Jurnal Teknologi

Full Paper

NUMERICAL APPROXIMATION OF SURFACE AREAS OF GEOMETRIES THROUGH INTEGRATION USING INTERACTIVE MATLAB GUI

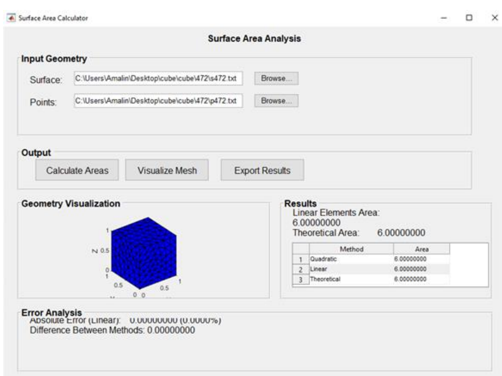
Raja Nur Amalin Puteri Raja Hussin, Nurfarrisha Asnida Khairul Akmar, Suzarina Ahmed Sukri*, Yeak Su Hoe, Taufiq Khairi Ahmad Khairuddin

Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia 81310 UTM Johor Bahru, Johor, Malaysia

Article history
Received
29 July 2025
Received in revised form
14 November 2025
Accepted
19 November 2025
Published Online
29 August 2026

*Corresponding author
suzarina@utm.my

Graphical abstract



Abstract

In computational geometry, numerical integration is a widely used approach to estimate the surface area of geometric shapes by subdividing complex curves and surfaces into manageable elements known as meshes. Estimating surface area is a fundamental problem in geometry with important implications in scientific and engineering applications. This paper presents a numerical framework for surface area approximation using discrete linear and quadratic elements applied to several geometric models. Meshes for each geometry are generated using the NETGEN mesh generator, and surface areas are computed through numerical integration. The results are compared with analytical solutions to evaluate accuracy and convergence behavior. To enhance user interaction and improve accessibility, an interactive graphical user interface (GUI) is developed using MATLAB. The GUI allows users to input geometry parameters, visualize the mesh, and perform surface area calculations automatically. Findings show that higher-order (quadratic) elements provide better accuracy than linear elements, especially for curved surfaces. This study contributes to the development of efficient and user-friendly computational tools for surface area approximation, particularly in educational and engineering contexts.

Keywords: Numerical Integration, surface area, Netgen mesh generator, Graphical User Interface

Abstrak

Dalam geometri komputasi, integrasi numerik adalah pendekatan yang digunakan secara meluas untuk menganggarkan luas permukaan bentuk geometri dengan membahagikan lengkung dan permukaan yang kompleks kepada elemen-elemen yang boleh dikendalikan yang dikenali sebagai mesh. Penganggaran luas permukaan adalah masalah asas dalam geometri dengan implikasi penting dalam aplikasi saintifik dan kejuruteraan. Kertas kerja ini mempersembahkan rangka kerja numerik untuk penganggaran luas permukaan menggunakan elemen linear dan kuadratik terpisah yang diterapkan pada beberapa model geometri. Mesh untuk setiap geometri dijana menggunakan penjana mesh NETGEN, dan luas permukaan dikira melalui integrasi numerik. Keputusan dibandingkan dengan penyelesaian analitik untuk menilai ketepatan dan tingkah laku penumpuan. Untuk meningkatkan interaksi pengguna dan meningkatkan aksesibiliti, antara muka pengguna grafik interaktif (GUI) dibangunkan menggunakan MATLAB. GUI membolehkan pengguna memasukkan parameter geometri, memvisualisasikan mesh, dan melakukan pengiraan luas permukaan secara automatik. Penemuan menunjukkan bahawa elemen bertaraf tinggi (kuadratik) memberikan ketepatan yang lebih baik berbanding elemen linear, terutamanya untuk permukaan yang melengkung. Kajian ini menyumbang kepada pembangunan alat pengkomputeran yang cekap dan mesra pengguna untuk penganggaran luas permukaan, terutamanya dalam konteks pendidikan dan kejuruteraan.

Kata kunci: Integrasi numerik, Luas permukaan, Penjana jala Netgen, Antara Muka Pengguna Grafik

1.0 INTRODUCTION

In the modern era, numerical methods have become fundamental tools in addressing complex geometrical problems that are difficult or impossible to solve analytically. Traditional analytical approaches are often limited when applied to irregular, curved, or composite geometries, particularly in real-world engineering and scientific contexts. As a result, numerical integration techniques have emerged as powerful alternatives for approximating solutions to such problems, especially in the computation of surface areas and other integral-based quantities [8], [14].

Numerical integration, or numerical quadrature, is an approximated definite integral by dividing the domain into intervals and estimating sums in terms of formulas. Used in approximating the surface area, the process is done by dividing complicated surfaces into simple elements or meshes and estimating the integral on the subdivisions [11]. These mesh-based discretization methods are commonly applied in computational geometry and form a critical step in the finite element analysis process. Numerical integration methods applied in algorithms not only decrease the computation time but can also provide high accuracy rates, thus being applicable in a variety of applications within biostatistics, fluid dynamics, materials science, structural engineering, and environmental modelling [21]. In [15] research study, the authors provide a numerical integration approach that employs quadratic element integration for the computation of an engineering issue called polarization tensor.

In a study [15], them introduced a numerical integration method which is using quadratic element integration to calculated an engineering problems called polarization tensor. The results inferred that when there was a high singular integrand, lower order methods are not practical compared to higher order method especially when there are irregular boundary domains. Nevertheless, extincting the research, the researcher continue their works by using lower order numerical integration Gauss quadrature with one point and three points, with the same type problems, demonstrates that relevant error is reduced and the convergence is improved [16].

Discretization, which is the art of breaking the continuous domain into finite subdomains (or elements of lower order), is at the heart of those numerical solutions. Discretization developed into more applied fields such as computational geometry and numerical methods. One of the most recognizable applications of discretization is known as the Finite Element Method (FEM). This will grant fluidity with meshing, as well as solutions to approximating partial differential equations (PDEs) in arbitrary domains and with arbitrary geometries. FEM is routinely applied with pre-and post-conditions, to both linear and nonlinear simulations. It garners attention due to its efficiency, precision, and performance [6][1].

Higher-order finite element methods (FEMs) have been introduced as a way to increase the accuracy of solutions with respect to a given shape by increasing the polynomial degree of the shape functions in finite element [13]. Improved accuracy and convergence with augmented FEM methods (with enriched interpolation), such as spectral elements, benefit particular problems which include transient dynamics, wave propagation, and multi-physics coupling [3][10]. The Summation-By-Parts or Simultaneous Approximation Term (SBP-SAT) stabilization techniques represent the very latest methods of addressing numerical instabilities and costs due to boundary errors, enabling more reliable simulations in stiff and nonlinear systems [1].

The variety of numerical methods that extend beyond standard FEM have provided interesting options in terms of geometric modelling and surface approximation. [4] introduced the phase-field method, which is appropriate for a boundary-less, dynamic geometric evolution, such as an application to fluid-structure interaction or morphing material models. Meanwhile, [17] used a multivariate polynomial interpolation based on a least squares minimize method to refine shape reconstruction and surface property estimation. Additionally, [2] provided the extended virtual element method (VEM) as a means to model linear elastic fractures while allowing for exact simulations in non-polynomial and discontinuous domains. Similarly, edge quality is also an important consideration for numerical surface approximations. Mesh generation tools, such as NETGEN [12], are used largely due to their ease of use, as well as their capability to automate the generation of high-quality meshes for complicated 3D geometries.Improvements in mesh generation have also facilitated a smooth transition for researchers into changing from surface meshes to volumetric meshes.

This was when [20] provided an example and comparison of different algorithms to convert surface meshes to volumetric meshes, a common step following geometric modelling and simulation.

Developments of user-friendly interfaces and toolkits following [7] have been able, by embedding mesh-based surface and volume processing inside MATLAB workspaces, to facilitate progression from algorithm development and testing through to the end-user experience, have substantially contributed to applied research as shown in [18] in marine engineering, also refer to [10] regarding composite materials and [19] regarding pandemic modeling, just to name a couple of examples. The work highlights the importance of reliable geometric representation and area determination for accurate simulation computations, which applied researchers have considered according to the level of sub surface mesh resolution.

The proliferation of these tools and methods makes it timely to investigate the surface area estimates obtained using numerical integration through linear and quadratic elements. More specifically, this study will consider FEM theory and integrate it together with modern mesh development software (NETGEN) with user-friendly MATLAB-based GUI interfaces for accessibility on a sustained basis and without losing the theory, math or skills that will ultimately underpin accurate numerical geometry. The results from this study will continue the work and effort undertaken here and more widely for the continued development of numerical geometry practice in order to improve the accessibility and accuracy of and with distance, computational geometry, particularly but not limited to educational and engineering contexts where surface area is an important input for design, especially considering validation and optimization.

2.0 METHODOLOGY

2.1 Linear and Quadratic Element Integration

Linear element integration is one of the fundamental techniques used in numerical surface area approximation. This method involves discretizing a given surface into triangular elements, where each triangle is defined by three nodes connected by straight edges. The simplicity of linear elements makes them computationally efficient and relatively easy to implement, especially for geometries with flat or gently curved surfaces. In this approach, the surface integral is approximated by summing the contributions from each triangle, with the area calculated based on the vertices' coordinates as in equation (1). Let the area of a surface be denoted as in equation (1).

$$\begin{aligned} I_1 &= \iint 1 dS, \\ &= \sum_{i=1}^{N} A_N, \end{aligned} \tag{1}$$

where S_N is the area of a triangles of a geometry for $i = 1, \ldots, N$ subdivision elements. Instead of finding the area of each triangles of a surfaces, other techniques that can be applied to find the area of the surface is to use a Trapezoidal Rule where the approximation of the integral for a surface where the projection of each region can be denoted as

$$I_1 = \iint_{R_{xy}} f(x,y)\,dxdy + \iint_{R_{yz}} f(y,z)\,dydz + \iint_{R_{xz}} f(x,z)\,dxdz, \tag{2}$$

for which the functions of $f(x,y), f(y,x)$ and $f(x,z)$ are the continuous functions on the global coordinates of the plane xy, yz and xz. In order to evaluate each of the triangles, all these functions need to be changed to a local coordinate containing ξ and η. Suppose by considering the first region R_{xy}, the global coordinates need to be changed to a local coordinates.

$$\iint_{R_{xy}} f(x,y)\,dxdy = \iint_{R_{\xi\eta}} f(\xi,\eta) \cdot S_p(\xi,\eta) \cdot \det\left(J(\xi,\eta)\right) d\xi d\eta. \tag{3}$$

From (3), $f(\xi,\eta)$ is the function in terms of local coordinates, $S_p(\xi,\eta)$ is the surface projection of region while $\det(J(\xi,\eta))$ is the determinant of the Jacobian matrix of ξ and η. Hence, for a linear element points (3 points), x and y can be represented as

$$\begin{aligned} x &= \sum_{k=1}^{3} x_k N_k, \\ y &= \sum_{k=1}^{3} y_k N_k, \\ z &= \sum_{k=1}^{3} z_k N_k, \end{aligned} \tag{4}$$

where N_k are the shape functions which are defined as

$$\begin{aligned} N_1 &= \xi, \\ N_2 &= \eta, \\ N_3 &= 1 - \xi - \eta. \end{aligned} \tag{5}$$

In order to find the Jacobian matrix, $J(\xi,\eta)$ where it is a $m \times n$ matrix which can be explicitly represented as

$$J(\xi,\eta) = \begin{bmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial y}{\partial \xi} & \dfrac{\partial z}{\partial \xi} \\ \dfrac{\partial x}{\partial \eta} & \dfrac{\partial y}{\partial \eta} & \dfrac{\partial z}{\partial \eta} \end{bmatrix}^T \tag{6}$$

Although linear elements may introduce approximation errors in highly curved regions, they provide a solid foundation for understanding the basics of mesh-based integration methods in computational geometry. Hence, the needing to cater the highly curved regions is essential where in this paper, quadratic element integration to compute the area of each geometry is introduced. In quadratic element integration, we still use the same equation as in equation (3) but the equation of the shape function for quadratic element integration can be defined as

$$\begin{aligned} N_1 &= 2\xi^2 - \xi, \\ N_2 &= 2\eta^2 - \eta, \\ N_3 &= (1-\xi-\eta)(2(1-\xi-\eta)-1), \\ N_4 &= 4\xi\eta, \\ N_5 &= 4(1-\xi-\eta)\eta, \\ N_6 &= 4\xi(1-\xi-\eta). \end{aligned} \tag{7}$$

The surface projection can be defined as

$$\begin{aligned} S_{P_{yz}}(\xi,\eta) &= 1, \\ S_{P_{xy}}(\xi,\eta) &= S_{P_{zx}}(\xi,\eta) = \frac{r}{\sqrt{r^2 - P_1^2 - P_2^2}}. \end{aligned} \tag{8}$$

where P_1 and P_2 is the coordinates subjected to the surface projection of each plane. In quadratic element integration, the determinant of the Jacobian matrix can be defined in terms of ξ and η as notated in equation (9)

$$J(\xi,\eta) = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix}, \tag{9}$$

where

$$\begin{aligned} J_{11}(\xi,\eta) &= (4\xi-1)x_1 - (4(1-\xi-\eta)+1)x_3 \\ &\quad +4\xi x_4 - 4\eta x_5 + (4-8\xi-4\eta)x_6 \\ J_{12}(\xi,\eta) &= (4\xi-1)y_1 - (4(1-\xi-\eta)+1)y_3 \\ &\quad +4(1-\xi-\eta)y_4 - 4(1-\xi-\eta)y_5 \\ &\quad +(4-8\xi-4\eta)y_6 \\ J_{21}(\xi,\eta) &= (4\eta-1)x_2 - (4(1-\xi-\eta)+1)x_3 \\ &\quad +4\xi x_4 + 4(1-\xi-2\eta)x_5 - 4\xi x_6 \\ J_{22}(\xi,\eta) &= (4\eta-1)y_2 - (4(1-\xi-\eta)+1)y_3 \\ &\quad +4\xi y_4 + 4(1-\xi-2\eta)y_5 - 4\xi y_6 \end{aligned} \tag{10}$$

To compute the surface area of a geometry, the first step is to evaluate the integral in (3) where the numerical solution of the surface area serves a benchmark for more complex shapes where many engineering problems involve domains with irregular boundaries. For instance, in aerospace design, surface area calculations are essential for optimizing aerodynamics and structural integrity, as aircraft and spacecraft often feature complex shape that can affect airflow and weight distribution [5]. Similarly, in structural engineering, accurate surface area calculations are necessary for assessing material costs, weight distribution, and stress points in structures like bridges and buildings, where irregular surfaces lead to non-uniform pressure distributions impacting stability.

3.0 RESULTS AND DISCUSSION

3.1 Analyzing Surface Area of Spherical Shape

The estimation of surface area for different types of geometry is essential to emphasize on the application of numerical integration, particularly in the case where the geometry is complex and the analytical solution are difficult to obtain. Table 1 presents the comparison between the area and the absolute error calculated for spherical shape where the mesh size generated by the NETGEN Mesh Generator are $N = 44, 72, 118, 230, 620, 1888, 2480$ and 9920 triangles. These meshes were generated automatically by the NETGEN Mesh Generator.

To evaluate the accuracy and the convergence of the surface area approximation, the error between the computed result and the exact solution were computed and the results are presented in Table 1. The exact surface area for a sphere, A_{exact} with radius r is given by the following formula:

$$A_{\text{Exact}} = 4\pi r^2$$

The results in Table 1 indicate that as the mesh size decreases, both linear and quadratic element integration converge towards the exact surface area.

Table 1 Comparison of the area of spherical shape with $r = 0.01$ evaluated using linear and quadratic element integration

N	Area (Linear)	Area (Quadratic)	Error (Linear)	Error (Quadratic)
44	0.001098	0.001214	1.5836×10^{-4}	4.2729×10^{-5}
72	0.001155	0.001225	1.0147×10^{-4}	3.2110×10^{-5}
118	0.001193	0.001237	6.3461×10^{-5}	1.9281×10^{-5}
230	0.001223	0.001246	3.3669×10^{-5}	1.0502×10^{-5}
620	0.001244	0.001252	1.2581×10^{-5}	4.2823×10^{-6}
1888	0.001252	0.001255	4.1425×10^{-6}	1.4159×10^{-6}
2480	0.001253	0.001256	3.1665×10^{-6}	1.1049×10^{-6}
9920	0.001256	0.001256	7.9297×10^{-7}	2.7744×10^{-7}

The convergence of the surface area of both methods can be analyzed by plotting the area and its corresponding errors as indicated in Figure 1 and Figure 2. Although by using both methods, the surface area of sphere converge to the exact solution, quadratic element integration method achieves higher level of accuracy with fewer mesh elements which demonstrating its good performance in approximating the surface area.

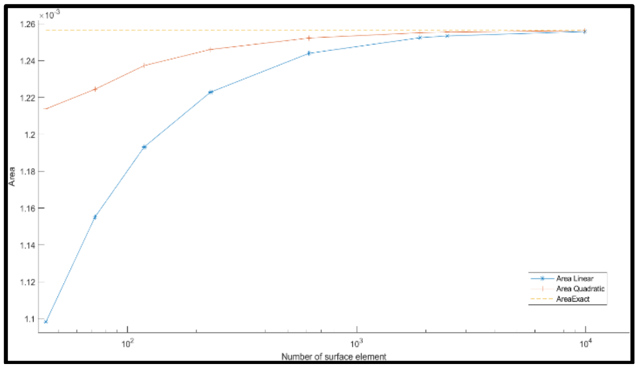


Figure 1 Convergence of the surface area of a sphere with radius $r = 0.01$ using linear and quadratic element integration compared to the exact solution

Figure 2 present the comparison of the absolute error for the surface area computed using both integration schemes, the error for the linear element integration approaching to 0 slowly compared to the quadratic element integration. This highlight that the higher order (quadratic element) is efficient in ensuring that the numerical results is accurate which will beneficial in reducing computational costs as well as in dealing with large scale of problems.

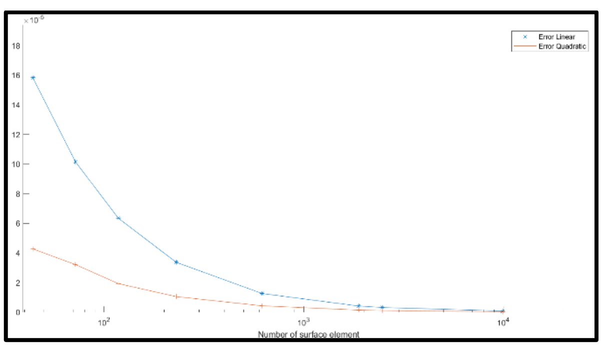


Figure 2 Comparison between the absolute error for a surface area of a sphere computed using linear and quadratic element integration

From the analysis, it is clear that the quadratic element integration method outperforms the linear element method in terms of both accuracy and convergence rate when applied to surface area calculations. The results show that as the mesh size increases, the errors decrease, with quadratic elements providing a faster and more accurate convergence to the exact surface area of the sphere. These findings suggest that for more complex geometries, where analytical solutions are difficult or impossible to obtain, quadratic elements will likely provide better results in terms of both computational efficiency and accuracy.

3.2 Analyzing Surface Area of Cylindrical Shape

This subsection presents a detailed analysis of calculation of cylindrical shape with radius $r = 0.5cm$ and height, $h = 3cm$ where the surface area is approximated using both linear as well as quadratic element integration. Hence from the given radius and height, the exact solution is approximately 7.8540 is used as a benchmark to assess the accuracy of the numerical results.

Table 2 Comparison of the area of cylindrical shape with $r = 0.5cm$ and height, $h = 3cm$ evaluated using linear and quadratic element integration

N	Area (Linear)	Area (Quadratic)	Error (Linear)	Error (Quadratic)
632	7.8035	7.8378	0.0064	0.0021
1676	7.8352	7.8474	0.0024	8.3335×10^{-4}
6704	7.8493	7.8523	5.9763×10^{-4}	2.0897×10^{-4}
9728	7.8506	7.8527	4.3671×10^{-4}	1.5906×10^{-4}
18432	7.8523	7.8534	2.1646×10^{-4}	7.6245×10^{-5}
26816	7.8528	7.8536	1.4943×10^{-4}	4.9833×10^{-5}

Table 2 shows the comparison between the area approximation obtained using linear and quadratic element, together with their respective absolute errors for increasing number of surface elements. From the approximation, increasing number of elements improves the accuracy of the area approximation for both methods and the numerical approximation gradually converge towards the exact value. However, the quadratic elements consistently depicted a more accurate numerical approximation across all levels of discretization. The curve for the quadratic elements demonstrates smoother and faster convergence to the exact solution compared to the linear elements, especially at lower mesh densities. This behavior depicts that higher accuracy of the shape function is essential in representing curved geometries like the surface of a cylinder.

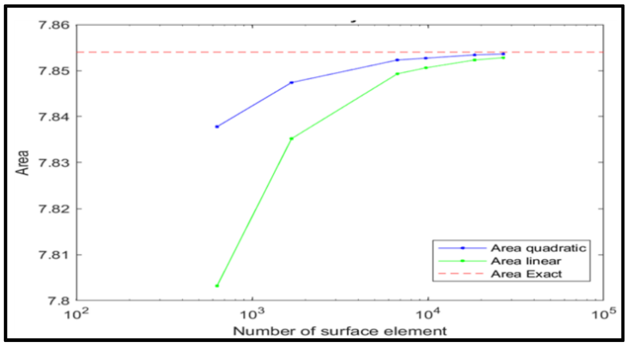


Figure 3(a) Graphical comparison of the surface area of a cylinder with radius $r = 0.5cm$ and height, $h = 3cm$

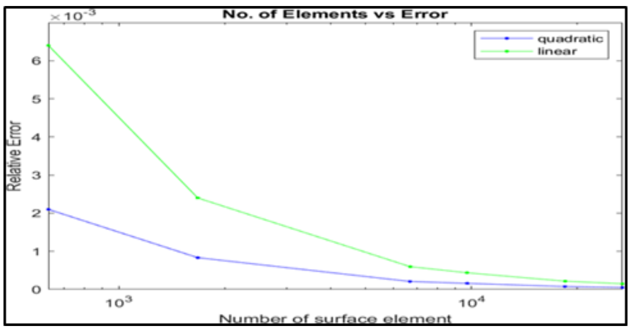


Figure 3(b) Absolute error for a cylinder computed using linear and quadratic element integration using linear and quadratic element integration compared to the exact solution

3.3 Analyzing Surface Area of Cube Shape

For the second geometric shape, the surface area of the cube was evaluated using both linear and quadratic triangular elements. By using the same step as cylinder, the results are shown as Table 3 where AreaL is the area of cube using linear element integration, AreaQ is the area of cube using quadratic element integration, ErrorL is absolute error of cube using linear elements integration and ErrorQ is absolute error of cube using quadratic elements integration. For this shape, the side length which is $S = 1$ was considered. The mesh that is generated for cube using NETGEN mesh generator are N=192,472,1888,3072 and 7552. The results shown as the following Table 2.

Table 3 Comparison of the area of cube with $S = 1$ evaluated using linear and quadratic element integration

N	Area (Linear)	Area (Quadratic)	Error (Linear)	Error (Quadratic)
192	6	6	0	0
472	6.000000000000008	6.000000000000007	7.9936×10^{-15}	7.1054×10^{-15}
1888	6.000000000000008	6.000000000000005	7.9936×10^{-15}	5.3291×10^{-15}
3072	6	6	0	0
7552	5.999999999999922	6.000000000000013	7.8160×10^{-15}	6.2978×10^{-14}

Based on the Table 3, it shows that both linear and quadratic finite elements provide near-perfect solutions for the surface area of a cube. The error from the exact solution are nearly equal to zero. Therefore, for simple geometries, linear elements are enough.

Then, the Figure 4(a) indicates that quadratic and linear element methods yield surface area results for the cube that are nearly identical to the analytical solution exact regardless of the number of surface elements used. There is no visible difference from the true area even at the less meshes, and this show that both methods are highly accurate for this geometry. This is due to the fact that the cube consists of all flat surfaces, for which linear triangular elements can be exactly matched without any creation of geometric approximation error as shown in Figure 4(b). Thus, increasing the number of elements or the use of higher-order elements does not yield visible accuracy improvements, since the geometry is already perfectly modeled. Quadratic elements, which generally good at curvature capture, are no better in this case. Therefore, for flat geometries like a cube, both elements are equally as good, as opposed to in curved geometries.

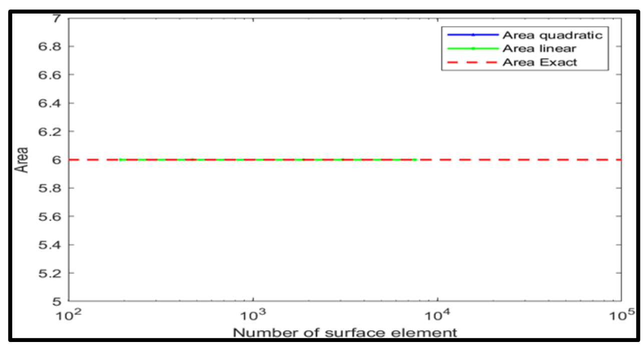


Figure 4(a) Graphical comparison of the surface area of a cube with side length $S = 1$

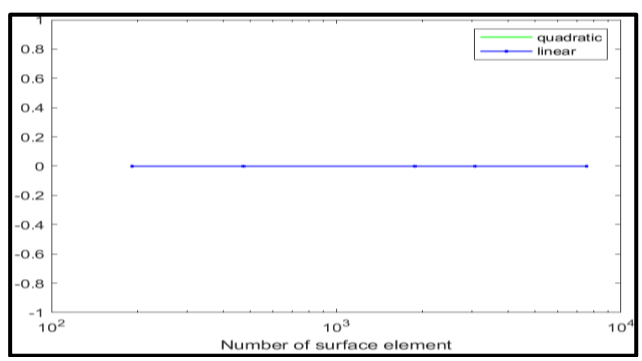


Figure 4(b) Absolute error for a cube computed using linear and quadratic element integration using linear and quadratic element integration compared to the exact solution

3.2 Interactive GUI for Surface Visualization: Cube

As an initiative, an interactive graphical user interface (GUI) is built in a graphical and as an intuitive way of exploring the surface of a cube. It allows for easy operations such as rotation, zooming, and inspection of geometric details to help users better comprehend surface properties and spatial structure. The graphical user interface (GUI) is a helpful device to make the computation of the surface area of a cube simpler. As it has an interactive and welcoming interface, the GUI allows the user to visualize the three-dimensional figure of the cube. The visual interaction allows the user simply to identify the individual surfaces that contribute to the total amount of the area. By making these calculations simple, the GUI reduces human errors and enhances productivity in surface area calculation, especially among users who are not mathematically oriented. This approach is particularly useful in learning and numerical analysis environments, where intuitive and accurate surface visualizations are highly essential.

3.2.1 Steps to run the Interactive GUI for Surface Visualization: Cube

The decision to initially implement the GUI for a cube shape is based on its simplicity and regularity, where it can be a useful starting points in validating the functionality of the GUI. The process of using the interactive GUI for surface area calculations involves a series of steps which are outlined in the following.

Steps 1: Selecting Surface Elements and Points

The first step in order to run the GUI is to generate the surface elements as well as the number of points for a cube surface. This can be achieved by generating both inputs in NETGEN Mesh Generator software by Joachim Schoberl [12] as in Figure 5. The output from the NETGEN Mesh Generator will be save in .txt file and will be imported to the developed GUI as in Figure 6 for which the user need to click the *Browse* button located in the *Input Geometry* section of the GUI interface.

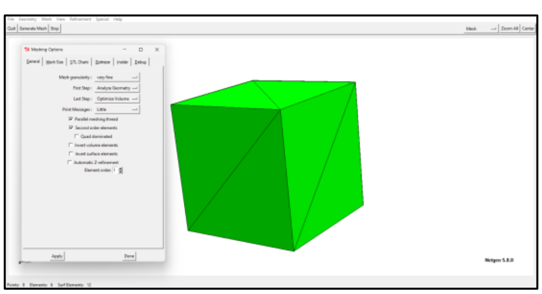


Figure 5 The interface of NETGEN Mesh Generator to generate the surface elements and points

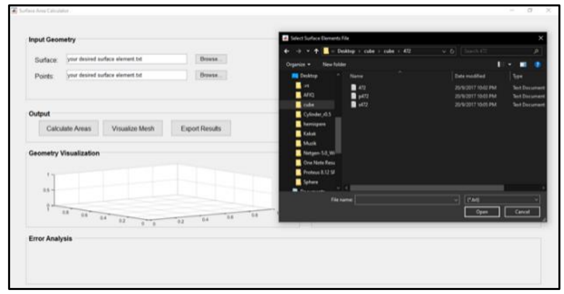


Figure 6 Selecting generated surface elements and points from NETGEN Mesh Generator

Steps 2: Calculating and Visualizing the Surface Area and Mesh

After selecting the surface element and points, the next step is to compute the surface area of the cube and visualize the mesh. To perform this calculation, the user must click the Calculate Areas button. Upon execution, the results of the calculation will be displayed in the Results Section of the interface, as shown in Figure 7. Additionally, to visualize the cube with its mesh, the user can click the Visualize Mesh button. This will generate a mesh visualization in the Geometry Visualization section of the GUI, as depicted in Figure 8. The mesh representation allows for a clearer understanding of the geometric configuration of the cube and the connectivity of its elements.

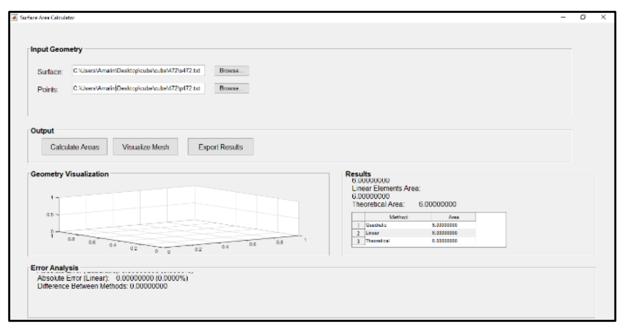


Figure 7 The output can be seen in the *Results* section showing quadratic and linear element areas

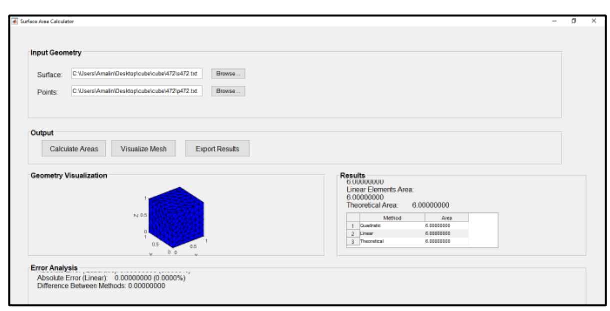


Figure 8 The absolute error and the geometry visualization can be seen in *Geometry Visualization* Section and *Error Analysis* section

Steps 3: Exporting the Results

Finally, to save both the surface area calculation results and the mesh visualization, the user must click the Export Results button. This will enable the results to be saved in a .txt file format for future reference or further analysis. The export functionality ensures that the data can be preserved and utilized beyond the GUI session.

4.0 CONCLUSION

Overall, the outcomes indicate that the choice of element type should be guided by the complexity of the geometry. For the curved surfaces of a cylinder, for instance, quadratic elements are more precise. For the flat surfaces of a cube, linear elements are more efficient and equally as precise, so they would be the best choice for simple geometries. For future work, it is recommended to apply numerical integration methods to complicated geometries such as the torus, which has a more complicated curved surface than the geometry investigated in this paper. That would challenge the stability of existing techniques and guide the development of meshing and integration for topologically complicated surfaces. Next is to explore the application of higher order elements like cubic or quartic to improve the accuracy of the surface area especially for curved geometries because it will utilize more nodes, then allowing it to apprehend complex curvature efficiently. As a result, it can reach the high accuracy and could benefit where precision is critical.

Acknowledgement

This research was funded by a grant from Universiti Teknologi Malaysia (UTM) for financial funding under UTM Encouragement Research (UTMER) grant scheme with reference number PY/2024/00856, cost center number Q.J130000.3854.42J43 and Kurita Water and Environment Foundation (KWEF) 2025 (25Pmy125) under International Grant R.J130000.7354.1U125.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

[1] Abgrall, R., J. Nordström, P. Öffner, and S. Tokareva. 2020. Analysis of the SBP-SAT Stabilization for Finite Element Methods, Part I: Linear Problems. *Journal of Scientific Computing*. 85(2): 1–29. https://doi.org/10.1007/s10915-020-01349-z.

[2] Benvenuti, E., A. Chiozzi, G. Manzini, and N. Sukumar. 2022. Extended Virtual Element Method for Two-Dimensional Linear Elastic Fracture. *Computer Methods in Applied Mechanics and Engineering*. 390: 114352. https://doi.org/10.1016/j.cma.2021.114352.

[3] Chai, Y., W. Li, and Z. Liu. 2022. Analysis of Transient Wave Propagation Dynamics Using the Enriched Finite Element Method with Interpolation Cover Functions. *Applied Mathematics and Computation*. 412: 126564. https://doi.org/10.1016/j.amc.2021.126564.

[4] Du, Q., and X. Feng. 2020. The Phase Field Method for Geometric Moving Interfaces and Their Numerical Approximations. *Handbook of Numerical Analysis*. 21: 425–508. https://doi.org/10.1016/bs.hna.2019.05.001.

[5] Elsheltat, S. F., A. A. Alshara, and W. Elshara. 2024. Aerodynamic Evaluation of a Symmetrical Airfoil Using Wind Tunnel Testing. *African Journal of Advanced Pure and Applied Sciences* (AJAPAS). 49–56.

[6] Geng, D., N. Dai, P. Guo, S. Zhou, and H. Di. 2021. Implicit Numerical Integration of Highly Nonlinear Plasticity Models. *Computers and Geotechnics*. 132: 103961. https://doi.org/10.1016/j.compgeo.2020.103961.

[7] Khairuddin, T. K. A., M. A. K. Ahmad, Z. Majid, A. M. Samad, and I. Ma'arof. 2024. Volume and Surface Area of 3D Mesh Objects: A MATLAB GUI. *Semarak International Journal of Fundamental and Applied Mathematics*. 1(1): 13–23. https://doi.org/10.37934/sijfam.1.1.1323.

[8] Lehrenfeld, C. 2016. High-Order Unfitted Finite Element Methods on Level Set Domains Using Isoparametric Mappings. *Computer Methods in Applied Mechanics and Engineering*. 300: 716–33. https://doi.org/10.48550/arXiv.1509.02762.

[9] Luo, Y., X. Liu, F. Chen, H. Zhang, and X. Xiao. 2023. Numerical Simulation on Crack–Inclusion Interaction for Rib-to-Deck Welded Joints in Orthotropic Steel Deck. *Metals*. 13(8): 1402. https://doi.org/10.3390/met13081402.

[10] Müzel, D. S., E. P. Bonhin, N. M. Guimarães, and E. S. Guidi. 2020. Application of the Finite Element Method in the Analysis of Composite Materials: A Review. *Polymers*. 12(4): 818. https://doi.org/10.3390/polym12040818.

[11] Ray, N., D. Wang, X. Jiao, and J. Glimm. 2012. High-Order Numerical Integration over Discrete Surfaces. *SIAM Journal on Numerical Analysis*. 50(6): 3061–83. https://doi.org/10.1137/110857404.

[12] Schöberl, J. 2009. *Getting Started: Netgen 4.X Mesh Generator*. https://netgen-mesher.sourceforge.net/docs/ng4.pdf.

[13] Scroggs, M. W., J. S. Dokken, C. N. Richardson, and G. N. Wells. 2022. Construction of Arbitrary-Order Finite Element Degree-of-Freedom Maps on Polygonal and Polyhedral Cell Meshes. *ACM Transactions on Mathematical Software*. 48(2): 1–23. https://doi.org/10.1145/3524456.

[14] Smyth, G. K. 1998. Numerical Integration. In *Encyclopedia of Biostatistics*. 3088–95.

[15] Sukri, S. A., Y. S. Hoe, and T. K. A. Khairuddin. 2020. Quadratic Element Integration of Approximated First-Order Polarization Tensor for Sphere. *Malaysian Journal of Fundamental and Applied Sciences*. 16(5): 560–65. https://doi.org/10.11113/mjfas.v16n5.1916.

[16] Sukri, S. A., Y. S. Hoe, and T. K. A. Khairuddin. 2021. Study of Different Order of Gaussian Quadrature Using Linear Element Interpolation in First-Order Polarization Tensor. *Malaysian Journal of Fundamental and Applied Sciences*. 17(4): 343–53. https://doi.org/10.11113/mjfas.v17n4.2051.

[17] Sukri, S. A., Y. S. Hoe, and T. K. A. K. Khairuddin. 2021. First-Order Polarization Tensor Approximation Using Multivariate Polynomial Interpolation Method via Least-Square Minimization Technique. *Journal of Physics: Conference Series*. 1988(1): 012013. https://doi.org/10.1088/1742-6596/1988/1/012013.

[18] Taborda, D. M., L. Zdravković, D. M. Potts, H. J. Burd, B. W. Byrne, K. G. Gavin, G. T. Houlsby, R. J. Jardine, T. Liu, C. M. Martin, and R. A. McAdam. 2020. Finite-Element Modelling of Laterally Loaded Piles in a Dense Marine Sand at Dunkirk. *Géotechnique*. 70(11): 1014–29. https://doi.org/10.1680/jgeot.18.PISA.006.

[19] Viguerie, A., G. Lorenzo, F. Auricchio, D. Baroli, T. J. Hughes, A. Patton, A. Reali, T. E. Yankeelov, and A. Veneziani. 2021. Simulating the Spread of COVID-19 via a Spatially Resolved Susceptible–Exposed–Infected–Recovered–Deceased (SEIRD) Model with Heterogeneous Diffusion. *Applied Mathematics Letters.* 111: 106617. https://doi.org/10.1016/j.aml.2020.106617.

[20] Wang, Y., Q. Pang, M. Li, and Y. Liu. 2024. Comparative Study of Algorithms for the Conversion from Surface Mesh Models to Volume Mesh Models. *Advances in Civil Engineering.* 2024(1): 4669855. https://doi.org/10.1155/2024/4669855.

[21] Wolfrom, R. 2023. An Introduction to Surface Area." *Particle Technology Labs*. https://particletechlabs.com/ptl-press/introduction-to-surface-area/.