Elementary applications of MATHEMATICA to the solution of elasticity problems by the finite element method

Nikolaos I. Ioakimidis

Division of Applied Mathematics and Mechanics, Department of Engineering Sciences, School of Engineering, University of Patras, GR-265 04 Patras, Greece

Publication details: This preprint was published in 1993 (after a review process, with the necessary amendments for its publication and in a definitive and appropriately formatted form) in the research journal Computer Methods in Applied Mechanics and Engineering published by the publishing house North-Holland (an imprint of Elsevier Science Publishers). The final publication details of this preprint as an article in the journal Computer Methods in Applied Mechanics and Engineering have as follows:

- DOI (Digital Object Identifier)-based link to the web page of this article at ScienceDirect: https://doi.org/10.1016/0045-7825(93)90139-O
- Copyright © 1991 by the author transferred to Elsevier Science Publishers B.V.

Therefore, whenever reference to the present preprint is to be made, this should be made directly to its above final and formal publication as an article in the journal Computer Methods in Applied Mechanics and Engineering.
ELEMENTARY APPLICATIONS OF MATHEMATICA TO THE SOLUTION OF ELASTICITY PROBLEMS BY THE FINITE ELEMENT METHOD

N. I. IOAKIMIDIS

Correspondence address: Prof. N. I. Ioakimidis
P. O. Box 1120
GR-261.10 Patras
Greece

Division of Applied Mathematics and Mechanics, School of Engineering, University of Patras, P. O. Box 1120, GR-261.10 Patras, Greece

Submitted 18 March 1991

The computer algebra system MATHEMATICA is used for the SAN (semi-analytical/numerical) solution of two simple elasticity problems, having been reduced to systems of linear algebraic equations by the finite element method. In both cases, one parameter was left in symbolic form and Taylor series expansions with respect to this parameter (either a material constant or a geometric parameter) were used in the SAN results. The Gauss-Seidel iterative method for systems of linear equations and its SOR variant were used for the solution of the systems of linear equations always in the SAN environment offered by MATHEMATICA. The SAN results were compared with the corresponding numerical results and they were found equally acceptable. The present approach, which can be generalized in a variety of ways, offers the advantage (over purely numerical techniques) that it permits the incorporation of symbolic parameters into the results of the finite element method.

1. Introduction—approach

The powerful finite element method for the solution of problems in elasticity, fluid mechanics and many additional areas of engineering (like heat transfer problems) is revisited. This method leads to the solution of a system of linear algebraic equations approximating the original elasticity (or any other) problem, which can be solved by standard direct or iterative methods. The disadvantage of this approach is simply that the obtained numerical results are valid only for concrete numerical values of the material constants (like the modulus of elasticity and the Poisson ratio), the loading conditions and, what is more important, the dimensions of the specimen under consideration. If anyone of the above
quantities changes, then the system of linear equations should be solved again. Moreover, the same approach does not permit the successful study of the influence of the parameters of the problem on the obtained results (for the stresses, displacements, stress concentration and intensity factors, etc.).

Because of the importance of the finite element method, we thought that it might be somewhat interesting to use computer algebra software during the solution of the systems of linear equations resulting by the finite element method. A variety of recent engineering applications of computer algebra software can be found in the books (ASME meeting proceedings) [1,2]. Furthermore, an excellent related review paper was also recently published [3]. The same kind of software has been already used in the finite element method since about twenty years. We have collected a moderate number of related papers [4–31], all of which are mainly concerned about the application of computer algebra software (or, better, concrete computer algebra systems) to the finite element method. These papers are mentioned in chronological order in the References section below. A large number of aspects in the finite element method is treated in sufficient detail in these papers by using all major computer algebra systems. We will not proceed to related details, some of which are indicated by the titles of these papers, but we can add that one of the most important problems having been studied in detail is that of the generation of the stiffness matrix (and related matrices) for the elements used, where a variety of symbolic computations and, mainly, symbolic integration and matrix operations, is necessary. The first papers in this area seem to have been published in 1971 and the related idea was suggested and referred to by Argyris, Zienkiewicz and Clough among others [6, p. 97; 9, p. 379]. Of course, beyond the evaluation of stiffness and related coefficients, computer algebra systems were also used in a variety of other cases inside the finite element method for the automatic preparation of the whole set of results just before the solution of the system of linear equations [4–31]. The symbolic system FINGER by Wang [13] is one of the most serious efforts for the automation of this first part in a finite element analysis. Very recently, Sharma and Wang [26] presented really admirable results on the application of computer algebra software to the finite element method including the iterative numerical solution of the resulting systems of linear equations. Yet, even these results seem to be restricted to numerical values for the unknown quantities in the finite element method.

Our sole aim here is to try to make a first step in the second part of a finite element analysis, that of the solution of the resulting system of linear equations. This step will permit us to obtain results of general validity (e.g., valid for varying elastic constants or dimensions of the specimen), a fact obviously interesting as a concept and even more useful in cases where the influence of such a parameter (material or geometric) on the results is of interest, possibly up to optimization techniques. We are completely unaware of related results, but, of course, we cannot exclude the existence of such results at least in the elementary closed-form level. In fact, also recently, Beltzer clearly indicated all of these possibilities [3, p. 124] although he restricted himself (at least in this review paper) to the evaluation of the stiffness matrix of a particular rectangular plate element (as far as finite elements are concerned).

Our approach is a generalization of related approaches having been used in other elasticity applications combined with numerical methods and leading to the so called SAN (semi-
analytical/numerical or symbolic-analytical-numerical) approach. Such results, where series expansions are used, were applied to the evaluation of stress intensity factors by the weight function approach [32–34] (where numerical integration is necessary) and to Cauchy-type singular integral equations [35,36] (where iterative algorithms were used). We can also add that three kinds of approximations have been already used: Taylor/Maclaurin series [32,35,36] (which are the simplest ones), Chebyshev series [33] and minimax approximations [34]. Finally, in a recent paper of us [37] we applied the SAN approach (with Taylor/Maclaurin approximations) to the solution of systems of linear equations. This method can be directly extended to the finite element method and this will be really done below. In passing, we can add that another review (beyond [3]) on computer algebra systems and their applications to mechanics can be found in [38].

In the next two sections, we will present the SAN results (by the above approach) in two concrete elementary finite element applications paying attention just to the solution of the resulting systems of linear equations. These elementary applications were obtained from [39]. More important is the selection of the computer algebra system MATHEMATICA [40,41] as our tool in the present applications. Beyond a slight recent experience of us in this particular system, we feel that it is a very powerful and modern computer algebra system essentially for every task in symbolic computations. But, what is much more important in our case, is that this system is very efficient in numerical computations as well, which are normally performed in the computer with an accuracy of about sixteen digits by using the computer’s own algorithms and not special algorithms as is the case with the competitive system MAPLE [42], an also efficient and recommended system (but much less rapid in the SAN environment, where numerical computations play a very significant rôle, essentially equivalent to that of symbolic computations).

Finally, following [37], we will use here the classical Gauss-Seidel method for the solution of linear equations and its successive overrelaxation (SOR) variant [43].

2. Variation of a material constant and of loading

As a first application, we consider a square plane isotropic elastic medium under symmetric conditions, having been divided into eight triangular elements. This problem is described in detail in [39, pp. 303-309] and concrete numerical applications are presented. We will not repeat the procedure here, but, rather, we will restrict ourselves to the resulting stiffness matrix $K$ in this problem, which has the form (found after an appropriate modification/generalization of the results of [39, p. 306] so that the Poisson ratio $\nu$ be a variable)

$$K = \frac{Eh}{2(1-\nu^2)} \begin{pmatrix} (3-\nu)/2 & (1+\nu)/2 & -(1-\nu)/2 & -(1-\nu)/2 \\ (1+\nu)/2 & (3-\nu)/2 & -(1-\nu)/2 & -(1-\nu)/2 \\ -(1-\nu)/2 & -(1-\nu)/2 & (3-\nu)/2 & (1+\nu)/2 \\ -(1-\nu)/2 & -(1-\nu)/2 & (1+\nu)/2 & (3-\nu)/2 \end{pmatrix}. \quad (2.1)$$

The system of linear equations has now the form
\[ KD = F, \]  
\hspace{1cm} (2.2) 

where \( D \) is the displacement vector \([39, \text{p. 306}]\)

\[ D = [v_1, u_3, u_4, v_4]^T \equiv 1/(Eh)[u[1], u[2], u[3], u[4]]^T \]  
\hspace{1cm} (2.3) 

(in our notation in the computer) and \( F \) is the loading vector assumed (contrary to \([39, \text{pp. 306–309}]\)) completely arbitrary, that is,

\[ F = [f_1, f_2, f_3, f_4]^T. \]  
\hspace{1cm} (2.4) 

Since the Poisson ratio \( \nu \) is of the order of 0.30, we found it convenient to use a Taylor/Maclaurin series (with 3 terms plus the constant term) for \( \nu \) about this point, that is,

\[ \nu = 0.30 + d\nu. \]  
\hspace{1cm} (2.5) 

We give below the whole MATHEMATICA procedure in the above problem by the Gauss-Seidel method, having also included a parameter \( \omega \) if we might wish to use its SOR variant \([43]\). In the present application, we do not use this variant of the Gauss-Seidel method (having assumed that \( \omega = 1 \)), since this was not found necessary. Moreover, \( m_0 \) denotes the maximum number of the permitted iterations in the Gauss-Seidel method.

```mathematica
FEPlane[omega_,m0_]:=Block[{},
 (* the Poisson ratio and the stiffness coefficients *)
 nu=0.30+dnu; aa=1/(2*(1-nu^2));
a1=aa*(3-nu)/2; a2=-aa*(1-nu)/2; a3=aa*(1+nu)/2;
 (* the stiffness matrix *)
 K=\{a1, a3, a2, a2\},
 \{a3, a1, a2, a2\},
 \{a2, a2, a1, a3\},
 \{a2, a2, a3, a1\};
 (* the dimension of the matrix and the zero vector *)
 n=Length[K]; tt=Table[0,\{i,n\}];
 (* the elements of the stiffness matrix *)
 Do[Do[k[i,j]=K[[i,j]],\{i,n\},\{j,n\}];
 (* initial (zero) values of the displacements *)
 Do[u[i]=0,\{i,n\}]; m=0;
 (* the main loop *)
 While[m<m0,\{m=m+1,\}
 Do[u[i]=Chop[Collect[u[i]+omega*Expand[N[Normal[Series[
 (f[i]-Sum[k[i,j]*u[j],\{j,n\}]/k[i,i],
 \{dnu,0,3\}]]],dnu]],\{i,n\}],
 (* test for the error *)
 error=Table[Chop[Collect[Expand[N[Normal[Series[
}}]
```
We present also below the convergence of the iterations for $u[1]$ only and for increasing values of the iteration index $m$ (up to $m = 10$):
\begin{align}
  u[1] &= 0 \\
  &\quad \text{for } m = 0, \\
  u[1] &= 1.3481 f_1 - 0.38957 d v f_1 - 1.6258 d v^2 f_1 - 0.60214 d v^3 f_1 \\
  &\quad \text{for } m = 1, \\
  u[1] &= 1.7320 f_1 - 0.51151 f_2 + 0.18123 f_3 + 0.34952 f_4 \\
  &\quad + d v(-0.040465 f_1 - 0.93286 f_2 - 0.43593 f_3 - 0.47086 f_4) \\
  &\quad + d v^2(-1.3238 f_1 + 0.78819 f_2 - 0.046833 f_3 - 0.45160 f_4) \\
  &\quad + d v^3(-1.1570 f_1 + 1.5606 f_2 + 0.46172 f_3 + 0.27877 f_4) \\
  &\quad \text{for } m = 2, \\
  u[1] &= 1.8082 f_1 - 0.72110 f_2 + 0.20609 f_3 + 0.22917 f_4 \\
  &\quad + d v(0.26330 f_1 - 1.2992 f_2 - 0.48340 f_3 - 0.47674 f_4) \\
  &\quad + d v^2(-0.75539 f_1 + 0.61442 f_2 + 0.10887 f_3 + 0.22905 f_4) \\
  &\quad + d v^3(-1.1386 f_1 + 1.2414 f_2 + 0.17044 f_3 + 0.18288 f_4) \\
  &\quad \text{for } m = 3, \\
  u[1] &= 1.8179 f_1 - 0.77299 f_2 + 0.19392 f_3 + 0.18261 f_4 \\
  &\quad + d v(0.39890 f_1 - 1.4553 f_2 - 0.45172 f_3 - 0.44715 f_4) \\
  &\quad + d v^2(-0.42287 f_1 + 0.19109 f_2 + 0.19707 f_3 + 0.30813 f_4) \\
  &\quad + d v^3(-0.74716 f_1 + 0.73092 f_2 + 0.0037383 f_3 + 0.070601 f_4) \\
  &\quad \text{for } m = 4, \\
  u[1] &= 1.8184 f_1 - 0.78120 f_2 + 0.18543 f_3 + 0.17755 f_4 \\
  &\quad + d v(0.44041 f_1 - 1.5202 f_2 - 0.44157 f_3 - 0.44946 f_4) \\
  &\quad + d v^2(-0.27861 f_1 - 0.058664 f_2 + 0.21992 f_3 + 0.23710 f_4) \\
  &\quad + d v^3(-0.39067 f_1 + 0.35761 f_2 + 0.029691 f_3 + 0.094659 f_4) \\
  &\quad \text{for } m = 5, \\
  u[1] &= 1.8184 f_1 - 0.78166 f_2 + 0.18242 f_3 + 0.17962 f_4 \\
  &\quad + d v(0.44956 f_1 - 1.5420 f_2 - 0.44463 f_3 - 0.45362 f_4) \\
  &\quad + d v^2(-0.22590 f_1 - 0.15835 f_2 + 0.21848 f_3 + 0.20212 f_4) \\
  &\quad + d v^3(-0.19469 f_1 + 0.12143 f_2 + 0.076446 f_3 + 0.095455 f_4) \\
  &\quad \text{for } m = 6, \\
  u[1] &= 1.8184 f_1 - 0.78150 f_2 + 0.18165 f_3 + 0.18095 f_4 \\
  &\quad + d v(0.45098 f_1 - 1.5476 f_2 - 0.44850 f_3 - 0.45365 f_4) \\
  &\quad + d v^2(-0.20931 f_1 - 0.19080 f_2 + 0.21231 f_3 + 0.19645 f_4) \\
  &\quad + d v^3(-0.11392 f_1 - 0.00062444 f_2 + 0.091467 f_3 + 0.078352 f_4) \\
  &\quad \text{for } m = 7, \\
\end{align}
\[ u[1] = 1.8185 f_1 - 0.78146 f_2 + 0.18151 f_3 + 0.18137 f_4 \\
+ \nu (0.45116 f_1 - 1.5486 f_2 - 0.45042 f_3 - 0.45251 f_4) \\
+ \nu^2 (-0.20479 f_1 - 0.20006 f_2 + 0.20751 f_3 + 0.19870 f_4) \\
+ \nu^3 (-0.086615 f_1 - 0.051273 f_2 + 0.088333 f_3 + 0.069413 f_4) \\
\text{for } m = 8, \quad (2.7i) \]

\[ u[1] = 1.8185 f_1 - 0.78147 f_2 + 0.18149 f_3 + 0.18147 f_4 \\
+ \nu (0.45123 f_1 - 1.5487 f_2 - 0.45108 f_3 - 0.45174 f_4) \\
+ \nu^2 (-0.20367 f_1 - 0.20244 f_2 + 0.20491 f_3 + 0.20108 f_4) \\
+ \nu^3 (-0.078551 f_1 - 0.068506 f_2 + 0.082251 f_3 + 0.069285 f_4) \\
\text{for } m = 9, \quad (2.7j) \]

\[ u[1] = 1.8185 f_1 - 0.78148 f_2 + 0.18148 f_3 + 0.18148 f_4 \\
+ \nu (0.45127 f_1 - 1.5487 f_2 - 0.45126 f_3 - 0.45142 f_4) \\
+ \nu^2 (-0.20339 f_1 - 0.20301 f_2 + 0.20379 f_3 + 0.20238 f_4) \\
+ \nu^3 (-0.076303 f_1 - 0.073475 f_2 + 0.078260 f_3 + 0.071751 f_4) \\
\text{for } m = 10. \quad (2.7k) \]

The convergence of the above values of \( u[1] \) to their correct values (in five digits), given by (2.6a), is clear from the above results (2.7).

Furthermore, the above results contain, beyond the main variable \( \nu \) (that is, the Poisson ratio), the loading vector \( \mathbf{F} \), given by (2.4), as well. Therefore, they are valid for any physically admissible loading conditions. Finally, the symbolic parameter (product) \( Eh \) should also be included in the displacement vector \( \mathbf{D} \), because of (2.3), but, of course, the main variables are just \( \nu \) and \( \mathbf{F} \).

3. Variation of a geometric parameter

As a second application of the present approach we consider the problem (in plate bending analysis) of a rectangular isotropic elastic plate with its edges simply supported and loaded with a uniformly distributed normal loading of intensity \( q \) per unit area acting over the whole area of the plate. This elementary problem was solved by the finite element method (by using four appropriate rectangular plate elements) in [39, pp. 425-426].

The set of the unknown quantities consists in the present case of the rotations \( \phi_2 \) and \( \theta_4 \) as well as the deflection at the centre of the plate (maximum deflection in the plate) \( w_3 \) [39, p. 425]. In [39] the plate was assumed to be square during the numerical solution with the ratio of the lengths \( 4a \) and \( 4b \) of its edges equal to 1 [39, p. 426]. Here we will consider again the corresponding system of linear equations, but without this assumption, permitting the ratio \( a/b \) to get an essentially arbitrary value. More explicitly, we assume the dimensions of the plate to be

\[ 4a = A(1 + d) \quad \text{and} \quad 4b = A \quad (3.1) \]
in such a way that the quantity $p$ in [39, p. 420] takes the value

$$p = \left(\frac{a}{b}\right)^2 = (1 + d)^2.$$  \hspace{1cm} (3.2)

As far as the Poisson ratio $\nu$ is concerned, we assumed it to be equal to 0.30, whereas all additional constants are assumed arbitrary although this is not of particular importance.

Now we obtain a typical system of three linear equations of the form (2.2), but with [39, p. 426]

$$\mathbf{K} = \begin{pmatrix} b^2(80p + 16 - 16\nu) & b(-60p - 6 + 6\nu) & 0 \\ b(-60p - 6 + 6\nu) & 60p + 60p^{-1} + 42 - 12\nu & a(-60p^{-1} - 6 + 6\nu) \\ 0 & a(-60p^{-1} - 6 + 6\nu) & a^2(80p^{-1} + 16 - 16\nu) \end{pmatrix},$$  \hspace{1cm} (3.3)

$$\mathbf{D} = [\phi_2, w_3, \theta_4]^T \equiv [u[1], u[2], u[3]]^T$$  \hspace{1cm} (3.4)

and

$$\mathbf{F} = (60qa^2b^2/D_0)[b/3, 1, a/3]^T \equiv [f[1], f[2], f[3]]^T,$$  \hspace{1cm} (3.5)

$D_0$ denoting—as usual—the flexural rigidity of the plate [39, pp. 286, 412].

For the solution of the above-described system of stiffness equations in our elementary application we found it convenient to use the SOR (successive overrelaxation) method, a variant of the Gauss-Seidel iterative method [43, pp. 437-438]. This is a usual practice in finite element problems. The whole MATHEMATICA procedure is displayed below, where $\omega$ denotes again the overrelaxation factor and $m_0$ the maximum permitted number of iterations. This procedure is sufficiently analogous to that in the previous section and, therefore, no comments are included here.

```plaintext
FEPlate[omega_,m0_] := Block[{},
  nu=0.30; b=A/4; a=b*(1+d); p=(a/b)^2;
  K={
    {b^2*(80*p+16-16*nu), b*(-60*p-6+6*nu), 0},
    {b*(-60*p-6+6*nu), 60*p+60/p+42-12*nu, a*(-60/p-6+6*nu)},
    {0, a*(-60/p-6+6*nu), a^2*(80/p+16-16*nu)}
  };
  n=Length[K]; tt=Table[0,{i,n}];
  Array[u,n]; Array[f,n]; Array[k,n,n];
  c=60*q*a^2(b^2/D0); f[1]=c*b/3; f[2]=c; f[3]=c*a/3;
  Do[k[i,j]=K[[i,j]],{i,n},{j,n}];
  Do[u[i]=0,{i,n}]; m=0;
  While[m<m0, {m=m+1},
    Do[u[i]=Chop[Collect[Collect[u[i]+omega*Expand[N[Normal[Series[
          (f[i]-Sum[k[i,j]*u[j],{j,n}]/k[i,i],
          {d,0,20}]]],d],{q,A,D0}]],{i,n}]],
    error=Table[Chop[Collect[Collect[Expand[N[Normal[Series[
          Sum[k[i,j]*u[j],{j,n}]-f[i],
          {d,0,20}]]],d],{q,A,D0}]]10^-6],{i,n}]
  ];
  If[error==tt,Break[]][]
  ]
```

8
We display below the results that we obtained by the SOR method (with \( \omega = 1.25 \)) in four significant digits. With zero initial values for \( u[i] \) (\( i = 1, 2, 3 \)) 27 iterations were required for the above accuracy in the series of the geometric parameter \( d \) (with powers up to \( \Delta d^{20} \)):

\[ \phi_2 \equiv u[1] = (A^3 q/D_0)[0.01768 + 0.03099d + 0.001557d^2 - 0.01353d^3 + 0.007787d^4 + 0.001436d^5 - 0.005998d^6 + 0.005548d^7 - 0.002831d^8 + 0.0001289d^9 + 0.001621d^{10} - 0.002351d^{11} + 0.002297d^{12} + 0.001743d^{13} + 0.0009635d^{14} - 0.0002016d^{15} - 0.0003704d^{16} + 0.0006789d^{17} - 0.0007450d^{18} + 0.0006484d^{19} - 0.0004821d^{20} + O(d^{21})], \]  

\( (3.6a) \)

\[ w_3 \equiv u[2] = (A^4 q/D_0)[0.005063 + 0.01013d + 0.0004946d^2 - 0.004569d^3 + 0.002898d^4 - 0.001518d^5 + 0.001518d^6 + 0.0008355d^7 + 0.0001534d^8 + 0.0002682d^9 - 0.0004293d^{10} + 0.0004077d^{11} - 0.00002811d^{12} + 0.0001163d^{13} + 0.00003113d^{14} + 0.0001238d^{15} - 0.0001499d^{16} + 0.0001499d^{17} - 0.0001215d^{18} + 0.0006458d^{19} - 0.000006458d^{20} + O(d^{21})], \]  

\( (3.6b) \)

\[ \theta_4 \equiv u[3] = (A^3 q/D_0)[0.01768 + 0.02206d - 0.007376d^2 + 0.001779d^3 + 0.007787d^4 - 0.009223d^5 + 0.004661d^6 + 0.0003499d^7 + 0.003094d^8 + 0.0035356d^9 - 0.002674d^{10} + 0.001319d^{11} - 0.0005568d^{12} - 0.0008087d^{13} + 0.001165d^{14} - 0.001067d^{15} + 0.0006840d^{16} - 0.0002195d^{17} - 0.0001572d^{18} + 0.0003574d^{19} - 0.0003771d^{20} + O(d^{21})]. \]  

\( (3.6c) \)

The above SAN results were found to be in complete agreement with the numerical results presented in [39, p. 426], but for \( d = 0 \) only, although (because of the fact that only four rectangular elements were used) their accuracy is very low. In fact, for \( w_3 \) the above finite element solution is about 25% in error (for \( d = 0 \) [32, p. 426]) compared with the corresponding exact value reported in [44, p. 120].

We tested further this error, taking always into account the results in [44, p. 120] for \( w_3 \) and comparing these results to ours. We found that the relative error \( \epsilon \) is equal to 24.7% for \( d = 0, 25.3\% \) for \( d = 0.1, 25.5\% \) for \( d = 0.2, 26.1\% \) for \( d = 0.3, 27.2\% \) for \( d = 0.4, 27.5\% \) for \( d = 0.5, 28.5\% \) for \( d = 0.6, 29.4\% \) for \( d = 0.7, 30.6\% \) for \( d = 0.8, 32.0\% \) for \( d = 0.9 \) and \( 34.3\% \) for \( d = 1.0 \). These results show a very slight increase of the relative error \( \epsilon \) with
increasing values of $d$. A small part of this increase is due to the finite number of terms in the series for $w_3$, which is rather slowly convergent in the present application. Better (smaller) values for $\epsilon$ were obtained with 31 terms in the power series (3.6b) for $d > 0.8$, e.g., $\epsilon = 32.3\%$ (instead of 34.3%) for $d = 1.0$.

4. Conclusions

We showed above that it is possible to derive SAN results including a parameter in its symbolic form in two elementary problems of finite element analysis. It is our opinion that this will permit a greater flexibility in the finite element method. Although we used Taylor series in our expansions, the use of alternative approaches is also possible. Similarly, although we used iterative algorithms for the solution of the systems of linear equations, the use of direct algorithms (like Gaussian elimination) seems not presenting any essential difficulty. Another generalization concerns the case of using two or more nontrivial symbolic parameters in our results. It seems that this case will require much more effort in the computer, but, essentially, it does not present any theoretical difficulty. Beyond the basic (and varying) parameter the loading may be completely arbitrary as well and additional symbolic parameters can appear too. Moreover, in the case of a geometric parameter, the dimensions of the finite elements are also variable!

Concerning computer time, it is acknowledged that the present approach requires much more computer time than the purely numerical approach. But, the generality of the obtained results, valid for a large interval in the selected symbolic parameter, as well as the continuous—we dare say monthly—significant increase of the potential of modern computers (about at least 30 MIPS are easily accessible in modern RISC workstations of really moderate value) permit us to claim that SAN computations in finite element analysis may become a useful tool in future. Finally, the really astonishing facilities offered by modern computer algebra systems and, especially, MATHEMATICA [40,41] and including, beyond strong symbolic and numerical possibilities, excellent graphics, \TeX{} and POSTSCRIPT output and the additional possibility for the direct preparation of papers and books (MATHEMATICA Notebooks), by mixing computer output (formulae and graphics) and text, clearly permit the prediction of a real boom in the area very soon. This author was already really surprised by the variety of related results contained in [1–3].

References


12


(Manuscript prepared by \texttt{\LaTeX-VANILLA};
the related computer files are \texttt{available}!)