Software

Better Finite-Element Analysis of Composite Shell Structures

A computer program implements a finite-element-based method of predicting the deformations of thin aerospace structures made of isotropic materials or anisotropic fiber-reinforced composite materials. The technique and corresponding software are applicable to thin shell structures in general and are particularly useful for analysis of thin beam-like members having open cross-sections (e.g. I-beams and C-channels) in which significant warping can occur.

In many popular commercial packages that offer the two-dimensional finite elements option, inaccuracies arise from node offsets and overlapping of elements; in other formulations utilizing one-dimensional discretization, angles of twist tend to be overestimated. In the present formulation, a shell element that incorporates "floating" reference surfaces on which nodal points reside is developed. This element concept facilitates the formation of proper connections among elements and thereby eliminates the inaccuracies attributable to element-overlapping and nodal offsets in the other methods.

This program was written by Gregory Clarke of Goddard Space Flight Center. Further information is contained in a TSP (see page 1).
GSC-14756-1

Computing Spacecraft-Pointing Vectors for Limb Tracking

LMBTRK is a computer program that is used together with two software libraries known as ERHAND and HYBRRD to generate spacecraft-pointing vectors for limb-tracking maneuvers needed for experiments on propagation of radio signals through planetary atmospheres. LMBTRK determines, as a function of time, the direction in which one must point a ray (representing a radio beam) emitted by a spacecraft in order to make the ray pass through a planetary atmosphere on its way to a receiving station at a known location. LMBTRK was written for Sun computers running the Solaris operating system and has been running on a cluster of such computers used in the Radio Science System of the Cassini Spacecraft mission.

This program was written by Nicole Rappaport and Essam Marouf of Caltech for NASA's Jet Propulsion Laboratory. Further information is contained in a TSP (see page 1).

This software is available for commercial licensing. Please contact Karina Edmonds of the California Institute of Technology at (626) 395-2322. Refer to NPO-40542.

Enhanced Master Controller Unit Tester

The Enhanced Master Controller Unit Tester (EMUT) software is a tool for development and testing of software for a master controller (MC) flight computer. The primary function of the EMUT software is to simulate interfaces between the MC computer and external analog and digital circuitry (including other computers) in a rack of equipment to be used in scientific experiments. The simulations span the range of nominal, off-nominal, and erroneous operational conditions, enabling the testing of MC software before all the equipment becomes available.

The EMUT software comprises a Win32-based set of three programs that run on different computers and are linked by Common Object Request Broker (CORBA) Ethernet communications. The main program, denoted the EMUT 1553 Application, generates a graphical user interface and is responsible for all communications via several MIL-STD-1553B data buses and for logging of output data. The second program, denoted the Analog Application, implements mathematical models of equipment (e.g., sensors and valves) and analog signals generated by the equipment and is responsible for all analog-signal communications with the MC. The third program, denoted the EMUT Model Viewer, provides a graphical interface for viewing the statuses of the aforementioned models.

This program was written by Patricia Benson, Yvette Johnson, and Brian Johnson of Marshall Space Flight Center; Philip Williams of Dynamic Concepts, Inc.; Geoffrey Burton of Madison Research Corp.; and Anthony McCoy of ERC. Further information is contained in a TSP (see page 1).
MFS-32172-1

Rover Graphical Simulator

Rover Graphical Simulator (RGS) is a package of software that generates images of the motion of a wheeled robotic exploratory vehicle (rover) across terrain that includes obstacles and regions of varying traversability. The simulated rover moves autonomously, utilizing reasoning and decision-making capabilities of a fuzzy-logic navigation strategy to choose its path from an initial to a final state. RGS provides a graphical user interface for control and monitoring of simulations.

The numerically simulated motion is represented as discrete steps with a constant time interval between updates. At each simulation step, a dot is placed at the old rover position and a graphical symbol representing the rover is redrawn at the new, updated position. The effect is to leave a trail of dots depicting the path traversed by the rover, the distances between dots being proportional to the local speed. Obstacles and regions of low traversability are depicted as filled circles, with buffer zones around them indicated by enclosing circles. The simulated robot is equipped with onboard sensors that can detect regional terrain traversability and local obstacles out to specified ranges. RGS won the NASA Group Achievement Award in 2002.

This program was written by Bruce Bon and Homayoun Seraji of Caltech for NASA's Jet Propulsion Laboratory. Further information is contained in a TSP (see page 1).

This software is available for commercial licensing. Please contact Karina Edmonds of the California Institute of Technology at (626) 395-2322. Refer to NPO-35223.