Proceedings of the International Conference for Agricultural Machinery & Process Engineering
October 19-22, 1993, Seoul, Korea

MICROCOMPUTER FEM ANALYSIS OF SOIL CUTTING PROCESS

J.Shen R.L.Kushwaha
Agricultural and Bioresource Engineering Department
University of Saskatchewan
Saskatoon, SK, Canada, S7N 0W0

ABSTRACT

Current finite element analysis programs for soil cutting process with tillage tools require mainframe computers. Several special treatments in developing a microcomputer FEM program were introduced to increase the capacity for solving large problems and reducing the total time cost. The program was evaluated by solving one 3-D example on a 486 microcomputer. The result showed a close agreement with the laboratory soil bin test.

INTRODUCTION

A large amount of energy is spent in soil preparation for crop production each year. Many attempts have been made to optimize tillage tool design. Due to the complexity of soil behavior, conventional analytical methods have not been successful in providing all the information about the tillage process. Many researchers used finite element method to carry out the prediction of soil cutting process(Yong and Hanna, 1977; Liu and Hou, 1985; Xie and Zhang, 1985; Chi and Kushwaha, 1989; Wang and Gee-clough, 1991). However, almost all of these programs were developed for use on the mainframe computers. These FEM programs limit their availability to persons who do not have access to the mainframe computers or not willing to pay the cost of using the mainframe computer in their daily design activities.

The memory and running speed of microcomputers are continuously increasing and have reached a level that allows large FEM problems to be solved after some special treatments. For example, Table 1 shows the time cost of two test programs under different hardware environment. The first program was a test for the running speed of mathematical operations in which 15000×15000 multiplications were carried out. The time cost of 486/50 microcomputer was only a little longer than SPARC station 2 in network. There is also 486/66 microcomputers available in the market now. According to Table 1, its running speed is expected the same as the SPARC station 2 in network. The second program was an I/O type test in which two hundred records were written to and read from an unformatted and directly-accessed file for twenty times, with each record including one hundred integers. The time cost of this kind of I/O operation on PC 486 was lower than that on both SPARC station 2 and VAX 3100. Therefore, it is possible to develop a microcomputer FEM analysis program for soil cutting without much extra time cost in comparison with workstations.

The objective of this paper is to introduce the authors' efforts in developing a microcomputer FEM program to increase its capacity for solving large problems and to reduce the time cost for the whole calculation under DOS operating system.

STRATEGIES FOR THE MEMORY MANAGEMENT

The DOS operation system was originally designed to run on Intel86-family processors in 16-bit real mode, which supported a physical address space of 1 megabyte. DOS divides the address space into 640 Kilobytes of RAM for application programs and 384 Kilobytes for ROM and hardware subsystems. For users' program, there are only about 500 Kilobytes available which approximately corresponds to the size of a double precision array A(62000) and is a rather small address space for today's sophisticated applications.

In this study, the variable band and LDL^T method were adopted as the solver of FEM problem. The capacity of variable band method can be extended by blocking the stiffness and mass arrays (Mondkar and Powell, 1974). Then, only two of these blocks are necessary in RAM at any one time instead of the entire set of equations. However, this will increase I/O operations considerably and thus increase the time cost.

An alternative to the blocked variable band method is frontal method. The frontal scheme works element by element, forming only the part of stiffness matrix belonging to the front. But the running speed is usually lower than the variable band method.

The Intel386 and 486 processors can operate in 32-bit protected mode, where the physical address space is 4 gigabytes, and where 32-bit operations are significantly faster than their 16-bit counterpart. Thus, in this study, a 32-bit Fortran compiler was chosen in which DOS extender placed the processor in 32-bit protected mode. Since the current prices of Microcomputer RAM are in the order of $50.00 per megabyte(MB), the total RAM in 386 or 486 can be extended to 32 MB without much cost. In this way, the 640K memory barrier was broken.

In addition, the code for virtual memory management (VM) in 32-bit Fortran compiler was chosen by default, where disk space was viewed as a logical extension to physical memory, just like on minicomputers and mainframes. The paging hardware of the 386 and 486 processors was used to map the virtual address to logical program addresses, allowing users' application programs of virtually unlimited size.

A FEM program should be able to solve both small and large problems. Thus, different size of arrays should be declared according to the size of the problem. A single array was partitioned to store all the main arrays in the program. Each main array was dynamically dimensioned to the size required for each problem by using a set of pointers. In this way, no space was wasted in data storage and a maximum amount of space was reserved to store the global arrays(Zienkievicz and Taylor, 1989).

STRATEGIES FOR REDUCING CALCULATING TIME

One existing problem with finite element analyses is that it takes a long time to solve a problem compared to the traditional analytical method. Even on the mainframe computers, the calculating time is still very long because each user can only share a part of the total CPU time. This would discourage designers to carry out comparative designs repeatedly in a short period. This is the reason that the designers have not adapted finite element method (FEM) in tillage tool designs. Hence, the calculating speed of the FEM analysis is a key factor in determining whether this method will be accepted by designers or not.

General specification

The following three criteria have been extensively used to reduce the computation time:

(1) In a program, multiplication and division operations usually take longer time than addition and substraction operations, although the detailed difference between these two kinds of operation varies among different computer architectures. The first effort has been made to reduce multiplication and division operations. The term, multiplication and division operations, has been abbreviated to m/d operations in the following text.

(2) Using high-dimensional array variables requires more time than using simple variables or low-dimensional array variables. In this program, the variables were used in the priority sequence of simple, low-dimensional array and high-dimensional array variable.

(3) All middle files written to the hard drive were opened in the unformatted form. Test result on PC 486/33 indicated that the time cost of I/O operations of a formatted directly-accessed file was 2.3 times as long as that of the corresponding unformatted file.

Shape function and inverse Jacobian matrix

The shape function of 8-node 3-D isoparametric element is

$$N_i = \frac{1}{8}(1+\xi_i\xi)(1+\eta_i\eta)(1+\zeta_i\zeta) \qquad (i=1,8) \tag{1}$$

where, N_i= shape function.

Eq.(1) can be changed to the following form in order to decrease one m/d operation:

$$N_i = (0.5+S_iS)(0.5+T_iT)(0.5+Z_iZ) \tag{2}$$

where, $S_i = 0.5*\xi_i$ = known constant
$S = \xi$
$T_i = 0.5*\eta_i$ = known constant
$T = \eta$
$Z_i = 0.5*\zeta_i$ = known constant
$Z = \zeta$

Similar changes can be made to the equation of $N_{i,\xi}$,$N_{i,\eta}$ and $N_{i,\zeta}$ to save another three m/d operations. Furthermore, by using mid-variables MS, MT and MZ in the following equation, the calculation amount of $N_{i,\xi}$,$N_{i,\eta}$ and $N_{i,\zeta}$ at one Gaussian point will be reduced from 16 m/d to 11 m/d operations.

$$\begin{aligned} MS &= 0.5 + S_iS \\ MT &= 0.5 + T_iT \\ MZ &= 0.5 + Z_iZ \\ N_i &= MS \times MT \times MZ \\ N_{i,\xi} &= S_i \times MT \times MZ \\ N_{i,\eta} &= MS \times T_i \times MZ \\ N_{i,\zeta} &= MS \times MT \times Z_i \end{aligned} \tag{3}$$

Instead of using the general subroutine to get inverse matrix, the inverse Jacobian matrix was directly derived as follows to save many m/d operations.

$$[J]^{-1} = \frac{1}{|J|}\begin{bmatrix} Y_{,\eta}Z_{,\zeta}-Z_{,\eta}Y_{,\zeta} & Z_{,\xi}Y_{,\zeta}-Z_{,\zeta}Y_{,\xi} & Z_{,\eta}Y_{,\xi}-Z_{,\xi}Y_{,\eta} \\ X_{,\zeta}Z_{,\eta}-Z_{,\zeta}X_{,\eta} & X_{,\xi}Z_{,\zeta}-Z_{,\xi}X_{,\zeta} & X_{,\eta}Z_{,\xi}-X_{,\xi}Z_{,\eta} \\ X_{,\eta}Y_{,\zeta}-Y_{,\eta}X_{,\zeta} & X_{,\zeta}Y_{,\xi}-X_{,\xi}Y_{,\zeta} & X_{,\xi}Y_{,\eta}-X_{,\eta}Y_{,\xi} \end{bmatrix} \tag{4}$$

where, $|J|$ = Jacobian determinant;

$[J]^{-1}$ = inverse Jacobian matrix.

$X_{,\xi} = \partial X/\partial \xi$ (other partial derivatives were defined in the same way)

Element stiffness matrix

In 3-D cases, element strain submatrix Bi, elasticity matrix D and element stiffness submatrix K_{ij} can be written as

$$[B_i] = \begin{bmatrix} N_{i,x} & 0 & 0 \\ 0 & N_{i,y} & 0 \\ 0 & 0 & N_{i,z} \\ N_{i,y} & N_{i,x} & 0 \\ 0 & N_{i,z} & N_{i,y} \\ N_{i,z} & 0 & N_{i,x} \end{bmatrix} \quad (i=1,8) \tag{5}$$

$$[D] = \begin{bmatrix} G_1 & G_2 & G_2 & 0 & 0 & 0 \\ G_2 & G_1 & G_2 & 0 & 0 & 0 \\ G_2 & G_2 & G_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & G_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & G_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & G_3 \end{bmatrix} \tag{6}$$

$$[K_{ij}] = \int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} [B_i]^T [D] \, [B_j] \, |J| \, d\xi \, d\eta \, d\zeta \tag{7}$$

where, $N_{i,x}$, $N_{i,y}$ and $N_{i,z}$= derivative of N to x,y and z respectively;

$G_1 = E_t(1-\nu)/(1+\nu)/(1-2\nu)$

$G_2 = E_t\nu/(1+\nu)/(1-2\nu)$

$G_3 = E_t/2/(1+\nu)$

ν = Poisson's ratio

If common matrix operation is used, the calculation of $[B_i]^T[D][B_j]$ will need 3x6x6x3=324 m/d operations. By noticing that there were many zeros in both B_i and D matrix, $[B_i]^T[D][B_j]$ was written as follow:

$$[B_i]^T[D][B_j] = \begin{bmatrix} N_{i,x}N_{j,x}G_1+N_{i,y}N_{j,y}G_2+N_{i,z}N_{j,z}G_3 & N_{i,x}N_{j,y}G_2+N_{i,y}N_{j,x}G_3 & N_{i,x}N_{j,z}G_2+N_{i,z}N_{j,x}G_3 \\ N_{i,y}N_{j,x}G_2+N_{i,x}N_{j,y}G_3 & N_{i,y}N_{j,y}G_1+N_{i,x}N_{j,x}G_3+N_{i,z}N_{j,z}G_3 & N_{i,y}N_{j,z}G_2+N_{i,z}N_{j,y}G_3 \\ N_{i,z}N_{j,x}G_2+N_{i,x}N_{j,z}G_3 & N_{i,z}N_{j,y}G_3 & N_{i,z}N_{j,z}G_1+N_{i,y}N_{j,y}G_3+N_{i,x}N_{j,x}G_3 \end{bmatrix} \quad (8)$$

According to Eq.(8), the calculation of $[B_i]^T[D][B_j]$ only required 42 m/d operations. In addition, by referring to the symmetry of element stiffness matrix, only half of element stiffness matrix needed to be calculated. This further reduced the m/d operations to 21.

Storage and solving of Global stiffness matrix K

Since K matrix was sparse, most elements in K matrix were zeros. Variable band method was adopted to store K matrix which excluded many useless zero elements. In addition, according to symmetry of K matrix, only half of K matrix needed to be stored. LDL^T triangle decomposition method was used to solve basic analytical equations.

In solving process, there were three layers of loops. In the innermost loop, the execution times of any kind of calculation is proportional to N^3 (cubic of equation number) operations. Therefore, in the program, calculations were tried to be performed in the outermost or middle loop which corresponded to N and N^2 operations respectively.

Element strain and stress calculations

The following equations were derived to reduce 72 and 24 m/d operations from 144 and 36 m/d operations of an ordinary matrix multiplication for strain and stress calculation respectively.

$$[B_i]\{\delta_i\} = \begin{bmatrix} N_{i,x} & 0 & 0 \\ 0 & N_{i,y} & 0 \\ 0 & 0 & N_{i,z} \\ N_{i,y} & N_{i,x} & 0 \\ 0 & N_{i,z} & N_{i,y} \\ N_{i,z} & 0 & N_{i,x} \end{bmatrix} \begin{Bmatrix} u_i \\ v_i \\ w_i \end{Bmatrix} = \begin{bmatrix} u_i N_{i,x} \\ v_i N_{i,y} \\ w_i N_{i,z} \\ u_i N_{i,y} + v_i N_{i,x} \\ v_i N_{i,z} + w_i N_{i,y} \\ u_i N_{i,z} + w_i N_{i,x} \end{bmatrix} \quad (i=1,8) \quad (9)$$

$$
\begin{bmatrix} d\sigma_x \\ d\sigma_y \\ d\sigma_z \\ d\tau_{xy} \\ d\tau_{yz} \\ d\tau_{zx} \end{bmatrix} = \begin{bmatrix} G_1 & G_2 & G_2 & 0 & 0 & 0 \\ G_2 & G_1 & G_2 & 0 & 0 & 0 \\ G_2 & G_2 & G_1 & 0 & 0 & 0 \\ G_2 & G_2 & G_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & G_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & G_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & G_3 \end{bmatrix} \begin{bmatrix} d\varepsilon_x \\ d\varepsilon_y \\ d\varepsilon_z \\ d\tau_{xy} \\ d\tau_{yz} \\ d\tau_{xz} \end{bmatrix} = \begin{bmatrix} G_1 d\varepsilon_x + G_2 d\varepsilon_y + G_2 d\varepsilon_z \\ G_2 d\varepsilon_x + G_1 d\varepsilon_y + G_2 d\varepsilon_z \\ G_2 d\varepsilon_x + G_2 d\varepsilon_y + G_1 d\varepsilon_z \\ G_3 d\tau_{xy} \\ G_3 d\tau_{yz} \\ G_3 d\tau_{xz} \end{bmatrix} \quad (10)
$$

FEM ANALYSIS AND RESULTS

A 3-D example of soil-cutting problem with a vertical flat blade was considered to demonstrate the capability of the program developed.

Material nonlinearity

In order to obtain information of every stages of soil-cutting process, an incremental method was used to solve soil nonlinearity. The total horizontal movement of flat blade was divided into several smaller increments. During each increment, a standard Newton iteration method was used to solve equilibrium equations.

Geometric nonlinearity

During continuous soil cutting with a blade, large soil displacement occurred and geometric non-linearity may make a larger analytic error to FEM prediction. Therefore, the update Lagrangian method (Bathe, 1976; Yamada, 1972) was used to make continuous node coordinate transformation after each increment.

Mesh information and boundary condition

Figure 1 shows a 3-D finite element mesh for simulating soil-cutting process with a 50 mm wide vertical blade operating at 100 mm depth. The boundary conditions of the 3-D analysis were as follows:

1. The nodes on bottom surface were fixed in vertical (Z) direction.
2. The nodes on front and rear surface were fixed in travel (X) direction.
3. The nodes on symmetric and left side surface were fixed in sideways (Y) direction.
4. The nodes on blade-soil interface had a specified displacement in X direction during each increment.
5. The nodes on side wall of furrow were fixed in Y direction.

6. All other nodes were free in three direction.

Analytical results

The FEM analysis was carried out on 33 MHz DataTrain DC 520 microcomputer with 80486 CPU. Source code was written in Fortran and compiled by SVS Fortran 2.8.2. The calculating time of the example for six incremental steps was 32 minutes and 48 seconds. If PC 486/66 is used, the time cost will approximately be reduced to 16 minutes. It may be stated that the calculating time was tolerable for designers to do comparative design work.

Fig.2 shows the draft prediction of narrow blade in the 3-D case shown in Fig.1. The dash line only stands for the maximum value of draft obtain from soil-bin test and the analytical result was close to this limit.

CONCLUSIONS

A microcomputer FEM program was written with the capability for solving large problems and reducing time costs. One 3-D case was calculated on 486 microcomputer. The result showed that the time cost was tolerable for designer to do comparative designs.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial assistance received from the Natural Sciences and Engineering Research Council of Canada and the President's NSERC funds for this research.

REFERENCES

Bathe, K.J. 1976. ADINA, static and dynamic geometric and material a nonlinear analysis, Report No. 82448-2, M.I.T., Cambridge, MA.

Chi,L. and Kushwaha,R.L. 1989. Finite element analysis of a plane soil blade. Can. Agric. Eng. Vol.30(2):125-130.

Chi,L. and Kushwaha,R.L. 1991. Finite element analysis of soil force on two tillage tools, Can. Agric. Eng. Vol.33(1):39-45.

Duncan, J.M. and Chang, C.Y. 1970. Nonlinear analysis of stress and strain in soils. J. Soil Mech. Found. Div., Am. Soc. Civil Engrs. Vol.96:1629-1653.

Liu,Y. and Hou,Z.M. 1985. Three dimensional nonlinear finite element analysis of soil cutting by narrow blades. Proc. of Int. Conf. on Soil dynamics, Auburn, Alabama, Vol.2:412-427.

Mondkar, D.P. and Powell, G.H. 1974. Towards optimal in-core eqution solving. Comp. Struc. 4:531-548.

Wang,J. and Gee-clough, D. 1991. Deformation and failure in Wet clay soil, II.

Simulation of Tine Soil Cutting. IAMC conference, Beijing, China, 2-219-2-226.

Xie,X.M. and Zhang,D.J. 1985. An approach to 3-D nonlinear FEM simulative method for investigation of soil-tool dynamic system. Proc. of Int. Conf. on soil dynamics, Auburn, Alabama. Vol.2:322-327.

Yamada, Y. 1972. Incremental formulation for problems with geometric and material nonlinearities. in: Advances in Computational Methods in Structural Mechanics Design, Univ. of Alabama Press, 325-355.

Yong,R.N. and A.W.Hanna. 1977. Finite element analysis of plane soil cutting. Journal of Terramechanics. Vol. 14(3):103-125.

Zienkiewicz, O.C. and Taylor, R.L. 1989. The Finite Element Method. McGraw-Hill Book Company.

Table 1 Time cost of two test programs under different hardware environment

Computer name	Status	Program 1 Time cost (m:s:s/100)	Program 2 Time cost (m:s:s/100)
PC 486/33	stand alone	5:04:23	0:11:53
PC 486/50	stand alone	3:23:94	
SPACstation2	in network	2:51:65	0:15:99
VAX 3100 Model 90	in network	0:54:42	0:27:70

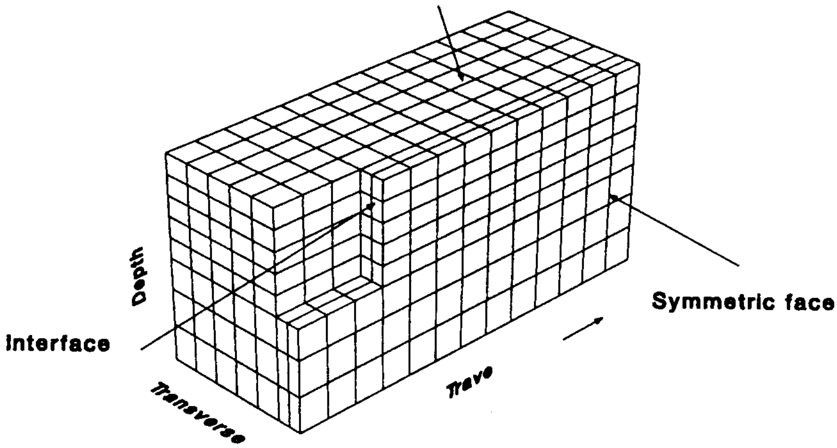

Figure 1. Three-dimensional FE mesh for soil cutting

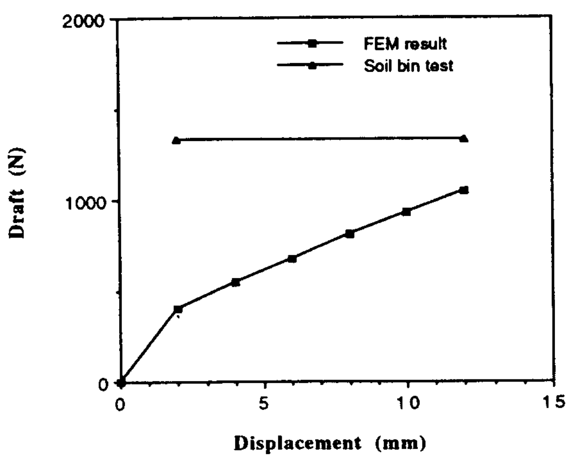

Figure 2. Draft prediction by 3-D FEM analysis for narrow vertical blade