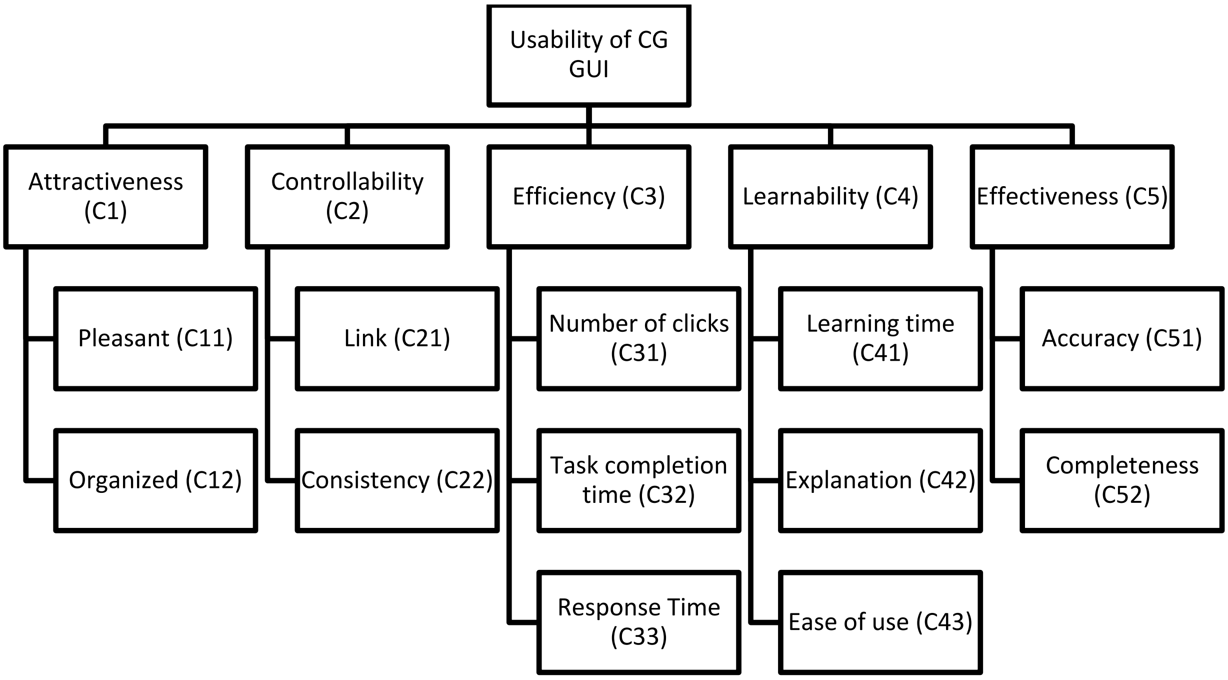

Fig. 2. The hierarchical for usability model

The descriptions of criteria and sub-criteria taken from the previous studies [19,26-28] are described as follows:

i. Attractiveness (C1): Attractiveness is a usability attribute essential to measuring the visual aspect of the GUI. It helps in measuring the direct interest of users towards the GUI, both in functional and non-functional aspects.
   - Pleasant (C11): Users are interested in helpful and valuable content.
   - Organised (C12): A sense of orderliness, structure and efficiency where things are arranged logically and efficiently.

ii. Controllability (C2): Controllability is an essential usability attribute that measures the GUI's navigational prospects. The score indicates how easily the user can navigate the GUI and complete the required task.
   - Link (C21): Text descriptions of links allow quick decisions when selecting the correct link.
   - Consistency (C22): Refers to the ability to produce consistent results.

iii. Efficiency (C3): Efficiency measures the ease of work. It signifies how easily users can complete the task with the limited resources available and effectively in terms of time and money.

- The number of clicks (C31): Defined as the users' navigational prospects (e.g., the frequency of mouse clicks made to complete their specific duties).
- Task completion time (C32): Defined as the time a specified user takes to complete any particular task.
- Response time (C33): Refers to the duration of time required for a user or system to respond.

iv. Learnability (C4): Defined as the effortless capacity of users to comprehend and master the GUI's functions.
- Learning time (C41): Refers to the time a user spends learning.
- Explanation (C42): Describe the contents.
- Ease of use (C43): An attribute that indicates the degree to which a new user may quickly and easily master the system's interface.

v. Effectiveness (C5): Effectiveness is the accuracy and completeness with which users achieve specified goals.
- Accuracy (C51): Information should be accurate and trustworthy.
- Completeness (C52): Defined as the capacity of the system to provide the functionalities necessary to implement the tasks intended by the user.

The AHP methodology mainly depends on evaluating multiple criteria through pairwise comparisons. This method allows for the independent evaluation of each criterion. In the AHP pairwise comparison, the criteria are ranked in order of significance using a scale from 1 to 9. The value of 1 denotes that the criteria possess "equivalent importance", and 9, the highest possible value, means "extremely strong importance" [20]. Each AHP score is described in depth in Table 2.

Table 2
Description of AHP scale

Rating	Description
1	Equivalent importance
2	Weak importance
3	Moderate importance
4	Moderate to strong importance
5	Strong importance
6	Strong to very strong importance
7	Very strong importance
8	Very strong to extremely strong importance
9	Extremely strong importance

The consistency of the pairwise comparisons is measured using the consistency ratio (CR), which compares the degree of inconsistency in the ratings to what would be expected by chance. The judgements are considered unreliable if the CR exceeds a certain threshold (usually 0.10 [29]). The users entered the input data, which, thus, should be reviewed and revised [30]. The consistency index (CI) and random matrix (RI) [20] are used in calculating AHP analysis, stated as follows,

$$CR = \frac{CI}{RI},$$

$$CI = \frac{\lambda_{\max} - n}{n-1},$$

$$CR = \frac{RI(\lambda_{\max} - n)}{n-1},$$

where $\lambda_{\max}$ is a principal eigenvalue. More details on AHP methodology can be referred to the previous studies [31,32].

3. Result and Discussion

3.1 Usability Criteria

The GUI was developed with the primary focus on enhancing its usability. The AHP ranking method was employed to assess critical criteria. A pairwise comparison is performed based on the five criteria, with a principal eigenvalue of 5.201 and a CR of 0.045. The analysis of each criterion's weight is summarized in Table 3.

Table 3
Weights for criteria of usability criteria

Level 1	Goal: Usability of CG GUI											
Level 2	C1		C2		C3			C4			C5	
	0.035		0.062		0.363			0.055			0.485	
	CR=0.045											
Level 3	C11	C12	C21	C22	C31	C32	C33	C41	C42	C43	C51	C52
	0.1	0.9	0.1	0.9	0.063	0.743	0.194	0.069	0.25	0.681	0.9	0.1
	CR=0		CR=0		CR=0.074			CR=0.01			CR=0	

The analysis for the criteria's weight yields the following sequence: C5>C3>C2>C4>C1. This sequence suggests that the graphical user interface (GUI) usability evaluation should prioritize effectiveness as the most crucial criterion, followed by efficiency, controllability, learnability, and attractiveness. After that, the calculation to determine the priority order of the sub-criteria is performed. In the calculation, the weight of each sub-criteria is multiplied by the weight of its respective criterion. Table 4 displays the finalized ranking for the sub-criteria.

Table 4
Ranking for sub-criteria

Priority	Sub-criteria	Ranking
0.4365	accuracy	1
0.269709	task completion time	2
0.070422	response time	3
0.0558	consistency	4
0.0485	completeness	5
0.037455	ease of use	6
0.0315	organized	7
0.022869	number of clicks	8
0.01375	explanation	9
0.0062	link	10
0.003795	learning time	11

Table 4 presents the results, demonstrating the descending order of values: C51, C32, C33, C22, C52, C43, C12, C31, C42, C21, C41, and C11. GUI usability is often assessed using six key criteria: accuracy, task completion time, response time, consistency, completeness, and ease of use.

3.2 User Interface

This research aims to develop a GUI based on MATLAB App Designer for solving UO test problems using the CG method with a strong Wolfe line search. The GUI was designed to satisfy all criteria for usability quality, as outlined in Tables 3 and Table 4. As depicted in Figure 3, this study devised a MATLAB-based GUI to solve UO test problems using the CG method. MATLAB is a powerful mathematical programming language with extensive applications in various fields, including engineering, physics, and finance, for numerical computation, algorithm development, data analysis, and visualisation [33]. In addition to its core functionality, the MATLAB software offers a GUI that allows users to interact with their code in a more user-friendly manner [34]. The GUI collects user input and modifies the state of its widgets, which may include icons, text fields, sliders, and other interactive elements [35]. The software also encourages users to experiment with diverse parameters and algorithms, visualise their data, and debug their code by providing user-friendly interfaces. This feature can be highly beneficial, especially for individuals who lack programming expertise or require accelerated prototype development or exploration of new ideas.

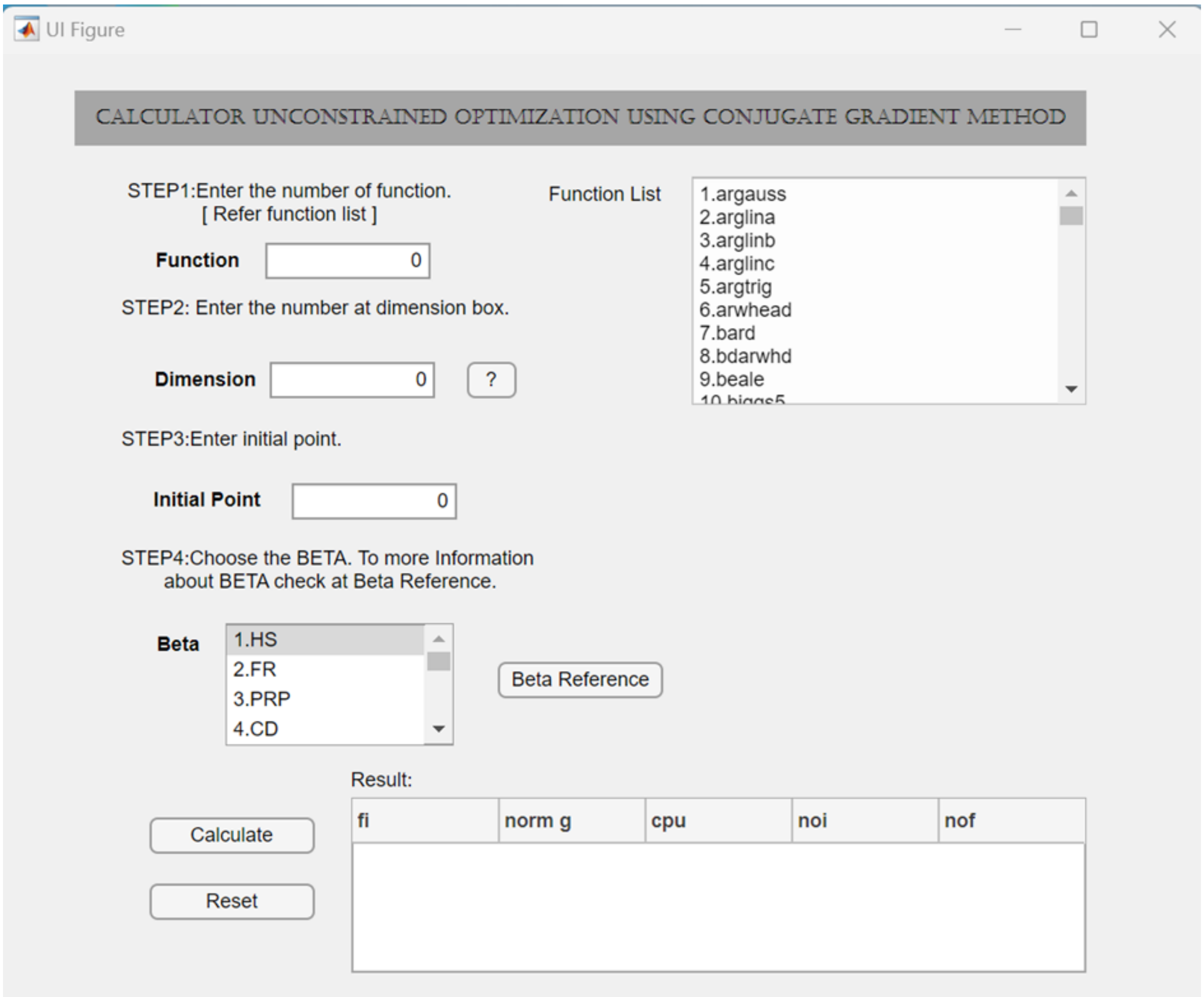

Fig. 3. MATLAB-based GUI interface

The interface was designed to test and benchmark CG algorithms using the Constrained and Unconstrained Testing Environment (CUTEst) set. The initial step in the data entry instructions for this interface is to prompt users to choose a test function from the interface's list of 134 CUTEst functions [36]. These test functions, which vary in complexity and characteristics, can assess the robustness and efficacy of various optimization methods. Optimization algorithm research and benchmarking frequently employ these. Rosenbrock, Powell, and Hock & Schittkowski are examples of the said functions.

Following the user's selection of test functions, the next step is to input the test function's dimension. The dimension parameter is crucial as it determines the number of variables that can be optimized through the CG algorithm. The third step entails inputting the initial point of the CG algorithm. The initial point is commonly used as the starting point for algorithm optimization and is crucial in determining the algorithm's convergence behaviour and final solution.

Lastly, the CG coefficient should be selected by the user. Users will access a comprehensive catalogue of CG coefficients upon selecting the Beta Reference button on the right-hand side of the Beta input. The Uniform Resource Locator (URL) specified is https://acrobat.adobe.com/link/review?uri=urn:aaid:scds:US:edf35b47-4683-368c-980a-6a762220e54b, which comprises a total of 42 coefficients for the CG model. Once these four pieces of information (test function, dimension, initial point, and CG coefficient) have been input by the user, the interface can execute the CG algorithm on the selected test functions. Users will be presented with results once the procedure has been completed.

By comparing the outcomes of earlier studies utilising the script file shown in Figure 1 with those acquired from the suggested MATLAB GUI, we evaluated the functionality of the GUI programme. As an example of simulation, Figure 4 demonstrates that the selected test function is Freuroth with four dimensions, the initial point x0 = (4, 4, 4, 4), and the CG algorithm is the HS. Based on the exhibited output, the HS CG algorithm determined function values (fi = 9.2229e-21) and norms (8.0253e-09). The computation time (CPU = 0.0093), number of iterations (noi = 5), and number of function evaluations (nof = 20) are displayed. This outcome indicates that the HS solved the Freuroth function with four dimensions and x0 = (4, 4, 4, 4) as the initial point in 0.0093 CPU time, five iterations, and 20 total function evaluations.

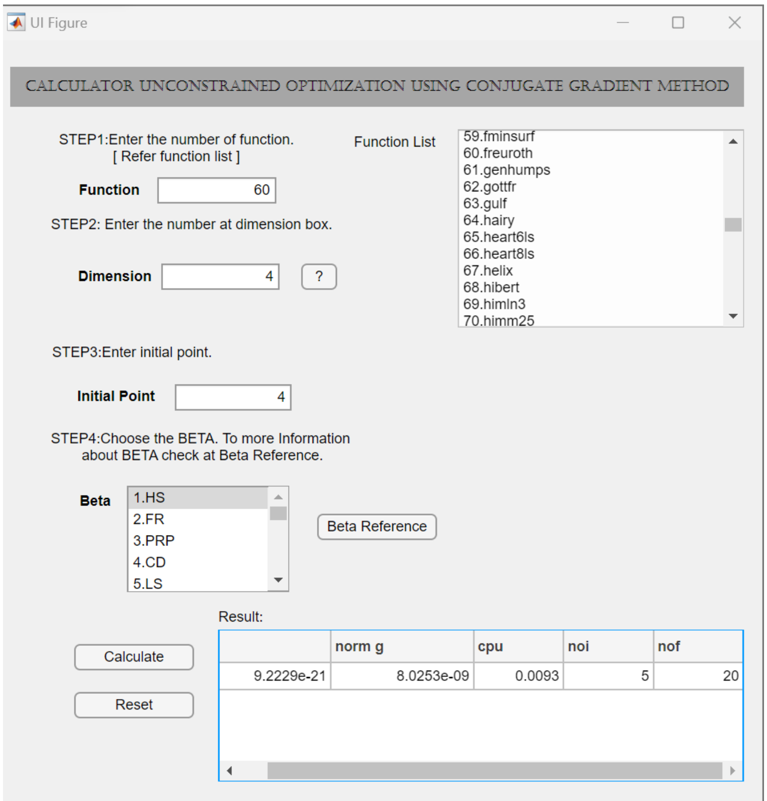

Fig. 4. Sample of results from GUI

The newly developed GUI has undergone testing using diverse inputs and outputs, and it has been confirmed that the outcomes generated are accurate. Figure 4 and Figure 5 demonstrate the same consistency between the number of iterations and the evaluated functions. Consequently, our GUI has satisfied the most important criterion: accuracy. The developed GUI is also designed to be user-friendly and intuitive, facilitating ease of navigation and enhancing its reliability and convenience as a tool for our clients.

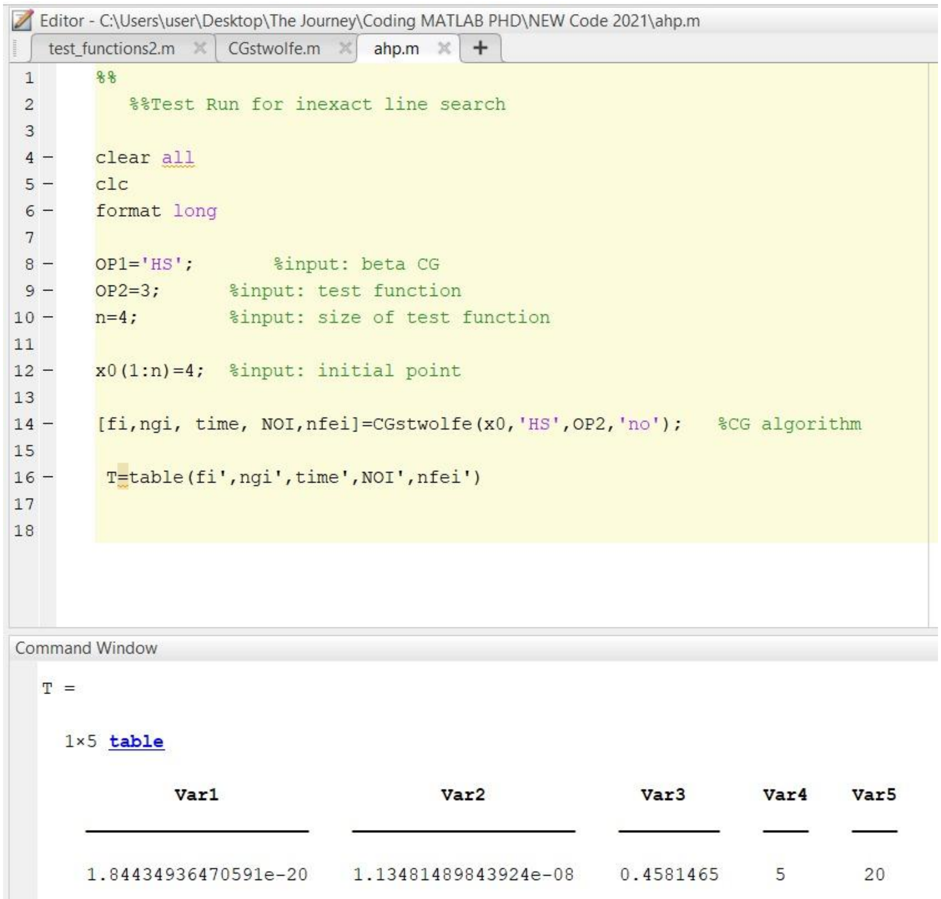

Fig. 5. Results from MATLAB script file

4. Conclusions

This study aims to design a new GUI for CG algorithms that meet essential usability criteria. In the numerical phase of CG research, it is essential to acquire data to assess the performance and accuracy of the proposed CG method. Using a GUI can expedite the data collection process and reduce human error. Furthermore, since the GUI could simplify the analysis process compared to the conventional one, it can be implemented in CG research more rapidly and with less effort.

The AHP was utilized in this research to prioritize the usability criteria that impact the GUI score. The AHP method can assist us in ensuring that the graphical user interface is effective, efficient, controllable, easy to learn, and attractive, thereby improving the user experience and satisfaction. The AHP method revealed that the five most essential criteria for evaluating GUI usability are accuracy, task completion time, response time, consistency, completeness, and ease of use. The new CG-MATLAB GUI has been observed to meet the most critical usability criteria.

The Technology Acceptance Model (TAM), which describes how people accept and use new technologies, may be considered for future research. The model was developed by Davis, F.D. [37]

and has been utilized extensively in information systems research [38,39]. As a result, it should be used in future studies to test and refine our GUI.

Acknowledgement
This research was funded by a grant from Ministry of Higher Education of Malaysia (FRGS/1/2019/STG06/UITM/02/3).

References

[1] Dongarra, Jack, and Francis Sullivan. "Guest editors introduction to the top 10 algorithms." *Computing in Science & Engineering* 2, no. 01 (2000): 22-23. https://doi.org/10.1109/MCISE.2000.814652
[2] Idalisa, Nur, Mohd Rivaie, Nur Hidayah Mohd Noh, Mohd Agos Salim Nasir, Nurul Hafawati Fadhilah, and Norma Alias. "A new three-term conjugate gradient method with application to regression analysis." *International Journal of Electrical & Computer Engineering (2088-8708)* 12, no. 5 (2022). https://doi.org/10.11591/ijece.v12i5.pp5248-5259
[3] Norddin, Nur Idalisa, Mohd Rivaie Mohd Ali, Nurul Hafawati Fadhilah, Nur Atikah, Anis Shahida, and Nur Hidayah Nohd Noh. "Multiple Linear Regression Model of Rice Production using Conjugate Gradient Methods." *Matematika* (2019): 229-236. https://doi.org/10.11113/matematika.v35.n2.1180
[4] Zullpakkal, Norhaslinda, N. 'Aini, N. H. A. Ghani, N. S. Mohamed, N. Idalisa, and M. Rivaie. "Covid-19 data modelling using hybrid conjugate gradient method." *Journal of Information and Optimization Sciences* 43, no. 4 (2022): 837-853. https://doi.org/10.1080/02522667.2022.2060610
[5] Sulaiman, I. M., M. Mamat, M. Y. Waziri, U. A. Yakubu, and M. Malik. "The convergence properties of a new hybrid conjugate gradient parameter for unconstrained optimization models." In *Journal of physics: Conference series*, vol. 1734, no. 1, p. 012012. IOP Publishing, 2021. https://doi.org/10.1088/1742-6596/1734/1/012012
[6] Hestenes, Magnus Rudolph, and Eduard Stiefel. *Methods of conjugate gradients for solving linear systems*. Vol. 49, no. 1. Washington, DC: NBS, 1952. https://doi.org/10.6028/jres.049.044
[7] Fletcher, Reeves, and Colin M. Reeves. "Function minimization by conjugate gradients." *The computer journal* 7, no. 2 (1964): 149-154. https://doi.org/10.1093/comjnl/7.2.149
[8] Polak, Elijah, and Gerard Ribiere. "Note sur la convergence de méthodes de directions conjuguées." *Revue française d'informatique et de recherche opérationnelle. Série rouge* 3, no. 16 (1969): 35-43. https://doi.org/10.1051/m2an/196903R100351
[9] Fletcher, Roger. *Practical methods of optimization*. John Wiley & Sons, 2000. https://doi.org/10.1002/9781118723203
[10] Liu, Y., and C. Storey. "Efficient generalized conjugate gradient algorithms, part 1: theory." *Journal of optimization theory and applications* 69 (1991): 129-137. https://doi.org/10.1007/BF00940464
[11] Dai, Yu-Hong, and Yaxiang Yuan. "A nonlinear conjugate gradient method with a strong global convergence property." *SIAM Journal on optimization* 10, no. 1 (1999): 177-182. https://doi.org/10.1137/S1052623497318992
[12] Idalisa, Nur, Mohd Rivaie, Nurul Hafawati Fadhilah, and Mohd Agos Salim Nasir. "Brief review on the performance of RMIL conjugate gradient methods." In *AIP Conference Proceedings*, vol. 2465, no. 1. AIP Publishing, 2022. https://doi.org/10.1063/5.0078462
[13] Zull, Norhaslinda, N. Aini, M. Rivaie, and M. Mamat. "A new gradient method for solving linear regression model." *International Journal of Recent Technology and Engineering* 7, no. 5 (2019): 624-630.
[14] Yadav, Rajesh, and Hifjur Raheman. "Development of an artificial neural network model with graphical user interface for predicting contact area of bias-ply tractor tyres on firm surface." *Journal of Terramechanics* 107 (2023): 1-11. https://doi.org/10.1016/j.jterra.2023.01.004
[15] Shneiderman, Ben, and Catherine Plaisant. *Designing the user interface: strategies for effective human-computer interaction*. Pearson Education India, 2010.
[16] Jančíková, Zora, Pavel Koštial, Daniela Bakošová, Ivan Ružiak, Karel Frydrýšek, Jan Valíček, Martina Farkašová, and Richard Puchký. "The study of electrical transport in rubber blends filled by single wall carbon nanotubes." *Journal of Nano Research* 21 (2013): 1-6. https://doi.org/10.4028/www.scientific.net/JNanoR.21.1
[17] Alghamdi, AHMED S., A. L. I. H. Al-Badi, R. O. O. B. A. E. A. Alroobaea, and P. A. M. J. Mayhew. "A comparative study of synchronous and asynchronous remote usability testing methods." *International Review of Basic and Applied Sciences* 1, no. 3 (2013): 61-97.
[18] Nielsen, Jakob. "Finding usability problems through heuristic evaluation." In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 373-380. 1992. https://doi.org/10.1145/142750.142834

[19] Ajibola, A. S., E. T. Abiodun, and L. Goosen. "Development of A New Model for the Usability Evaluation of M-commerce Applications." *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* 14, no. 3 (2022): 1-9. https://doi.org/10.54554/jtec.2022.14.03.001
[20] Saaty, Thomas L. "A scaling method for priorities in hierarchical structures." *Journal of mathematical psychology* 15, no. 3 (1977): 234-281. https://doi.org/10.1016/0022-2496(77)90033-5
[21] Roy, Sharmistha, Prasant Kumar Pattnaik, and Rajib Mall. "Quality assurance of academic websites using usability testing: an experimental study with AHP." *International Journal of System Assurance Engineering and Management* 8 (2017): 1-11. https://doi.org/10.1007/s13198-016-0436-0
[22] Pamučar, Dragan, Željko Stević, and Edmundas Kazimieras Zavadskas. "Integration of interval rough AHP and interval rough MABAC methods for evaluating university web pages." *Applied soft computing* 67 (2018): 141-163. https://doi.org/10.1016/j.asoc.2018.02.057
[23] Muhammad, AbdulHafeez, Ansar Siddique, Quadri Noorulhasan Naveed, Uzma Khaliq, Ali M. Aseere, Mohd Abul Hasan, Mohamed Rafik N. Qureshi, and Basit Shahzad. "Evaluating usability of academic websites through a fuzzy analytical hierarchical process." *Sustainability* 13, no. 4 (2021): 2040. https://doi.org/10.3390/su13042040
[24] Ramanayaka, Kokila Harshan, Xianqiao Chen, and Bing Shi. "Application of extent analysis FAHP to determine the relative weights of evaluation indices for library website usability acceptance model." *Iet Software* 13, no. 1 (2019): 86-95. https://doi.org/10.1049/iet-sen.2018.5185
[25] Gong, Jia-Wei, Hu-Chen Liu, Xiao-Yue You, and Linsen Yin. "An integrated multi-criteria decision making approach with linguistic hesitant fuzzy sets for E-learning website evaluation and selection." *Applied Soft Computing* 102 (2021): 107118. https://doi.org/10.1016/j.asoc.2021.107118
[26] Du, Yinfeng, Dun Liu, Juan Antonio Morente-Molinera, and Enrique Herrera-Viedma. "A data-driven method for user satisfaction evaluation of smart and connected products." *Expert Systems with Applications* 210 (2022): 118392. https://doi.org/10.1016/j.eswa.2022.118392
[27] Byun, Dae-Ho, and Gavin Finnie. "An AHP method for evaluating usability of electronic government portals." *Electronic government, an international journal* 8, no. 4 (2011): 343-362. https://doi.org/10.1504/EG.2011.042811
[28] Alonso-Ríos, David, Ana Vázquez-García, Eduardo Mosqueira-Rey, and Vicente Moret-Bonillo. "Usability: a critical analysis and a taxonomy." *International journal of human-computer interaction* 26, no. 1 (2009): 53-74. https://doi.org/10.1080/10447310903025552
[29] Saaty, Roseanna W. "The analytic hierarchy process—what it is and how it is used." *Mathematical modelling* 9, no. 3-5 (1987): 161-176. https://doi.org/10.1016/0270-0255(87)90473-8
[30] Tahri, Meryem, Mohamed Maanan, Haytham Tahri, Jan Kašpar, Ratna Chrismiari 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
[31] Saaty, Thomas L. "Fundamentals of the analytic hierarchy process." *The analytic hierarchy process in natural resource and environmental decision making* (2001): 15-35. https://doi.org/10.1007/978-94-015-9799-9_2
[32] Saaty, Thomas L. "How to make a decision: the analytic hierarchy process." *Interfaces* 24, no. 6 (1994): 19-43. https://doi.org/10.1287/inte.24.6.19
[33] He, Song, and Peiyue Li. "A MATLAB based graphical user interface (GUI) for quickly producing widely used hydrogeochemical diagrams." *Geochemistry* 80, no. 4 (2020): 125550. https://doi.org/10.1016/j.chemer.2019.125550
[34] Nasir, Sharifah Noha Zahirah Syed Abdul, Nurul Ain Ab Wahab, and Mohd Agos Salim Nasir. "Graphical User Interface for Solving Non-Linear Equations for Undergraduate Students." *International Journal of Advanced Research in Future Ready Learning and Education* 30, no. 1 (2023): 25-34.
[35] Banerjee, Ishan, Bao Nguyen, Vahid Garousi, and Atif Memon. "Graphical user interface (GUI) testing: Systematic mapping and repository." *Information and Software Technology* 55, no. 10 (2013): 1679-1694. https://doi.org/10.1016/j.infsof.2013.03.004
[36] Gratton, Serge, and Philippe L. Toint. "OPM, a collection of optimization problems in Matlab." *arXiv preprint arXiv:2112.05636* (2021).
[37] Davis, Fred D. "Technology acceptance model: TAM." *Al-Suqri, MN, Al-Aufi, AS: Information Seeking Behavior and Technology Adoption* 205 (1989): 219.
[38] Marangunić, Nikola, and Andrina Granić. "Technology acceptance model: a literature review from 1986 to 2013." *Universal access in the information society* 14 (2015): 81-95. https://doi.org/10.1007/s10209-014-0348-1
[39] Davis, Fred D., Andrina Granić, and Nikola Marangunić. "The technology acceptance model 30 years of TAM." *Technology* 1, no. 1 (2023): 1-150. https://doi.org/10.1007/978-3-030-45274-2_1