

## Accurate Evaluation Schemes for Triangular Domain Integrals

Farzana Hussain<sup>1</sup>, M. S. Karim<sup>2</sup>

<sup>1,2</sup> (Department of Mathematics, Shahjalal University of Science and Technology, Sylhet -3114, Bangladesh.)

---

**ABSTRACT :** This paper mainly presents higher order Gaussian quadrature formulae for numerical integration over the triangular surfaces. In order to show the exactness and efficiency of such derived quadrature formulae, it shows the effective use of available Gaussian quadrature for square domain integrals to evaluate the triangular domain integrals. Then it presents  $n \times n$  points and  $\frac{n(n+1)}{2} - 1$  points (for  $n > 1$ ) Gaussian quadrature formulae for triangle utilizing  $n$ -point one-dimensional Gaussian quadrature. By use of simple but straight-forward algorithms, Gaussian points and corresponding weights are calculated and presented for clarity and reference. The proposed  $\frac{n(n+1)}{2} - 1$  point formula completely avoids the crowding of Gaussian points. Finally, it presents an error formula which can evaluate the error in monomial/polynomial integration using  $m \times n$  point method successfully. The error calculated by the new formula and the error in calculation of integrals by the proposed methods are found to be in good agreement. Thus, it overcomes all the drawbacks in view of accuracy and efficiency for the numerical evaluation of the triangular domain integrals of any arbitrary functions encountered in the realm of science and engineering.

**Keywords:** Extended Gaussian quadrature, Triangular domain, Numerical accuracy, Convergence, Error formula.

---

### I. Introduction

The integration theory extends from real line to the plane and three dimensional spaces by the introduction of multiple integrals. Integration procedures on finite domains underlie physically acceptable averaging process in engineering. In probabilistic estimations and in spatially discretized approximations, e.g., finite and boundary-element methods, evaluation of integrals over arbitrary-shaped domain  $\Omega$  become the pivotal task. In practice, most of the integrals (encountered frequently) either cannot be evaluated analytically or the evaluations are very lengthy and tedious. Thus, for simplicity numerical integration methods are preferred and the methods extensively employ the Gaussian quadrature technique that was originally designed for one dimensional cases and the procedure naturally extends to two and three-dimensional rectangular/cubic domains according to the notion of the Cartesian product. Gaussian quadratures are considered as the best method of integrating polynomials because they guarantee that they are exact for polynomials less than a specified degree.

In order to obtain the result with the desired accuracy, Gaussian integration points and weights necessarily increase and there is no computational difficulty except time in evaluating any domain integral when the two and three-dimensional regions are bounded respectively, by systems of parallel lines and parallel planes.

Analysts cannot ignore at all the randomness in material properties and uncertainty in geometry that are frequently encountered in complex engineering systems. Specifically, the vital components are rated during quality control inspections according to reliability indices calculated from the average probability density functions that model failure. This entails the evaluation of an integral of the function (say joint probability frequency function) over the volume  $\Omega$  of the component. In general, the  $\Omega$ -shape-class is very irregular in two and three-dimensional geometry. For non parallelogram quadrilateral, very frequent in finite-element modeling, there is no consistent procedure to select the sampling point to implement a Gaussian quadrature on the entire element.

Special integration schemes, e.g., reduced integration over quadrilaterals have been successfully developed in [1] and are widely used in commercial programs. There is no methodical way to design such approximate integration schemes for polygons with more than four sides. An attempt to distribute the sampling points according to the governing perspective transformation fails to assure the error order germane to the quadrature formula. The reason can be traced to the crowding of quadrature points and this numerical computational difficulty persists in all non parallelogram polygonal finite elements [2]. A considerable amount of research has been performed to attain perfect results of domain integration for plane quadrilateral elements where numerical quadrature techniques are employed [3]. The accuracy of a selected quadrature strategy is indicated by compliance with the patch-test proposed in [4].

The overall error in a finite element calculation can be reduced by not relying so heavily on artificial tessellation, which requires the deployment of elements with large number of sides. An elegant systematic procedure to yield shape functions for convex polygons of arbitrary number of sides developed in [5] by which the energy density can be obtained in closed algebraic form in terms of rational polynomials. However, a direct Gaussian

quadrature scheme to numerically evaluate the domain integral on n-sided polygons cannot be constructed to yield the exact results, even on convex quadrilaterals. In two-dimension, n-sided polygons can be suitably discretized with linear triangles rather than quadrilaterals (Fig. 1) and hence triangular elements are widely used in finite element analysis. Another advantage is to be mentioned that there is no difficulty with triangular elements as the exact shape functions are available and the quadrature formulas are also exact for the polynomial integrands [6].

Integration schemes based on weighted residuals are prone to instability since the accuracy goal cannot be controlled. In deterministic cases the underlying averaging process may be inconsistent, which was stated as a variational crime [7]. In stochastic differential equation literature [8-9], such averaging processes are termed dishonest [10]. Thus, the high accuracy integration method is demanded and it is meaningful when the shape functions are the very best. Therefore, there has been considerable interest in the area of numerical integration schemes over triangles [11-24]. It is explicitly shown in [21, 24] that the most accurate rules are not sufficient to evaluate the triangular domain integrals and for some element geometry these rules are not reliable also.

To address all these short comings, to make a proper balance between accuracy and efficiency, to avoid the crowding of quadrature points this article proposed  $n \times n$  points and  $\frac{n(n+1)}{2} - 1$  points higher order Gaussian quadrature formulae to evaluate the triangular domain integrals. It is thoroughly investigated that the  $\frac{n(n+1)}{2} - 1$  point formulae are totally free of crowding of Gauss points and also appropriate in view of accuracy and efficiency. Finally, it presents an error formula which can evaluate the error in monomial/polynomial integration using  $m \times n$  point method successfully. The error calculated by the new formula and the error in calculation of integrals by the proposed methods are found to be in good agreement. Hence we believe that the formulae will find better place in numerical solution procedure of continuum mechanics problems.

## II. Problem Statement

In finite and boundary element methods for two-dimensional problems, a pivotal task is to evaluate the integral of a function  $f$ :

$$I_1 = \iint_{\Omega} f d\Omega ; \quad \Omega: \text{element domain} \tag{2.1}$$

Observe that  $I_1$  can be calculated as a sum of integrals evaluated over simplex divisions'  $\Delta_i$ :

$$\Omega = \bigcup_i \Delta_i ; \quad \Delta_i \text{ completely covers } \Omega \tag{2.2}$$

$\Delta_i$  = triangle for two dimensional domain (see Fig. 1).

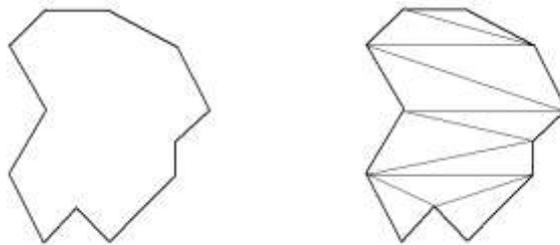


Fig 1: Original and triangulated domain of Integral.

Now (2.1) can be written as

$$I_1 = \iint_{\Omega} f d\Omega = \sum_i \iint_{\Delta_i} f d\Delta_i \tag{2.3}$$

To evaluate the integral  $I_1$  in (2.3), it is now required to evaluate the triangular domain integral:

$$I_2 = \iint_{\Delta} f(x, y) dx dy ; \quad \Delta: \text{triangle (arbitrary)} \tag{2.4}$$

Integration over triangular domains is usually carried out in normalized co-ordinates. To perform the integration, first map one vertex (vertex 1) to the (-1,-1), the second vertex (vertex 2) to point (1,-1) and the third vertex (vertex 3) to point (-1, 1). This geometrical transformation to standard triangle is most easily accomplished by use of shape functions as:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{pmatrix} \begin{pmatrix} N_1 \\ N_2 \\ N_3 \end{pmatrix} \tag{2.5}$$

$$\text{Where } N_1 = -\frac{1}{2}(s+t), \quad N_2 = \frac{1}{2}(s+1), \quad N_3 = \frac{1}{2}(t+1), \tag{2.6}$$

The original and the transformed triangles are shown in Fig. 2.

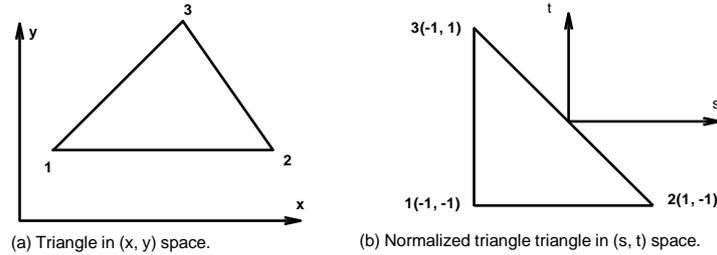


Fig. 2: Original and Transformed triangle.

Form (2.5) using (2.6), we obtain

$$\begin{aligned} x(s,t) &= \frac{1}{2}(x_2 + x_3) + \frac{1}{2}(x_2 - x_1)s + \frac{1}{2}(x_3 - x_1)t \\ y(s,t) &= \frac{1}{2}(y_2 + y_3) + \frac{1}{2}(y_2 - y_1)s + \frac{1}{2}(y_3 - y_1)t \end{aligned} \tag{2.7}$$

and hence

$$\frac{\partial(x,y)}{\partial(s,t)} = \frac{1}{4}[(x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)] = \frac{\text{Area}}{4} \tag{2.8}$$

Finally (2.4) reduces to

$$I_2 = \frac{\text{Area}}{4} \int_{s=-1}^1 \int_{t=-1}^{-s} f(x(s,t), y(s,t)) dt ds \tag{2.9}$$

One can simply verify that

$$I_2 = \frac{\text{Area}}{4} \int_{t=-1}^1 \int_{s=-1}^{-t} f(x(s,t), y(s,t)) ds dt \tag{2.10}$$

Here, we wish to mention that the evaluation of integrals  $I_2$  in (2.9) and (2.10) by the existing Gaussian quadrature (i.e. 7-point and 13-point) will yield the same results. Thus, any one of these two can be evaluated numerically. Influences of these integrals will be investigated later to present new quadrature formulae for triangles.

### III. Numerical Evaluation Procedures

In this section, we wish to describe different procedures to evaluate the integral  $I_2$  numerically and new Gaussian quadrature formulae for triangles.

#### 3.1 Procedure-1 (use of Gaussian quadrature for triangle, GQT):

Gaussian quadrature for triangle in [11-24] can be employed as

$$I_2 = \frac{\text{Area}}{4} \sum_{i=1}^{NGP} \sum_{j=1}^{NGP} W_i W_j f(x(s_i, t_j), y(s_i, t_j)) \tag{3.1}$$

where  $(s_i, t_j)$  are the  $ij$ -th sampling points  $W_i, W_j$  are corresponding weights and  $NGP$  denotes the number of gauss points in the formula. It is thoroughly investigated that in some cases available Gaussian quadrature for triangle cannot evaluate the integral  $I_2$  exactly [11, 21, 24].

#### 3.2 Procedure-2 (use of Gaussian quadrature for square domain integral, IOST):

Integration over the normalized (standard) triangle can be calculated as a sum of integrals evaluated over

three quadrilaterals (Fig – 3). Then  $I_2$  can be written as:

$$\begin{aligned}
 I_2 &= \frac{Area}{4} \int_{s=-1}^1 \int_{t=-1}^{-s} f(x(s,t), y(s,t)) dt ds \\
 &= \frac{Area}{4} \sum_{j=1}^3 \iint_{e_j} f(x(s,t), y(s,t)) dt ds \\
 &= \frac{area}{96} \int_{\xi=-1}^1 \int_{\eta=-1}^1 [f(X_1, Y_1)(4 - \xi + \eta) + f(X_2, Y_2)(4 - \xi - \eta) + f(X_3, Y_3)(4 + \xi - \eta)] d\xi d\eta
 \end{aligned}
 \tag{3.2}$$

(3.2) is obtained after transforming each quadrilaterals in to a square in  $(\zeta, \eta)$  space where

$$\begin{aligned}
 X_1 &= \frac{1}{24} [a_{11} + a_{12}\xi + a_{13}\eta + a_{14}\xi\eta] & Y_1 &= \frac{1}{24} [b_{11} + b_{12}\xi + b_{13}\eta + b_{14}\xi\eta] \\
 X_2 &= \frac{1}{24} [a_{21} + a_{22}\xi + a_{23}\eta + a_{24}\xi\eta] & Y_2 &= \frac{1}{24} [b_{21} + b_{22}\xi + b_{23}\eta + b_{24}\xi\eta] \\
 X_3 &= \frac{1}{24} [a_{31} + a_{32}\xi + a_{33}\eta + a_{34}\xi\eta] & Y_3 &= \frac{1}{24} [b_{31} + b_{32}\xi + b_{33}\eta + b_{34}\xi\eta]
 \end{aligned}
 \tag{3.3}$$

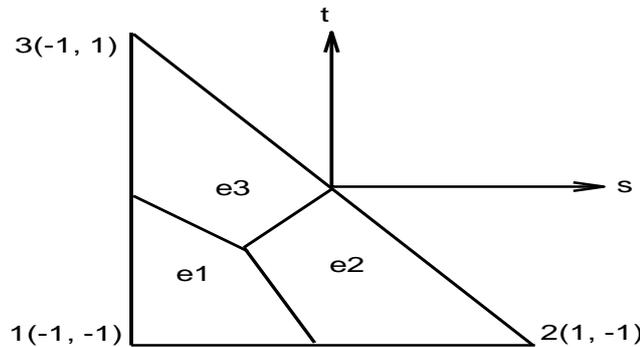


Fig 3: Standarded triangle splitted into three quadrilaterals.

and

$$\begin{aligned}
 a_{11} &= [5x_1 + 5x_2 + 14x_3] & b_{11} &= [5y_1 + 5y_2 + 14y_3] \\
 a_{12} &= [-x_1 + 5x_2 - 4x_3] & b_{12} &= [-y_1 + 5y_2 - 4y_3] \\
 a_{13} &= [-5x_1 + x_2 + 4x_3] & b_{13} &= [-5y_1 + y_2 + 4y_3] \\
 a_{14} &= [x_1 + x_2 - 2x_3] & b_{14} &= [y_1 + y_2 - 2y_3] \\
 a_{21} &= [14x_1 + 5x_2 + 5x_3] & b_{21} &= [14y_1 + 5y_2 + 5y_3] \\
 a_{22} &= [-4x_1 + 5x_2 - x_3] & b_{22} &= [-4y_1 + 5y_2 - y_3] \\
 a_{23} &= [-4x_1 - x_2 + 5x_3] & b_{23} &= [-4y_1 - y_2 + 5y_3] \\
 a_{24} &= [2x_1 - x_2 - x_3] & b_{24} &= [2y_1 - y_2 - y_3] \\
 a_{31} &= [5x_1 + 14x_2 + 5x_3] & b_{31} &= [5y_1 + 14y_2 + 5y_3] \\
 a_{32} &= [-5x_1 + 4x_2 + x_3] & b_{32} &= [-5y_1 + 4y_2 + y_3] \\
 a_{33} &= [-x_1 - 4x_2 + 5x_3] & b_{33} &= [-y_1 - 4y_2 + 5y_3] \\
 a_{34} &= [x_1 - 2x_2 + x_3] & b_{34} &= [y_1 - 2y_2 + y_3]
 \end{aligned}$$

Now right hand side of (3.2) with (3.3) can be evaluated by use of available higher order Gaussian quadrature for square. For clarity, we mention that each quadrilaterals in Fig. 3 is transformed into 2- square in  $(\xi, \eta) \in \{(-1,-1), (1,-1), (1,1), (-1,1)\}$  space through isoperimetric transformation and that resulted the equivalent integral  $I_2$  in (3.2).

**3.3 Procedure-3:**

In this section, we wish to present two new techniques to evaluate the integrals over the triangular surface and to calculate Gaussian points and corresponding weights for triangle.

Using mathematical transformation equations:

$$s = \xi, \quad t = \frac{(1 - \xi)(1 + \eta)}{2} - 1 \tag{3.4}$$

Then, the integral  $I_2$  of (2.9) is transformed into an integral over the surface of the standard square  $\{(\xi, \eta) | -1 \leq \xi, \eta \leq 1\}$  and the (2.7) reduces to

$$\begin{aligned} x &= \frac{1}{2}(x_2 + x_3) + \frac{1}{2}(x_2 - x_1)\xi + \frac{1}{4}(x_3 - x_1)(\eta - \xi - \xi\eta - 1) \\ y &= \frac{1}{2}(y_2 + y_3) + \frac{1}{2}(y_2 - y_1)\xi + \frac{1}{4}(y_3 - y_1)(\eta - \xi - \xi\eta - 1) \end{aligned} \tag{3.5}$$

Now the determinant of the Jacobean and the differential area are:

$$\frac{\partial(s,t)}{\partial(\xi,\eta)} = \frac{\partial s}{\partial \xi} \frac{\partial t}{\partial \eta} - \frac{\partial s}{\partial \eta} \frac{\partial t}{\partial \xi} = \frac{1}{2}(1 - \xi) \tag{3.6}$$

$$\text{and } ds dt = dt ds = \frac{\partial(s,t)}{\partial(\xi,\eta)} d\xi d\eta = \frac{1}{2}(1 - \xi) d\xi d\eta \tag{3.7}$$

Now using (3.4) and (3.7) into (2.9), we get

$$\begin{aligned} I_2 &= \text{Area} \int_{-1}^1 \int_{-1}^1 f\left(x\left(\xi, \frac{(1 - \xi)(1 + \eta)}{2} - 1\right), y\left(\xi, \frac{(1 - \xi)(1 + \eta)}{2} - 1\right)\right) \left(\frac{1 - \xi}{8}\right) d\xi d\eta \\ &= \text{Area} \int_{-1}^1 \int_{-1}^1 f\left(\xi, \frac{(1 - \xi)(1 + \eta)}{2} - 1\right) \left(\frac{1 - \xi}{8}\right) d\xi d\eta \end{aligned} \tag{3.8}$$

In order to evaluate the integral  $I_2$  in (3.8) efficient Gaussian quadrature co-efficient (points and weights) are readily available so that any desired accuracy can be readily obtained [21, 24].

**3.3.1 New quadrature formula, GQSTS:**

In this section we are straightly computing Gaussian quadrature formula for standard triangles (GQSTS). The Gauss points are calculated simply for  $i = 1, m$  and  $j = 1, n$ . Thus the  $m \times n$  point Gaussian quadrature formula for (3.8) gives

$$I_2 = \text{Area} \sum_{i=1}^m \sum_{j=1}^n \frac{1 - \xi_i}{8} W_i W_j f\left(\xi_i, \frac{(1 - \xi_i)(1 + \eta_j)}{2} - 1\right) = \text{Area} \sum_{r=1}^{m \times n} G_r f(u_r, v_r) \tag{3.9}$$

where  $(u_r, v_r)$  are the new Gaussian points,  $G_r$  is the corresponding weights for triangles. Again, if we consider the integral  $I_2$  of (2.10) and substitute

$$s = \frac{(1 + \xi)(1 - \eta)}{2} - 1, \quad t = \eta$$

Then one can obtain (on the same line of (3.9))

$$I_2 = \text{Area} \int_{-1}^1 \int_{-1}^1 f\left(\frac{(1 + \xi)(1 - \eta)}{2} - 1, \eta\right) \frac{1 - \eta}{8} d\xi d\eta = \text{Area} \sum_{r=1}^{m \times n} G'_r f(u'_r, v'_r) \tag{3.10}$$

where  $G'_r$  and  $(u'_r, v'_r)$  are respectively weights and Gaussian points for triangle. All the Gaussian points and corresponding weights can be calculated simply using the following algorithm:

- step1.  $r \rightarrow 1$
- step2.  $i = 1, m$
- step3.  $j = 1, n$

$$G_r = \left(\frac{1-\xi_i}{8}\right) W_i W_j, \quad u_r = \xi_i, \quad v_r = \frac{(1-\xi_i)(1+\eta_j)}{2} - 1$$

$$G'_r = \left(\frac{1-\eta_j}{8}\right) W_i W_j, \quad u'_r = \frac{(1+\xi_i)(1-\eta_j)}{2} - 1, \quad v'_r = \eta_j$$

$r = r + 1$

- step4. compute step3
- step5. computestep2

For clarity and reference, computed Gauss points and weights (for  $n = 3, 7$ ) based on above algorithm listed in Table-1 and Fig. 4a shows the distribution of Gaussian points for  $n = 10$ . In figure- 4a, it is seen that there is a crowding of points at least at one point within the triangle and that is one of the major causes of error germen in the calculation. To avoid this crowding further modification is needed. This modification is obtained in the next section.

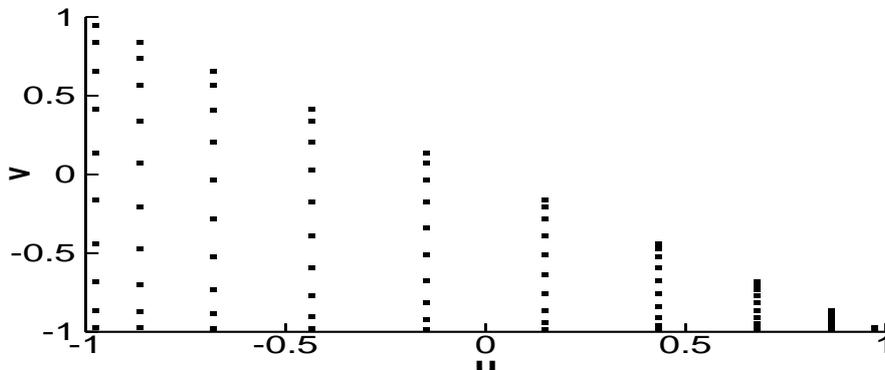


Fig 4a. Distribution of 10 X 10 Gauss Points.

### 3.3.2 New quadrature formula, GQSTM:

It is clearly noticed in the (3.9) that for each  $i$  ( $i = 1, 2, \dots, m$ )  $j$  varies from 1 to  $n$  and hence at the terminal value  $i = m$  there are  $n$  crowding points as shown in Table-1 and Fig.- 4a. To overcome this situation, we can use the advantage of (3.9) by making  $j$  dependent on  $i$  for the calculation of new gauss points and corresponding weights. To do so, we wish to calculate gauss points and weights for  $i = 1, m-1$  and  $j = 1, m + 1 - i$  that is  $\frac{m(m+1)}{2} - 1$  points Gaussian quadrature formulae from (3.9) as:

$$I_2 = Area \sum_{i=1}^{m-1} \sum_{j=1}^{m-i+1} \frac{(1-\xi_i)}{8} W_i W_j f\left(\xi_i, \frac{(1-\xi_i)(1+\eta_j)}{2} - 1\right) = Area \sum_{r=1}^{\frac{m(m+1)}{2}-1} L_r f(p_r, q_r) \quad 3.11$$

where  $(p_r, q_r)$  are the new Gaussian points,  $L_r$  is the corresponding weights for triangles. Similarly, (3.10) can be written as

$$I_2 = Area \int_{-1}^1 \int_{-1}^1 f\left(\frac{(1+\xi_i)(1-\eta_j)}{2} - 1, \eta_j\right) \left(\frac{1-\eta_j}{8}\right) d\xi d\eta = Area \sum_{r=1}^{\frac{m(m+1)}{2}-1} \{L'_r f(p'_r, q'_r)\} \quad 3.12$$

Where  $p'_r, q'_r$  are the new Gaussian points,  $L'_r$  are respectively weights of Gaussian points for triangle. All the Gaussian points and corresponding weights can be calculated simply using the following algorithm:

step1.  $r \rightarrow 1$   
 step2.  $i = 1, m-1$   
 step3.  $j = 1, m+1-i$   

$$L_r = \left(\frac{1-\xi_i}{8}\right)W_iW_j, \quad p_r = \xi_i, \quad q_r = \frac{(1-\xi_i)(1+\eta_j)}{2} - 1$$
  
 step4.  $j = 1, m-1$   
 step5.  $i = 1, m+1-j$   

$$L'_r = \left(\frac{1-\eta_j}{8}\right)W_iW_j, \quad p'_r = \frac{(1+\xi_i)(1-\eta_j)}{2} - 1, \quad q'_r = \eta_j$$
  

$$r = r + 1$$
  
 step6. compute step3  $\rightarrow$  step2  
 step7. compute step5  $\rightarrow$  step4

Thus, the new  $\frac{m(m+1)}{2} - 1$  points Gaussian quadrature formulae is now obtained which is crowding free.

For clarity and reference, computed Gauss points and weights (for  $n = 5, 9$ ) based on above algorithm listed in table-2 and Fig. 4b shows the distribution of Gaussian points for  $n = 10$  i.e. 54-points formula.

### IV. Application Examples

To show the accuracy and efficiency of the derived formulae, following examples with known results are considered:

$$I_1 = \int_{y=0}^1 \int_{x=0}^{1-y} (x+y)^{\frac{1}{2}} dx dy = 0.4$$

$$I_2 = \int_{y=0}^1 \int_{x=0}^{1-y} (x+y)^{-\frac{1}{2}} dx dy = 0.6666667$$

$$I_3 = \int_{y=0}^1 \int_{x=0}^y (x^2 + y^2)^{-\frac{1}{2}} dx dy = 0.881373587$$

$$I_4 = \int_{y=0}^1 \int_{x=0}^y \exp^{|x+y-1|} dx dy = 0.71828183$$

Computed values (by use of three procedures) are summarized in Table-3. Some important remarks from the Table – 3 are:

- Usual Gauss quadrature (GQT) for triangles e.g. 7-point and 13-point rules cannot evaluate the integral of non-polynomial functions accurately.
- Splitting standard triangle into quadrilaterals (IOST) provides the way of using Gaussian quadrature for square and the convergence rate is slow but satisfactory in view of accuracy.
- New Gaussian quadrature formulae for triangle (GQSTS & GQSTM) are exact in view of accuracy and efficiency and GQSTM is faster in calculation of non-polynomial function.

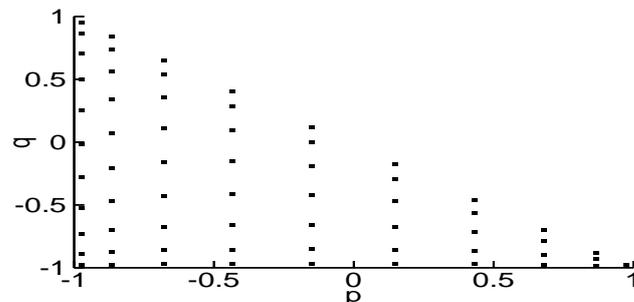


Fig 4b. Distribution of 54 Gauss points ( $m = 10$ ).

Again, we consider the following integrals of rational functions due to [24] to test the influences of formulae in (3.9), (3.10), (3.11) and (3.12) described in procedure-3. Consider

$$II^{p,q} = \int_{y=0}^1 \int_{x=0}^{1-y} \frac{x^p y^q}{(\alpha + \beta x + \gamma y)} dx dy$$

Example-1:  $II^{r,0} = \int_{y=0}^1 \int_{x=0}^{1-y} \frac{x^r}{(0.375-0.375x)} dx dy, \quad \beta \neq \gamma = 0$

Example-2:  $II^{0,r} = \int_{y=0}^1 \int_{x=0}^{1-y} \frac{y^r}{(0.375-0.375y)} dx dy, \quad \gamma \neq \beta = 0$

Example-3: 
$$II^{0,0} = \int_{y=0}^1 \int_{x=0}^{1-y} \frac{1}{12 + 21.5367983 kx - 8.82106723 ly} dx dy, \quad \beta \neq 0, \gamma \neq 0$$

Example-4: 
$$II^{0,0} = \int_{y=0}^1 \int_{x=0}^{1-y} \frac{1}{12 + 9.941125498(x+y)} dx dy, \quad \beta = \gamma \neq 0$$

Results are summarized in Tables-(4, 5, 6). In Tables (4 - 6) for method GQSTS, Formula 1 is for (3.9) and Formula 2 is for (3.10) are written, for method GQSTM, Formula 1 is for (3.11) and Formula 2 is for (3.12) are written. These tables substantiated the influences of numerical evaluation of the integrals as described in section-3. Some important comments may be drawn from the Tables (4 - 6).

- For the integrand  $\frac{x^r}{\alpha + \beta x + \gamma y}$  with  $\beta \neq \gamma = 0$  the formulae in (3.9) and (3.11) described in procedure-3 is more accurate and rate of convergence is higher. But the new formula in (3.11) requires very less computational effort.
- Similarly for the integrand  $\frac{y^r}{\alpha + \beta x + \gamma y}$  with  $\gamma \neq \beta = 0$  the second formula in (3.10) and (3.12) as described in procedure-3 is more accurate and convergence is higher. Here also the new formula in (3.12) requires very less computational effort.
- Similar influences of these formulae in procedure-3 may be observed for different conditions on  $\beta, \gamma$ .
- General Gaussian quadrature e.g. 7-point and 13-point rules cannot evaluate the integral of rational functions accurately.

It is evident that the new formulae e.g. (3.11) and (3.12) are very fast and accurate in view of accuracy and equally applicable for any geometry that is for different values of  $\alpha, \beta$  and  $\gamma$ . We recommend this is appropriate quadrature scheme for triangular domain integrals encountered in science and engineering.

### V. Error Analysis:

Consider the integral  $\int_{x=-1}^1 \int_{y=-1}^1 (\sum_{i=0}^N x^i y^{N-i}) dy dx$

We are using m points along x direction and n points along y direction, the error in this method is found to be of the form

$$\frac{2^{2m+2} [m!]^4}{(2m+1)[(2m)!]^3} f_x^{2m}(x_1, y_1) + \frac{2^{2n+2} [n!]^4}{(2n+1)[(2n)!]^3} f_y^{2n}(x_2, y_2) \tag{5.1}$$

Where  $f_x^{2m}(x_1, y_1)$  is the (2m) th partial derivative of the function with respect to x,  $f_y^{2n}(x_2, y_2)$  is the (2n) th partial derivative of the function with respect to y and  $(x_1, y_1)$  and  $(x_2, y_2)$  are points somewhere in the domain  $[-1, 1] \times [-1, 1]$ . The computed results for the errors are given in Table 7.

We know, the n-point Gaussian Quadrature rule gives exact results for polynomials of degree at most 2n-1. Thus with n = 2, we have a rule with 4 nodes which is exact for any polynomial of degrees at most 3 in x and y separately, so the total degree of this monomial is at most 6. But this rule is not exact for all monomials of degree at most 6, which includes  $x^6, x^5 y, x^4 y^2, x^3 y^3, x^2 y^4, x y^5, y^6$ . Thus  $p \times q$  -point rule cannot give exact result for all monomial of order  $2(p+q)-2$ . Let N be the maximum value of  $p+q$  for each term in the polynomial. Then N-point rule can calculate all the polynomials of order up to  $2p-1$  or  $2q-1$  in x and y separately. Hence  $N \times N$ - point rule can evaluate all the monomials of degree at most  $2(p+q)-2$  ie  $2N-2$ .

The method GQSTS is tested on the integral of all monomials  $x^i y^j$  where  $i, j$  are non-negative integers such that  $i+j \leq N$ . In Table-8, we present the absolute error over corresponding polynomial integral of order up to 30. Table 8 is in good agreement with the above statement. The results are compared with the results of [26] and it is observed that the new method GQSTS is always accurate for monomial/polynomial functions in view of both accuracy and efficiency and hence a proper balance is observed.

### VI. Conclusions:

In continuum mechanics and in spatially discretized approximations, e.g., finite- and boundary-element methods, evaluation of integrals over arbitrary-shaped domain  $\Omega$  is the important and pivotal task. Most of the integrals defy our analytical skills and we are resort to numerical integration schemes. Among all the numerical integration schemes Gaussian quadrature formulae are widely used for its simplicity and easy incorporation in computer.

In general, the  $\Omega$  –shape-class is very irregular in two and three-dimensional geometry. If the domain  $\Omega$  is subdivided into quadrilaterals or into hexahedron respectively in two and three-dimensions, higher order Gaussian quadrature formulae are readily available. Furthermore, reduced integrations techniques compliance with the patch-test is also available [1, 4]. It is notable that there is no methodical way to design such approximate integration schemes for polygons with more than four sides. Generally simplexes e.g., triangle and tetrahedron are popular finite elements to discretized the arbitrary domain  $\Omega$ . Though these are the widely used elements in FEM and BEM, Gaussian quadrature formulae for the triangular/tetrahedral domain integrals are not developed as compared to the square domain integrals. To achieve the desired accuracy of the integral it is necessary to increase the number of points and corresponding weights. Therefore, it is an important task to make a proper balance between accuracy and efficiency of the calculations.

For the necessity of the exact evaluation of the integrals, this article shows first the integral over the triangular domain can be computed as the sum of three integrals over the square domain. In this case the readily available quadrature formulae for the square can be used for the desired accuracy. The results obtained are found accurate in view of accuracy and efficiency. Secondly, it presented new techniques to derive quadrature formulae utilizing the one dimensional Gaussian quadrature formulae and that overcomes all the difficulties pertinent to the higher order formulae. The first technique (GQSTS) derives two-types of  $m \times m$  point quadrature formula utilizing the one dimensional m-point Gaussian quadrature formula. Finally, in the second technique (GQSTM) two-types of  $\frac{m(m+1)}{2} - 1$  points quadrature formulae are derived utilizing the m-point one dimensional Gaussian quadrature formula. It is observed that the  $\frac{m(m+1)}{2} - 1$  points scheme is appropriate for the triangular domain integrals as it requires less computational effort for desired accuracy. Through practical application examples, it is demonstrated that the new appropriate Gaussian quadrature formula for triangles are accurate in view of accuracy and efficiency and hence a proper balance is observed.

Finally, an error formula for numerical double integral using Gaussian quadrature is described. The error calculated by the new formula (5.1) and the error in calculation of integrals by the proposed methods are found to be in good agreement.

Thus, we believe that the newly derived appropriate quadrature formulae for triangles and error formula will ensure the exact evaluation of the integrals in an efficient manner and enhance the further utilization of triangular elements for numerical solution of field problems in science and engineering.

### References

- [1]. T.J.R. Hughes, *The Finite Element Method*, (Englewood Cliffs, N.J: Prentice-Hall, 1966).
- [2]. D.F. Rogers and J.A. Adams, *Mathematical Elements of Computer Graphics*, (McGraw-Hill, New York, 1990).
- [3]. K.J. Bathe, (1996), *Finite Element Procedures*, (Englewood Cliffs, N.J: Prentice-Hall, 1966).
- [4]. Irons, B.M. and Razzaque, A. (1972), Experience with the patch-test for convergence of finite element method, Academic press, New York.
- [5]. E.L. Wachspress, *A rational finite element basis*, (Academic press, San Diego, 1975).
- [6]. O.C. Zienkiewicz and R.L. Taylor, *The finite element method*, Vol. 1, 5<sup>th</sup> Ed., (Butterworth- Heinemann. Elsevier Science, Burlington, Mass, 2000).
- [7]. G. Strang and G.J. Fix, (1973), *An analysis of the finite element method*, (Englewood Cliffs, N.J: Prentice-Hall, 1973).
- [8]. J.B. Keller, Stochastic Equations and Wave propagation in Random Media, *Proc., Symposium on Applied Mathematics*, Vol. 16, American Mathematical Society, Providence, R.I, 1964.
- [9]. J.B. Keller and H.P. McKean, "SIAM-AMS Proceedings." *Stochastic differential equations*, Vol.VI, American Mathematical Society, Providence, R.I., 1973.
- [10]. J.E. Molyneux, "Analysis of 'dishonest' methods in the theory of wave propagation in a random medium". *J. Opt. Soc. Am.*, 58, 1968, 951-957.
- [11]. P. C. Hammer, O. J. Marlowe and A. H. Stroud, Numerical integration over simplex and cones, *Math Tables Other Aids Computation*, **10**, 1956, 130 – 136.
- [12]. P. C. Hammer and A. H. Stroud, Numerical integration over simplexes, *Math Tables Other Aids Computation*, **10**, 1956, 137 – 139.
- [13]. P. C. Hammer and A. H. Stroud, Numerical evaluation of multiple integrals, *Math Tables Other Aids Computation*, **12**, 1958, 272 – 280.
- [14]. G. R. Cowper, Gaussian quadrature formulas for triangles, *International journal on numerical methods and engineering*, **7**, 1973, 405 – 408.
- [15]. J. N. Lyness and D. Jespersen, Moderate degree symmetric quadrature rules for triangle, *J. Inst. Math. Applications*, **15**, 1975, 19 – 32.
- [16]. F. G. Lannoy, Triangular finite element and numerical integration. *Computers Struct*, **7**, 1977, 613.
- [17]. D. P. Laurie, Automatic numerical integration over a triangle, *CSIR Spec. Rep. WISK 273*, National Institute for Mathematical Science, Pretoria, 1977.
- [18]. M. E. Laursen and M. Gellert, Some criteria for numerically integrated matrices and quadrature formulas for triangles, *International journal for numerical methods in engineering*, **12**, 1978, 67 – 76.
- [19]. F. G. Lether, Computation of double integrals over a triangle, *Journal comp. applic. Math.*, **2**, 1976, 219 – 224.
- [20]. P. Hillion, Numerical integration on a triangle, *International journal on numerical methods and engineering*, **11**, 1977, 797 – 815.
- [21]. G. Lague and R. Baldur, Extended numerical integration method for triangular surfaces, *International journal on numerical methods and engineering*, **11**, 1977, 388 – 392.
- [22]. C. T. Reddy, Improved three point integration schemes for triangular finite elements, *International journal on numerical methods and engineering*, **12**, 1978, 1890 – 1896.
- [23]. C. T. Reddy and D. J. Shippy, Alternative integration formulae for triangular finite elements, *International journal for numerical methods in engineering*, **17**, 1981, 133 – 139.

[24]. H. T. Rathod and M. S. Karim, An explicit integration scheme based on recursion for curved triangular finite elements, *Computer structure*, **80**, 2002, 43 – 76.  
 [25]. M. Abramowitz and I. A. Stegun, *Handbook of mathematical functions*, (Dover, 1974).  
 [26]. S. Wandzura and H. Xiao, Symmetric quadrature rules on a triangle, *Computers and mathematics with applications*, **45**, 2003, 1829 – 1840.

Table-1: Computed Gauss points (u, v) and weights G for n×n-point formula (GQSTS):

n	u	v	G
3	-0.7745966692414834E+00	-0.8000000000000000E+00	0.2738575106854138E+00
	-0.7745966692414834E+00	-0.1127016653792583E+00	0.4381720170966623E+00
	-0.7745966692414834E+00	0.5745966692414834E+00	0.2738575106854138E+00
	0.0000000000000000E+00	-0.8872983346207417E+00	0.2469135802469134E+00
	0.0000000000000000E+00	-0.5000000000000000E+00	0.3950617283950617E+00
	0.0000000000000000E+00	-0.1127016653792583E+00	0.2469135802469134E+00
	0.7745966692414834E+00	-0.9745966692414834E+00	0.3478446462322783E-01
	0.7745966692414834E+00	-0.8872983346207417E+00	0.5565514339716456E-01
	0.7745966692414834E+00	-0.8000000000000000E+00	0.3478446462322783E-01
7	-0.9491079123427585E+00	-0.9504029146358146E+00	0.1633971902233038E-01
	-0.9491079123427585E+00	-0.7481081943789639E+00	0.3529604741916732E-01
	-0.9491079123427585E+00	-0.4209640416964355E+00	0.4818316692765079E-01
	-0.9491079123427585E+00	-0.2544604382862081E-01	0.5274230535088134E-01
	-0.9491079123427585E+00	0.3700719540391937E+00	0.4818316692765079E-01
	-0.9491079123427585E+00	0.6972161067217224E+00	0.3529604741916732E-01
	-0.9491079123427585E+00	0.8995108269785730E+00	0.1633971902233038E-01
	-0.7415311855993945E+00	-0.9556849211223279E+00	0.3153707751100920E-01
	-0.7415311855993945E+00	-0.7749342496082218E+00	0.6812443847836622E-01
	-0.7415311855993945E+00	-0.4826304010243249E+00	0.9299769892288504E-01
	-0.7415311855993945E+00	-0.1292344072003028E+00	0.1017972322343420E+00
	-0.7415311855993945E+00	0.2241615866237192E+00	0.9299769892288504E-01
	-0.7415311855993945E+00	0.5164654352076163E+00	0.6812443847836622E-01
	-0.7415311855993945E+00	0.6972161067217224E+00	0.3153707751100920E-01
	-0.4058451513773972E+00	-0.9642268026617967E+00	0.3475337161903307E-01
	-0.4058451513773972E+00	-0.8183164352463221E+00	0.7507207749196586E-01
	-0.4058451513773972E+00	-0.5823551434482712E+00	0.1024820257759689E+00
	-0.4058451513773972E+00	-0.2970774243113015E+00	0.1121789753788726E+00
	-0.4058451513773972E+00	-0.1179970517433171E-01	0.1024820257759689E+00
	-0.4058451513773972E+00	0.2241615866237192E+00	0.7507207749196586E-01
	-0.4058451513773972E+00	0.3700719540391937E+00	0.3475337161903307E-01
	0.0000000000000000E+00	-0.9745539561713792E+00	0.2705971537896379E-01
	0.0000000000000000E+00	-0.8707655927996972E+00	0.5845271854796313E-01
	0.0000000000000000E+00	-0.7029225756886985E+00	0.7979468810555958E-01
	0.0000000000000000E+00	-0.5000000000000000E+00	0.8734493960849647E-01
	0.0000000000000000E+00	-0.2970774243113015E+00	0.7979468810555958E-01
	0.0000000000000000E+00	-0.1292344072003028E+00	0.5845271854796313E-01
	0.0000000000000000E+00	-0.2544604382862081E-01	0.2705971537896379E-01
	0.4058451513773972E+00	-0.9848811096809618E+00	0.1468787955288007E-01
	0.4058451513773972E+00	-0.9232147503530723E+00	0.3172784626693875E-01
	0.4058451513773972E+00	-0.8234900079291260E+00	0.4331216169277282E-01
	0.4058451513773972E+00	-0.7029225756886985E+00	0.4741040083224655E-01
	0.4058451513773972E+00	-0.5823551434482712E+00	0.4331216169277282E-01
	0.4058451513773972E+00	-0.4826304010243249E+00	0.3172784626693875E-01
	0.4058451513773972E+00	-0.4209640416964355E+00	0.1468787955288007E-01
	0.7415311855993945E+00	-0.9934229912204305E+00	0.4680565643230238E-02
	0.7415311855993945E+00	-0.9665969359911727E+00	0.1011066754980332E-01
	0.7415311855993945E+00	-0.9232147503530723E+00	0.1380222483601957E-01
	0.7415311855993945E+00	-0.8707655927996972E+00	0.1510820486158432E-01
	0.7415311855993945E+00	-0.8183164352463221E+00	0.1380222483601957E-01
	0.7415311855993945E+00	-0.7749342496082218E+00	0.1011066754980332E-01
	0.7415311855993945E+00	-0.7481081943789639E+00	0.4680565643230238E-02
0.9491079123427585E+00	-0.9987049977069439E+00	0.4266374414229460E-03	
0.9491079123427585E+00	-0.9934229912204305E+00	0.9215957350721234E-03	

0.9491079123427585E+00	-0.9848811096809618E+00	0.1258084244262349E-02
0.9491079123427585E+00	-0.9745539561713792E+00	0.1377125407046231E-02
0.9491079123427585E+00	-0.9642268026617967E+00	0.1258084244262349E-02
0.9491079123427585E+00	-0.9556849211223279E+00	0.9215957350721234E-03
0.9491079123427585E+00	-0.9504029146358146E+00	0.4266374414229460E-03

Table-2: Computed Gauss points  $(p, q)$  and weights  $L$  for  $\frac{m(m+1)}{2} - 1$  point formula (GQSTM):

points	p	q	L
(m =5) 14 point	-0.8611363115940526E+00	-0.9126939522585497E+00	0.7669385858827389E-01
	-0.8611363115940526E+00	-0.5705142370613163E+00	0.1549333650457879E+00
	-0.8611363115940526E+00	-0.6943184420297377E-01	0.1841508361811174E+00
	-0.8611363115940526E+00	0.4316505486553688E+00	0.1549333650457879E+00
	-0.8611363115940526E+00	0.7738302638526020E+00	0.7669385858827389E-01
	-0.3399810435848563E+00	-0.9069626449468782E+00	0.1519885905918008E+00
	-0.3399810435848563E+00	-0.5577935549985240E+00	0.2849424819989606E+00
	-0.3399810435848563E+00	-0.1022254014166197E+00	0.2849424819989606E+00
	-0.3399810435848563E+00	0.2469436885317342E+00	0.1519885905918008E+00
	0.3399810435848563E+00	-0.9256147644301332E+00	0.1195633790398805E+00
	0.3399810435848563E+00	-0.6699905217924281E+00	0.1913014064638090E+00
	0.3399810435848563E+00	-0.4143662791547230E+00	0.1195633790398805E+00
	0.8611363115940526E+00	-0.9706546497379454E+00	0.2415220341283323E-01
	0.8611363115940526E+00	-0.8904816618561072E+00	0.2415220341283323E-01
(m = 9) 44 point	-0.9602898564975363E+00	-0.9687924202367442E+00	0.8063933990653976E-02
	-0.9602898564975363E+00	-0.8392867214556355E+00	0.1792366417936781E-01
	-0.9602898564975363E+00	-0.6210479706453962E+00	0.2585743793853289E-01
	-0.9602898564975363E+00	-0.3376704201677756E+00	0.3099065107963482E-01
	-0.9602898564975363E+00	-0.1985507175123180E-01	0.3276589850173939E-01
	-0.9602898564975363E+00	0.2979602766653120E+00	0.3099065107963482E-01
	-0.9602898564975363E+00	0.5813378271429326E+00	0.2585743793853289E-01
	-0.9602898564975363E+00	0.7995765779531718E+00	0.1792366417936781E-01
	-0.9602898564975363E+00	0.9290822767342806E+00	0.8063933990653976E-02
	-0.7966664774136268E+00	-0.9643270581779194E+00	0.2022265498028209E-01
	-0.7966664774136268E+00	-0.8173387381173184E+00	0.4442556514902738E-01
	-0.7966664774136268E+00	-0.5737699931387190E+00	0.6266989030061808E-01
	-0.7966664774136268E+00	-0.266452197773301E+00	0.7245416447754392E-01
	-0.7966664774136268E+00	0.6311867519095693E-01	0.7245416447754392E-01
	-0.7966664774136268E+00	0.3704364705523460E+00	0.6266989030061808E-01
	-0.7966664774136268E+00	0.6140052155309454E+00	0.4442556514902738E-01
	-0.7966664774136268E+00	0.7609935355915465E+00	0.2022265498028209E-01
	-0.5255324099163290E+00	-0.9611812354352877E+00	0.3098378782547042E-01
	-0.5255324099163290E+00	-0.8028487233396140E+00	0.6692925642225962E-01
	-0.5255324099163290E+00	-0.5467987609586445E+00	0.9136613786352844E-01
	-0.5255324099163290E+00	-0.2372337950418355E+00	0.1000112912703450E+00
	-0.5255324099163290E+00	0.7233117087497343E-01	0.9136613786352844E-01
	-0.5255324099163290E+00	0.3283811332559430E+00	0.6692925642225962E-01
	-0.5255324099163290E+00	0.4867136453516165E+00	0.3098378782547042E-01
	-0.1834346424956498E+00	-0.9600410418417249E+00	0.3676731142740568E-01
	-0.1834346424956498E+00	-0.7995317256959110E+00	0.7742169799018377E-01
	-0.1834346424956498E+00	-0.5494777843396608E+00	0.1004172673430814E+00
	-0.1834346424956498E+00	-0.2670875731646896E+00	0.1004172673430814E+00
	-0.1834346424956498E+00	-0.1703363180843920E-01	0.7742169799018377E-01
	-0.1834346424956498E+00	0.1434756843373746E+00	0.3676731142740568E-01
	0.1834346424956498E+00	-0.9616948561788959E+00	0.3508354238981680E-01
	0.1834346424956498E+00	-0.8115650136036089E+00	0.7087414012328669E-01
	0.1834346424956498E+00	-0.5917173212478248E+00	0.8423964820917546E-01
	0.1834346424956498E+00	-0.3718696288920410E+00	0.7087414012328669E-01
0.1834346424956498E+00	-0.2217397863167538E+00	0.3508354238981680E-01	
0.5255324099163290E+00	-0.9670568402059502E+00	0.2588799002096551E-01	
0.5255324099163290E+00	-0.8434211981700837E+00	0.4853382811049090E-01	

0.5255324099163290E+00	-0.6821112117462453E+00	0.4853382811049090E-01
0.5255324099163290E+00	-0.5584755697103789E+00	0.2588799002096551E-01
0.7966664774136268E+00	-0.9770839733770847E+00	0.1256042196994617E-01
0.7966664774136268E+00	-0.8983332387068135E+00	0.2009667515191388E-01
0.7966664774136268E+00	-0.8195825040365421E+00	0.1256042196994617E-01
0.9602898564975363E+00	-0.9916082592691212E+00	0.2009899851317608E-02
0.9602898564975363E+00	-0.9686815972284151E+00	0.2009899851317608E-02

Table-3: Calculated values of the integrals  $I_1, I_2, I_3$  and  $I_4$

Method	Gauss Point	Test example			
		$I_1$	$I_2$	$I_3$	$I_4$
GQT	7×7	0.4001498818	0.6606860757	0.8315681219	0.6938790083
	13×13	0.4000451564	0.6637058258	0.8501738309	0.7238717079
IOST	7×7	0.4000006725	0.6664256210	0.8755247309	0.7178753433
	10×10	0.4000001234	0.6665789279	0.8783900003	0.7180745324
GQSTS	7×7	0.4000037499	0.6659893974	0.8696444431	0.7184323939
	10×10	0.4000006929	0.6664193645	0.8753981854	0.7182531970
GQSTM	54	0.4000009417	0.6663718426	0.8742865042	0.7175459725
	90	0.4000002469	0.6665339400	0.8772635782	0.7180958214
Exact value		0.4	0.6666667	0.881373587	0.71828183

Table 4: Computed results of Example - 1

Method	TP	Value of $II^{r,0}$					
		r = 2		r = 4		r = 6	
GQT	7×7	0.7288889289		0.3733333349		0.2209523767	
	13×13	0.7883351445		0.4327795803		0.2803986370	
IOST	5×5	0.8536515995		0.4980960513		0.3457150911	
	6×6	0.8636423911		0.5080868305		0.3557058624	
	7×7	0.8699174628		0.5143619067		0.3619809556	
	8×8	0.8741142348		0.5185586841		0.3661777177	
	9×9	0.8770583374		0.5215027742		0.3691218199	
	10×10	0.8792029273		0.5236473748		0.3712664246	
GQSTS		Formula 1	Formula 2	Formula 1	Formula 2	Formula 1	Formula 2
	5×5	0.8888889003	0.8189709	0.5333333421	0.4634153	0.3809523939	0.3110344
	6×6	0.8888888979	0.8386859	0.5333333394	0.4831303	0.3809523751	0.3307493
	7×7	0.8888889008	0.8511113	0.5333333320	0.4955557	0.3809523895	0.3431748
	8×8	0.8888888889	0.8594404	0.5333333473	0.5038848	0.3809523887	0.3515039
	9×8	0.8888888960	0.8652928	0.5333333366	0.5097372	0.3809523945	0.3573563
GQSTM	10×10	0.8888888916	0.8695606	0.5333333260	0.5140051	0.3809523860	0.3616241
	14	0.8888888885	0.7979759	0.5333333288	0.4424203	0.3809523780	0.2900394
	44	0.8888888823	0.8620172	0.5333333369	0.5064616	0.3809523803	0.3540807
	77	0.8888888823	0.8738937	0.5333333366	0.5183381	0.3809523815	0.3659572
	90	0.8888888888	0.8761310	0.5333333301	0.5205754	0.380952300	0.3699649
		0.8888888		0.5333333		0.3809523	

Table 5: Computed results of Example - 2.

Method	TP	Value of $II^{0,r}$		
		r = 2	r = 4	r = 6
GQT	7×7	0.7288889289	0.3733333349	0.2209523767
	13×13	0.7883350849	0.4327795803	0.2803986370
IOST	5×5	0.8536515995	0.4980960513	0.3457150911
	6×6	0.8636423911	0.5080868305	0.3557058624
	7×7	0.8699174628	0.5143619067	0.3619809556

	8×8	0.8741142348		0.5185586841		0.3661777177	
	9×9	0.8770583374		0.5215027742		0.3691218199	
	10×10	0.8792029273		0.5236473748		0.3712664246	
GQSTS		Formula 1	Formula 2	Formula 1	Formula 2	Formula 1	Formula 2
	5×5	0.8189709	0.8888889003	0.4634153	0.5333333421	0.3110344	0.3809523939
	6×6	0.8386859	0.8888888979	0.4831303	0.5333333394	0.3307493	0.3809523751
	7×7	0.8511113	0.8888889008	0.4955558	0.5333333320	0.3431748	0.3809523895
	8×8	0.8594405	0.8888888889	0.5038849	0.5333333473	0.3515039	0.3809523887
	9×9	0.8652927	0.8888888960	0.5097372	0.5333333366	0.3573562	0.3809523945
	10×10	0.8695606	0.8888888916	0.5140051	0.5333333260	0.3616241	0.3809523860
GQSTM	14	0.7979759	0.8888888885	0.4424203	0.5333333288	0.2900394	0.3809523780
	44	0.8620172	0.8888888823	0.5064616	0.5333333369	0.3540807	0.3809523803
	77	0.8738937	0.8888888823	0.5183381	0.5333333366	0.3659572	0.3809523815
	90	0.8779014	0.8888888888	0.5223459	0.5333333301	0.3699649	0.3809523797
Exact Value	0.8888888		0.5333333		0.3809523		

Table 6: Computed results of Example -3, 4

Method	TP	Example -3		Example -4	
GQT	7×7	0.03669412062		0.02731705643	
	13×13	0.03688941523		0.02731722965	
IOST	5×5	0.03694724351		0.02731723353	
	6×6	0.03694789990		0.02731723339	
	7×7	0.03694799397		0.02731723359	
	8×8	0.03694800816		0.02731723343	
	9×9	0.03694800990		0.02731723344	
	10×10	0.03694801028		0.02731723331	
	GQSTS		Formula 1	Formula 1	Formula 2
5×5		0.0369262	0.0369568	0.0273172	0.0273172
6×6		0.0369429	0.0369500	0.0273172	0.0273172
7×7		0.0369467	0.0369485	0.0273172	0.0273172
8×8		0.0369477	0.0369481	0.0273172	0.0273172
9×9		0.0369479	0.0369480	0.0273172	0.0273172
GQSTM	10×10	0.0369479	0.0369480	0.0273172	0.0273172
	14	0.0368514	0.0369856	0.0273172	0.0273172
	44	0.0369477	0.0369481	0.0273172	0.0273172
	77	0.0369480	0.0369480	0.0273172	0.0273172
	90	0.0369480	0.0369480	0.0273172	0.0273172
Exact value	0.03694801040		0.02731723349		

Table 7: Error calculated in  $\int_{x=-1}^1 \int_{y=-1}^1 \sum_{i+j \leq N} x^i y^j dy dx$  using the new error formula.

Points	N	Error calculated using Eq (5.1)	Error using the proposed method
2×2	4	0.71111111	0.71111111
3×3	6	0.18285711	0.18285714
4×4	8	0.04643999	0.04643993
5×5	10	0.01172724	0.01172679

Table 8: Absolute error in  $\int_{x=0}^1 \int_{y=0}^{1-x} \sum_{i+j \leq N} x^i y^j dy dx$ ;  $N = 30$  for formula 1 and 2.

n	N	Formula 1	Formula 2
2	1	0.27755575615628913511E-15	0.27755575615628913511E-15
	2	0.13877787807814456755E-15	0.13877787807814456755E-15
3	3	0.83266726846886740532E-15	0.80491169285323849181E-15
	4	0.61062266354383609723E-15	0.61062266354383609723E-15
4	5	0.87617413324636572725E-13	0.87610474430732665496E-13
	6	0.56996074526693973894E-13	0.56996074526693973894E-13
6	9	0.80144224590128487762E-15	0.80144224590128487762E-15
	10	0.70429773124658368033E-15	0.70429773124658368033E-15
10	17	0.76934986159571394637E-14	0.76934986159571394637E-14
	18	0.65156213757688874466E-14	0.65164887375068758502E-14
16	29	0.12595827159067596313E-13	0.12595827159067596313E-13
	30	0.11583616010835129373E-13	0.11583616010835129373E-13