

## DRAG METHOD AS A FINITE ELEMENT MESH GENERATION SCHEME

S. PARK and C. J. WASHAM

Control Data Corporation, St. Paul, MN 55112, U.S.A.

(Received 18 May 1978)

**Abstract**—The "drag mesh" method for automatic generation of finite elements is presented. Highlights of the technique are: (1) simple, efficient element and node generation in regions of a structural model where a similarity of cross section is maintained; (2) exact model coordinate computation for surfaces and volumes of revolution and for many other doubly curve regions; (3) flexible user control of element and node numbering; (4) simultaneous generation of 1-, 2-, and 3-dimensional finite elements.

### INTRODUCTION

Most available methods for automatic generation of finite elements address specific classes of problems. For example, some schemes determine the interior node points of a region by interpolating from the distribution of nodes on the boundary; these methods deal well only with regions that are topologically equivalent to basic geometric shapes (cube, wedge, square, etc.).

One class of problems not addressed directly by many mesh generation methods is that in which there is a general cross-sectional similarity throughout portions of the model. The transition from one cross section cut to another is often easily expressed by a simple translation or rotation. This characteristic is particularly noticeable in many solid element models and also in models involving a surface or volume of revolution.

One method that has been used to address this class of problems is the two-step geometry/connectivity generation: first all the nodes are generated in space by the user and next the element connectivities are derived making use of the "proper" node numbering scheme. This method compromises usability by imposing a cumbersome node number bookkeeping on the user and also by severe requirements for model regularity, limiting the number of problems that can be addressed. Simple deviations from the prescribed requirements can lead to considerable difficulty and frustration in the modeling.

To make this particular class of problems easier to address, the authors have synthesized the "drag" automatic mesh generation method. Simply, this technique generates the prism-like finite elements described by the motion of a cross-sectional pattern as it is dragged in steps through space.

### GENERATRIX

**Generatrix:** a point, line, or surface whose motion generates a line, surface, or solid. (*Webster's New Collegiate Dictionary*). The motion of a generatrix of dimension  $n$  thus describes a prismatic element of dimension  $n + 1$ . For example, when a 4-node quadrilateral generatrix is moved through space in several increments, a sequence of 8-node hexahedron finite elements can be described as in Fig. 1. Note that the information for generating the hexahedron is completely contained in the topology of the associated quadrilateral generatrix and that there is no dependence on the node numbering of the generatrix.

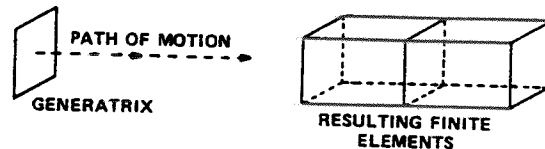


Fig. 1. Hexahedron generation.

### DRAG MESH

The basis of the drag mesh method is a pattern set of 0-dimension (point), 1-dimension (line), and 2-dimension (area) generatrix elements and a set of displacement (step, increment) control vectors. Starting with its original position, the generatrix set is dragged through a sequence of positions as determined by the succession of displacement control vectors: the first vector is added to the nodes in the original pattern position to determine the first transient position, the second vector is added to that position to determine the next position, and so on. Two successive positions of a generatrix element form two opposing faces/edges/ends of the new finite element and connecting the corresponding nodes of the pair forms the remaining faces/edges.

Two modes of dragging, rectangular and cylindrical, are easily defined. In rectangular drag, the set of displacement control vectors is taken to define a sequence of rectangular increments  $\Delta_i = (\Delta x_i, \Delta y_i, \Delta z_i)$  to be added successively to the pattern of generatrix elements. Figure 2 illustrates a rectangular drag of two 1-dimensional generatrix elements ( $G_i$  = generatrix,  $N_i$  = node,  $E_i$  = element).

In cylindrical drag, the set of displacement control vectors defines a sequence of cylindrical increments  $\Delta_i = (\Delta r_i, \Delta \theta_i, \Delta z_i)$  relative to a user-specified reference axis. Local cylindrical coordinates relative to the reference axis are used in applying these increments to compute the node positions. Figure 3 illustrates a cylindrical drag. Note that in the cylindrical mode, appropriate selection of the displacement control vectors can result in pattern motion which is strictly radial, angular, or axial or is any combination thereof. One very important feature of the cylindrical mode of drag is that a mesh on a doubly curved surface of volume of revolution can be computed exactly; poor results are normally obtained in methods that determine interior nodes by interpolating from boundary nodes.

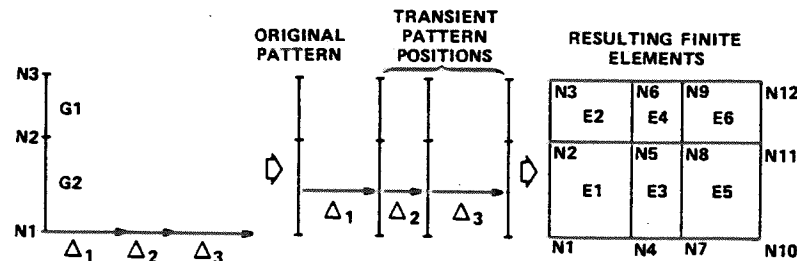


Fig. 2. Rectangular mode drag.

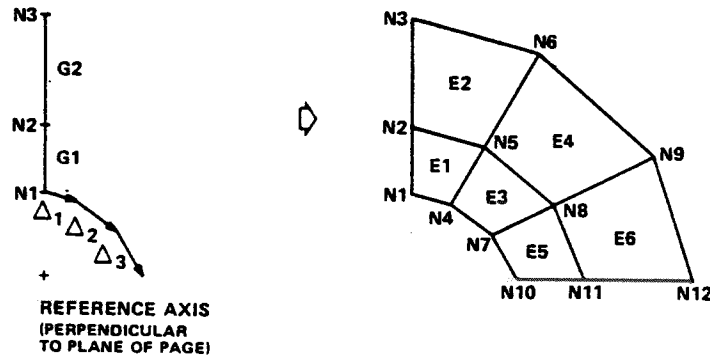


Fig. 3. Cylindrical mode drag.

The key to the simplicity and power of the drag mesh generation method is the generatrix. First, there is no dependence (and hence no resulting restrictions) on the geometry of the generatrix; all the information for determining the connectivity of a finite element created is contained in the topology/connectivity of the one corresponding generatrix. Further, there is no dependence on the numbering of nodes in the generatrix. Each generatrix is treated completely separately, so that multiple finite element types can be created simultaneously by including multiple generatrix types, even of different dimensions, in the pattern set.

The generatrix is also the key to the efficiency of the drag mesh method. The only information required to compute a layer of elements is the layer of nodes (current pattern position) and the displacement control vector. Further, all data is treated sequentially, reducing not only array size requirements but also page faults and thrashing in a virtual memory programming environment.

The drag mesh generation method is modular in the treatment of geometry and topology. First (geometry), the next layer of nodes is computed from the last layer; next (topology), the connectivity of each finite element is derived from the nodes on the two positions of the corresponding generatrix. As a result, the algorithm is easily expandable. To implement a new method for determining the motion of the nodes requires only the replacement/expansion of an isolated portion of the entire algorithm. Similarly, new types of generatrix elements can be easily implemented by enhancing only a portion of the topology code.

No restrictions need be placed on the geometry of the input parameters for the drag mesh generation. For instance, in solid element generation the two-dimensional generatrices of the pattern are not required to lie all in one plane. Further, for a given drag application, the generatrices do not have to form a connected pattern.

#### IMPLEMENTATION

First-order solid element generation via the drag mesh method has been implemented in the UNISTRUC<sup>TM</sup> system[1], a Control Data Corporation interactive time-sharing graphics finite element pre/post processor available through the CYBERNET<sup>®</sup> Data Services network. 8-node hexahedron and 6-node wedge finite elements, and 4-node tetrahedrons in some degeneracy cases, can be created by dragging a pattern of 4-node quadrilateral and 3-node triangular generatrices. The pattern can be defined by any combination of several methods. Within the UNISTRUC system, generatrices can be created (a) via the automatic triangular and quadrilateral element generators for meshing surfaces in three dimensions bounded by two, three, or four lines or by a line/point combination; (b) by individually specifying the connectivity for each generatrix; or (c) from the faces of specific existing solid elements. The generatrix elements may also be created off-line and read into the UNISTRUC system for use in solid element generation. Once defined, the pattern set of generatrices can be previewed and modified before dragging.

The sequence of displacement control vectors is defined by the vectors connecting the sequence of points on a displacement control line. This line can be created within the UNISTRUC system or read in from an external source, e.g. from a digitizer. The reference axis is simply a line joining any two user-specified locations in three-dimensional space.

In addition to simplicity of input as described above (a predefined pattern set of generatrices, a displacement control line, and optionally a reference axis), some further highlights of the implementation of the drag mesh generation method in the UNISTRUC system are described below:

1. Flexible control of the node-layer positions (transient positions of the pattern)—The points of the dis-

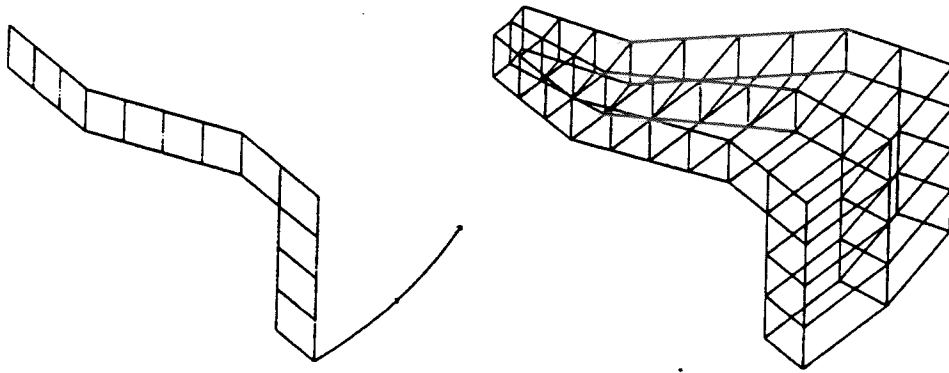


Fig. 4. Drag mesh cylindrical mode with angular increments.

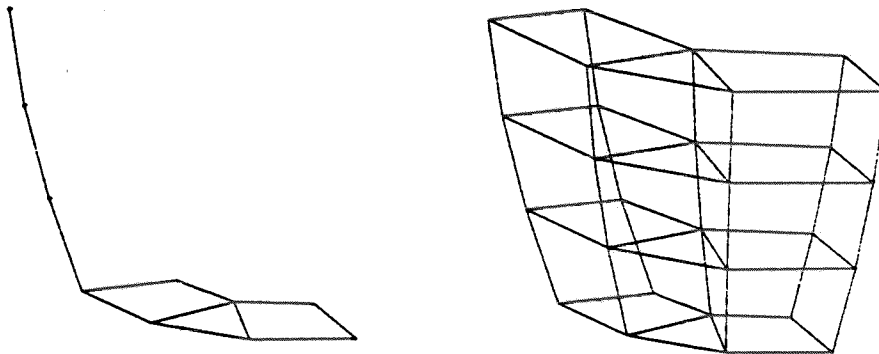


Fig. 5. Drag mesh cylindrical mode with radial and axial increments.

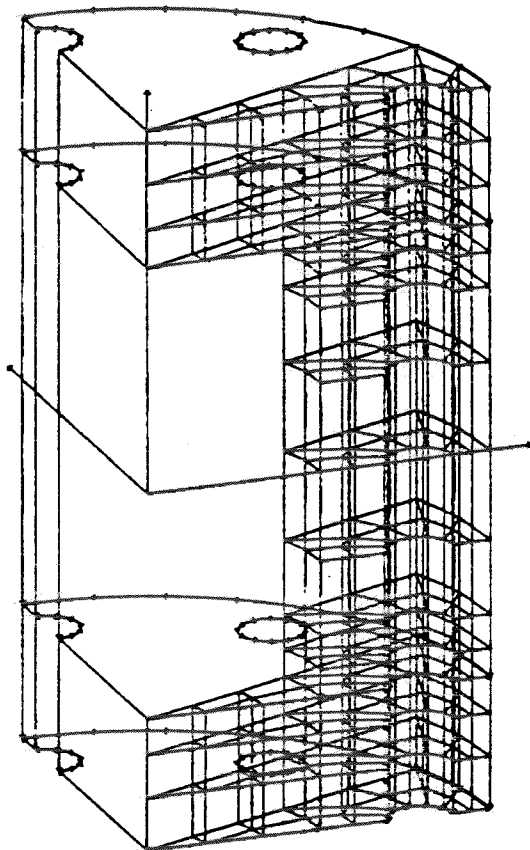


Fig. 6. 1/16 section pressure vessel model via  $\Delta z$  drag.

placement control line can be easily defined so that the intermediate layers of nodes are positioned at predetermined locations (e.g. to fit with weld-points, to meet element-layer thickness requirements, or to match an adjoining mesh).

2. Flexible control of element and node numbering—Several alternatives exist for determining element and node numbering. The following convention is established in the UNISTRUC system. Elements in a layer are numbered in the same order that the generatrices for the pattern were specified in the user input. Nodes in a layer are numbered in the same numerical order as the nodes in the original pattern; the nodes of the first layer are those of the original pattern.

3. Degeneracy processing—In the cylindrical drag mode degeneracy results if nodes of the pattern fall on the reference axis. For example, a 4-node quadrilateral generatrix will create a 6-node wedge instead of the usual 8-node hexahedron if the generatrix pivots about two adjacent nodes. Such collapsing is easily sensed by the algorithm, which then determines the proper connectivity.

4. Coincident layers—In the cylindrical drag mode, if the sequence of displacement control vectors defines a closed loop around the reference axis, the nodes of final layer are just those of the original position layer. This condition is also easily sensed and the original nodes are used to close the mesh.

Figures 4–8 illustrate finite element generation via drag mesh as implemented in the UNISTRUC system. Figures 4 and 5 show both the prepared pattern set/displacement control line combination and the resulting finite element

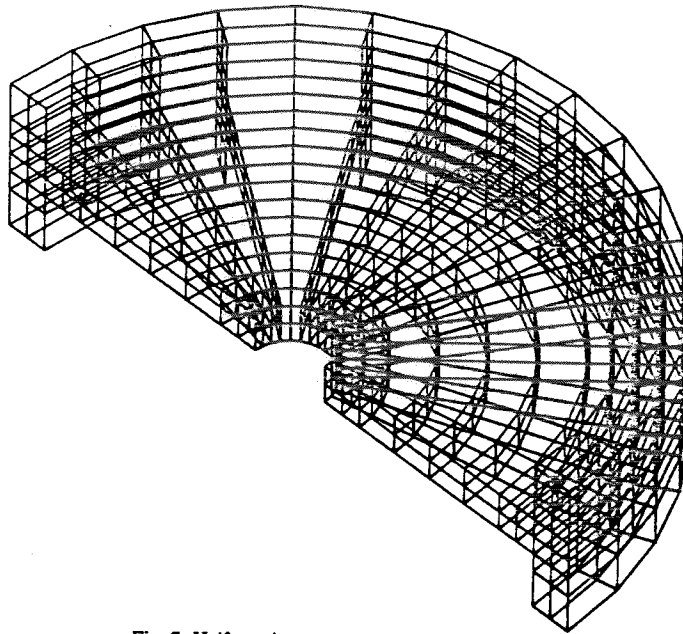


Fig. 7. Half section castorwheel F. E. model.

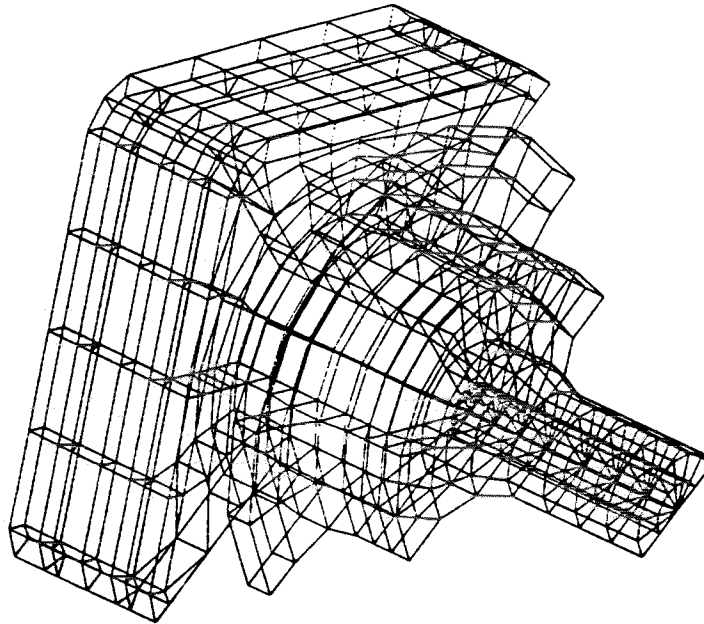


Fig. 8. Half section shaft F. E. model.

mesh. Figure 6 shows 1/16 reactor pressure vessel model highlighted with basic geometric lines of the structure.

For a quantitative performance measurement of the drag mesh generation method, the 8-node hexahedron model shown in Fig. 7 which consists of 765 nodes and 448 elements, was modeled using the UNISTRUC system and also with a similar interactive graphics finite element preprocessor in wide use today in industry. Total modeling time was approximately half an hour on both systems. However, the computer cost for the UNISTRUC system was \$120 compared with \$360 for the other system.

#### FUTURE PROJECTIONS AND RESEARCH DIRECTIONS

The drag mesh generation method described above has been applied in the generation of first-order (linear) elements. However, due to the simplicity and flexibility of both the input and the algorithm, the most beneficial use

of the drag method is projected to be in the generation of higher-order finite elements. What remains is the definition of an appropriate set of generatrix elements.

Further areas for extension of the drag mesh method include: (a) a projection mode, wherein the pattern is projected in increments onto a surface/line/point; and (b) multiple displacement control parameters, each governing the motion of a set of nodes in the pattern, with the motion of the remaining nodes being interpolated.

*Acknowledgements*—The authors wish to give special thanks to Mr. D. J. Oden for many valuable suggestions and Mr. C. S. Seum for his refinements and coding in implementing the drag mesh method in the UNISTRUC system. Thanks are also due to Ms. K. M. Stoffer for preparation of the manuscript.

#### REFERENCE

1. *UNISTRUC Reference Manual*, Control Data Corporation Pub. no. 76079600.