



CST0015

Performance and accuracy investigation of the closed-form quadrilateral element for heat conduction problem

Benjapa Yontsakul, Pramote Dechaumphai, and Nippon Wansophark*

Department of Mechanical Engineering, Faculty of Engineering, Chulalongkorn University, Phayathai Road, Patumwan, Bangkok, 10330, Thailand

* Corresponding Author: nippon.w@eng.chula.ac.th

Abstract. This paper presents the finite element method for analyzing two-dimensional heat conduction problem using the closed-form quadrilateral element. In general, Gauss-Legendre or numerical integration is the most widely used method in deriving the matrices for heat conduction problems. However, the accuracy of the result depends on the number of Gauss points which in turn has an effect on the CPU time. The closed-form expression is developed by using Mathematica symbolic manipulation and hand-post processing. The results show that the derived closed-form matrices can improve not only the solution accuracy but also the CPU time required for computation.

1. Introduction

Finite element method is the acknowledged numerical analysis technique due to its performance and accuracy for solving the complex engineering problems. When considering the real world engineering problems, the computational domain always be an irregular geometry and arbitrary condition; hence, the analytical solution is required a lot of effort or sometimes it is impossible to determine. That is the reason why finite element method is widely used.

Generally, element's shape of two dimensional problem can be divided into two element types; triangular element and quadrilateral element. Quadrilateral element gives the better solution than triangular element because the triangular element presents the distribution of the solution as flat plane; on the other hand, the quadrilateral element produces the bilinear interpolation function which obtain the higher accuracy. However, the integration over quadrilateral element is very difficult and complex that the numerical integration is commonly applied. However, applying such numerical integration always causes numerical error in the solution.

Gauss-Legendre is common numerical integration technique applied for the quadrilateral element. This method evaluates solution by multiplication of weight and value of function at Gauss point location. The accuracy of solution depends on number of Gauss point. The more number of Gauss point is applied, the more accuracy will be obtained. Number of Gauss point affects to the CPU time consuming which cannot be denied.

Nowadays, algebraic and symbolic manipulation programs are continuously developed and the computer hardware is greater efficiency than the past [1]. Therefore, researchers take these advantages to develop and improve element stiffness matrix calculation. In 1989, Kikuchi [2] used symbolic manipulation to direct integration the stiffness matrix for 4 node quadrilateral element with elastic-

plastic problem which give the high accurate result but required the long CPU time. Yagawa et al. [3] worked with the modified numerical integration scheme based on symbolic manipulation which can reduce CPU time and accurate solution. According to review article of symbolic computation in structural engineering [4], there is considerable attempt to develop stiffness matrix in closed-form expression by using symbolic language; nevertheless, it is not obviously state how FEM closed-form expression were. Then, Videla et al. [5] concerned with 8 node quadrilateral element on elastic problem in 2007. This paper concludes that the solution is much closed to the exact solution but stiffness matrix integration spent large computer memory and integral equations of stiffness matrix were very large and complex. In 2014, Roque [6] presented symbolic analysis on plate bending by using MATLAB to create two programs: generating expression and numerical calculation program. However, no published works presents explicitly closed-form expression and considers shape element to categorize FEM equation.

This paper develops the closed-form quadrilateral element for heat conduction problems. The solution accuracy and computational time of developed finite element matrices are investigated by using three cases which are single element matrix, heat conduction without heat generation and heat conduction with heat generation.

2. Finite Element Method for Heat Transfer Analysis

2.1. Governing equations for heat transfer problem

For two-dimensional heat transfer problem, the temperature distribution of pure heat conduction is governed by the differential equation as;

$$\frac{\partial}{\partial x} \left(k_x \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k_y \frac{\partial T}{\partial y} \right) + Q = 0 \quad (2.1)$$

where k_x , k_y are the thermal conductivity in x and y direction, respectively.

Q is the internal heat generation.

2.2. Finite element formulation for quadrilateral element

The method of weighted residuals is applied with equation (2.1) along with the isotropic material condition. Therefore, the finite element equation can be written as shown below;

$$\int_{\Omega} N \left(\frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) + Q \right) d\Omega = 0 \quad (2.2)$$

Temperature distribution on the quadrilateral element in the form of multiplication between interpolation function and nodal temperatures is shown in equation (2.3).

$$T(x, y) = \sum_{i=1}^4 N_i(x, y) T_i = \underset{1 \times 4}{[N]} \underset{4 \times 1}{\{T\}} \quad (2.3)$$

After substitute equation (2.3) into equation (2.2) and then apply Green's theorem. The finite element equation for heat transfer problem can be rewritten in matrix form as;

$$[K_c] \{T\} = \{Q\} \quad (2.4)$$

where $[K_c] = \iint_A k \left(\left\{ \frac{\partial N}{\partial x} \right\} \left[\frac{\partial N}{\partial x} \right] + \left\{ \frac{\partial N}{\partial y} \right\} \left[\frac{\partial N}{\partial y} \right] \right) t dx dy \quad (2.5a)$

$$\{Q\} = \iint_A Q \{N\} t dx dy \quad (2.5b)$$

2.3. Interpolation function of quadrilateral element

The quadrilateral element in Cartesian coordinate system (x, y) can be transformed to the natural coordinate (ξ, η) in range of $-1 < \xi < 1$ and $-1 < \eta < 1$. Thus, the interpolation function, N_i , as shown in equation (2.3) can be presented as follow;

$$\begin{aligned} N_1 &= \frac{1}{4}(1-\xi)(1-\eta) & N_3 &= \frac{1}{4}(1+\xi)(1+\eta) \\ N_2 &= \frac{1}{4}(1+\xi)(1-\eta) & N_4 &= \frac{1}{4}(1-\xi)(1+\eta) \end{aligned} \quad (2.7)$$

As the interpolation function is in natural coordinate, the chain rule is employed and rewritten in the matrix form as below;

$$\begin{aligned} \frac{\partial N}{\partial \xi} &= \frac{\partial N}{\partial x} \frac{\partial x}{\partial \xi} + \frac{\partial N}{\partial y} \frac{\partial y}{\partial \xi} \\ \frac{\partial N}{\partial \eta} &= \frac{\partial N}{\partial x} \frac{\partial x}{\partial \eta} + \frac{\partial N}{\partial y} \frac{\partial y}{\partial \eta} \end{aligned} \longrightarrow \begin{Bmatrix} \frac{\partial N}{\partial \xi} \\ \frac{\partial N}{\partial \eta} \end{Bmatrix} = \underbrace{\begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix}}_{\substack{[J] \\ (2 \times 2)}} \begin{Bmatrix} \frac{\partial N}{\partial x} \\ \frac{\partial N}{\partial y} \end{Bmatrix} \quad (2.8)$$

Hence, the first derivative of interpolation function in Cartesian coordinate can be written in natural coordinate as;

$$\begin{Bmatrix} \frac{\partial N}{\partial x} \\ \frac{\partial N}{\partial y} \end{Bmatrix} = [J]^{-1} \begin{Bmatrix} \frac{\partial N}{\partial \xi} \\ \frac{\partial N}{\partial \eta} \end{Bmatrix} = [J^*] \begin{Bmatrix} \frac{\partial N}{\partial \xi} \\ \frac{\partial N}{\partial \eta} \end{Bmatrix} \quad (2.9)$$

After substitution of equation (2.9) into equation (2.5a) and (2.5b), the finite element matrix can be rewritten in the natural $\xi - \eta$ coordinate as

$$[K_C] = \int_{-1}^1 \int_{-1}^1 k [B(\xi, \eta)]^T [B(\xi, \eta)] t |J(\xi, \eta)| d\xi d\eta \quad (2.10a)$$

$$\{Q\} = \int_{-1}^1 \int_{-1}^1 Q \{N(\xi, \eta)\} t |J(\xi, \eta)| d\xi d\eta \quad (2.10b)$$

where $[B]$ is the first derivative of interpolation function matrix;

$$[B(\xi, \eta)] = \begin{bmatrix} J_{11}^* & J_{12}^* \\ J_{21}^* & J_{22}^* \end{bmatrix} \begin{bmatrix} \frac{\partial N_1}{\partial \xi} & \frac{\partial N_2}{\partial \xi} & \frac{\partial N_3}{\partial \xi} & \frac{\partial N_4}{\partial \xi} \\ \frac{\partial N_1}{\partial \eta} & \frac{\partial N_2}{\partial \eta} & \frac{\partial N_3}{\partial \eta} & \frac{\partial N_4}{\partial \eta} \end{bmatrix} \quad (2.11)$$

2.4. Conventional quadrilateral matrix

For conventional technique, Gauss Legendre integration is applied to compute the conduction matrix and heat generation vector which can be written as the following;

$$[K_C] = \sum_{i=1}^{NG} \sum_{j=1}^{NG} W_i W_j k [B(\xi_i, \eta_j)]^T [B(\xi_i, \eta_j)] \cdot t \cdot |J(\xi_i, \eta_j)| \quad (2.12a)$$

$$\{Q\} = \sum_{i=1}^{NG} \sum_{j=1}^{NG} W_i W_j Q \{N(\xi_i, \eta_j)\} \cdot t \cdot |J(\xi_i, \eta_j)| \quad (2.12b)$$

where NG is number of Gauss point

W_i, W_j are weights in ξ and η direction, respectively

ξ_i, η_j are location of Gauss point

2.5. Closed-form quadrilateral matrix

To derive the closed-form quadrilateral matrix, the heat conduction matrix (equation (2.10a)) can be rewritten to the form as shown in equation (2.13).

$$[K_C] = k \cdot t \cdot [I] \quad (2.13)$$

where
$$[I] = \int_{-1}^1 \int_{-1}^1 [B(\xi, \eta)]^T [B(\xi, \eta)] |J| d\xi d\eta \quad (2.14)$$

The multiplication of matrix $[B]^T [B] |J|$ in equation (2.14) can be calculated by hand and grouped with the intermediate parameters as shown below;

$$I_{ij} = \int_{-1}^1 \int_{-1}^1 \frac{h_{ij} + f_{ij}\xi + g_{ij}\eta + l_{ij}\xi^2 + m_{ij}\eta^2 + n_{ij}\xi\eta}{p + q\xi + r\eta} d\xi d\eta \quad (2.15)$$

while h, f, g, l, m, n, p, q and r are rational function of nodal coordinate.

Then, equation (2.15) can be integrated directly using symbolic manipulation in Mathematica program to obtain the closed-form matrix of heat conduction. After considering the shape of element, some intermediate parameters in equation (2.15) become zero when the opposite sides of element are paralleled. Therefore, the closed-form matrix of heat conduction can be divided into four cases based on the parallelism of the side of element.

2.5.1. No parallel side.

For this case, all variable are valid, thus the conduction matrix can be noted as following:

$$K = kt \int_{-1}^1 \int_{-1}^1 \frac{h + f\xi + g\eta + l\xi^2 + m\eta^2 + n\xi\eta}{p + q\xi + r\eta} d\xi d\eta \quad (2.16)$$

$$K = kt \left(A + \frac{B_1 + B_2 + B_3 + B_4}{6q^3 r^3} \right) \quad (2.17)$$

where

$$A = \frac{2(-3g^2r + 2mpq^2 + 3fqr^2 + npqr - 4lpr^2)}{3q^2r^2} + \frac{4(gr - mp)}{r^2}$$

$$B_1 = E_1 \log E_1 \left\{ \frac{r(r(3q(2hq - fE_3) + 2l(C_1 + p(q - 2r) - qr)) - 3gq^2E_2 + nq(C_2 + p(q + r) + 2qr))}{+2mq^2(C_1 - p(2q - r) - qr)} \right\}$$

$$B_2 = -E_2 \log E_2 \left\{ \frac{r(r(3q(2hq - fE_4) + 2l(C_1 + p(q - 2r) - qr)) - 3gq^2E_2 + nq(C_2 + p(q + r) + 2qr))}{+2mq^2(C_1 + p(2q - r) - qr)} \right\}$$

$$B_3 = E_3 \log E_3 \left\{ \frac{r(-r(3q(2hq - fE_1) + 2l(C_1 - p(q + 2r) + qr)) + 3gq^2E_4 + nq(-C_2 + p(q - r) + 2qr))}{+2mq^2(-C_1 - p(2q + r) - qr)} \right\}$$

$$B_4 = -E_4 \log E_4 \left\{ \frac{r(-r(3q(2hq - fE_2) + 2l(C_1 - p(q - 2r) + qr)) + 3gq^2E_3 + nq(-C_2 + p(q + r) - 2qr))}{+2mq^2(-C_1 - p(2q - r) + qr)} \right\}$$

$$E_1 = p - q - r, E_2 = p - q + r, E_3 = p + q - r, E_4 = p + q + r$$

$$C_1 = p^2 + q^2 + r^2, C_2 = p^2 - 2q^2 - 2r^2$$

2.5.2. One pair of parallel sides (between line 1-2 and line 3-4)

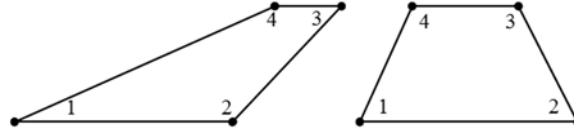


Figure 2.1. The element that has line $\overline{12}$ parallels with line $\overline{34}$.

As line $\overline{12}$ is parallel to line $\overline{34}$, both slopes are equal and give $q = 0$. Hence, the conduction matrix equation can be rewritten as equation (2.18) and corresponding algebraic equation be noted as equation (2.19)

$$K = kt \int_{-1}^1 \int_{-1}^1 \frac{h + f\xi + g\eta + l\xi^2 + m\eta^2 + n\xi\eta}{p + r\eta} d\xi d\eta \quad (2.18)$$

$$K = \frac{kt}{3r^3} \left[-12mpr + 12gr^2 - (6mp^2 - 6gpr + 6hr^2 + 2lr^2)(\log(p - r) - \log(p + r)) \right] \quad (2.19)$$

2.5.3. One pair of parallel sides (between line 1-4 and line 2-3)

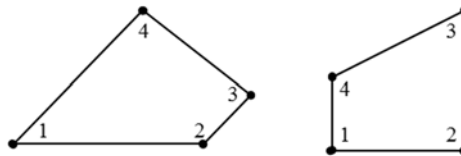


Figure 2.2. The element that has line $\overline{14}$ parallel with line $\overline{23}$.

As line $\overline{14}$ is parallel to line $\overline{23}$, both slopes are equal and give $r = 0$. Hence, the conduction matrix equation can be rewritten as equation (2.20) and corresponding algebraic equation be noted as equation (2.21)

$$K = kt \int_{-1}^1 \int_{-1}^1 \frac{h + f\xi + g\eta + l\xi^2 + m\eta^2 + n\xi\eta}{p + q\xi} d\xi d\eta \quad (2.20)$$

$$K = \frac{kt}{3q^3} \left[-12lpq + 12fq^2 - (6lp^2 - 6fpq + 6hq^2 + 2mq^2)(\log(p - q) - \log(p + q)) \right] \quad (2.21)$$

2.5.4. *Two pairs of parallel sides.*

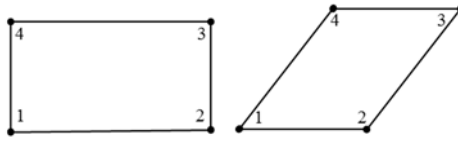


Figure 2.3. The element that has two pairs of parallel sides.

When there are two parallels, both q and r become zero. Hence, the conduction matrix equation can be rewritten as equation (2.22) and corresponding algebraic equation be noted as equation (2.23)

$$K = kt \int_{-1}^1 \int_{-1}^1 \frac{h + f\xi + g\eta + l\xi^2 + m\eta^2 + n\xi\eta}{p} d\xi d\eta \quad (2.22)$$

$$K = kt \left(\frac{4(3h + l + m)}{3p} \right) \quad (2.23)$$

3. Results

To evaluate the efficiency of the derived matrix, the investigation in both accuracy and computational time compared between closed-form expression and conventional method are considered. Three different cases are selected for this paper; 1) single element matrix, 2) heat conduction problem without heat generation and 3) heat conduction problem with heat generation.

3.1. Single element matrix

Accuracy and performance of a derived conduction matrix is evaluated by several different shape elements. The accuracy is measured by the total percentage difference of all 16 members in a conduction matrix [2] as shown in equation (3.1). While the performance is measured by CPU time consuming to calculate single element matrix. The time consuming is presented in ratio of time used by Gauss-Legendre integration to closed-form expression.

$$\sqrt{\sum_{i=1}^4 \sum_{j=1}^4 \left(\frac{K_{ij}^{ClosedForm} - K_{ij}^{GaussLegendre}}{K_{ij}^{ClosedForm}} \right)^2} \times 100\% \quad (3.1)$$

The several different shape elements and their description are as following:

Case A : Rectangular element (2 pairs of parallel side)

Case B : Parallelogram or Rhomboid

Case C : Trapezoid (1 pair of parallel side, 2 right angles, 1 obtuse angle and 1 acute angle)
Case D : Trapezoid (1 pair of parallel side, 2 obtuse angles and 2 acute angles)
Case E : Trapezium (1 right angles, 2 obtuse angle and 1 acute angle)
Case F : Trapezium (1 right angles, 1 obtuse angle and 2 acute angle)
Case G : Trapezium (1 obtuse angle and 3 acute angle)
Case H : Trapezium (2 obtuse angle and 2 acute angle)
Case I : Poor shape trapezium

Table 3.1. Different between closed-form expression and Gauss Legendre integration.


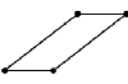







	Case A	Case B	Case C	Case D	Case E	Case F	Case G	Case H	Case I
									
NG=2	0.00	0.00	200.03	34.74	9.81	136.02	14.40	36.17	193.20
NG=3	-	-	6.07	9.91	0.23	8.17	0.44	3.65	93.40
NG=4	-	-	0.18	2.76	0.01	0.55	0.02	0.52	55.41
NG=5	-	-	0.01	0.76	0.00	0.04	0.00	0.09	36.77
NG=6	-	-	0.00	0.21	-	0.00	-	0.02	26.21
NG=7	-	-	-	0.06	-	-	-	0.00	19.63
NG=8	-	-	-	0.02	-	-	-	-	15.25
NG=9	-	-	-	0.00	-	-	-	-	12.20
NG=10	-	-	-	-	-	-	-	-	9.98

Table 3.2. Ratio of CPU time between Gauss Legendre integration to closed-form expression.

	Case A	Case B	Case C	Case D	Case E	Case F	Case G	Case H	Case I
NG=2	2.09	2.08	2.09	1.98	1.51	1.47	1.51	1.48	1.48
NG=3	-	-	2.08	2.06	1.55	1.56	1.55	1.52	1.57
NG=4	-	-	2.20	2.16	1.61	1.61	1.61	1.59	1.60
NG=5	-	-	2.38	2.26	1.71	1.65	1.71	1.72	1.68
NG=6	-	-	2.49	2.47	-	1.80	-	1.81	1.78
NG=7	-	-	-	2.59	-	-	-	1.95	1.93
NG=8	-	-	-	2.78	-	-	-	-	2.11
NG=9	-	-	-	3.05	-	-	-	-	2.24
NG=10	-	-	-	-	-	-	-	-	2.42

Table 3.1 shows that some shape elements are required high number of Gauss points to achieve the same accuracy as closed-form expression such as 9×9 Gauss point for case D and 10×10 Gauss point for case I. The results demonstrate that the closed-form element matrix provides more solution accuracy compared to the conventional technique.

From Table 3.2, Gauss-Legendre always spends more time than the closed-form expression, for example, 2.42 and 3.05 times for case I and D, respectively. When considering the cases based on parallelism of element, if the element has parallel sides, it would spent 25% less than general or non-parallelism cases.

3.2. Heat conduction problem without heat generation.

Considering heat conduction on rectangular plate with insulation along left and bottom side while the other sides are assigned with specific temperature as shown in Figure 3.1. The exact solution for temperature distribution of this problem can be calculated by equation (3.2). The rectangular domain is divided into 118 elements by using AUTOMESH2D [7] as shown in Figure 3.2. Gauss-Legendre method with 2×2 Gauss point is used in this problem to compare the accuracy and performance with closed-form element matrices.

$$T(x, y) = \frac{\cosh\left(\frac{\pi y}{8}\right) \cos\left(\frac{\pi x}{8}\right)}{\cosh\left(\frac{\pi}{4}\right)} \quad (3.2)$$

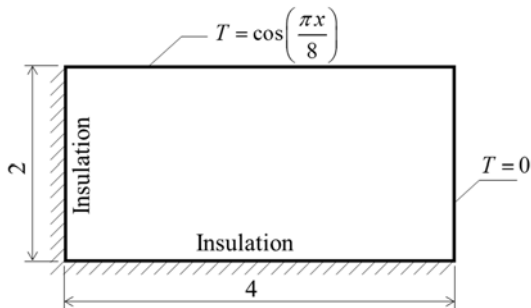


Figure 3.1. Problem statement.

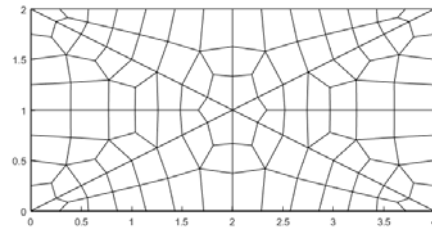


Figure 3.2. Mesh element.

The result of temperature distribution using the closed-form expression is presented in Figure 3.3. Figure 3.4 shows the temperature distribution along the line of symmetry ($y = 1$) from closed-form expression and Gauss-Legendre compared with the exact solution. The accuracy is measured by the mean absolute percentage error (MAPE) (equation (3.3)) and maximum percentage error. While the performance is measured by CPU time required to calculate the system conduction matrix. The accuracy and performance of both closed-form and Gauss-Legendre are summarized in Table 3.3.

$$MAPE = \frac{\sum_{n=1}^{NN} \left| \frac{T_n - T_n^{Exact}}{T_n^{Exact}} \right|}{NN} \times 100\% \quad (3.3)$$

where NN is number of node.

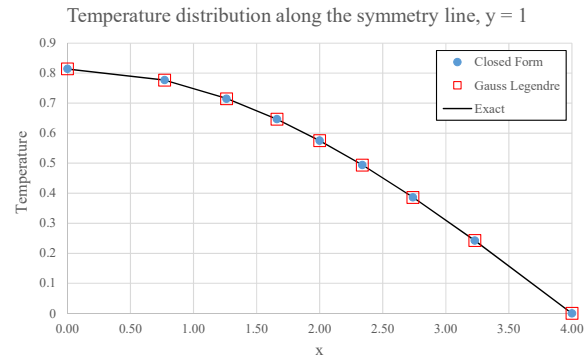
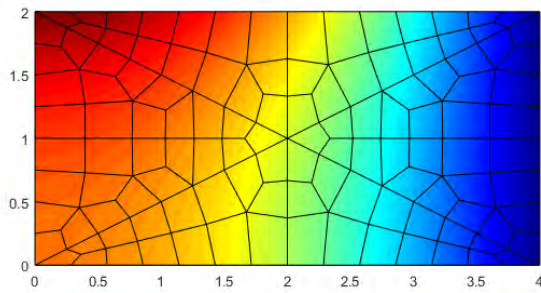


Figure 3.3. Temperature distribution on rectangular plate. **Figure 3.4.** Temperature along the symmetry line.

From Figure 3.4, temperature from both closed-form expression and Gauss-Legendre are in the same trend as the exact solution. Both mean absolute percentage error and maximum percentage error from closed-form expression are less than Gauss-Legendre integration. Closed-form expression also spends 27.22% less CPU time than Gauss-Legendre integration.

Table 3.3. Result of Closed form expression and Gauss Legendre

Item	Unit	Closed Form	Gauss-Legendre
Mean absolute percentage error	%	0.05478	0.05515
Maximum percentage error	%	0.6251	0.6310
CPU time	sec	0.006468	0.008887

3.3. Heat conduction problem with heat generation.

Considering heat conduction on the triangular plate with zero temperature on all edges with unit heat generation as shown in Figure 3.5. The exact solution for temperature distribution of this problem can be calculated by equation (3.3). The triangular domain is divided into 320 elements by using AUTOMESH2D as shown in Figure 3.5b and then do re-meshing to 413 and 534 elements in order to see the effect of the refined mesh. Gauss-Legendre method with 2×2 Gauss point is used in this problem to compare the accuracy and performance with closed-form element matrices.

$$T(x, y) = \frac{Q}{4k} (y - 2 + \sqrt{3}x)(y - \sqrt{3}x)y \quad (3.3)$$

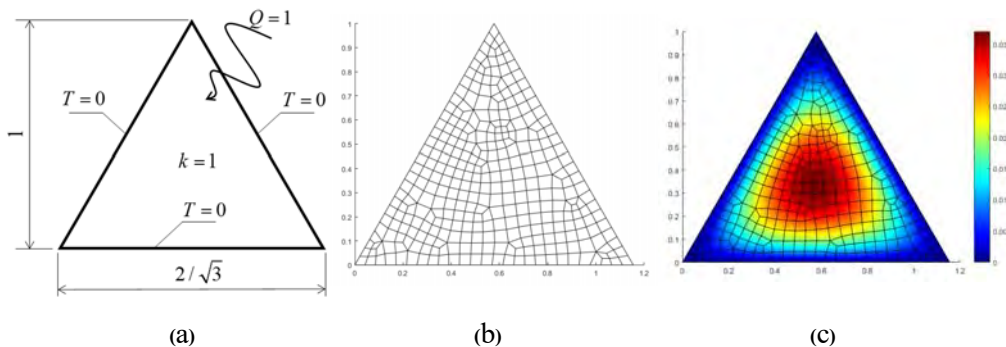


Figure 3.5. a) Problem statement, b) Mesh element, c) Temperature distribution on triangular plate

For comparing the accuracy of closed-form expression and Gauss-Legendre method, the value of mean absolute and maximum percentage error are considered. Table 3.4 shows the mean percentage error of all nodes of closed-form expression is less than Gauss-Legendre integration. Furthermore, closed-form expression gives the lower maximum percentage error than Gauss-Legendre integration. On the other hand, closed-form expression spends 28.44% less CPU time than Gauss-Legendre as presented in Table 3.4.

Table 3.4. Result of closed-form expression and Gauss Legendre.

Item	Unit	Closed-Form	Gauss-Legendre
Mean absolute percentage error	%	0.3343	0.3352
Maximum percentage error	%	5.1184	5.1261
CPU time	Sec	0.0156	0.0218

While the number of mesh increasing, Table 3.5 shows that the closed-form element matrices give more accurate result both mean and maximum percentage error.

Table 3.5. Percentage error of different number of element.

No. of Element	Mean absolute percentage error		Maximum percentage error	
	Closed-Form	Gauss-Legendre	Closed-Form	Gauss-Legendre
320	0.3343	0.3352	5.1184	5.1261
415	0.2784	0.2790	5.1130	5.1219
534	0.2462	0.2473	5.0995	5.1076

4. Conclusion

This paper explicitly presents the closed-form stiffness matrix of the quadrilateral element for heat conduction problems. The evaluation of the accuracy and computational time between both closed-form expression and Gauss-Legendre methods is demonstrated by three cases. The investigations are confirmed that the accuracy of closed-form expression is better than Gauss-Legendre integration, especially in single element testing. Considering a poor shape element, Gauss-Legendre method requires more than 10×10 Gauss points in order to obtain the same accuracy as closed-form expression. Moreover, the CPU time consuming of Gauss-Legendre method is more than the proposed method 1.5 to 3 times. For heat transfer problems, the results of the closed-form expression and Gauss-Legendre are comparable. However, the computational time of closed-form expression is faster than Gauss-Legendre almost 30%. This conclude that the closed-form expression of quadrilateral element developed in this paper can improve the performance and solution accuracy of the finite element method.

References

- [1] Wolfram Research, Inc., Mathematica, Version 10.4, Champaign, IL, 2016.
- [2] Kikuchi, M., Application of the symbolic mathematics system to the finite element program. *Computational mechanics*, 1989. 5(1): p. 41-47.
- [3] Yagawa, G., G.W. Ye, and S. Yoshimura, A numerical integration scheme for finite element method based on symbolic manipulation. *International journal for numerical methods in engineering*, 1990. 29: p.1539-1549.
- [4] M.N. Pavlovic, Symbolic computation in structural engineering. *Computer and Structure*, 2003. 81(22-23) : p.2121-2136.
- [5] Videla, L., Baloa, T., Griffiths D.V., and Cerrolaza, M., Exact integration of the stiffness matrix of an 8-node plane elastic element by symbolic computation. *Numerical Methods for Partial Differential Equations*, 2008. 24(1): p.249-261.
- [6] C.M.C.Roque, Symbolic and numerical analysis of plates in bending using Matlab. *Journal of symbolic computation*, 2014. 61-62: p.3-11.
- [7] Ma, X.W., Zhao, G.Q., and Sun, L., Automesh-2D/3D: robust automatic mesh generator for metal forming simulation. *Materials research innovations*, 2011. 15(s1): p. S482-s486.
- [8] Dechaumphai, P., *Finite element method: Fundamentals and applications*, Alpha Science International, Oxford, London, 2010.