Improved finite difference method for equilibrium problems based on differentiation of the partial differential equations and the boundary conditions

M. Arad, R. Segev, G. Ben-Dor

To cite this version:


HAL Id: hal-01068219
https://hal.archives-ouvertes.fr/hal-01068219
Submitted on 25 Sep 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
IMPROVED FINITE DIFFERENCE METHOD FOR EQUILIBRIUM PROBLEMS BASED ON DIFFERENTIATION OF THE PARTIAL DIFFERENTIAL EQUATIONS AND THE BOUNDARY CONDITIONS

M. ARAD, R. SEGEV AND G. BEN-DOR
Pearlstone Center for Aeronautical Engineering Studies, Department of Mechanical Engineering, Ben-Gurion University of the Negev, Beer Sheva, Israel

SUMMARY
A numerical algorithm for producing high-order solutions for equilibrium problems is presented. The approximated solutions are improved by differentiating both the governing partial differential equations and their boundary conditions.

The advantages of the proposed method over standard finite difference methods are: the possibility of using arbitrary meshes; the possibility of using simultaneously approximations with different (distinct) orders of accuracy at different locations in the problem domain; an improvement in approximating the boundary conditions; the elimination of the need for 'fictitious' or 'external' nodal points in treating the boundary conditions.

Furthermore, the proposed method is capable of reaching approximate solutions which are more accurate than other finite difference methods, when the same number of nodal points participate in the local scheme.

A computer program was written for solving two-dimensional problems in elasticity. The solutions of a few examples clearly illustrate these advantages.

KEY WORDS: numerical methods; finite differences; plane elasticity

1. INTRODUCTION
The rapid growth in the use of computers in the past decade gave rise to the development of advanced numerical methods. The two basic ones are the finite element and the finite difference methods. The advantages of the finite element method over the finite difference method were presented by Zienkiewicz. They are:

(a) the ease of arbitrary positioning of nodal points;
(b) the infinite possibilities of generation of 'improved' elements by simply increasing the number of element parameters;
(c) the improvement in the boundary value approximation due to its integral form; and
(d) the ease with which different types and sizes of elements can be adopted.

Since the pioneering work of Southwell in which the finite difference method was comprehensively formulated, many works, attempting to improve the method, have been published. For example, Forsyth and Wasow suggested to treat curved boundaries by the use of interpolations between points on the boundaries and nodal points adjacent to them. Wang and Collatz...
reated curved boundaries having Dirichlet conditions by using Taylor series expansions. A different approach for treating curved boundaries, still using a square or rectangular mesh, was forward by Varga. In this approach, sometimes referred to as the control volume approach, an integration is performed over the mesh areas using Green's theorem. This approach has been improved and applied in many areas e.g. by Griffin and Kellogg, Havir and Staton, Havir, Johnson, and others. As shown by Jensen, one of the difficulties arising from the use of a square or rectangular mesh with curved boundaries is the need to increase the number of the nodes over the number actually necessary to achieve a particular accuracy in the solution. One way of overcoming this shortcoming is the use of a triangle mesh. Triangle mesh was probably first proposed by MacNeal who investigated current flows through sheet applications of triangle mesh which uses finite differences derived from the variational approach and Ritz method appeared in the works of Kellogg and his coworkers. A different approach for using triangular mesh originally proposed by Window was generalized by Chu. The idea was to replace the rectangular (or square) mesh with an equilateral triangle mesh plane. The replacement involved an "oblique transformation" of the entire domain. A somewhat different approach, which in fact is an improvement, was patented by Frey who used the concept of isoparametric elements. He introduced flexible finite difference stencils of arbitrary shape mapped into a regular square mesh.

Jensen proposed a different approach for generating finite difference approximations in an arbitrary mesh by using two-dimensional Taylor series expansions. He showed that by using this approach, the error in the approximated Taylor series expansions converges to zero when the nodes density is increased. Second-order expansions were used by him on a six-point control scheme (template). The main disadvantage of his approach was frequent singularity or ill-conditioning of the six-point control scheme. More details regarding this singularity are given subsequently. Several authors tried to develop an automatic procedure which avoids incorrect schemes and thus improves the accuracy of the finite difference formula. Pereira and Kao suggested that additional nodes in the six-point scheme should be considered and an averaging process for the generation of finite difference coefficients applied. Liski and Orkisz proposed to increase the number of the neighbouring nodes in the schemes forming an over-determined set of linear equations. Their solution was obtained by a minimization of a norm that they derived. In this way a set of five equations with five unknowns was obtained. Noye and Arnold developed a more accurate finite difference scheme for approximating the derivatives at points that are not on a curved boundary along which a Neumann boundary condition applies. There is no need to reshape the curved boundary, and the approximations at adjacent grid points have the same accuracy as schemes commonly used for Dirichlet boundary conditions. Dow et al. improved the boundary modelling for finite difference applications in solid mechanics. The improvement was expressed by introducing the physical nature of the deformations into the equations by which the solution is approximated. This is achieved by expressing the coefficients of the Taylor series expansions which approximate the displacements in terms of rigid body motions, strains and derivatives of strains. As a result the fictitious nodal points were automatically incorporated into the mesh. This approach enabled one to use an arbitrary mesh provided the problem domain was rectangular. In their example they used Taylor series expansions with nine coefficients using Pascal's triangle, i.e. second-order Taylor expansions with three additional coefficients (note that this is identical to the regular nine-point finite element approximation). In a subsequent modification Dow and Hardaway applied the approach to curved boundaries and concave corners. More details regarding the derivation of the displacement expansions can be found in References 28–31. Kochavi et al. used non-conforming Taylor discretization that allowed discrepancy between values resulting from the Taylor series expansions about distinct nodes if the discrepancy
is in the same order of magnitude as the estimated error resulting from the discretization. This method was capable of producing results that were more accurate than those obtained by traditional conforming discretizations. In addition, unnecessary computations might be avoided by adjusting the accuracy of the solution of the algebraic equations to the truncation error in the Taylor series expansions.

The main disadvantage of all these studies is that they approximate the solutions by second-order Taylor series expansions in which the errors of the third order, i.e. $O(h^3)$ where $h$ is the grid size. Thus, the errors associated with the approximations of the second-order derivatives in an arbitrary mesh (which usually appear in the conservation equations) are very large, as they are of the order of $O(h)$.

In general, Jensen showed that with this formulation the derivatives of order $i$ can be calculated with an accuracy $O(h^{r-1})$, by using $N_x = \frac{1}{2}(m+1)(m+2)$ neighbouring values ($m$ is the order of the Taylor series expansion). For example, the difference expressions for all derivatives up to the fourth order require at least $N_x = 15$ neighbour values and consequently the inversion of a $15 \times 15$ nodal matrix for each point! In view of the computational effort required to invert a $15 \times 15$ matrix it is appropriate to look for ways of alleviating this problem.

The new method which will be presented in the next section has all the advantages associated with the finite element method as introduced by Zienkiewicz. Furthermore, in the proposed method, approximations using Taylor series expansions of the $m$th order require at most $N_x = 3m + 2$ neighbour values, instead of the above-mentioned requirement of $N_x = \frac{1}{2}(m+1)(m+2)$ neighbour values.

THE PROPOSED METHOD

The proposed method which is applicable to equilibrium problems is an improved Differentiation Based Finite Difference Method (DFD-method). In the DFD-method both the governing equations and their boundary conditions are differentiated. The model is a mathematical one and independent of a specific physical problem. In order to simplify its introduction, it will be applied to problems in elasticity. However, it should be emphasized here that it could be applied to equilibrium problems from other fields e.g. steady viscous flows, etc.

INTRODUCTION OF THE PROBLEM TO BE SOLVED

In the following the DFD-method will be illustrated by applying it to the solution of the equilibrium equations in terms of displacements $u$ and $v$ in plane elasticity (for details see p. 265 of Reference 33). The equilibrium equations are

$$
\begin{align}
\frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) - \frac{\partial}{\partial y} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) &= f_x \quad \text{in } \Omega \\
- \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) - \frac{\partial}{\partial y} \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) &= f_y
\end{align}
$$

the boundary conditions can be either primary (displacements)

$$
\begin{align}
u &= \tilde{u} \\
v &= \tilde{v}
\end{align}
$$

on $\Gamma_1$.
or natural traction

\[
\begin{align*}
(c_{11} \frac{\partial u}{\partial x} + c_{12} \frac{\partial v}{\partial y}) n_x + c_{13} \frac{\partial u}{\partial y} n_x + c_{23} \frac{\partial v}{\partial x} n_y & = f_x & \text{on } \Gamma_2 \\
(c_{12} \frac{\partial u}{\partial y} + c_{22} \frac{\partial v}{\partial y}) n_x + c_{23} \frac{\partial v}{\partial x} n_y & = f_y & \text{on } \Gamma_2
\end{align*}
\]  

(3a)  

(3b)

where \( u \) and \( v \) are the displacements in the \( x \) and \( y \) directions, respectively, are the unknown variables; \( f_x \) and \( f_y \) denote the body forces in the \( x \) and \( y \) directions, respectively; \( n_x \) and \( n_y \) are the \( x \) and \( y \) components of a unit vector \( \mathbf{n} \) normal to the boundary \( \Gamma \); \( \Gamma_1 \) and \( \Gamma_2 \) are (disjoint) portions of the boundary \( \Gamma \); \( f_x \) and \( f_y \) do not overlap except for a small number of discrete points—singular points; \( f_x \) and \( f_y \) denote specified boundary (traction) forces in the \( x \) and \( y \) directions, respectively; \( \tilde{u} \) and \( \tilde{v} \) are specified displacements in the \( x \) and \( y \) direction, respectively.

Figure 1. Schematic illustration of two-dimensional plane elasticity problems. (a) plane stress; (b) plane strain.
\( c_{ij} \) are the elasticity (material) constants. For an isotropic elastic body they are given in terms of the modulus of elasticity \( E \) and the Poisson's ratio \( \nu \) as follows:

for plane stress

\[
\begin{align*}
    c_{11} &= c_{22} = w \frac{E}{1 - \nu^2} & \quad c_{12} &= w \frac{\nu E}{1 - \nu^2} & \quad c_{44} &= w \frac{E}{2(1 + \nu)} \\
\end{align*}
\]

(4a)

for plane strain

\[
\begin{align*}
    c_{11} &= c_{22} = w \frac{E(1 - \nu)}{1 + \nu(1 - 2\nu)} & \quad c_{12} &= w \frac{\nu E}{1 + \nu(1 - 2\nu)} & \quad c_{23} &= w \frac{E}{2(1 + \nu)}
\end{align*}
\]

(4b)

where \( w \) is the thickness of the plate. For the reader's convenience the above definitions are shown in Figure 1.

For an isotropic elastic body without body forces, the above equations can be simplified to the following form:

\[
\begin{align*}
    c_{11} \partial u_x \partial + c_{12} \partial u_y \partial + 4(\nu + 1) u_x u_y &= 0 \quad (5a) \\
    c_{44} \partial u_x \partial + c_{11} \partial u_y \partial + (c_{12} + c_{44}) \partial u_y \partial &= 0 \quad (5b)
\end{align*}
\]

where

\[
\begin{align*}
    u_x &= \frac{\partial u}{\partial x}, & \quad u_y &= \frac{\partial u}{\partial y}, & \quad u_{xx} &= \frac{\partial^2 u}{\partial x^2}, \quad \text{etc.}
\end{align*}
\]

These derivatives are written in the following condensed form:

\[
\begin{align*}
    u_{x x x x} &= \frac{\partial^4 u}{\partial x^4},
\end{align*}
\]

In the following the solution of equations (5a) and (5b) will be approximated for the boundary conditions given by equations (2) or (3).

**METHOD OF SOLUTION**

The first step of the iteration method is to scatter \( N \) nodal points in the computation domain and along the boundary as shown in Figure 2. The way of positioning the nodal points in order to avoid singularity and ill-conditioning will be described subsequently. The \( i \)th nodal point will be defined by its \( x \) and \( y \) co-ordinates \((x_i, y_i)\). Around any nodal point one can derive Taylor series expansions of any order \((e.g., m)\) which will approximate the unknown variables (i.e. the displacements \( u \) and \( v \) in the present example) A similar approach was adopted, for example, by Dow et al.\(^{14}\) These expansions can be written in the following condensed form:

\[
\begin{align*}
    u_i(x, y) &= \sum_{i=1}^{N} \sum_{j=0}^{m} \left\{ U_{i,j} \right\} X^{i-j} Y^j \quad (i = 1, \ldots, N) \\
    V_i(x, y) &= \sum_{i=1}^{N} \sum_{j=0}^{m} \left\{ V_{i,j} \right\} X^{i-j} Y^j \quad (i = 1, \ldots, N)
\end{align*}
\]

(6a)

(6b)

where \( U_i(x, y) \) and \( V_i(x, y) \) approximate the displacements \( u \) and \( v \), respectively, in the vicinity of the \( i \)th nodal point \((x_i, y_i)\), and \([U_{i,j}]\) and \([V_{i,j}]\) are the coefficients of the Taylor series expansions and \( m \) is its order. \( N \) is the number of the nodal points, \( X \) and \( Y \) are the local co-ordinates with respect
to the $r$th nodal point, i.e.,

$$X = x - x_r \quad \text{and} \quad Y = y - y_r,$$

where $x$ and $y$ are the global co-ordinates.

Note that the number of each of the coefficients $[U_{i,j}]$, and $[P_{i,j}]$, is $N_r = \frac{1}{2}(m + 1)(m + 2)$.

The error of the above approximations is of the order $O(h^{m+1})$ where, as mentioned earlier, $h$ is the grid size. For example, for a fourth-order Taylor series expansion (i.e. $m = 4$) equation (6a) would explicitly read:

$$U(x,y) = \left[ U_{0,0} \right] + \left[ U_{1,0} \right] X + \left[ U_{2,0} \right] Y + \frac{1}{2} \left[ U_{1,1} \right] X^2 + \left[ U_{1,1} \right] X Y + \frac{1}{2} \left[ U_{2,2} \right] Y^2 + \frac{1}{6} \left[ U_{1,2} \right] X^3 + \frac{1}{6} \left[ U_{1,2} \right] X^2 Y + \frac{1}{2} \left[ U_{1,1} \right] Y^3 + \frac{1}{2} \left[ U_{2,1} \right] X Y^2 + \frac{1}{2} \left[ U_{2,1} \right] Y^3 + \frac{1}{3} \left[ U_{0,2} \right] Y^3.$$

It should also be noted here that the above-mentioned coefficients are indeed approximations of the unknown variables and their derivatives at the nodal points, i.e.

$[U_{0,0}]$ approximates $u$ if $(x,y)$, i.e. at $X = 0$ and $Y = 0$. 
Local governing equations

The most common method for obtaining the values of the above-mentioned 56 unknown variables (i.e. the coefficients) of the Taylor series expansions which approximate the solution, \( [U_{n-1}, \ldots, U_{n+1}] \) and \( [V_{n-1}, \ldots, V_{n+1}] \) is to:

1. Substitute the approximating Taylor series expansions [equations (6a) and (6b)] into the governing equations [equations (5a) and (5b)] for each nodal point while setting the local co-ordinates, \( X \) and \( Y \), to be equal to zero, and
2. Require that the Taylor series expansions from the neighbouring points of a given nodal point will have the same value at that point.

As mentioned earlier, based on Jensen, this requires the use of localized schemes (templates) having 28 neighbouring points. (Recall that sixth-order Taylor series expansions are used in this example.)

In order to obtain the 56 values of the unknown variables by using templates with less than 28 neighbouring points, we propose to:

1. successively differentiate the governing equations [equations (5a) and (5b)] and
2. substitute the approximating Taylor series expansions [equations (6a) and (6b)] into both the governing equations and their differentiated equations, while setting the local coordinates, \( X \) and \( Y \), to be equal to zero.

The governing equations should be successively differentiated until the order of the derivatives equals the order of the Taylor series expansions. (E.g. six in the present example.) Further differentiation will result in, after the substitution of the approximating Taylor series expansions, trivial equations.

Following the above description, the governing equations [equations (5a) and (5b)] were successively differentiated to generate the following set of 28 equations.

\[
\begin{align*}
\frac{\partial^3 U}{\partial x^3} + c_1 \frac{\partial U}{\partial x} + c_2 \frac{\partial^2 U}{\partial x^2} + (c_{12} + c_{13}) \frac{\partial^2 U}{\partial x \partial y} & = 0 \\
\frac{\partial^3 V}{\partial x^3} + c_1 \frac{\partial V}{\partial x} + c_2 \frac{\partial^2 V}{\partial x^2} + (c_{12} + c_{13}) \frac{\partial^2 V}{\partial x \partial y} & = 0
\end{align*}
\]

(7a)
differentiation by $\partial^2/\partial x\partial y$ yields
\[
\begin{align*}
7b(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7b(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7c(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7c(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7d(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7d(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7e(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7e(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7f(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7f(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7g(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7g(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7h(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7h(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7i(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7i(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7j(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7j(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7k(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7k(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7l(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7l(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7m(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7m(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 \\
7n(a) & \quad c_{11}u_{xy} + c_{33}u_{yy} + (c_{12} + c_{33})u_{yx} = 0 \\
7n(b) & \quad c_{22}u_{xy} + c_{11}u_{yx} + (c_{12} + c_{33})u_{xy} = 0 
\end{align*}
\]
differentiation by $\frac{d^2y}{dx^2}$ yields
\[ c_{11}u_{x,xx} + c_{23}u_{x,xx} + (c_{12} + c_{33})u_{x,xx} = 0 \]  \( (7m) \)

differentiation by $\frac{d^3y}{dx^3}$ yields
\[ c_{11}u_{x,xxx} + c_{23}u_{x,xxx} + (c_{12} + c_{33})u_{x,xxx} = 0 \]  \( (7n) \)

It can be seen that the number of the equations which results from the performance of the above illustrated differentiation procedure together with the appropriate governing equations is:
\[ N_s = \frac{(m-k) + 1}{2} \frac{(n-k) + 2}{2} \]

for each one of the governing equations [equations (5a) and (5b)]. Here $m$ is the order of the Taylor series expansions and $k$ is the order of the derivative of the unknown variables in the governing equations. For example, it is evident from equations (5a) and (5b) that in the present example $k = 2$, in addition since we chose $m = 6$, the above expression implies that $N_s = 15$ as is the case in equations (5) and (7). It is interesting to note that development of the Modified Equations Partial Differential Equations (MEPEDE) involved successive differentiation of the kind used to develop equations (7a)–(7n).

Recall that as shown in the foregoing discussion, $[U_{x,xx}]$ and $[V_{x,xx}]$ approximate the values $u_{x,xx}$ and $v_{x,xx}$, respectively, at $(X, Y)$ i.e. at $X = 0$ and $Y = 0$. Therefore, if the approximations given by equations (6a) and (6b) are substituted into the governing equations (5a) and (5b) and their differentiated equations (7a)–(7n) and $x$ and $y$ are set to zero for the nodal point (i.e. the Taylor series approximations satisfy the differential equations at the nodal points) then one obtains a set of algebraic equations identical to equations (5) and (7) in which the coefficients $[U_{x,xx}]$, and $[V_{x,xx}]$, replace the derivatives $u_{x,xx}$, and $v_{x,xx}$, respectively. These set of equations will be referred to as the local governing equations.

For example, the governing equations (5a) and (5b) will become
\[ c_{11}[U_{x,xx}] + c_{23}[V_{x,xx}] + (c_{12} + c_{33})[U_{x,xx}] = 0 \]
\[ c_{11}[V_{x,xx}] + c_{23}[U_{x,xx}] + (c_{12} + c_{33})[V_{x,xx}] = 0 \]

Similarly, the differentiated equations given by (7a) will read
\[ c_{11}[U_{x,xxx}] + c_{23}[V_{x,xxx}] + (c_{12} + c_{33})[U_{x,xxx}] = 0 \]
\[ c_{11}[V_{x,xxx}] + c_{23}[U_{x,xxx}] + (c_{12} + c_{33})[V_{x,xxx}] = 0 \]

At this stage one has $N_s$ coefficients and $N_s$ local governing equations (e.g. 28 coefficients and 15 equations for each variable $u$ or $v$) in the present example. Consequently, in order to have a solvable set of equations one must generate $(N_s^2 - N_s)$ additional equations (e.g. 13 equations in the present example) for each variable.

The procedure for generating the additional 13 equations for each unknown is to use 14-point templates (e.g. central nodal point and 13 neighbouring points). This approach is common when finite difference methods are applied to an arbitrary mesh. For example see References 13, 25 and 26. For the reader’s clarity this is shown in the following section.
The fitting equations

The procedure for generating these additional equations is illustrated in Figure 2. At the first step, \((N_s - N_r)\) neighboring points indicated by \(J\) are selected for each central point \(I\). A way of selecting ensuring no impurity or ill-conditioning is described subsequently. Let us denote the horizontal and vertical distances from the central point, \(I\), to each neighboring point, \(J\), by \(h_x\) and \(h_y\), respectively. Since the values of \(u\) and \(v\) at point \(J\), as calculated by the Taylor series expansions from points \(I\) to \(J\), should be equal (within the error associated with the Taylor series expansions), one may write, in general, for each one of the neighboring points

\[
U_I(x_{I,J}, y_{I,J}) - U_I(x_{I,J}, y_{I,J}) = 0 \pm \epsilon \\
V_I(x_{I,J}, y_{I,J}) - V_I(x_{I,J}, y_{I,J}) = 0 \pm \epsilon
\]

(6a)

(6b)

where \(U_I(x_{I,J}, y_{I,J})\) and \(V_I(x_{I,J}, y_{I,J})\) can be simply obtained from equations (6a) and (6b), respectively, by inserting into them \(X = h_x\) and \(Y = h_y\). Similarly, \(U_J(x_{I,J}, y_{I,J}) = U_J(x_{I,J}, y_{I,J})\) and \(V_J(x_{I,J}, y_{I,J}) = V_J(x_{I,J}, y_{I,J})\) since for the neighboring point \(J\), \(X = 0\) and \(Y = 0\). These will result in the following set of \(J\) fitting equations for each variable.

\[
\sum_{s=1}^{N_s} \sum_{t=0}^{N_r} \frac{[U_{I+s}, t] - [U_{I-s}, t]}{w - kt} \cdot \frac{h_x^2}{h_y^2} = 0 \pm \epsilon \\
\sum_{s=1}^{N_s} \sum_{t=0}^{N_r} \frac{[V_{I+s}, t] - [V_{I-s}, t]}{w - kt} \cdot \frac{h_x^2}{h_y^2} = 0 \pm \epsilon
\]

(9a)

(9b)

where, for simplicity, \(h_x\) and \(h_y\), as denoted in Figure 2, are replaced by \(h_x\) and \(h_y\), respectively. The error, \(\epsilon\), is of the order \(O(\varepsilon^{j+1})\) where \(\varepsilon^n = h_x^n + h_y^n\).

The foregoing procedure of generating the fitting equations [(9a) and (9b)] should be applied to any internal nodal point (e.g. point \(I\) in Figure 2). For nodal points which are located on the boundary, a different procedure which eventually results in a further reduction in the total number of the required neighboring points is suggested in the following.

The use of the boundary conditions

Primary (displacement) boundary condition: Let us first consider a nodal point located on the boundary whose boundary conditions are given by equation (2). Point \(A\) in Figure 2 is such a nodal point. Since the displacements \(u\) and \(v\) (which are the unknown variables) given at this point are \(\delta\) and \(\delta\), respectively, we simply write

\[
[U_{I+A}] = \delta \\
[V_{I+A}] = \delta
\]

Hence, if point \(A\) is selected to be the neighboring point to an internal point, \([U_{I+s}, t]\) and \([V_{I+s}, t]\), in equations (6a) and (6b) are, respectively, replaced by \([U_{I+s}, t]\) and \([V_{I+s}, t]\).

Natural (traction) boundary condition: The case is different when the boundary conditions of a nodal point which is located on the boundary are given by equation (3). In this case we distinguish between a point at a regular boundary and a point at a corner (e.g. points \(B_1\) and \(B_2\), respectively, in Figure 2). A similar distinction was noted by Dow et al.\cite{7}
Regular boundary

For a nodal point such as \( B_k \) we have the following traction boundary conditions:

\[
\begin{align*}
(c_{11} u_x + c_{12} u_y) n_x + c_{22} (u_x + v_y) n_y - t_x &= 0, \\
(c_{33} u_x + c_{32} u_y) n_x + c_{34} (u_x + v_y) n_y - i_x &= 0.
\end{align*}
\]

Since the distributions of the boundary conditions along the boundary are also given and since the boundary conditions are differential equations having an order \( k \), where \( k > 1 \) is in the order of the governing equations e.g., \( k = 2 \) in equations (5a) and (5b) and \( k = 1 \) in equations (10a) and (10b), we can successively differentiate the boundary conditions \( (m - k) \) times along the boundary. This yields the following set of \( (m - k) \) equations for each one of the boundary conditions given by equations (10a) and (10b). Here, \( s \) is the tangential direction of the boundary at the point \( B_k \) as shown in Figure 2.

\[
\begin{align*}
\frac{d^m}{ds^m} \left[ (c_{11} u_x + c_{12} u_y) n_x + c_{22} (u_x + v_y) n_y \right] &= 0, \\
\frac{d^m}{ds^m} \left[ (c_{33} u_x + c_{32} u_y) n_x + c_{34} (u_x + v_y) n_y \right] &= 0.
\end{align*}
\]

Inserting the approximations given by equations (10a) and (10b) into equations (10) and (11) and setting \( X = 0 \) and \( Y = 0 \) results in a set of \( N_e = (m - k) + 1 \) equations for each of the unknown variables. These equations will be referred to in the following as the local boundary condition equations. Thus, for a nodal point on the boundary such as \( B_k \) (see Figure 2), one has \( N_e \) local governing equations and \( N_b \) local boundary condition equations. Consequently \( \{N_e - N_l + N_b\} \) neighboring points should be selected in order to generate the appropriate listing equations e.g., \( N_e = 28 \), \( N_l = 15 \), \( N_b = 6 \) and hence seven neighboring points are required in the present example. This results in local schemes (templates) of eight points.

The foregoing description of generating the local boundary condition equations is illustrated, for the reader's convenience, in the following example.

Let us consider a straight oblique boundary as the one shown in Figure 3. It could be shown that for any differentiable and continuous function \( f \) one can write

\[
\frac{df}{ds} = \frac{df}{dx} \frac{dx}{ds} + \frac{df}{dy} \frac{dy}{ds}
\]

It is clear from Figure 3 that

\[
\frac{dx}{ds} = -\sin \theta = -s_x, \quad \frac{dy}{ds} = \cos \theta = s_y
\]

Thus

\[
\frac{df}{ds} = -\frac{df}{dx} s_x + \frac{df}{dy} s_y
\]

Let us assume that the boundary conditions are given by

\[
\begin{align*}
I_x &= 3x + \sin y + y^2 + 2x^2, \\
I_y &= 5x^2 + \cos x + 4x^2y^2 + 6y + 12
\end{align*}
\]
Equations (11a) and (11b) imply for $n = 1$:

\[
-\frac{\partial}{\partial x} \left[ \nabla \cdot \mathbf{u} + c_{12} \rho \left( \mathbf{u} + \mathbf{v}_s \right) n_z \right] + c_{13} \left( \mathbf{u} + \mathbf{v}_s \right) n_x n_z
\]

\[
+ \frac{\partial}{\partial y} \left[ \nabla \cdot \mathbf{u} + c_{12} \rho \left( \mathbf{u} + \mathbf{v}_s \right) n_z \right] + c_{13} \left( \mathbf{u} + \mathbf{v}_s \right) n_y n_z
\]

\[
= -\frac{\partial}{\partial x} \left( \nabla \cdot \mathbf{u} \right) n_z + \frac{\partial}{\partial y} \left( \nabla \cdot \mathbf{u} \right) n_z
\]

Similarly,

\[
-\frac{\partial}{\partial x} \left[ \nabla \cdot \mathbf{u} + c_{12} \rho \left( \mathbf{u} + \mathbf{v}_s \right) n_z \right] + c_{13} \left( \mathbf{u} + \mathbf{v}_s \right) n_y n_z
\]

\[
+ \frac{\partial}{\partial y} \left[ \nabla \cdot \mathbf{u} + c_{12} \rho \left( \mathbf{u} + \mathbf{v}_s \right) n_z \right] + c_{13} \left( \mathbf{u} + \mathbf{v}_s \right) n_x n_z
\]

\[
= -\frac{\partial}{\partial x} \left( \nabla \cdot \mathbf{u} \right) n_z + \frac{\partial}{\partial y} \left( \nabla \cdot \mathbf{u} \right) n_z
\]

Inserting the boundary conditions given in the above example into these two expressions results in:

\[- c_{11} n_x n_y n_x + c_{11} n_x^2 - c_{33} n_y^2 n_x + c_{13} n_x n_y n_z \]

\[- c_{33} n_y^2 n_x + c_{33} - c_{12} n_x n_y n_z + c_{13} n_y n_z n_z \]

\[- (3 + 4xy) n_y + (cos y + 2y + 2x^2)n_y \]

and

\[- c_{11} n_x^2 n_y + c_{12} - c_{33} n_x n_y + c_{13} n_z^2 n_y \]

\[- c_{33} n_y^2 n_x + c_{33} - c_{12} n_x n_y + c_{13} n_y n_z n_z \]

\[- (10x - 3x^3 + 8xy) n_y + (8x^2 y + 6)n_y \]
Inserting the approximating Taylor series expansion given by equations (6a) and (6b) while setting $X = 0$ and $Y = 0$, i.e. $x = \lambda$, and $y = \mu$, since as mentioned earlier $X = x = \lambda$, and $Y = y = \mu$ yields

$$
- c_{11} n_{x} u_{1}(x, \lambda) + (c_{12} n_{x}^{2} - c_{13} n_{x} n_{y}) u_{2}(x, \lambda) + c_{21} n_{y} u_{1}(x, \lambda) = - (3 + 4 \lambda) n_{x} + (\cos \beta_{1} + 2 \lambda) n_{y},
$$

and

$$
- c_{11} n_{y}^{2} u_{2}(x, \lambda) + (c_{12} - c_{13} n_{x} n_{y}) u_{1}(x, \lambda) + c_{21} n_{x} u_{2}(x, \lambda) = - (10 \lambda x - 9 n_{y} + 6 n_{x}^{2}) n_{x} + (6 \lambda n_{y} + 6 n_{x}) n_{y}.
$$

Recall that in order to generate the full set of the local boundary conditions, equations (11a) and (11b) should be executed also for $n = 2, \ldots, 5$, since in the above example $m = 6$ and $k_{1} = 1$.

**Corner**

For a corner nodal point such as $B_{2}$, we have two sets of traction boundary conditions, one set to its left, $L$, and the other to its right, $R$. Each set is identical to equations (10a) and (10b) with its appropriate $n_{x}$ and $n_{y}$ (recall that $B_{2}$ is located at a corner).

Careful inspection of equations (10a) and (10b) indicates that they contain the following three unknowns: the normal stress $\sigma_{xx} = c_{11} n_{x} + c_{12} n_{y}$ and $\sigma_{yy} = c_{11} n_{y} + c_{12} n_{x}$ and the shear stress $\tau_{xy} = c_{13} n_{x} n_{y} + \tau_{yx}$. Thus, using the two sets (left and right) for a corner nodal point will result in four equations with only three unknowns. In fact these four equations are linearly dependent. Furthermore, it can be shown that any pair of them are linearly independent. Therefore, one of them should be arbitrarily dropped. Then differentiating each set with respect to $\lambda$ or $\mu$, respectively, (see Figure 2) results in $(4N_{x} - 1)$ local boundary condition equations $4N_{x}$ as a result of differentiating equations (10a) and (10b) to the left and to the right of the corner nodal point and $-1$ due to the arbitrary drop of redundant boundary conditions. Hence, for a corner nodal point such as $B_{2}$, one has $2N_{x}$ local governing equations and $(4N_{x} - 1)$ local boundary condition equations. Consequently, $[2N_{x} - (2N_{x} + 4N_{x} - 1)]$ fitting equations should be added [e.g. $N_{x} = 3$, $N_{y} = 15$, $N_{x} + 6$ and only three fitting equations are required in the present example]. This is done by selecting two neighbouring points and fitting to one of them only one of the two equations given by equation (9a) and (9b). This nodal point acts in fact like a 'half' neighbouring point.

In summary, for the illustrated case, i.e. the equilibrium equations in terms of displacements in plane elasticity, as given by (5a) and (10b), and the appropriate boundary conditions, there is a need for 13 neighbouring points (14-point templates) for each internal nodal point (such as point $l$ in Figure 2), 7 neighbouring points (6-point templates) for a nodal point along a regular boundary (such as point $B_{1}$ in Figure 2) and 11 neighbouring points (2-point templates) for a corner nodal point (such as point $2$ in Figure 2). This is summarized for the reader's convenience in Table I.

It should be emphasized that for nodal points located on a boundary with natural (traction) boundary conditions, the proposed method requires to fit the approximating Taylor series expansion (6a) and (6b) to the governing equations (5a) and (5b), their differentiated equations (7a) and (7b), the boundary conditions (10a) and (10b) and their differentiated equations (11a) and
### Table 1. Summary of the required equations for the solution of various nodal points

<table>
<thead>
<tr>
<th></th>
<th>$A_i$</th>
<th>$B_i$</th>
<th>$C_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local governing equations (5) and (7)</td>
<td>$2N_a = 2\frac{1}{2}(m - k + 1)(m - k + 2)$</td>
<td>$2N_a = 2\frac{1}{2}(m - k + 1)(m - k + 2)$</td>
<td>$2N_a = 2\frac{1}{2}(m - k + 1)(m - k + 2)$</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>Local boundary condition equations (10) and (11)</td>
<td>$N_a = 0$</td>
<td>$2N_k = 2(m - k_1 + 1)$</td>
<td>$4N_k - 1 = 4(m - k_1 + 1) - 1$</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>Fitting equations (9)</td>
<td>$N_a = 2(N_a - N_s)$</td>
<td>$N_a = 2(N_a - N_s - N_r)$</td>
<td>$N_a = 2(N_a - N_s - 2N_r)$</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>Required number of neighbour points</td>
<td>$N_a = (N_a - N_s)$</td>
<td>$N_a = (N_a - N_s - N_r)$</td>
<td>$N_a = N_s - N_s - 2N_r + \frac{1}{2}$</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>23</td>
<td></td>
</tr>
</tbody>
</table>

$m$—the order of the Taylor expansion ($m = 6$ in the numerical example)

$k$—the order of the differential governing equations ($k = 2$ in the numerical example)

$k_1$—the number of differential boundary condition equations ($k_1 = 1$ in the numerical example) note $k_1 < k$

$k_s$—the required number of neighbour points

$r$—the number of coefficients of the sixth Taylor series ($r = 28$) for each variable

$s$—the number of local governing equations for each governing equation

$N_a$—the number of local boundary conditions; equations that can be obtained from one traction boundary condition equation for each surface

$N_s$—the required number of fitting equations ($N_s = 2N_r$)
In addition, there is a requirement to fit the various Taylor series expansions which are developed from its neighbouring points and for one developed from it (i.e., the fitting equations). The above is accomplished without the need to use fictitious nodal points. Note that the use of fictitious nodal points increases the order of the error in addition to the redundant computations associated with them. This was clearly described by Noye and Arndt (1999). The above-mentioned requirements together with the fact that the proposed method does not require the use of fictitious nodal points results in an improvement in the boundary value approximation with respect to ordinary finite difference methods.

**Solution Method**

The foregoing described procedure enables one to generate 6 equations for each one of the N nodal points. Thus, the $N^6$ unknown coefficients can be solved, e.g., by inverting a $56N 	imes 56N$ matrix. For any reasonable value of $N$ this would be a very inefficient and ineffective way. Due to this complexity it is common to define a local matrix for each nodal point ($56 \times 56$ in the present illustration) and then to further reduce it.

Kochi and others (11) gave a procedure by which the solution can be reduced to the inversion of a $54 \times 54$ local matrix for each one of the $N$ nodal points and eventually obtain 2N equations with 2N unknowns, i.e., $[U_{n0}]$, and $[Y_{n0}]$, for each one of the $N$ nodal points. Similar approaches can be seen in References 21 and 26.

In the following we present and formulate the method by which the solution is further reduced to the inversion of a $26 \times 26$ matrix for a type $I$ nodal point, $14 \times 14$ matrix for a type $II$ nodal point and $3 \times 3$ matrix for a type $III$ nodal point. (Recall that the nodal types are defined in Figure 2.) Note that the matrix size is identical to the number of the fitting equations which are required. The formulation should not be regarded as a conditional part of our proposed method.

In the authors' view it reduces the computation volume. However, in principle, any other way of solving the equations can be adopted.

**Proposed solution method**

The set of equations at each nodal point which is composed of local governing equations, local boundary condition equations and fitting equations can in general be written as

$$[K][A] = [F]$$

where $[K]$ is a matrix of order $56 \times (56 + N_f)$, here $N_f$ is the number of the fitting equations (for details, see Table 1). $[A]$ is a column vector of dimension $(56 + N_f)$ containing the $56$ unknowns of the nodal point and the $N_f$ unknowns of the neighboring points, i.e., $[U_{n0}]$, and $[Y_{n0}]$, for $j = 1, \ldots, N_f$, where $N_f$ the number of the neighboring points is equal to $4N_f$.

We now divide the vector $[A]$ into three subvectors; $[A_1]$ contains all the $[U_{n0}]$ and $[Y_{n0}]$ values (both of the ith nodal points and its $N_s$ neighboring points) and its dimension is $2(N_s + 1)$ or alternatively $(N_f + 2)$. The rest of the unknowns (i.e., $[U_{n1}]$ and $[Y_{n1}]$ for which $p + q = 0$) are divided in the following way: the first $2N_f$ unknowns of each of two Taylor series expansions given by equations (6a) and (6b) form the vector $[A_2]$ which therefore has a dimension $N_{f}$ and the rest of them form the vector $[A_3]$ whose dimension is $(54 - 16)$. 

\[11\]
As a result of this division, equation (2) breaks into the following form:

\[
\begin{bmatrix}
N_1 + 2 & N_4 & 54 - N_4 \\
N_1 & N_4 & K_{12} & K_{33} & \{A_1\} & \{F_1\} \\
N_1 & N_4 & K_{22} & K_{33} & \{A_2\} & \{F_2\} \\
54 - N_4 & K_{33} & \{A_3\} & \{F_3\}
\end{bmatrix}
\]  

(13)

Alternatively, this can be split into the following three matrix equations:

\[
[K_{11}] \{A_1\} + [K_{12}] \{A_2\} + [K_{13}] \{A_3\} = \{F_1\}  
\]  

(14a)

which are the two approximations of the governing equations (5a) and (5b).

\[
[K_{22}] \{A_1\} + [K_{23}] \{A_2\} + [K_{23}] \{A_3\} = \{F_2\}  
\]  

(14b)

which are the fitting equations (9a) and (9b), and

\[
[K_{33}] \{A_1\} + [K_{33}] \{A_2\} + [K_{33}] \{A_3\} = \{F_3\}  
\]  

(14c)

which are the remaining local governing equations (7a) and (7b), and the local boundary condition, equations (10a), (10b), (11a) and (11b), when the nodal point is on the boundary.

Prior to proceeding with the solution of equations (14a) and (14b) it should be noted that both \([K_{11}]\) and \([K_{13}]\) are in general zero matrices. This is due to the fact that based on the procedure by which \([K_{11}]\) is generated, it contains only derivatives of the unknowns. Equation (14a) which contains only low-order derivatives, i.e. those appearing in the governing equations, implies that in general \([K_{11}] = 0\) (e.g. second-order derivatives in the present example). In addition, it could be seen from the fitting equations (9a) and (9b) that the vector \([F_2] = 0\).

Inserting these facts into equation (14a) and (14b) and solving the entire set for \([A_2]\) finally results in:

\[
([K_{11}] - [K_{12}] [K_{22}]^{-1} [K_{12}]^{-1} [K_{11}] - [K_{13}] [K_{33}]^{-1} [K_{13}]^{-1} [K_{11}]^{-1} [K_{22}] [K_{33}]^{-1} [K_{23}] [K_{33}]^{-1} [K_{23}]^{-1} [K_{33}]^{-1} [K_{33}]^{-1} [F_1]) = \{A_1\} 
\]  

(15)

The matrix equation (15) which applies to each of the \(N\) nodal points consists of two equations which, as mentioned earlier, contain only \([U_{0,0}]\) and \([V_{0,0}]\) unknowns. Thus at this stage we have \(2N\) equations with \(2N\) unknowns. These equations can be written in the following matrix form:

\[
[G] [\vec{x}] = [S]  
\]  

(16)

where \([\vec{x}]\) is the unknown vector (it contains \([U_{0,0}]\) and \([V_{0,0}]\)) and \([G]\) and \([S]\) are simply obtained from equation (15). These equations can be solved by standard solution methods (e.g. iterative or direct methods).

Once the values of \([U_{0,0}]\) and \([V_{0,0}]\), are obtained (i.e. \([A_1]\) is known) for all the \(N\) nodal points, the appropriate approximations of the derivatives \([U_{1,0}]\) and \([V_{1,0}]\) where \(p + q \neq 0\) could be obtained from equations (14a), (14b) and (14c) by solving for \([A_2]\) and \([A_3]\).

It should be pointed out here that the coefficient matrices \([K_{11}]\), \([K_{12}]\), \([K_{22}]\), \([K_{23}]\), \([K_{33}]\) and \([K_{33}]\), which arise from the approximations of the governing equations are the same for all the nodal points; the coefficient matrices \([K_{11}]\), \([K_{12}]\), \([K_{22}]\) and \([K_{23}]\) are the same for all the \(I\)-type points (see Figure 2a) and only the coefficient matrices \([K_{33}]\), \([K_{33}]\) and \([K_{33}]\) which arise from the fitting equations differ from point to point as they depend on the geometry of the investigated problem, or on the
relative location of the neighbouring points. This minor complexity can be simplified by
arranging the nodal points in such a way that identical geometrical patterns of neighbouring
points (templates) are generated. Therefore, the coefficient matrices \([ K_{11} ]\), \([ K_{12} ]\) and \([ K_{13} ]\)
should be calculated only once, the coefficient matrices \([ K_{11} ]\), \([ K_{12} ]\) and \([ K_{13} ]\) should be
calculated only once for all the \(t\)-type nodal points. Note that this implies that \([ K_{13} ]\) which
appears in equation (15) should be inverted at most three times. The coefficient matrices \([ K_{21} ]\),
\([ K_{22} ]\) and \([ K_{23} ]\) should be calculated once for each geometrical pattern (template). The above
discussion implies that \([ K_{22} ] = [ K_{22} ] [ K_{13} ]^{-1} [ K_{23} ]\) which also appears in equation (15)
should be calculated and inverted only once for each template.

Note, that as mentioned earlier, the size of this coefficient matrix is \(26 \times 26\) for an \(t\)-type nodal
point, \(14 \times 14\) for a \(B_l\)-type nodal point and \(3 \times 3\) for a \(B_l\)-type nodal point. It is the only matrix
which should be inverted for each one of the different geometrical patterns. As a consequence, the
coefficient matrices in equation (13) are identical for each of the different geometrical patterns.
The only change between the various geometrical patterns is in the vectors \([ A ]\) and \([ f ]\).

**SINGULARITIES OR ILL-CONDITIONS IN MATRICES**

In general, two problems related to singularities or ill-conditions in matrices exist in numerical
methods. The first problem can appear in the local matrix while the second in the global one. In
the following they will be discussed separately.

*Local matrix*

The local matrix in the present solution method is the one marked by \([ X ]\) in equation (12). As
seen in equation (15) the proposed solution method involved the inversion of
\[ [ K_{13} ] = [ K_{13} ] [ K_{11} ]^{-1} [ K_{12} ]^{-1} \]

Therefore one must assure that the above matrix is not singular or ill-conditioned. (Recall that as
mentioned earlier \([ K_{13} ]\), which does not depend on the local schemes, cannot be singular.)

"Perone and Kao\(^{23}\) treated the singularity and ill-conditioning of their local matrix (which was
slightly different from the present one) and suggested a technique (criterion) to avoid the
singularity. Liszka and Orkisz\(^{24}\) proposed a different technique which they claimed was simpler
and quicker in addition to being more accurate.

Unfortunately, Perone and Kao\(^{23}\) criterion is limited to maximum eight neighbouring points
(i.e. templates of nine nodal points), and Liszka and Orkisz\(^{24}\) criterion does not always
guarantee non-singularity.

Perone and Kao\(^{23}\) criterion is illustrated in Figure 4. The area around the central nodal point,
\(C\), is divided into eight different zones each of which is a pie-shaped segment with a central
angle of \(45^\circ\). Segments I and II are bisected by the \(X\)-axis and segments III and IV are bisected by
the \(Y\)-axis. Segments V, VI, VII, VIII are bisected by 45° lines through the nodal point, \(C\). For an \(N\)-point
template (in which \(5 \leq N \leq 9\)) the criterion suggests to locate the first four neighbouring points in
segments I–IV and the rest of them in the other segments (at most one point per segment).

Since our method allows templates having more than nine points we propose to distribute the
neighbouring points in the following way. They should occupy all the eight segments and should
be located as far as possible from each other while being close enough to the central nodal point.
Global matrix

The global matrix, i.e., \([G]\) in equation (16), cannot be singular. It can, however, be ill-conditioned. The ill-conditioning can be weakened by ensuring that

\[|u_j| \geq |u_i|\]

for all the rows – i.e., \((\hat{u}_j, \hat{y}_j)\) are the terms of \([G]\). Note that the above requirement is not necessary. However, the more rows fulfilling it, the further away the matrix from ill-conditioning. By redistributing the neighbouring points, problematic rows which do not fulfill the above-mentioned requirement can be improved.

EXAMPLES, VERIFICATION AND DISCUSSION

In the following, two examples for which the analytical solution is known will be solved using the proposed method.

The first example is shown in Figure 5. A thin rectangular plate, having a dimension of \(l \times b\), is fixed at its left edge. The boundary conditions along its four edges are given by

- at \(x = 0, 0 \leq y \leq b\) \(\hat{u} = 0\) and \(\hat{\varepsilon} = 0\)
- at \(0 \leq x \leq l, y = 0\)

\[I_x = -\frac{2(2 + 3\eta)}{0} fx^3\] \(\text{and}\) \[I_y = -\frac{2(2 + 3\eta)}{(3 + 2\eta)} ea^3\]
at \( 0 \leq x \leq l, \ y = h \)

\[
\begin{align*}
\dot{i}_x &= \frac{2(2 + 3d) f^3}{v} - 10c \dot{x}^4 + \frac{20(1 + 2c)}{v} f h^3 x^3 \\
&\quad + \frac{20(1 + 2c)}{(3 + 2c)} c h^3 \dot{x}^2 \quad - 10f h^3 x - \frac{1}{(3 + 2c)} c h^3
\end{align*}
\]

and

\[
\begin{align*}
\dot{i}_x &= \frac{2(2 + 3d) x^3}{v} + \frac{10(2 + 2c)}{v} f h x^2 + 20d h^2 x^2 \\
&\quad - \frac{20(1 + 2c)}{(3 + 2c)} c h^3 \dot{x}^2 - \frac{10d}{(3 + 2c)} c h^3 x + 3bh^3
\end{align*}
\]

at \( x = l, \ 0 \leq y \leq b \)

\[
\begin{align*}
\dot{i}_x &= \frac{2}{v} f^3 \dot{y} + \frac{10}{(3 + 2c)} c h^3 + 20(2 + 2c) y^2 - \frac{20(1 + 2c)}{(3 + 2c)} c h^3 y \\
&\quad - \frac{10(1 + 2c)}{v} f h^3 y + 2c h^3
\end{align*}
\]

\[
\begin{align*}
\dot{i}_x &= \frac{2}{v} (2 + 3d) f^3 \dot{y} - 10h^3 x^3 + \frac{20(2 + 2c)}{(3 + 2c)} c h^3 y \\
&\quad + \frac{20(1 + 2c)}{(3 + 2c)} c h^3 y^2 - \frac{10c h^3 x - 2(2 + 3d) f}{c}
\end{align*}
\]

where \( f \) and \( b \) are shown in Figure 5, \( c \) and \( f \) can be arbitrarily chosen.

The analytical solutions of this problem for the displacements \( u(x, y) \) and \( v(x, y) \) could be obtained using the polynomials method as described in Chapter 3 of Reference 34. Following
their method, we derived the following analytical solution:

\[
\sigma(x, y) = \frac{(u + 1)^2}{(3 + 2\eta)E} \left[ - \frac{2(\eta + 1)(\varepsilon + 1)}{\eta} x^2 y^2 - \frac{10(\eta + 1)^2}{(3 + 2\eta)E} x^3 y^2 + \frac{2\alpha(t - 1)}{\eta} s_y \right]
\]

and

\[
\sigma(x, y) = \frac{4(\eta + 1)(\varepsilon + 2)}{(3 + 2\eta)E} x^2 y^2 - \frac{10(\eta + 1)^2}{(3 + 2\eta)E} x^3 y^2 + \frac{20(\eta + 1)(\varepsilon + 3)}{(3 + 2\eta)E} x^4 y^2
\]

The analytical solutions for the stress components are

\[
\sigma_x = 2x^2 y^2 - \frac{10(\eta + 1)^2}{(3 + 2\eta)E} x^3 y^2 - \frac{20(\eta + 1)}{(3 + 2\eta)E} x^4 y^2
\]

\[
\sigma_y = -\frac{2(\eta + 1)(\varepsilon + 1)}{\eta} x^2 y^2 + \frac{10(\eta + 1)^2}{(3 + 2\eta)E} x^3 y^2 + 20(\eta + 1)(\varepsilon + 3) x^4 y^2
\]

\[
\tau_{xy} = \frac{212 + 30(\varepsilon + 1)}{(3 + 2\eta)E} x^2 y^2 - \frac{20(\eta + 1)(\varepsilon + 3)}{\eta} x^3 y^2
\]

Note that the above solutions are for a case in which the body forces are assumed to be negligibly small.

The second example shown in Figure 6. As a matter of fact it is a part cut out from the plane of the first example (see dashed line in Figure 5). Its boundary conditions were \(u = 0\) and \(f = 0\) at \(x = 0\), \(v \leq y \leq b\). Along the other boundaries, the natural (traction) boundary conditions were calculated from the analytical solution given by equation (19).

This choice of the second example implies that its analytical solution is identical to that of the first example, i.e. equation (18). The second example was chosen in order to verify the applicability of the proposed method to inclined boundaries.

Both examples were solved numerically using the proposed method for \(c = 1\) and \(f = 1\). Sixty six nodal points were used in the first example and 52 in the second one. Their distributions in the appropriate computational domains are shown in Figures 5 and 6, respectively. Sixth-order Taylor series expansions were used in both solutions.

The numerical solutions of both examples were practically identical to the analytical ones. The only noticeable errors were due to the computer's round-off errors. These should not be surprising since the analytical solution [see equation (18)] is a sixth-order polynomial as are the Taylor series expansion approximations.
The use of standard finite difference methods or finite element methods (with eight-point elements) would inherently lead to such large errors since their order of approximations are lower (usually 2 or 3). The use of sixth-order Taylor series expansion approximations with standard finite element or finite difference methods would result in very complex local schemes (21 neighbouring points for every element or nodal point).

The above-described two problems were solved using MATLAB which very conveniently treats algebraic manipulations with matrices. For this reason the CPU time required for a complete calculation was not checked. The aim of the present paper is to present an idea for a possible improvement of the finite difference method and not to compete with computational time of other methods or computer programs. A good programmer may adopt the proposed method to generate a very efficient code.

Consequently, it is probable that in spite of the improvements of our proposed method over other standard methods, there might be cases in which similar results can be achieved using the same computer time but less programming effort, by increasing the number of nodal points with standard methods. However, it is clear that there will be situations when the new proposed method will provide the only feasible method when computing power is limited. In addition to solving engineering problems, the proposed method can be used as a tool for comparison between other numerical methods.

It should be noted here that although straight and oblique boundaries were used in the presented two examples, the method can be used with curved boundaries as well. Owing to the difficulty in differentiating the natural (traction) boundary condition equations along curved boundaries such an example was not demonstrated here.

CONCLUSIONS

A new idea for the numerical solution of partial differential equations of equilibrium problems was forwarded and presented. The idea is based on differentiation of both the governing and the
boundary condition equations. The proposed method was verified by comparing its results to the analytical results of two problems in elasticity. Since only two examples were checked and since the idea, to the best of our knowledge, is new, it is difficult to determine at this stage the full applicability of the proposed method.

The method would be most useful in equilibrium problems where the geometrical boundaries are curved and the boundary conditions along them have large gradients. This is because in the proposed method both the boundary conditions and their derivatives are approximated.

The advantages of the proposed method over others are: The order of approximations of the solution at nodal points located on the boundaries are the same as those of internal nodal points (in contrast to other finite difference methods in which the approximation of boundary nodal points are one order of magnitude lower than internal nodal points); fictitious nodal points are not required; The numerical algorithm for the approximation of the derivatives depends on the entire differential solution and therefore provides better approximations than those obtained from the fixed numerical algorithms of finite difference methods. The method uses more local equations and hence the solution depends less on the size of the mesh. The solution procedure requires the inversion of smaller matrices than those inverted in other numerical methods when the same order of approximations are used and the grid is not regular. The size of the band width of the finally generated global matrix is smaller than that of other numerical methods this is due to the use of less neighbouring points, i.e. smaller local schemes. Consequently, the improvement of the proposed solution procedure over other procedures becomes more and more pronounced as the number of the nodal points increases.

The disadvantages of the proposed method in relation to others are: It is relatively a cumbersome method in its implementation. Prior to its application it requires a large volume of analytical calculations, i.e. successive differentiation of both the governing and boundary condition equations. It requires a manual selection of the local schemes (i.e. templates) for each nodal point. (Note that this can be overcome by appropriate programming which will account for the criterion suggested to avoid singularities and ill-conditioning)

Because the new method describes a method for solving equilibrium problems when the spatial region is irregular in shape, and provides consistent accuracy everywhere, it will be useful to many researchers and engineers.

Finally, it should be noted that the present method (idea) has not been applied yet to non-linear differential equations where convergence must be guaranteed.

REFERENCES