GENERAL PURPOSE COMPUTER PROGRAM FOR DYNAMIC NONLINEAR ANALYSIS

by

D.P. Mondkar[I] and G.H. Powell[II]

SYNOPSIS

This paper presents a brief description of the concepts, features, organization and usage of a general purpose finite element computer program ANSR (Analysis of Nonlinear Structural Response) for the static, dynamic and earthquake response analysis of nonlinear structures. The program is intended ultimately to form a practical analysis tool for use in design studies as well as to satisfy research needs in various aspects of nonlinear analysis.

INTRODUCTION

Because of the need for rational investigations of nonlinear structures subjected to severe loading conditions such as would occur in strong motion earthquakes, a study was initiated with the objective of developing a general purpose computer program for nonlinear structural analysis. This study progressed through two phases. Phase one consisted of review and development of theories, computational techniques and algorithms that can be applied in nonlinear structural analysis, and phase two resulted in development of a general purpose computer code based on the studies in phase one. The results and findings of these two study phases have been documented in [1,2]. The purpose of this paper is to summarize the features of the computer program ANSR [2].

THEORETICAL BASIS

The program is based on the finite element discrete equations obtained from the variational form of the incremental equations of motion described in a Lagrangian coordinate system. In the Lagrangian description, the kinematics of deformation of a body is described by three configurations, namely, the initial configuration, current deformed configuration and a neighboring configuration; the initial undeformed configuration C_o (time $t = 0$) is taken as the reference configuration and the kinetic variables are the conjugate pair consisting of the second (symmetric) Piola-Kirchhoff stress and Lagrangian strain.

The incremental equations can be used to study nonlinearities due to large displacements, large strain effects and/or nonlinear material behavior. The material nonlinearity is specified by an appropriate relationship between stress and strain.

[I] Assistant Research Engineer, Division of Structural Engineering and Structural Mechanics, Department of Civil Engineering, University of California, Berkeley, California, U.S.A.

[II] Professor of Civil Engineering, University of California, Berkeley, California, U.S.A.

The dynamic response is obtained by a step-by-step numerical integration of the equations of motion using Newmark's generalized implicit operator. Equilibrium iterations can be performed, if desired, in each time step to satisfy equilibrium subject to a specified tolerance. A variety of iterative schemes are included so as to give considerable flexibility to the analyst to design a solution scheme giving results as accurately as possible with minimum computation times.

PROGRAM FEATURES AND LIMITATIONS

A. Structural Idealization

(1) The structure to be analyzed is idealized as an assemblage of discrete finite elements connected at nodes. The theory and solution procedure are based on the finite element formulation of the displacement method, with the nodal displacements as the unknown field variables.

(2) Each node may possess up to six displacement degrees of freedom, as in a typical three dimensional frame analysis.

(3) Certain degrees of freedom can be deleted or combined. This feature provides the program user ample flexibility in the idealization of the structure, and may permit the size of the problem to be substantially reduced.

(4) The structure mass is assumed to be lumped at the nodes only, so that the mass matrix is diagonal. The program could be modified to consider a coupled (consistent) mass matrix.

(5) Viscous damping effects proportional to mass, initial elastic stiffness and/or tangent stiffness can be included. These effects may be specified to vary in magnitude from one group of elements to the next.

B. Static and Dynamic Loadings

(1) Loads are assumed to be applied only at the nodes. Static and/or dynamic loads may be specified; however, static loads, if any, must be applied prior to the dynamic loads.

(2) For static analysis, a number of static force patterns must be specified. Static loads are then applied in a series of load increments, each load increment being specified as a linear combination of the static force patterns. This feature permits nonproportional loads to be applied. Each load increment can be specified to be applied in a number of equal steps.

(3) The dynamic loading may consist of earthquake ground accelerations time dependent nodal loads, and prescribed initial values of the nodal velocities and accelerations. These dynamic loadings can be specified to act singly or in combination.

(4) Earthquake excitations are defined by time histories of ground acceleration. Three different time histories may be specified, one for each of the X, Y and Z axes with which the structure geometry is defined. For any given axis, all support points are assumed to move identically

and in phase. The accelerations for any time history may be specified at equal time intervals (as in an artificially generated earthquake record) or at unequal time intervals (as in a measured earthquake record).

(5) Any number of time histories of dynamic force may be specified. As with the earthquake records, these time histories may be input at equal or unequal time intervals. Any dynamic force record may be prescribed to act at a node or a group of nodes, either as forces in the X, Y or Z directions, or as moments about the X, Y or Z axes.

(6) Values of initial translational and/or rotational velocity and acceleration may be specified at each node. Structures subjected to impulsive loads can be analyzed by prescribing appropriate initial velocities. For the case of static analysis followed by dynamic analysis, the displacements at the start of the dynamic analysis are assumed to be those at the end of the static analysis.

C. Finite Element Library

(1) The finite element library is limited at the time of writing, consisting of the following.

(a) Three dimensional truss element, which may yield in tension and yield or buckle elastically in compression. Large displacement effects may be included.

(b) Three dimensional beam-column element, which may be of variable cross section and strength, and which yields through the formation of concentrated plastic hinges at its ends. The inelastic behavior is considered using interaction between axial force and major axis bending moment only, and this interaction may be specified to be either for cross sections of steel or concrete. Eccentricities at the element ends may be specified for analysis of coupled frame-shear wall structures.

(c) Two dimensional variable number (4-to-8) node finite element for plane stress and plane strain analysis. Large displacement effects may be included. The material description may be specified to be isotropic linearly elastic, orthotropic linearly elastic, or isotropic elasto-plastic with either von Mises or Drucker-Prager yield function.

(d) Two dimensional variable number (4-to-8) node finite element for axisymmetric solid or shell analysis. Large displacement effects may be included. The material descriptions available are the same as those for two dimensional plane stress/strain analysis.

(e) Three dimensional variable number (8-to-20) node solid element for analysis of solids and thick shells/plates. Large displacement effects may be included. The material law may be specified to be isotropic linearly elastic, orthotropic linearly elastic or isotropic elastic-plastic with von Mises yield function.

(f) Boundary (spring) element for use in specifying flexible supports, prescribed displacements, and for idealizing gaps between the structure and supporting system.

(2) The program is organized to permit the addition of new finite elements to the library with relative ease, without modifying the original program. Also, for the currently developed elements in the library, new material constitutive laws can be added by developing a few new "material" subroutines.

(3) Nonlinearities are introduced at the element level only, and may be due to large displacements, large strains and/or nonlinear materials. The programmer adding a new element may include any type or degree of nonlinearity in the behavior of the element.

D. Solution Procedure

(1) The program incorporates a solution strategy defined in terms of a number of control parameters. By assigning appropriate values to these parameters, a wide variety of solution schemes, including step-by-step iterative and mixed schemes, may be constructed. This permits the program user considerable flexibility in selecting optimal solution schemes for particular types of nonlinear behavior.

(2) For static analysis, a different solution scheme may be employed for each load increment. The use of this feature can reduce the solution time for structures in which the response must be computed more precisely for certain ranges of loading than for others. In such cases a sophisticated solution scheme with equilibrium iteration might be used for the critical ranges of loading, whereas a simpler step-by-step scheme without iteration might suffice for other loading ranges.

(3) The dynamic response is computed by step-wise time integration of the incremental equations of motion using Newmark's β-γ-δ operator. A variety of integration operators may be obtained by assigning appropriate values to the parameters β and γ. However, the most commonly used scheme will be the "constant average acceleration" scheme ($\beta=1/4$, $\gamma=1/2$, $\delta=0$). Viscous damping effects may be introduced by specifying a positive value to the parameter δ. In most cases, however, damping effects will be introduced more explicitly, in mass proportional and/or stiffness proportional form.

E. Other Features

(1) Data checking runs may be made prior to execution runs. During data checking, the program reads and prints all input data, and also prints any generated data, but performs no substantial analysis.

(2) In its current version the program requires that the stiffness matrix be stored in core. This matrix is stored column-wise in a compacted form omitting most zero elements. Because the stiffness matrix is updated, rather than completely reformed, as the tangent stiffness changes, it is also necessary to save a duplicate stiffness matrix. Updating of the matrix requires least numerical operations if the duplicate stiffness matrix can be held in core. However, this may not always be possible, and hence provision is made for the user to specify whether the duplicate matrix should be stored in core or on disc. Modifications for very large systems, using

blocking and out-of-core storage of the stiffness matrix, are planned for a future version of the program.

(3) During solution, the decomposition (triangularization) of the structure stiffness matrix is carried out on only that part of the updated stiffness matrix which follows the first modified coefficient. Significant savings in solution time can sometimes be obtained by numbering those nodes connecting nonlinear elements to be last, so that the operations on the structure stiffness matrix are limited to the end of the matrix.

(4) The element information is stored either in core or on disc. When stored on disc, the information is blocked to minimize input/output cost for data transfer between core and disc storages. Thus, the number of finite elements is not limited by the available core storage, except that this storage must be sufficient to hold the information for at least one element.

(5) Because the structure stiffness matrix is stored in compacted form, rather than in banded form, there will be relatively small penalties in storage requirements and equation solving time if there are local increases in the matrix band width. Hence, if a few nodes are to be added to a structure for which an input data deck has already been prepared, the additional computational cost incurred by numbering these nodes last may be less than the man-hour cost involved in renumbering all of the nodes and preparing a new input data deck.

(6) Because all nonlinearities are introduced at the element level, and because the structure stiffness matrix is updated, not reformed, due to nonlinearities, there is no loss of efficiency in computing dynamic response of purely linear structures.

(7) At present no restart capability is included in the program. Such a capability will be added in a future version.

PROGRAM STRUCTURE

The program is divided into two parts, namely, (1) the base program consisting of a series of subroutines performing specific tasks required for static and dynamic analysis, and (2) a number of auxiliary programs, each program consisting of a package of subroutines required for a specific type of finite element in the element library. The base program reads and prints the structure geometry and loading data, carries out a variety of bookkeeping operations, assembles the structure stiffness matrix and loading, and using the user-specified solution scheme computes the displacement response of the structure. The base program is then combined with element subroutines of the auxiliary programs to produce a complete package. The auxiliary program for any one specific type of element performs four main functions, namely, (1) reading and printing of element data, (2) computing tangent stiffness, (3) calculating new state, and (4) outputing response results. Each of these functions is performed by a separate subroutine, and information returned to the base program. The rules for the linkage and information transmittal between the base program and the auxiliary program have been simplified, so that auxiliary programs for new structural elements can be developed and added to the base program relatively easily. The currently developed auxiliary programs also have the capability of accepting new material constitutive laws. This is achieved by performing all computations associated with a particular material in a package of "material" subroutines, and returning the information to the auxiliary program. Again, rules for the linkage and transmittal of information between the "material" subroutines and the auxiliary program have been designed to give the programmer the ability to code new material models with relative ease. The capability to add new structural elements and new material models is an important feature of the program for use as a research tool as well as a practical analysis tool.

The available core storage is allocated dynamically at execution time. As mentioned previously, the element data is blocked to reduce input/output cost for data transfer between disc and core storage. If all element data can be held in core then no data transfer operations are required. The program incorporates efficiently designed computational algorithms for equation solving and stress computations.

SAMPLE APPLICATIONS

The program has been used to analyze a number of structures with widely different nonlinear characteristics. The main objective of these analyses has been verification of the program features and the various solution schemes implemented in the program. Wherever possible the results of these analyses have been compared with the analytical, numerical and/or experimental studies of other investigators, and close agreement has been obtained; see reference [2] for descriptions of these analyses.

FUTURE DEVELOPMENTS

A number of new capabilities will be added to the program. These include restart options, new constitutive material laws such as elasto-plastic isotropic and kinematic hardening models, a parallel component model, and soil material models. The program will also be modified to handle out-of-core blocking of the structure stiffness matrix for use in large scale structures. A special purpose version of the program is also being developed for exclusive use in three dimensional nonlinear analysis of building systems. Thermal stress analysis capability including some simple thermo-plastic material models have been developed. This paper should be regarded as a preliminary documentation paper. It is anticipated that the program ANSR will be continually updated to satisfy both research and industrial needs in nonlinear analysis.

ACKNOWLEDGEMENTS

The work presented in this paper was supported partly by the National Science Foundation, under Grant NSF-GI-36387. Computer Facilities were provided by the Computer Centers of the University of California, Berkeley and the Lawrence Berkeley Laboratory.

REFERENCES

1. Mondkar, D.P., and Powell, G.H., "Static and Dynamic Analysis of Nonlinear Structures", Report No. EERC 75-10, Earthquake Engineering Research Center, University of California, Berkeley, March 1975.
2. Mondkar, D.P., and Powell, G.H., "ANSR-I, General Purpose Program for Analysis of Nonlinear Structural Response", Report No. EERC 75-37, Earthquake Engineering Research Center, University of California, Berkeley, December 1975.