

# IS YOUR MESH REFINED ENOUGH? Estimating Discretization Error using GCI

Leonard E Schwer

Schwer Engineering & Consulting Services, Windsor, CA, USA

## Summary:

The importance of estimating discretization error has been sadly neglected in the computational solid mechanics community. Quite the opposite is true in the computational fluid dynamics community where discretization error estimation is required to publish numerical results in many CFD journals, including the ASME *Fluids Engineering Journal*. It is suggested that if more solid mechanics types would estimate their discretization error, they would be surprised at the magnitude of the error in their solution attributable to meshing.

In this paper the Grid Convergence Index is presented and applied to the simulation of a simple cantilever beam under a uniform distributed load. Estimates of the error in the beam end deflection are provided. To illustrate the effectiveness of CGI, the analytical solution for the cantilever beam is provided. However, an analytical solution is not required by GCI to estimate the discretization error.

## Keywords:

Discretization error, mesh refinement, Grid Convergence Index (GCI), Verification.

## 1 Introduction

Numerical methods in engineering are used to solve coupled partial differential equations, which describe the key physics, using discretization of the time and space variables. LS-DYNA users are familiar with criteria for time step stability, e.g. Courant criteria, which provide the minimum time step size required for a stable solution. But what is the corresponding criterion for spatial discretization to obtain a converged solution?

All discrete methods introduce discretization error, i.e. error in the solution due to the choice of finite time and space resolution. Even if your numerical solution agrees with an experiment, there is still discretization error in your solution. So then the question is, what other error is present in the simulation that is cancelling the discretization error, and how large is that error?

Traditional methods of estimating discretization error, such as Richardson Extrapolation<sup>1</sup>, rely on mesh doubling, i.e. the mesh size changes by a factor of two. Recent development of the Grid Convergence Index (GCI) [Roache, 1998] provides a discretization error estimate even when the successive mesh refinements are not integer multiples, i.e. doubling the mesh is not required.

The Grid Convergence Index is stated as an error percentage and provides a confidence bound on the estimated error band within which the numerically converged solution will likely lie. GCI error estimates can be used with a minimum of two mesh solutions, but provide a better error estimate when used with three mesh solutions.

## 2 Rectangular Constant Cross Section Cantilever Beam

To motivate the presentation, and application, of the Grid Convergence Index, a uniform rectangular solid cross section cantilever beam with a uniform distributed load over the length is solved using LS-DYNA. The problem is (quasi)static and use is made of the implicit solver. Solutions are presented for several forms of user defined beam cross section integration.

The end displacement of this beam is known to every undergraduate mechanics student:

$$w = \frac{qL^4}{24EI} \quad (1)$$

For a beam of length  $L = 2000$  with a uniform-solid-rectangular cross section of dimensions  $b = 200$  and  $h = 50$ , then  $I_u = \frac{bh^3}{12} = 2.083 \times 10^6$ , loaded by a uniform distributed load of magnitude  $q = 0.5$  /length and using  $E = 68.9 \times 10^3$ , the displacement from Equation (1) is  $w = 6.9666$ .

Figure 1 shows an LS-PrePost view of the six element, along the length, discretization of the cantilever beam with a uniform distributed load. Note: a free end moment is required to provide the work equivalent discrete loading corresponding to a distributed load, viz.

$$M_{end} = \frac{q(dL)(dL/2)}{12} \quad (2)$$

where  $dL$  is the beam element length, and all the other work equivalent moments cancel each other as internal moments.

<sup>1</sup> See the Wikipedia reference Richardson Extrapolation:  
[http://en.wikipedia.org/wiki/Richardson\\_extrapolation](http://en.wikipedia.org/wiki/Richardson_extrapolation)

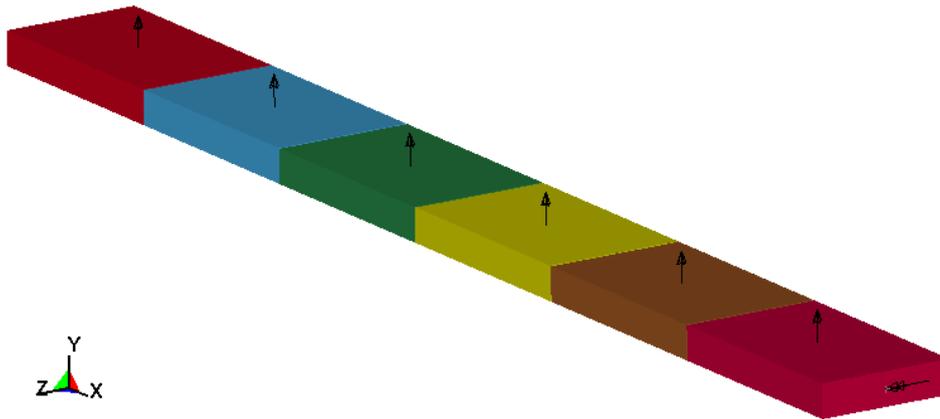


Figure 1 Six beam element mesh of a uniformly loaded rectangular cross section beam.

## 2.1 Numerical Results

The end deflection of the beam is solved for using the Belytschko-Schwer beam element with two types of cross section integration: resultant, ELFORM=2, and user/Gauss, ELFORM=4. Table 1 summarizes the cantilever beam end deflections results; Table 2 provides the results expressed using relative error.

Table 1 Uniform cantilever beam summary of end deflections.

NELE	$h$	User Integration		Gauss	Resultant
		K=0 (9 pts)	K=4 (49 pts)	2X2	
2	1000	8.16406	7.40813	7.25694	7.25810
4	500	7.91909	7.18584	7.03919	7.04032
6	333.33	7.87373	7.14468	6.99887	6.99990
8	250	7.85786	7.13028	6.98476	6.98585
Analytical		6.96662			

Table 2 Uniform cantilever beam summary of end deflection relative errors.

NELE	$h$	User Integration		Gauss	Resultant
		K=0 (9 pts)	K=4 (49 pts)	2X2	
2	1000	17.2%	6.3%	4.2%	4.2%
4	500	13.7%	3.1%	1.0%	1.1%
6	333.33	13.0%	2.6%	0.5%	0.5%
8	250	12.8%	2.3%	0.3%	0.3%

The first two columns of end displacements in Table 1 are the results obtained using the default cross section integration<sup>2</sup> for a trapezoidal cross section (ICST=7), see Figure 2, with dimensions corresponding to the rectangular cross section for the present beam, i.e.  $D1 = D4 = 50$  and  $D3 = 200$ . As shown in Figure 2, the number of cross section integration points in each direction is  $n = (K + 3)$  and thus the total number of integrations points is  $n^2$ .

<sup>2</sup> The default cross sections in LS-DYNA all use a trapezoidal rule for cross section integration.

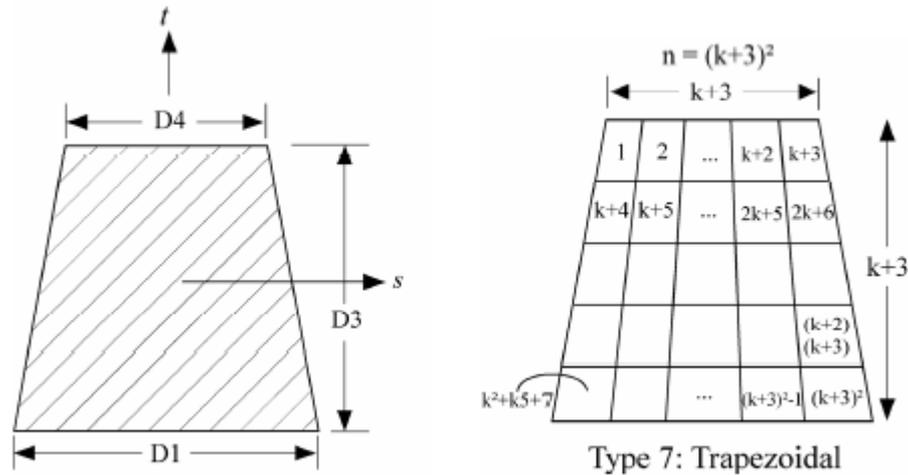


Figure 2 LS-DYNA trapezoidal standard cross section integration; figure from LS-DYNA User Manual.

The end displacements in the column labeled Gauss 2x2 of Table 1 are obtained by specifying QR=2 on the \*SECTION\_BEAM keyword; this is the default integration for solid rectangular cross sections. The user defined and Gauss integrations used the Belytschko-Schwer full cross section integration (ELFORM=4) beam formulation. The last column in Table 1, labeled Resultant, uses the resultant form (ELFORM=2) of the Belytschko-Schwer beam with the cross section properties:  $A=1.0E4$ ,  $ISS=3.333E7$ ,  $ITT=2.083E6$ ,  $IRR=1.417E8$  and  $SA=1.0E9$ ; where the shear area, SA, is set to a large number to minimize shear effects and provide a better comparison with the analytical solution which ignores shear effects.

Focusing first on the most refined mesh, i.e. Mesh-8 with eight elements along the length NELE=8, results in Table 1, the end displacement is seen to vary from 7.85786 for the user defined integration with K=0, to 6.98476 for Gauss 2x2 integration<sup>3</sup>. Comparing these end deflections with the analytical solution provides relative errors of 12.8% to 0.3%, respectively, when eight beam elements are used to represent a simple rectangular cross section cantilever beam; see Table 2. While the 0.3% relative error from the Gauss 2x2 integration is not a concern, the 12.8% relative error from the user integration rule is a concern, as this is the form of cross section integration used for all default beam element cross section integrations, with the exception of the simple rectangular and circular cross sections that can also be integrated using Gauss integration. Increasing the number of integration points via K=4 yields a smaller relative error of 2.3%, but at the cost of increasing the number of stress evaluation from 9 with K=0 to 49 with K=4, a factor of more than 5, with a corresponding increase in CPU time.

## 2.2 Traditional Error Convergence Analysis

Most error analyses start by assuming a relation between the exact solution,  $f_{exact}$ , and the approximate numerical solution,  $f(h)$ , as given in the following equation:

$$f_{exact} = f(h) + Ah^p + HOT \quad (3)$$

<sup>3</sup> The Gauss 2x2 and Resultant end deflections agree to three significant figures in all cases. Only the Gauss 2x2 integration results will be used in the remainder of this section. The Resultant end deflections served as a check on both the user defined and Gauss 2x2 integration. Also, Gauss 3x3 results were generated and were found to be identical to the Gauss 2x2 results.

where  $h$  is a measure of the mesh discretization,  $A$  is a constant,  $p$  is the order (rate) of convergence, and  $HOT$  are Higher Order Terms involving powers of  $h$ . If the meshes to be considered are sufficiently refined, i.e. the results are in the asymptotic region and not changing rapidly, the  $HOT$  can be neglected. Then the discretization error can be approximated as

$$E(h) = f_{exact} - f(h) \approx Ah^p \tag{4}$$

The order of convergence can be found by plotting the discretization error,  $E(h)$ , as a function of the mesh discretization parameter  $h$ . This is most easily done by taking the logarithm of both sides of Equation (4) and the corresponding form provides  $p$  as the slope of the resulting line plotted in log-log space, viz.

$$\log E(h) = \log A + p \log h \tag{5}$$

### 2.3 Application of Traditional Analysis

Figure 3 shows a log-log plot of the cantilever beam end displacement error versus the mesh size for the three integration methods summarized in previously shown Table 1. Estimating the observed order of convergence from the slope of the last two points (left most pair), i.e. Meshes 6 & 8 the most refined meshes, provides the values listed in Table 3. Also listed in this table are the initial observed rates of convergence, i.e. from the slope of the first two points (right most pair), Meshes 2 & 4 the most coarse meshes.

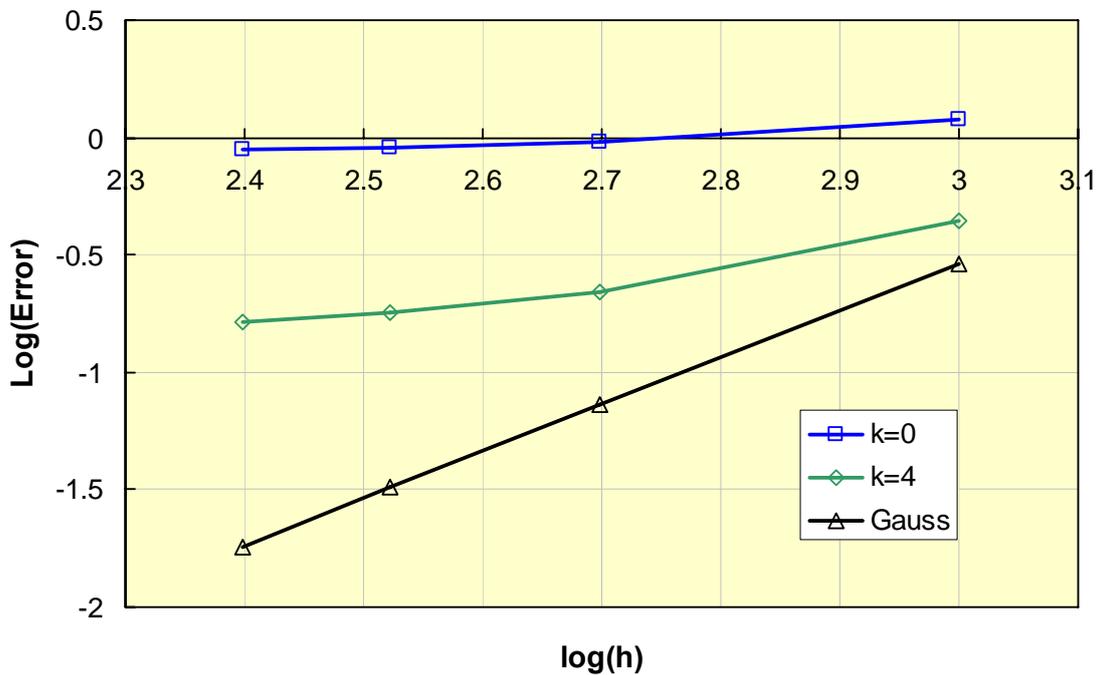


Figure 3 Cantilever beam error versus mesh size for three integration methods.

Table 3 Observed rates of convergence for three integration methods.

$p$	User Integration		Gauss
	K=0 (9 pts)	K=4 (49 pts)	2X2
Mesh 2 - 4	0.3302	1.0101	2.0002
Mesh 6 - 8	0.0614	0.2931	2.0000

The observed order of convergence values listed in Table 3 show the Gauss 2x2 integration method has the largest rate of convergence,  $p = 2$ , and maintains that rate over the entire range of mesh refinements. The two user defined (trapezoidal) integration methods show lower rates of convergence, and the rates decrease with an increasing mesh refinement.

## 2.4 Grid Convergence Index

The discretization error will be estimated using a method popular in the Computational Fluid Dynamics (CFD) community called the Grid Convergence Index (GCI), due to Roache [1994]. Readers new to the concept of discretization error estimation are encouraged to review the very nice summary provided on the web at

<http://www.grc.nasa.gov/WWW/wind/valid/tutorial/spatconv.html>

In the preceding traditional error convergence illustration, use was made of the exact (analytical) solution to determine the error, and then the observed rate of convergence was determined graphically. In most practical problems the exact solution is unknown. The question then is how to estimate the discretization error and observed rate of convergence?

Consider again Equation (4) where there are three unknowns: the exact solution  $f_{exact}$ , the constant  $A$ , and the observed order of convergence  $p$ . It should be possible to solve for these three unknowns by performing three mesh refinements, i.e. three different values of  $h$ . Determining these three unknowns is the basis of the Grid Convergence Index (GCI).

### 2.4.1 Constant Grid Refinement Ratio

For the special case where the three meshes are constructed with a constant grid refinement ratio,  $r$ , e.g.  $r = h_3 / h_2 = h_2 / h_1 = const$  where  $h_1 < h_2 < h_3$ , then Equation (5) can be written three times and solved in closed form for the observed convergence rate:

$$p = \frac{\log\left(\frac{f_3 - f_2}{f_2 - f_1}\right)}{\log r} \quad (6)$$

Once the observed order-of-convergence is known, an estimate of the error between the fine grid solution and the unknown exact solution can be estimated by writing Equation (4) twice for each of the grids  $h_1$  and  $h_2$ , eliminating the unknown constant  $A$ , and solving for  $f_{exact}$ , viz.

$$f_{exact} \approx f_1 - \frac{f_2 - f_1}{r_{21}^p - 1} \quad (7)$$

or

$$\frac{f_{exact} - f_1}{f_1} \approx \frac{f_1 - f_2}{r_{21}^p - 1} \frac{1}{f_1} = \frac{e_{21}}{r_{21}^p - 1} \quad (8)$$

where the relative error between the two finest grids is given by

$$e_{21} = \left| \frac{f_2 - f_1}{f_1} \right| \quad (9)$$

Equation (8) provides an estimate of the *exact relative error* with respect to the fine grid solution. This estimate is based on the two finest grid solutions,  $f_1$  and  $f_2$ , the mesh refinement ratio,  $r_{21}$ , and observed order of convergence,  $p$ .

#### 2.4.2 Nonuniform Grid Refinement Ratio

Requiring a constant mesh refinement ratio, over three meshes, can become a computational burden, especially for three dimensional problems. Practical experience [Roache, 1998] has shown that grid refinement ratios need only be greater than 1.3, e.g.  $r = h_2 / h_1 > 1.3$  to obtain good results using GCI.

Relaxing the requirement that the mesh refinement ratios be constant, Equation (5) can be written three times, then solving this system of equations results in a transcendental equation for the observed convergence:

$$p = \frac{|\ln |f_{32} / f_{21}| + q(p)|}{\ln r_{21}}$$

$$q(p) = \ln \left( \frac{r_{21}^p - s}{r_{32}^p - s} \right) \quad (10)$$

$$s = \text{sign}(f_{32} / f_{21})$$

where  $f_{ij} = f_i - f_j$  and  $r_{ij} = h_i / h_j$ . Note: the first of Equations (10) is to be solved iteratively using an initial guess for  $q(p) = 0$ ; the first of Equations (10) with  $q(p) = 0$  is identical to the constant grid refinement ratio observed convergence rate, see Equation (6).

The Grid Convergence Index (GCI) provides an *estimate* of the amount of discretization error in the finest grid solution relative to the *converged numerical solution*<sup>4</sup>. The GCI is given by

$$GCI_{21} = F_s \frac{e_{21}}{r_{21}^p - 1} \quad (11)$$

Where  $F_s$  is a 'safety factor' multiplying the relative error term, defined previously by Equation (9). The safety factor,  $F_s = 1.25$ , is based on experience applying GCI in many situations, Roache [1994]. The safety factor should be thought of as representing a 95% confidence bound on the estimated relative error. Example: say  $f_1 = 5$  with a GCI=2%, then the following statement can be made:

*The converged numerical solution lies in the interval [4.02,5.10], i.e.  $[f_1(1 - GCI), f_1(1 + GCI)]$ , with a 95% confidence level.*

Finally, based upon the two finest mesh solutions, and the estimate of the observed convergence rate, an extrapolation of the numerical solution is possible:

<sup>4</sup> Note the discretization error is with respect to the convergence numerical solution, as the exact solution is generally unknown.

$$f_{21}^* = \frac{r_{21}^p f_1 - f_2}{r_{21}^p - 1} \quad (12)$$

The extrapolated solution,  $f_{21}^*$ , provides a useful estimate of the numerically converged solution.

## 2.5 Application of GCI

The results of applying the above GCI method to the three coarsest grid cantilever beam end deflections, listed precisely in Table 1, are provided in Table 4. There are two striking similarities among the results for the three integration methods:

- The observed rates of convergence are identical, i.e.  $p = 2.0002$
- The GCI values are identical,  $GCI_{21} = 0.00576$

Recall the observed rates of convergence differed markedly when the traditional error convergence, using the analytical solution, was applied; see Table 3. So why then are the estimated rates of convergence identical? The key to the answer lies in the extrapolated solution values  $f_{21}^*$ .

Table 4 Estimated rates of convergence and GCI for three integration methods.

	User Integration		Gauss
	K=0 (9 pts)	K=4 (49 pts)	2X2
$P$	2.0002	2.0002	2.0002
$f_{21}^*$	7.83745	7.11176	6.9662
$GCI_{21}$	0.00576	0.00576	0.00576
95% Interval	[7.828,7.919]	[7.104,7.186]	[6.959,7.039]
Mesh 8	7.85786	7.13028	6.98476

When the traditional error convergence is applied, the exact solution,  $f_{exact}$ , is required. For the estimated rates of convergence, obtained from GCI, the exact solution is not required, indeed it is typically unknown. However, the extrapolated solution serves as an estimate of the converged numerical solution. Thus, the conclusion from the results in Table 4 is the three integration methods are converging to *different numerical solutions*. The two trapezoidal cross section integration results are converging, however due to the limitations of trapezoidal integration, to cantilever beam problems other than the defined problem; an appendix explains the limitations of trapezoidal cross section integration.

As already demonstrated, with the traditional error convergence, and as indicated in Table 4, the Gauss integration results are converging to the analytical solution  $f_{exact} = 6.96662$ .

Finally, looking at the 95% confidence level intervals, obtained from the Mesh-6 solutions, and the corresponding GCI values, it is obvious that the tip displacement results for next mesh refinement, i.e. Mesh-8, do indeed lie in the 95% confidence intervals.

## 3 Conclusions

While there is a lot to be said about using engineering judgment to design computational solid mechanics meshes, this judgment is derived mostly from experience with very few rules-of-thumb. In turn, both experience and rules-of-thumb are established by trial-and-error, with an emphasis on the latter, i.e. *error*. Analysts lacking experience, and those with experience seeking confirmation of their engineering judgment, are urged to embrace the method and concepts comprising the Grid Convergence Index.

Those familiar with the rapidly growing emphasis on Verification & Validation in computational mechanics, will recognize the Grid Convergence Index as a key step in verification, where the question asked is "How much discretization error is in my solution?" which is referred to in the V&V literature as "solution" or "calculation" verification. As management and sponsoring agencies become more aware of the importance of V&V, more analysts will be required to adopt V&V procedures and guidelines. Among all the various V&V topics, the most easily addressed is discretization error via mesh refinement. Modern pre-processor are capable of generating variations on a basic mesh with the change of a few parameters, if the basic mesh is designed with this goal in mind.

#### 4 References

Roache, P. J., (1993), "A Method for Uniform Reporting of Grid Refinement Studies," Proceedings of Quantification of Uncertainty in Computation Fluid Dynamics, Edited by Celik, et al., ASME Fluids Engineering Division Spring Meeting, Washington D.C., June 230-240, ASME Publ. No. FED-Volume 158.

Roache, P.J., (1994), "Perspective: A Method for Uniform Reporting of Grid Refinement Studies," ASME Journal of Fluids Engineering, Vol. 116, September.

Roache, P. J. (1998), *Verification and Validation in Computational Science and Engineering*, Hermosa Publishers, Albuquerque.

Schwer, L., (2003), "LS-DYNA Beam Elements: Default and User Defined Cross Section Integration," Proceedings of the 4<sup>th</sup> European LS-DYNA Users Conference, Ulm, Germany.  
[http://www.dynalook.com/documents/4th\\_European\\_Is-dyna/LS-DYNA\\_ULM\\_H-I-31.pdf](http://www.dynalook.com/documents/4th_European_Is-dyna/LS-DYNA_ULM_H-I-31.pdf)

#### 5 Appendix: Limitations of Trapezoidal Cross Section Integration

As shown previously in Figure 2, trapezoidal cross section integration is based upon dividing the cross section into small rectangular regions, each surrounding an integration point. The integrals over the cross section, e.g. for the moments, are then approximated by summing the small areas multiplied by their distance from the local cross sectional axis. For example the bending moments are approximated by

$$M_y = \int \sigma z dA \approx \sum \sigma_i z_i A_i$$

If the stress  $\sigma$  is replaced by the coordinate  $z$  in the above integral, the approximation then provides an estimate of the cross section moment of inertia, viz.

$$I_{yy} = \int z^2 dA \approx \sum z_i^2 A_i$$

When compared to the exact cross section integration, see Schwer [2003], the above trapezoidal cross section integration approximation suffers from two limitations:

1. The contribution of each small area's local moment of inertia is ignored,
2. Integration points on the local cross sectional axis are ignored, e.g.  $z = 0$ .

The second limitation is significant, if there are an odd number of integration points, since there will be integration points on the local cross sectional axis; see Figure 4 which shows the locations of integration areas (points) for 3x3 and 4x4 trapezoidal cross section integration of a rectangular shape.

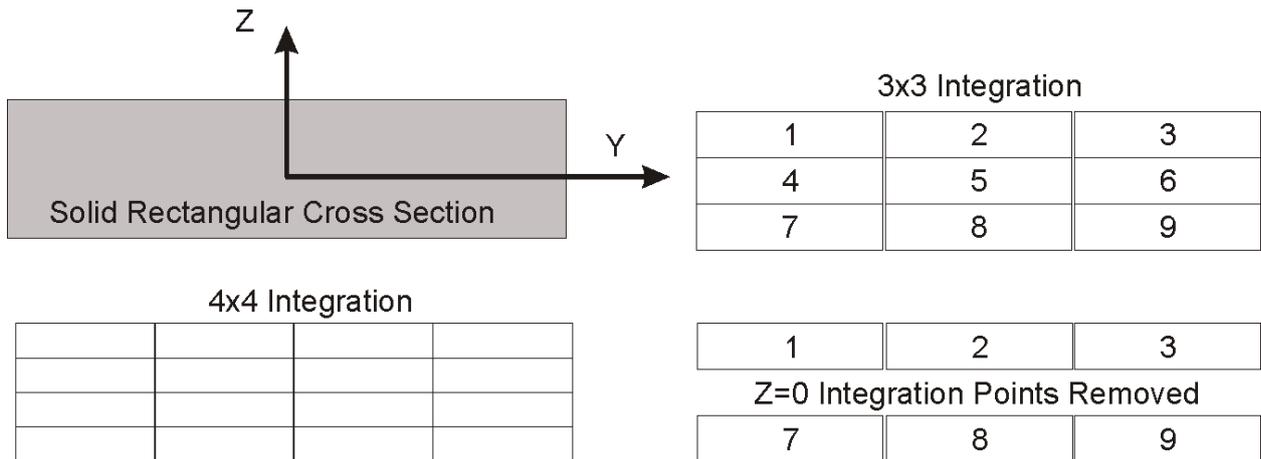


Figure 4 Illustration of 3x3, and 4x4, trapezoidal cross section integration.

Referring to Figure 4, for the case of user integration with  $K=0$ , i.e.  $NPTS=3 \times 3=9$ , there are three integration points, in each direction, with a zero coordinate. The beam cross section then consists of two strips of width 200 and thickness 16.66 ( $=50/3$ ) separated by a distance of 16.66 ( $=50/3$ ), for the calculation of the cross sectional moment of inertia of interest. For this case the cross section moment of inertia from the trapezoidal rule yields:

$$I \approx 2Ad^2 = 2(2000 \times 16.66)(16.66)^2 = 1.852 \times 10^6$$

The exact cross section moment of inertia is  $2.083 \times 10^6$ , thus the approximate cross sectional moment of inertia is about 11% smaller. Using the *approximate* cross sectional moment of inertia in the exact solution, Equation (1), provides an end deflection of 7.83745 which, to six significant figures, is exactly the same as the extrapolated value provided in Table 4, i.e.  $f_{21}^* = 7.83745$ .