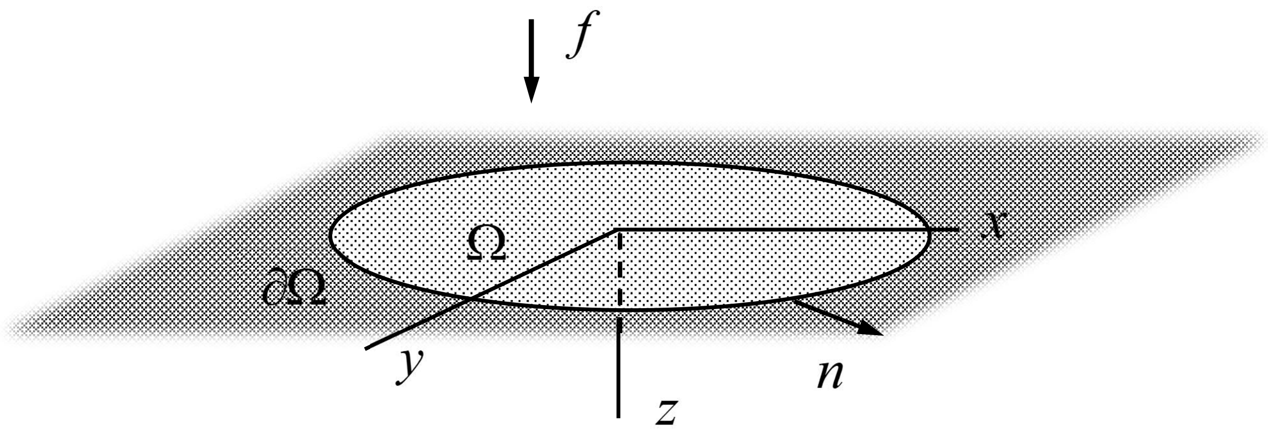
6 FEM FOR MEMBRANE MODEL

MEMBRANE EQUATIONS

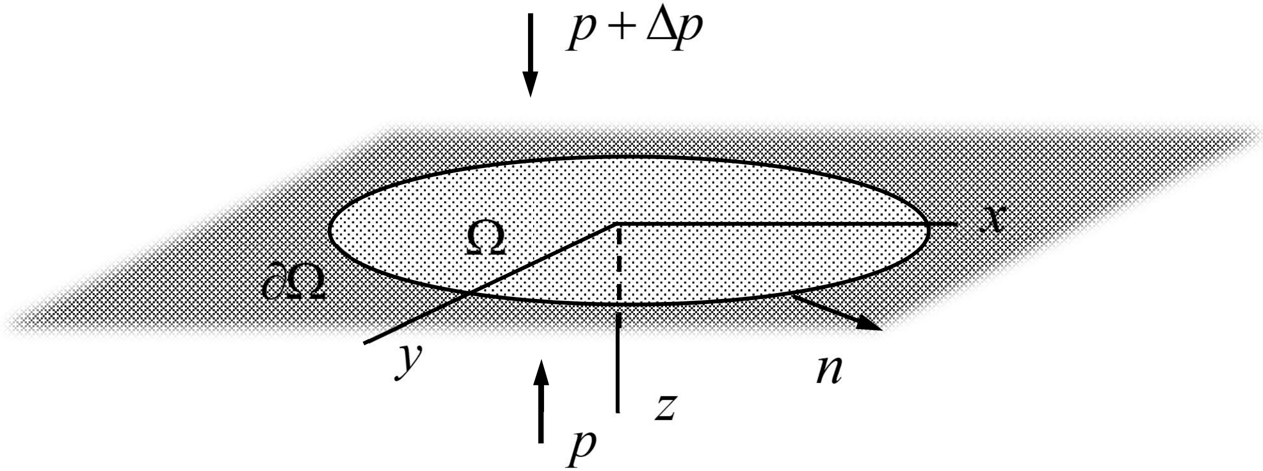


Equation of motion $S'(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2}) + f' = m'\frac{\partial^2 w}{\partial t^2} \quad (x, y) \in \Omega \quad t > 0,$

Boundary conditions $w = \underline{w}$ or $S'(n_x \frac{\partial w}{\partial x} + n_y \frac{\partial w}{\partial y}) = F' \quad (x, y) \in \partial\Omega \ t > 0,$

Initial conditions $w = g$ and $\frac{\partial w}{\partial t} = h \quad (x, y) \in \Omega \quad t = 0.$

EXAMPLE A circular membrane of radius R, fixed edges, and constant tightening S' (force per unit length) is loaded by pressure $p+\Delta p$ acting on the upper surface and p acting on the lower surface. Find the transverse displacement assuming that the solution is rotation symmetric.



Answer $w(r)=\dfrac{\Delta p}{S'}\dfrac{1}{4}(R^2-r^2)$

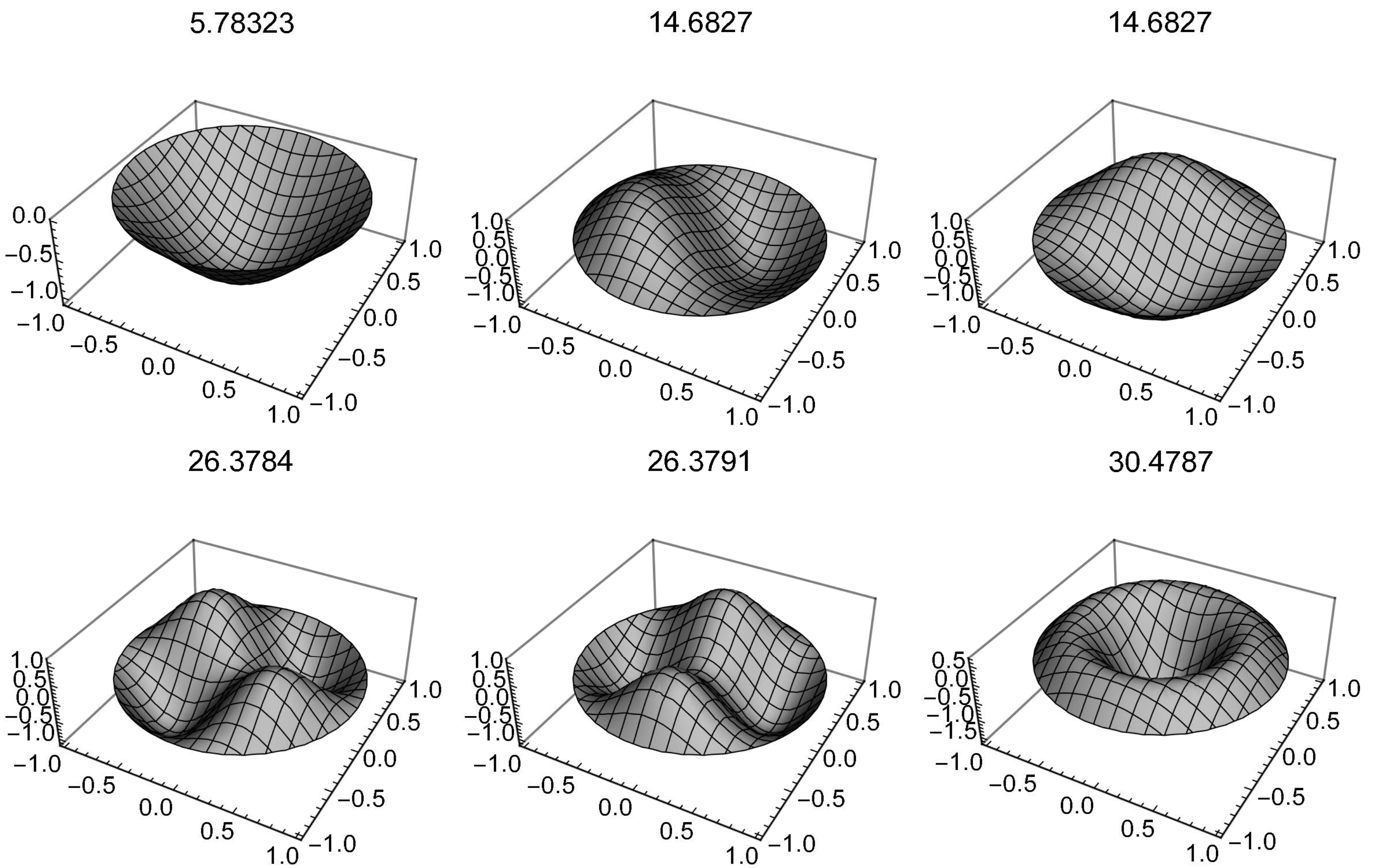
According to the problem description, solution is rotational symmetric so it can depend only on the distance from the centerpoint $r=\sqrt{x^2+y^2}$. Then, membrane equilibrium equation simplifies to the ordinary differential equation (see: Laplace operator in polar coordinate system)

$$S'\frac{1}{r}\frac{d}{dr}(r\frac{dw}{dr})+\Delta p=0 \quad \Leftrightarrow \quad w(r)=-\frac{\Delta p}{S'}\frac{1}{4}r^2+a\ln r+b.$$

Solution should be bounded at the origin so $a=0$. The value of the second parameter follows from the boundary condition $w(R)=0$. Therefore

$$w(r)=\frac{\Delta p}{S'}\frac{1}{4}(R^2-r^2) \ .$$

VIBRATION PATTERNS OF DRUMHEAD

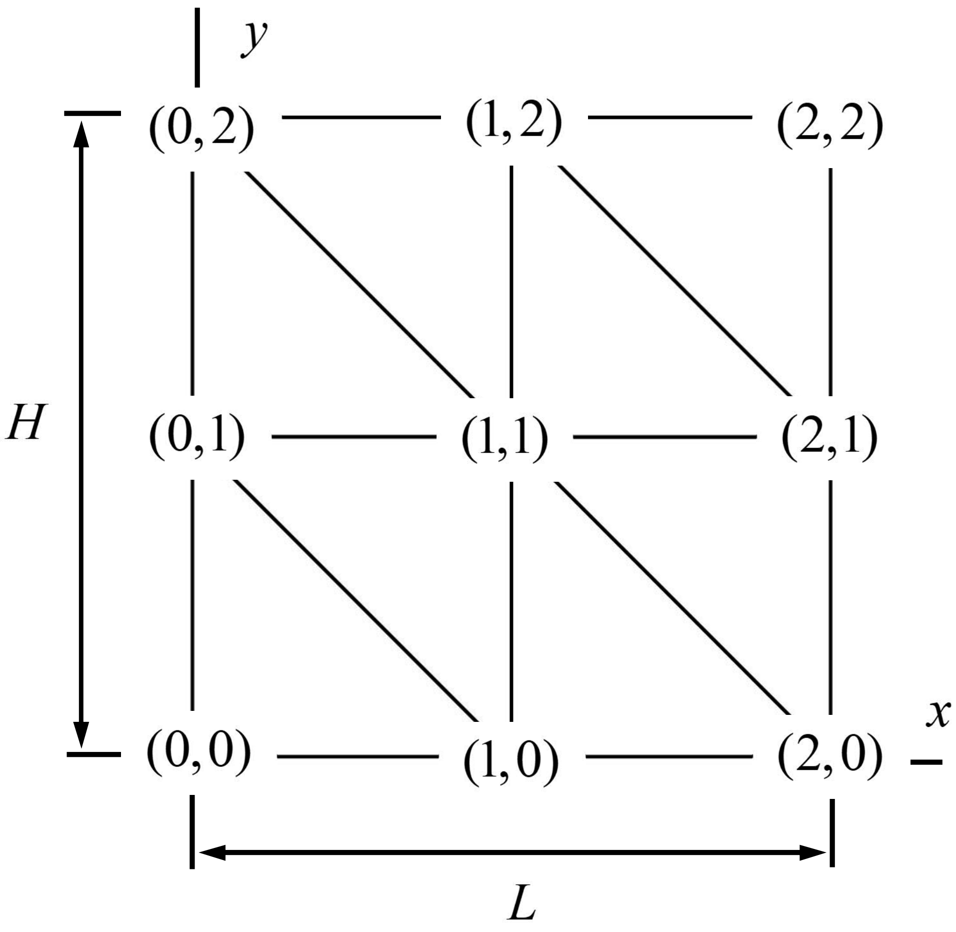


PRINCIPLE OF VIRTUAL WORK

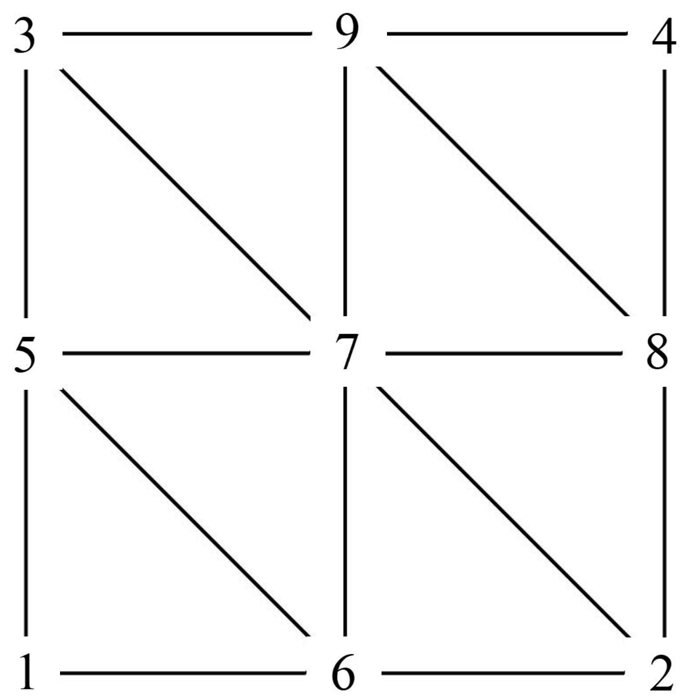
Principle of virtual work for particle and continuum models is just a concise representations of equations-of-motion and boundary conditions of the models.

Virtual work	String	Membrane
δW^{int}	$-\int_\Omega S(\frac{\partial \delta w}{\partial x}\frac{\partial w}{\partial x})dx$	$-\int_\Omega S'(\frac{\partial \delta w}{\partial x}\frac{\partial w}{\partial x}+\frac{\partial \delta w}{\partial y}\frac{\partial w}{\partial y})dA$
δW^{ext}	$\int_\Omega (\delta w f)dx$	$\int_\Omega (\delta w f')dA$
δW^{ine}	$-\int_\Omega (\delta w \rho A \frac{\partial^2 w}{\partial t^2})dx$	$-\int_\Omega (\delta w \rho t \frac{\partial^2 w}{\partial t^2})dA$

6.1 INTERPOLANT AND APPROXIMATION



2-index labeling



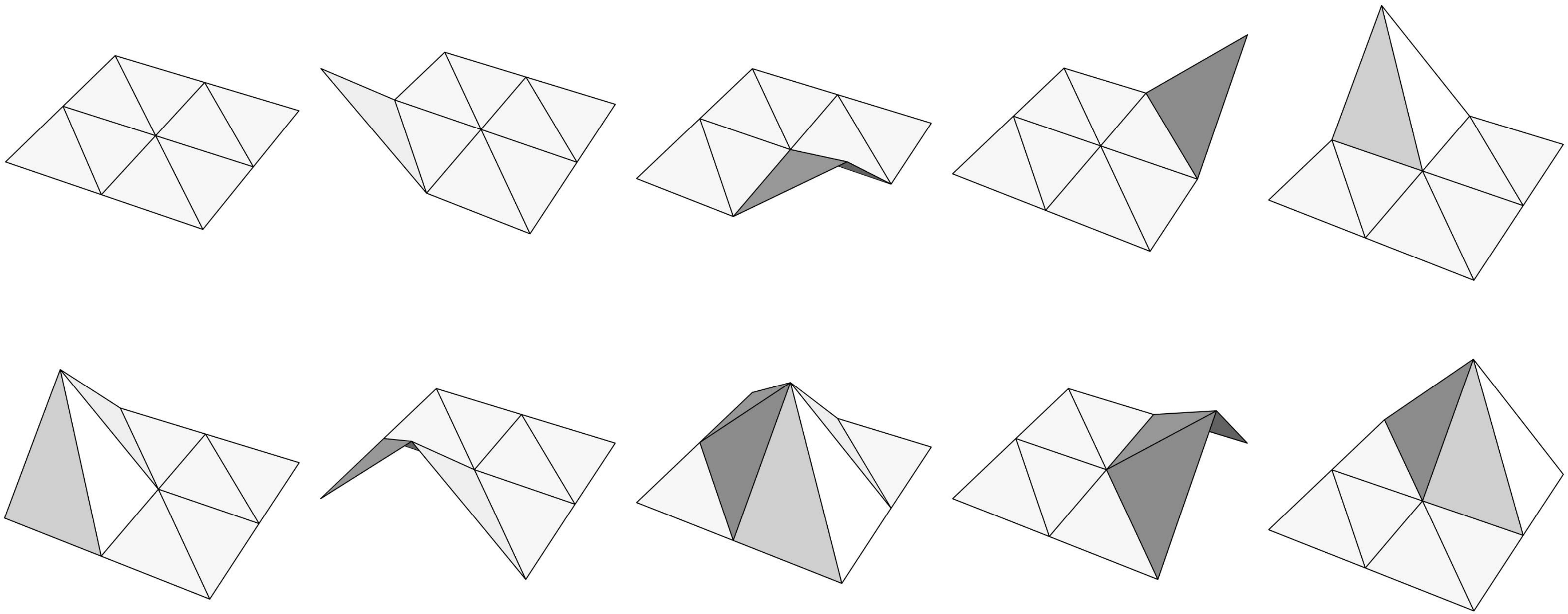
1-index labeling

In Finite Element Method, grid points and triangles, having the grid points as the vertices, are called as the nodes and elements, respectively. The representation of geometry or dataset uses a list of triangle vertex labels and separate lists of coordinates and function values. The regular grid representations used in Particle Surrogate Method and Finite Difference Method are particular cases of the more generic representation.

Piecewise linear interpolant $p(x, y)$ to the dataset $\{\ldots,(x_i, y_{i,} f_i),\ldots\}$ (one-index labeling) uses a triangle representation having the grid points as vertices. Regular triangle representation on a regular grid of the dataset repeats the same triangle element pattern for all interior points. Assuming that the dataset is sampling of function $f(x, y)$ at the grid points, $p(x, y)$ can also be considered as an approximation to $f(x, y)$. Piecewise linear interpolation with the triangle division works also without a regular grid.

SHAPE FUNCTIONS

Piecewise linear interpolants to datasets $\{\ldots,(x_i, y_i, f_i),\ldots\}$, where f_i is chosen to be one at one of the grid points the remaining being zeros, are called as the piecewise linear shape functions $N_i(x, y)$:



With the shape function concept, the linear interpolant to dataset $\{\ldots,(x_i, y_i, f_i),\ldots\}$ (x_i, y_i, f_i) can be represented in the same form as in the one-dimensional case

$$p(x,y) = \sum_{i\in I} f_i N_i(x,y)$$

in which f_i $i \in I$ are the nodal values and I is the labelling set. In a typical triangle element of the vertex nodes (i, j, k), only the shape functions of nodes i, j, and k are non-zeros. The expression of the shape functions and the interpolant are

$$\begin{Bmatrix} N_i(x,y) \\ N_j(x,y) \\ N_k(x,y) \end{Bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ x_i & x_j & x_k \\ y_i & y_j & y_k \end{bmatrix}^{-1} \begin{Bmatrix} 1 \\ x \\ y \end{Bmatrix} \quad \text{and} \quad p(x,y) = \begin{Bmatrix} f_i \\ f_j \\ f_k \end{Bmatrix}^{\mathrm{T}} \begin{Bmatrix} N_i(x,y) \\ N_j(x,y) \\ N_k(x,y) \end{Bmatrix},$$

respectively.

6.2 WEIGHTED RESIDUAL APPROXIMATION

Finding an approximation $g(x,y)$ to function $f(x,y)$ is one the basic tasks in numerical mathematics. In the Least Squares Method and Weighted Residual Methods, the nodal values g_i of approximation $g(x) = \sum g_i N_i(x) = \mathbf{N}^{\mathrm{T}}\mathbf{g}$ follow from the steps

Distance: $\Pi(\mathbf{g}) = \frac{1}{2}\int_{\Omega} (g - f)^2 dA = \frac{1}{2}\int_{\Omega} (\mathbf{N}^{\mathrm{T}}\mathbf{g} - f)^2 dA,$

Minimizer: $\mathbf{Kg} - \mathbf{F} = \mathbf{0}$ where $\mathbf{K} = \int_{\Omega} \mathbf{N}\mathbf{N}^{\mathrm{T}} dA$ and $\mathbf{F} = \int_{\Omega} \mathbf{N}f \, dA,$

Nodal values: $\mathbf{g} = \mathbf{K}^{-1}\mathbf{F}.$

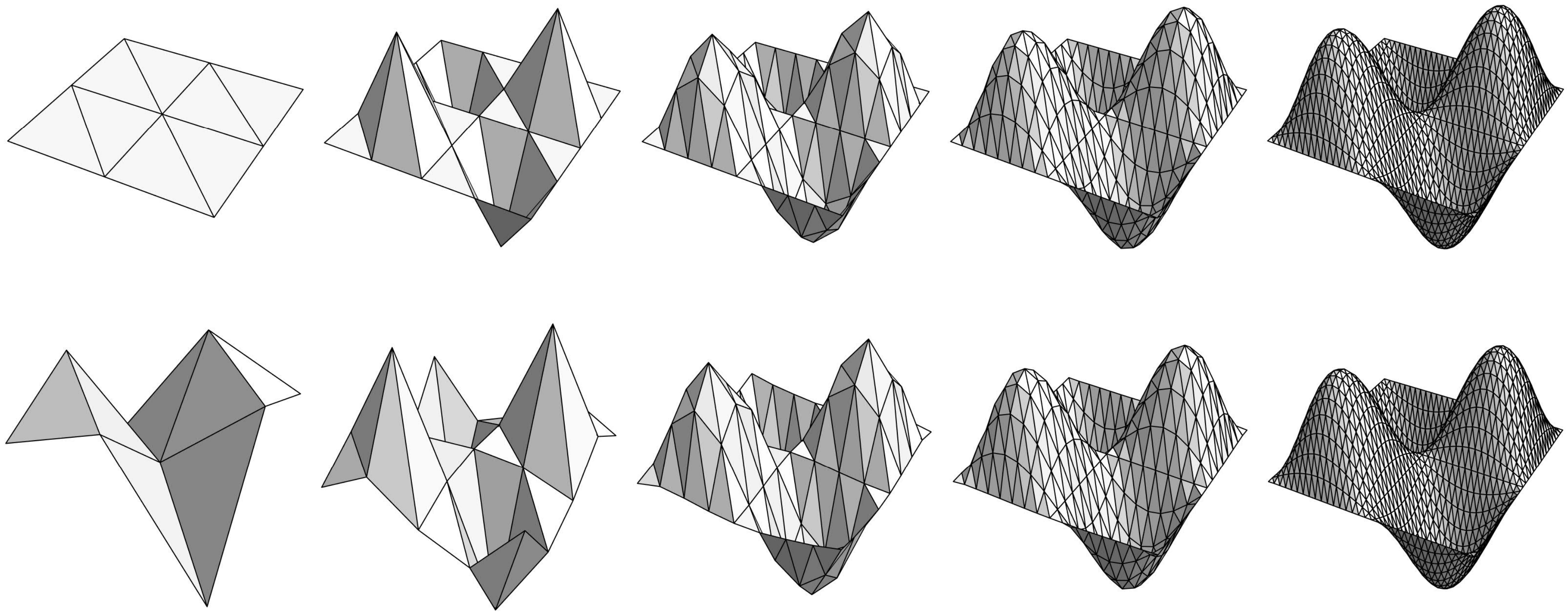
In practice, the nodal values $\mathbf{g}$ are solved from the linear equation system without matrix inversion (to avoid excess computational work). The method works in the same manner irrespective of the series approximation used.

Least Squares Method is useful in various tasks in numerical mathematics. One of the applications is related with the condition for the minimum of Π, which can be written in the form

$$\int_{\Omega} N_i R dA = 0 \quad i \in \{0,1,\ldots\}$$

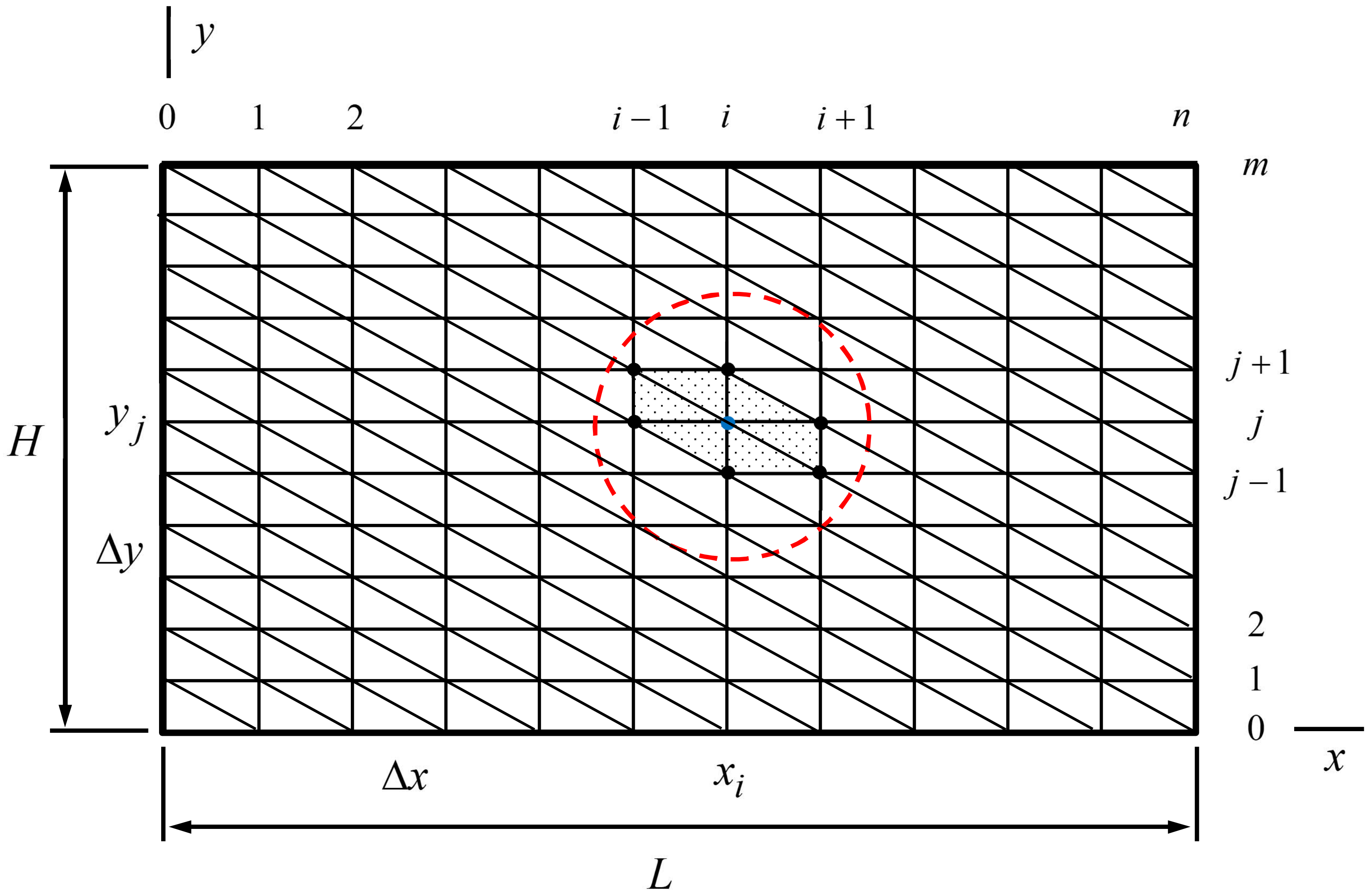
where $R = g(x,y) - f(x,y)$ is called as the residual. In the weighted residual interpretation of the method, linear equations giving the values of the approximation are obtained as the weighted residuals with the shape functions. The idea extends to residuals of differential equations and is one of the starting points for the Finite Element Method.

EXAMPLE Let us consider function $f(x,y)=\sin(2\pi x/L)\sin(\pi y/L)/2$ on the square domain $(x,y)\in[0,L]\times[0,L]$. Using regular triangle elements on a regular grid of points (nodes), piecewise linear interpolant $p(x,y)$, and the least-squares approximation $g(x,y)$ to $f(x,y)$ are:



Interpolant is accurate on the grid points but the interpolation error at the other points is not controlled. As nodal values on the rough 3×3 point grid vanish, also the interpolant is identically zero. Least squares method considers all points of the domain and control the error everywhere. On the rough grid, the piecewise linear approximation is not particularly accurate but better than that given by the interpolant. When the number of elements is increase, both approximations converge to $f(x,y)$.

REGULAR GRID OF TRIANGLES



APPROXIMATION TO DERIVATIVES

The weighted average, using $N_{(i,j)}$, of a derivative of the piecewise linear interpolant $a = \sum_{(i,j)\in I\times I} N_{(i,j)} a_{(i,j)}$ where $I = \{0,1,\ldots,n\}$ is interpreted as an approximation to derivative at the interior grid points (multiplied by ΔA due to the integration).

Term	Weighted residual
$a(x,y)$	$\int_\Omega N_{(i,j)} a \, dA$
$\frac{\partial a}{\partial x}$	$\int_\Omega N_{(i,j)} \frac{\partial a}{\partial x} dA$
$\frac{\partial^2 a}{\partial x^2} + \frac{\partial^2 a}{\partial y^2}$	$-\int_A \left(\frac{\partial N_{(i,j)}}{\partial x}\frac{\partial a}{\partial x} + \frac{\partial N_{(i,j)}}{\partial y}\frac{\partial a}{\partial y}\right) dA$

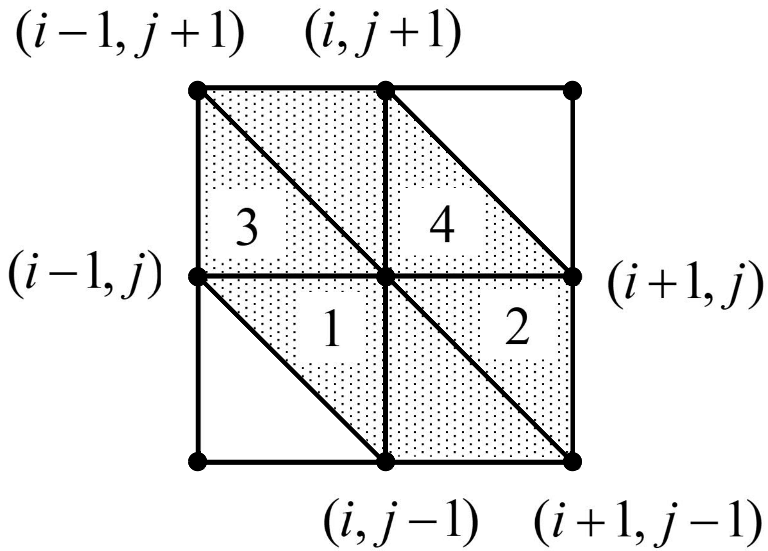
As an example, let us consider the approximation

$$\Delta A\left(\frac{\partial^2 a}{\partial x^2}\right)_{(i,j)} \approx -\int_{\Omega} \left(\frac{\partial N_{(i,j)}}{\partial x}\frac{\partial a}{\partial x}\right)dA = -\sum_e \int_{\Omega^e} \left(\frac{\partial N_{(i,j)}}{\partial x}\frac{\partial a}{\partial x}\right)dA,$$

where the sum is over the elements having the grid point (i, j) in common ($N_{(i,j)}$ vanishes elsewhere). Considering the 6 elements separately and using the fact the derivatives of the shape functions of a piecewise linear interpolation are piecewise constants, and placing the origin of the coordinate system at (i, j):

1 and 3 : $\frac{\partial a}{\partial x} = \frac{a_{(i,j)} - a_{(i-1,j)}}{\Delta x}, \quad \frac{\partial N_{(i,j)}}{\partial x} = \frac{1}{\Delta x}$

2 and 4: $\frac{\partial a}{\partial x} = \frac{a_{(i+1,j)} - a_{(i,j)}}{\Delta x}, \quad \frac{\partial N_{(i,j)}}{\partial x} = -\frac{1}{\Delta x}$



Therefore, the outcome

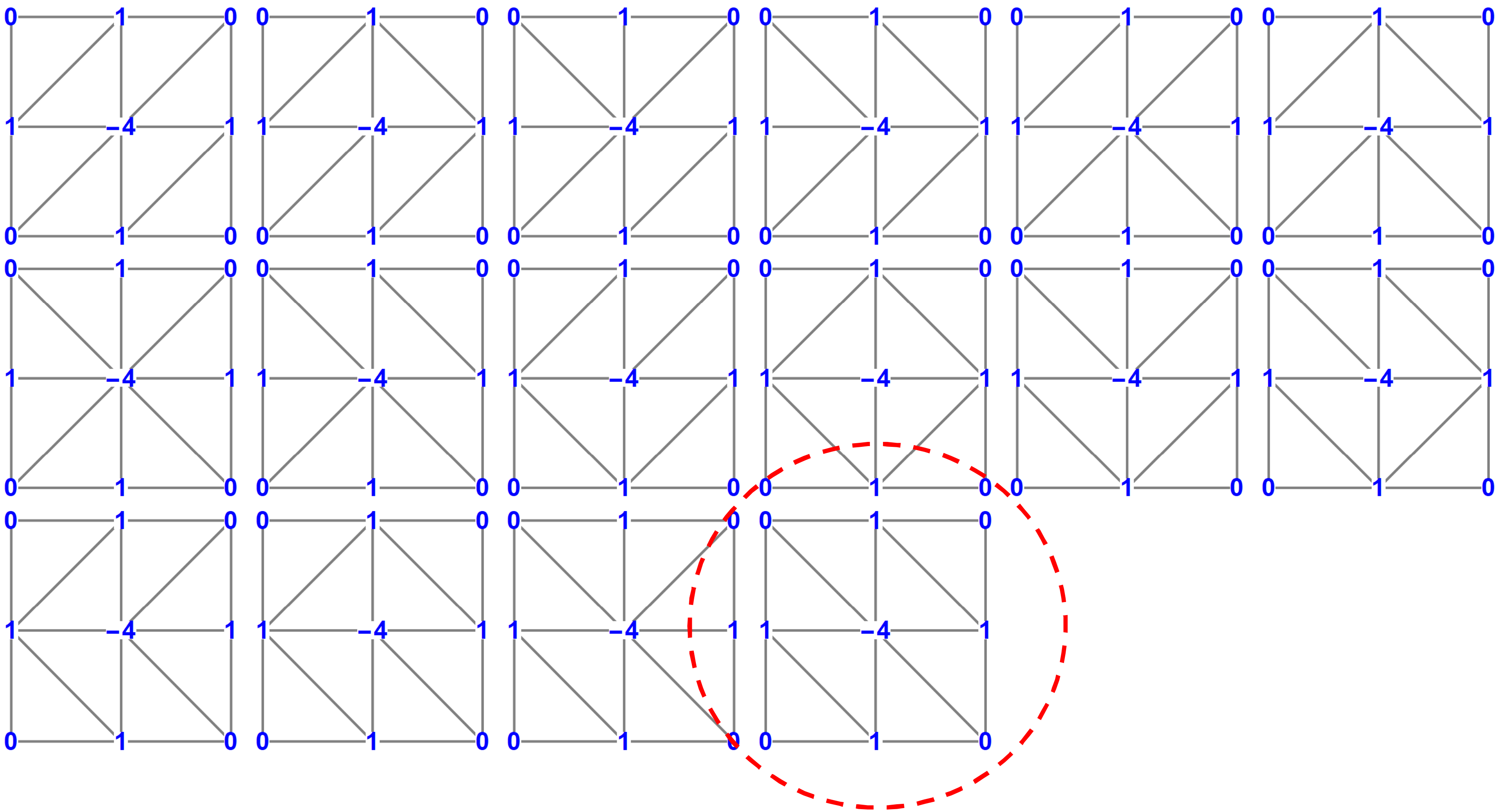
$$\Delta A(\frac{\partial^2 a}{\partial x^2})_{(i,j)} \approx -\sum_e \int_{A^e} (\frac{\partial N_{(i,j)}}{\partial x}\frac{\partial a}{\partial x})dA \approx \frac{\Delta y}{\Delta x}[a_{(i-1,j)} - 2a_{(i,j)} + a_{(i+1,j)}])$$

differs from the expression by the Finite Difference Method by multiplier

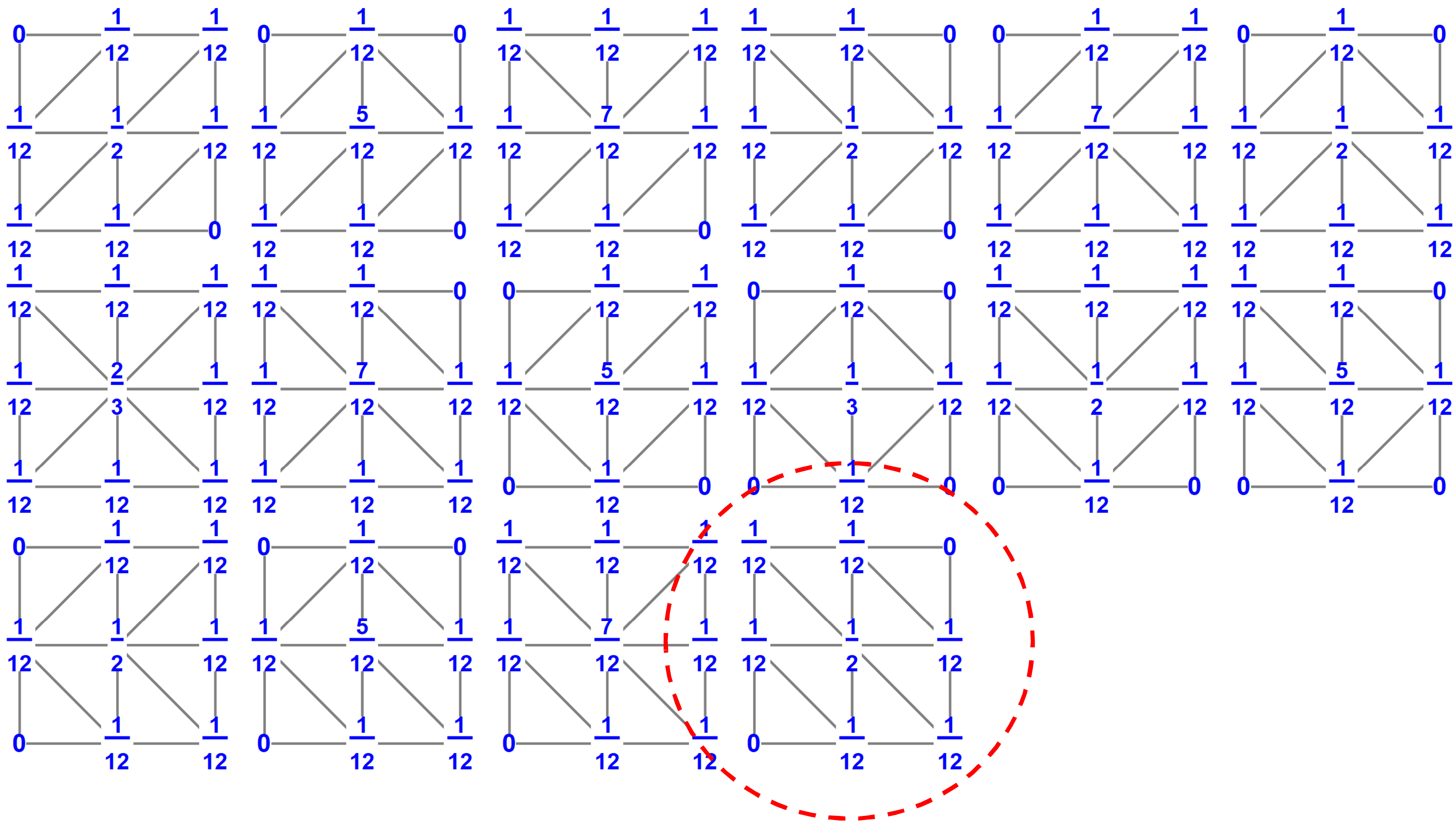
$$\Delta A \approx \int_\Omega N_{(i,j)} dA = \Delta x \Delta y$$

so the weighted residual approximation can be considered as the shape function weighted average. The rule for finding the difference approximation by the weighted residual method is the same for any division of the domain into triangle elements but the outcomes may differ. For the regular triangle division used in the example (geometry is the same for all interior grid points) the outcome can be described by a stencil of constant weights.

STENCIL OF LAPLACIAN

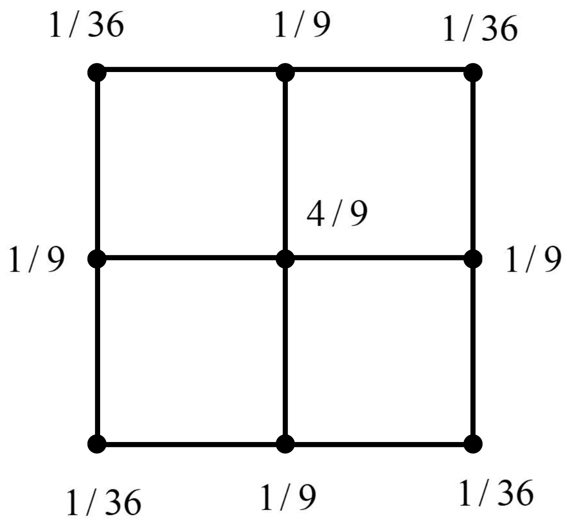


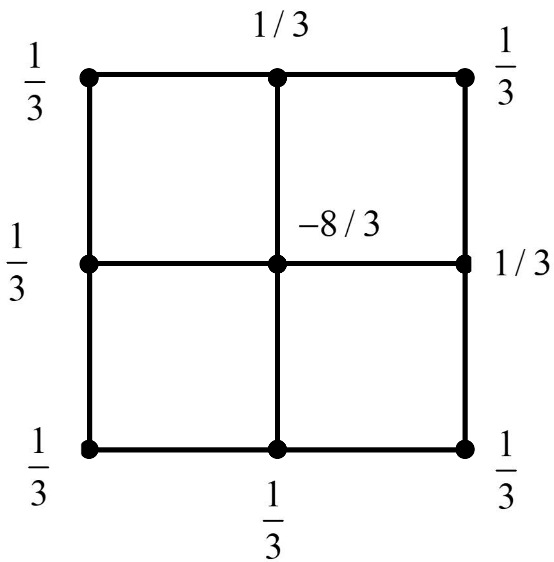
STENCIL OF IDENTITY MAPPING



OTHER STENCILS

Even on a regular grid, derivative approximations based on the weighted residual expression depend on the interpolation used. For a bi-linear interpolation (another common interpolation type in the Finite Element Method) based on four vertex points of rectangular elements, the stencils for the identity and Laplacian operators take the forms:





In each element, bi-linear approximation is combination of 1, x, y, and xy. The shape functions take the value one at one grid point and vanishes at all the other grid points.

WEIGHTED RESIDUAL EXPRESSIONS FOR MEMBRANE

Difference equation approximation to the membrane model follow from the principle of virtual work, triangle representation of the domain, piecewise linear approximation to the transverse displacement, and using the shape functions as the weights:

Virtual work	String	Membrane
δW^{int}	$-\int_{\Omega} S(\frac{\partial N_i}{\partial x}\frac{\partial w}{\partial x})dx$	$-\int_{\Omega} S'(\frac{\partial N_{(i,j)}}{\partial x}\frac{\partial w}{\partial x}+\frac{\partial N_{(i,j)}}{\partial y}\frac{\partial w}{\partial y})dA$
δW^{ext}	$\int_{\Omega} (N_i f)dx$	$\int_{\Omega} (N_{(i,j)} f')dA$
δW^{ine}	$-\int_{\Omega} (N_i \rho A\frac{\partial^2 w}{\partial t^2})dx$	$-\int_{\Omega} (N_{(i,j)}\rho t\frac{\partial^2 w}{\partial t^2})dA$

6.3 FINITE ELEMENT METHOD

Considering a regular grid of points of constant spacing $\Delta x = \Delta y = h$ and linear interpolation using a regular triangle division, the weighted residual expression for the membrane problem give the difference equations (2-index notation)

$$S'[w_{(i-1,j)} + w_{(i,j-1)} - 4w_{(i,j)} + w_{(i+1,j)} + w_{(i,j+1)}] + h^2 f' =$$

$$\frac{\rho t h^2}{12}[\ddot{w}_{(i-1,j-1)} + \ddot{w}_{(i-1,j)} + \ddot{w}_{(i,j-1)} + 6\ddot{w}_{(i,j)} + \ddot{w}_{(i+1,j)} + \ddot{w}_{(i,j+1)} + \ddot{w}_{(i+1,j+1)}] \quad (i,j) \in I,$$

$$w_{(i,j)} = 0 \quad (i,j) \in \partial I,$$

$$w_{(i,j)} - g_{(i,j)} = 0 \quad \text{and} \quad \dot{w}_{(i,j)} - h_{(i,j)} = 0 \quad (i,j) \in I,$$

where the interior grid point are denoted by I and the boundary grid points by ∂I.

Finite Element Method replaces the differential equations by difference equations with a compact and generic recipe for stencils of Identity, Laplacian etc. operators on regular and irregular grid of points for all dimensions. The starting point is the weighted residual expression implied by the principle of virtual work.

Using a regular triangle element representation of the solution domain, piecewise linear interpolation to the transverse displacements on the spatial grid, considering the displacement values $w_i(t)$ as functions of time, and assuming constant properties, for $(i, j) \in \{0,1,\dots,n\} \times \{0,1,\dots,n\}$, the weighted residual expression

$$-\int_{\Omega} S'\left(\frac{\partial N_{(i,j)}}{\partial x}\frac{\partial w}{\partial x} + \frac{\partial N_{(i,j)}}{\partial y}\frac{\partial w}{\partial y}\right)dA + \int_{\Omega} N_{(i,j)} f' dA = \int_{\Omega} N_{(i,j)} \rho t \frac{\partial^2 w}{\partial t^2} dA$$

gives the difference equation for the interior grid points $I = \{1, 2\dots, n-1\} \times \{1, 2\dots, n-1\}$ (nodes)

$$S'[w_{(i-1,j)} + w_{(i,j-1)} - 4w_{(i,j)} + w_{(i+1,j)} + w_{(i,j+1)}] + h^2 f' =$$

$$\frac{\rho t h^2}{12}[\ddot{w}_{(i-1,j-1)} + \ddot{w}_{(i-1,j)} + \ddot{w}_{(i,j-1)} + 6\ddot{w}_{(i,j)} + \ddot{w}_{(i+1,j)} + \ddot{w}_{(i,j+1)} + \ddot{w}_{(i+1,j+1)}]. \quad t > 0$$

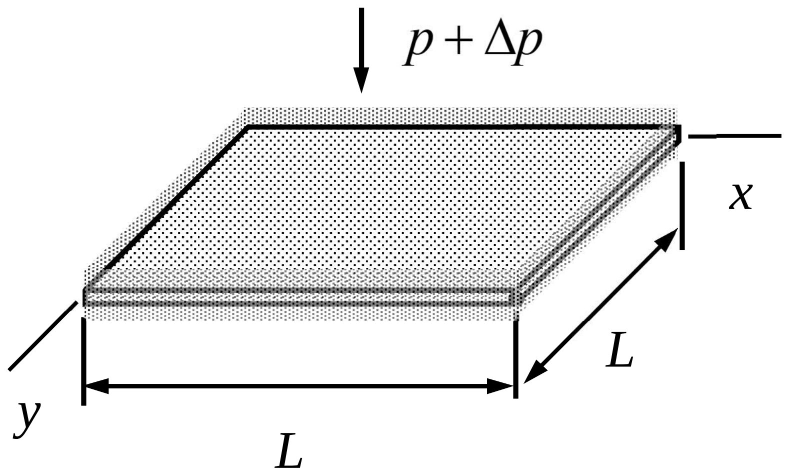
On the boundary points ∂I, displacement vanishes (the second option, force loading on the boundary, is not considered in this simplified setting) so

$$w_{(i,j)} = 0 \;\; t > 0.$$

Assuming that g and h of initial conditions are of the same form as the approximation, the initial conditions

$$w_{(i,j)} - g_{(i,j)} = 0 \quad \text{and} \quad \dot{w}_{(i,j)} - h_{(i,j)} = 0 \qquad (i, j) \in I .$$

EXAMPLE A rectangular membrane of fixed edges and constant tightening s (force per unit length) is loaded by pressures $p+\Delta p$ acting on the upper surface and p acting on the lower surface. Find the solution to the transverse displacement using the Finite Element Method, regular triangle division of the domain with the regular grid $(i, j) \in \{0,1,2\} \times \{0,1,2\}$, and a piecewise linear approximation to the transverse displacement.



Answer $w_{(1,1)} = \dfrac{1}{16}\dfrac{\Delta p L^2}{S'} \approx 0.0625\dfrac{\Delta p L^2}{S'}$ (exact to the model $0.0737\dfrac{\Delta p L^2}{S'}$)

Finite Element Method is based on the principle of virtual work, element representation of the domain, intrpolation of the grid values inside the elements, and using the shape functions of the interpolation as the weights. In the present problem the interior and boundary grid points and the equations for the grid points are

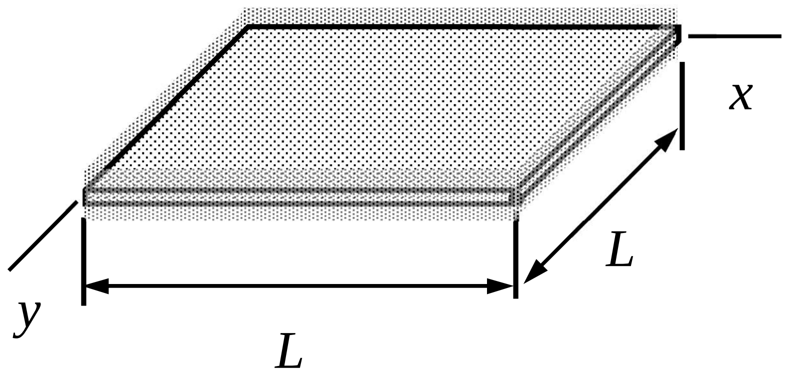
$$S'[w_{(i-1,j)} + w_{(i,j-1)} - 4w_{(i,j)} + w_{(i+1,j)} + w_{(i,j+1)}] + \Delta p h^2 = 0 \quad (i,j) \in I = \{(1,1)\}$$

$$w_{(i,j)} = 0 \quad (i,j) \in \partial I = \{0,1,2\} \times \{0,1,2\} \setminus \{(1,1)\} \quad \text{(interior point excluded)}$$

where $\Delta x = \Delta x = h = L/2$. Eliminating the displacements of the boundary points from the equation for the interior point

$$S'[-4w_{(1,1)}] + \Delta p \frac{L^2}{4} = 0 \quad \Rightarrow \quad w_{(1,1)} = \frac{1}{16}\frac{\Delta p L^2}{S'}.$$

EXAMPLE Consider a rectangular (side length L) drumhead of fixed edges, constant tightening S' (force per unit length) and density ρt (per unit area). Find the frequencies of the free vibrations by using the Finite Element Method on a regular grid $I = \{0,1,2\} \times \{0,1,2\}$ with piecewise linear approximation to the transverse displacement.



Answer $f = \dfrac{2}{\pi L}\sqrt{2\dfrac{S'}{\rho t}} \approx 0.90\dfrac{1}{L}\sqrt{\dfrac{S'}{\rho t}}$ (exact to the model $\approx 0.71\dfrac{1}{L}\sqrt{\dfrac{S'}{\rho t}}$)

Finite Element Method uses the principle of virtual work, element representation of the domain, interpolation of the grid point values inside the elements, and the shape functions of the approximation as the weights. In the present problem the interior and boundary grid points and the equations for the grid points are

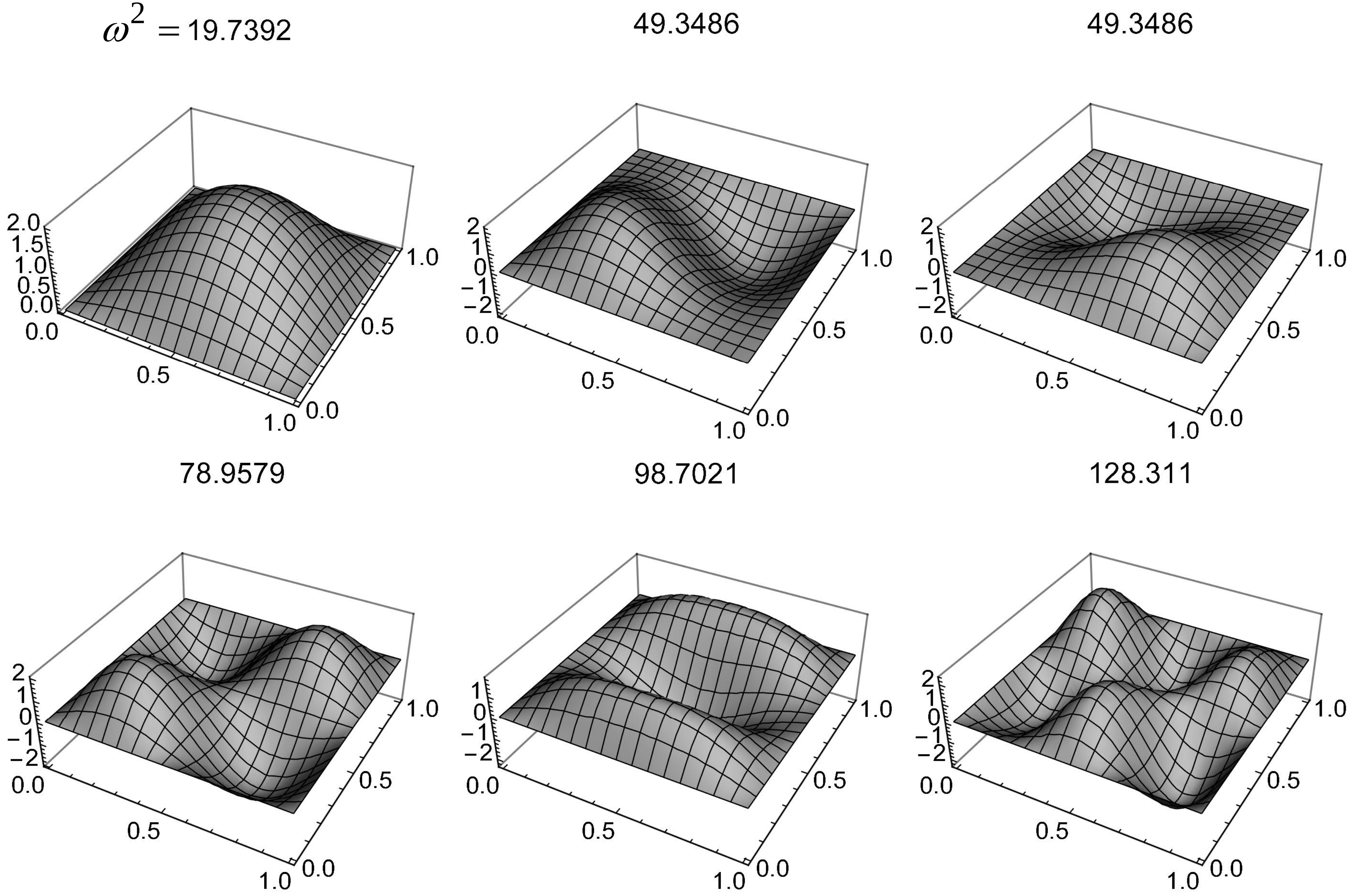
$$S'[w_{(i-1,j)} + w_{(i,j-1)} - 4w_{(i,j)} + w_{(i+1,j)} + w_{(i,j+1)}] =$$

$$\frac{\rho t h^2}{12}[\ddot{w}_{(i-1,j-1)} + \ddot{w}_{(i-1,j)} + \ddot{w}_{(i,j-1)} + 6\ddot{w}_{(i,j)} + \ddot{w}_{(i+1,j)} + \ddot{w}_{(i,j+1)} + \ddot{w}_{(i+1,j+1)}] \quad (i,j) = (1,1),$$

$$w_{(i,j)} = 0 \quad (i,j) \in \{0,1,2\} \times \{0,1,2\} \setminus \{(1,1)\} \quad \text{(interior point excluded)}$$

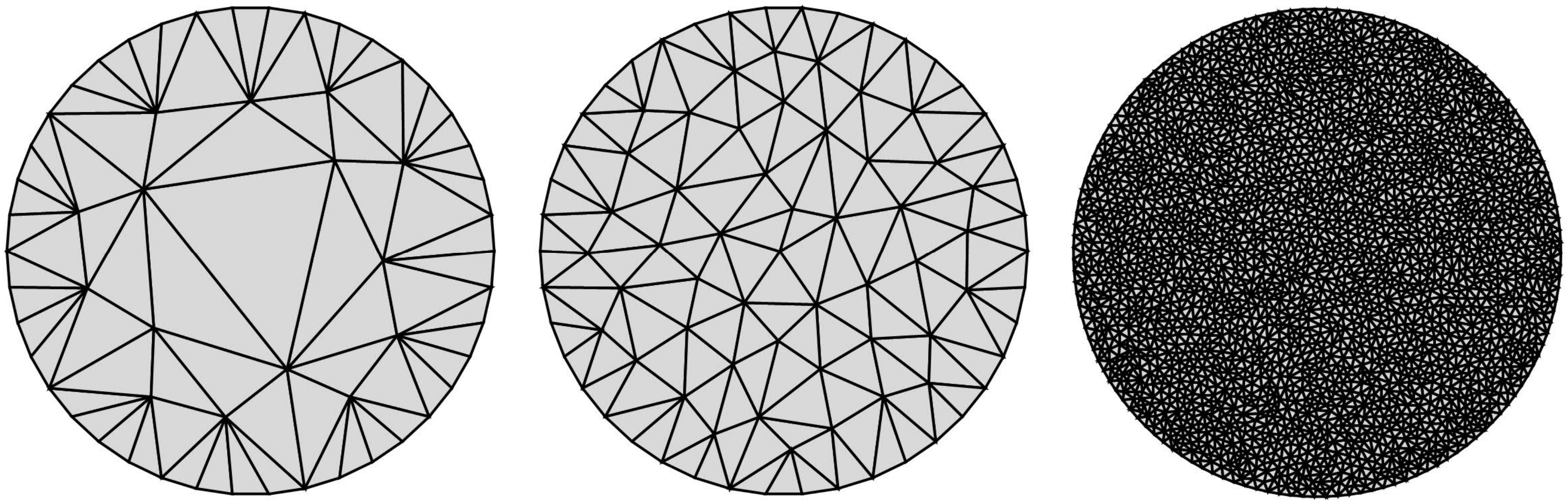
where $\Delta x = \Delta x = h = L/2$. Eliminating the displacements of the boundary points from the equation for the interior point

$$\ddot{w}_{(1,1)} + \omega^2 w_{(1,1)} = 0 \quad \text{where} \quad \omega = 2\pi f = \frac{4}{L}\sqrt{2\frac{S'}{\rho t}} \quad \text{so} \quad f = \frac{2}{\pi L}\sqrt{2\frac{S'}{\rho t}} \ .$$ ⇦

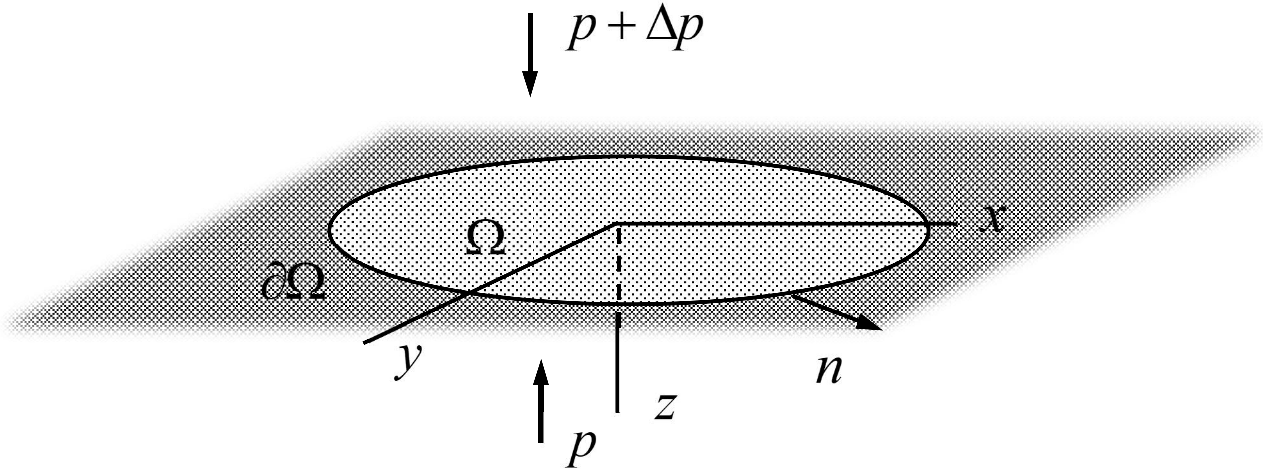

$\omega^2 = 19.7392$
49.3486
49.3486
78.9579
98.7021
128.311
2.0
1.5
1.0
0.5
0.0
0.0
0.5
1.0
0.0
0.5
1.0
2
1
0
−1
−2

6.4 NON-REGULAR GRIDS

Finite Element Method does not impose restrictions on the solution domain geometry or require regularity of the grid or elements having the grid points as vertices. No matter the case, function values at the grid points are interpolated inside the elements and the integrals are calculated elementwise. The outcome is a set algebraic equations or ordinary differential equations that can be solved with matrix methods.



EXAMPLE A circular membrane of radius R, fixed edges, and constant tightening S' (force per unit length) is loaded by pressure $p+\Delta p$ acting on the upper surface and p acting on the lower surface. Find the transverse displacement at the centerpoint by using the Finite Element Method and piecewise linear approximations on triangle elements.



Answer (Exact $w_{\max} = \frac{1}{4}\frac{\Delta p R^2}{S'}$)

In numerical calculations, the problem parameters need to be given values unless the number of grid points is small and the code used allows symbolic calculations (e.g., Mathematica does). Let us consider a combination for which the dimensionless group $\Delta p R^2 / S' = 4$ and let Mathematica to do the triangulation and find the interior and boundary points

```
Ω = Disk[{0, 0}];
ℛ = DiscretizeRegion[Ω, MaxCellMeasure → 0.05]
bℛ = RegionBoundary[ℛ];
bp = MeshCells[bℛ, 0] /. Point[any_] → any;
ip = Complement[Table[i, {i, 1, MeshCellCount[ℛ, 0]}], bp];
```

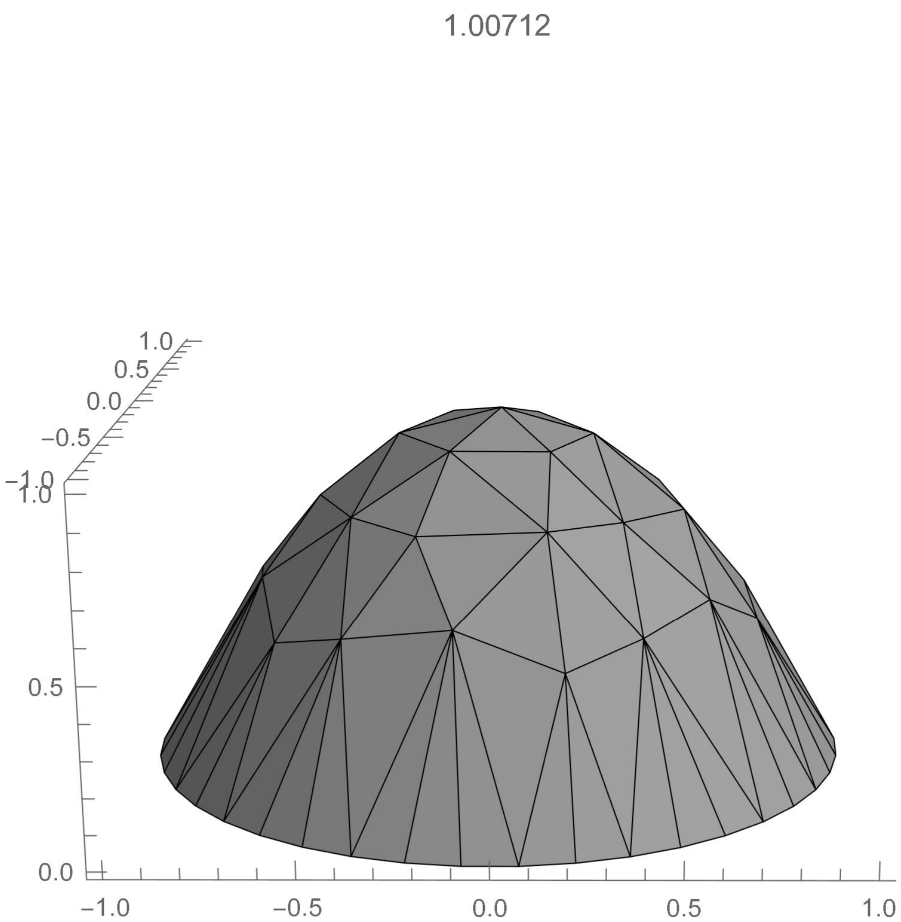
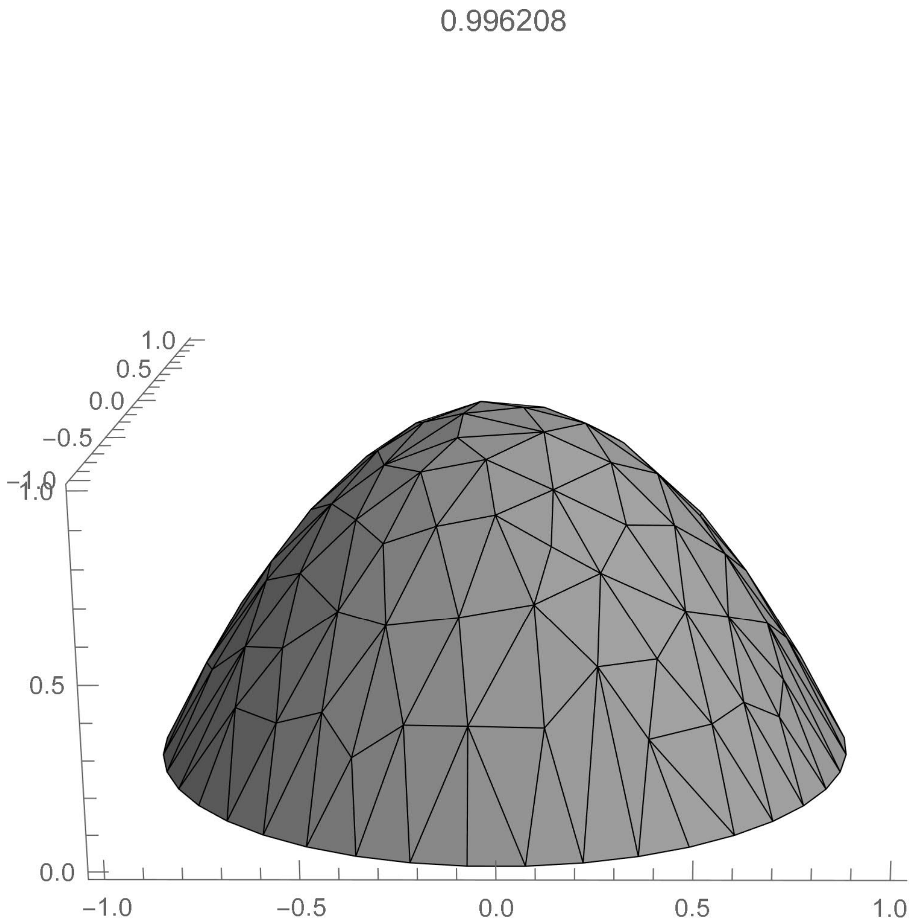
After that, use the stencils of Laplacian and loading at the interior points (given by the weighted residual expression and piecewise linear approximation to the transverse displacement on the triangle representation) to find the matrix representation of the equilibrium equations and solve for the displacements

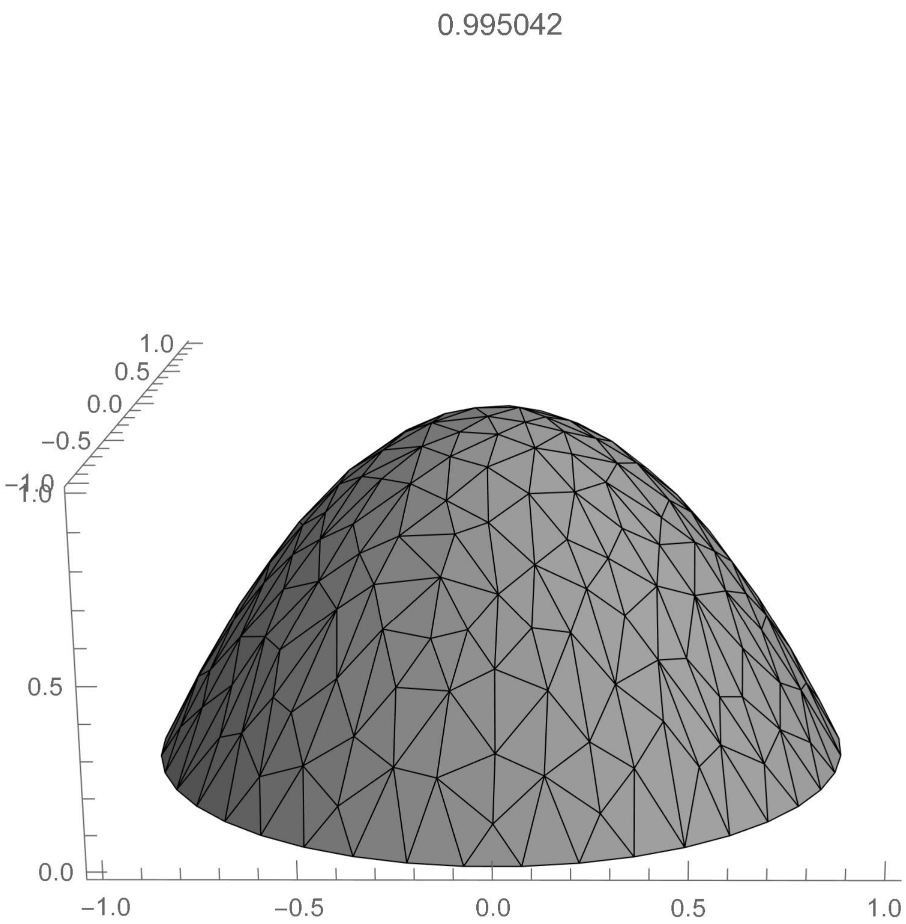
```
FF = LOAD[ℛ, 4];
KK = LAPLACIAN[ℛ];
iw = LinearSolve[KK[[ip, ip]], -FF[[ip]]]
ww = Table[0, {i, 1, MeshCellCount[ℛ, 0]}];
ww[[ip]] = iw;
```

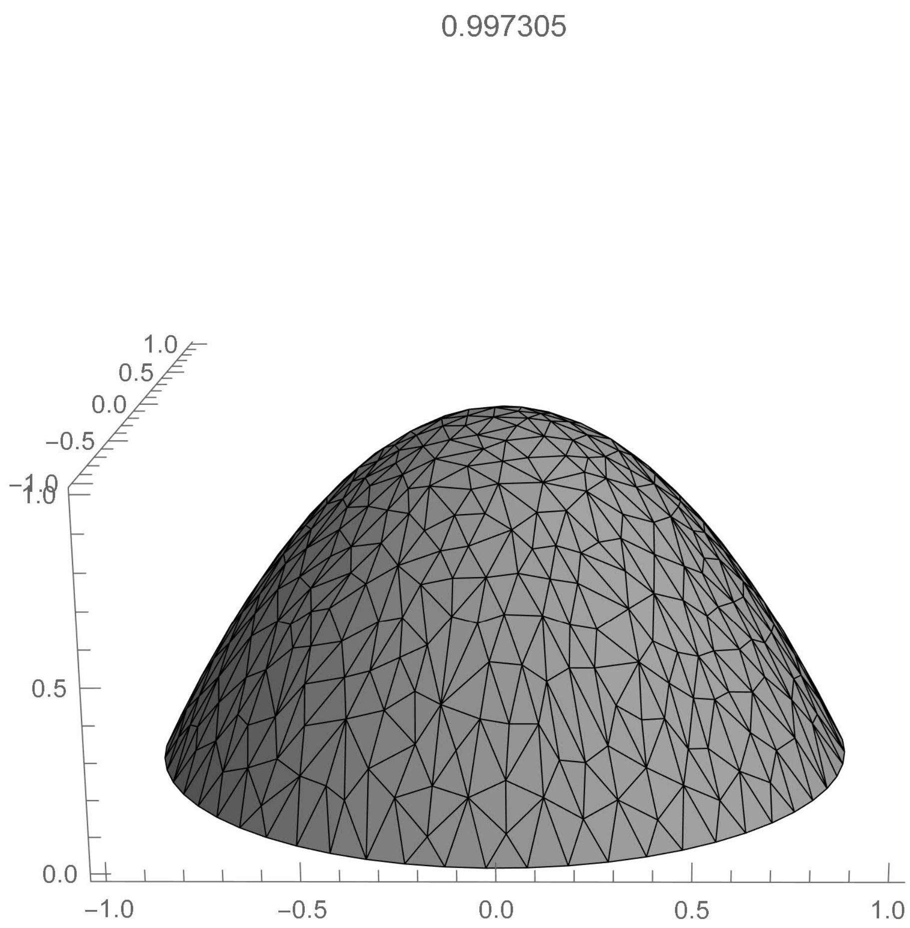
Finally, some post-processing to check the outcome and find the maximal value of the displacement

```
xyw = Transpose[Append[Transpose[MeshCoordinates[ℛ]], ww]];
wℛ = MeshRegion[xyw, MeshCells[ℛ, 2]];
Show[HighlightMesh[wℛ, {Style[{2}, Gray], Style[{1}, Black]}], Axes → True, PlotLabel -> Max[ww]]
```

For a picture about the discretization error, problem can be solved a few times by reducing the size of the elements (reducing the parameter MaxCellMeasure in the beginning of the Mathematica code):


1.00712
1.0
0.5
0.0
−0.5
−1.0
1.0
0.5
0.0
−1.0
−0.5
0.0
0.5
1.0

0.996208
1.0
0.5
0.0
−0.5
−1.0
1.0
0.5
0.0
−1.0
−0.5
0.0
0.5
1.0


0.995042
1.0
0.5
0.0
−0.5
−1.0
1.0
0.5
0.0
−1.0
−0.5
0.0
0.5
1.0


0.997305
1.0
0.5
0.0
−0.5
−1.0
1.0
0.5
0.0
−1.0
−0.5
0.0
0.5
1.0