Facilitating Navigation Through Large Archives

Automated Visual Access (AVA) is a computer program that effectively makes a large collection of information visible in a manner that enables a user to quickly and efficiently locate information resources, with minimal need for conventional keyword searches and perusal of complex hierarchical directory systems. AVA includes three key components: (1) a taxonomy that comprises a collection of words and phrases, clustered according to meaning, that are used to classify information resources; (2) a statistical indexing and scoring engine; and (3) a component that generates a graphical user interface that uses the scoring data to generate a visual map of resources and topics. The top level of an AVA display is a pictorial representation of an information archive. The user enters the depicted archive by either clicking on a depiction of subject area cluster, selecting a topic from a list, or entering a query into a text box. The resulting display enables the user to view candidate information entities at various levels of detail. Resources are grouped spatially by topic with greatest generality at the top layer and increasing detail with depth. The user can zoom in or out of specific sites or into greater or lesser content detail.

This program was written by Robert O. Shelton of **Johnson Space Center**, *Stephanie L. Smith of LinCom, and Dat Truong and Terry R. Hodgson of Science Applications International Corp.*

The code is copyrighted and is available for licensing. For further information, contact the Johnson Technology Transfer Office at (281) 483-3809.
MSC-23542

Program for Weibull Analysis of Fatigue Data

A Fortran computer program has been written for performing statistical analyses of fatigue-test data that are assumed to be adequately represented by a two-parameter Weibull distribution. This program calculates the following:

- Maximum-likelihood estimates of the Weibull distribution;
- Data for contour plots of relative likelihood for two parameters;
- Data for contour plots of joint confidence regions;
- Data for the profile likelihood of the Weibull-distribution parameters;
- Data for the profile likelihood of any percentile of the distribution; and
- Likelihood-based confidence intervals for parameters and/or percentiles of the distribution.

The program can account for tests that are suspended without failure (the statistical term for such suspension of tests is "censoring"). The analytical approach followed in this program for the software is valid for type-I censoring, which is the removal of unfailed units at pre-specified times. Confidence regions and intervals are calculated by use of the likelihood-ratio method.

This program was written by Timothy L. Krantz of the Vehicle Technology Center of the U.S. Army Research Laboratory for **Glenn Research Center**. *Further information is contained in a TSP (see page 1).*

Inquiries concerning rights for the commercial use of this invention should be addressed to NASA Glenn Research Center, Commercial Technology Office, Attn: Steve Fedor, Mail Stop 4–8, 21000 Brookpark Road, Cleveland, Ohio 44135. Refer to LEW-17401-1.

Comprehensive Micromechanics-Analysis Code — Version 4.0

Version 4.0 of the Micromechanics Analysis Code With Generalized Method of Cells (MAC/GMC) has been developed as an improved means of computational simulation of advanced composite materials. The previous version of MAC/GMC was described in "Comprehensive Micromechanics-Analysis Code" (LEW-16870), *NASA Tech Briefs*, Vol. 24, No. 6 (June 2000), page 38. To recapitulate: MAC/GMC is a computer program that predicts the elastic and inelastic thermomechanical responses of continuous and discontinuous composite materials with arbitrary internal microstructures and reinforcement shapes. The predictive capability of MAC/GMC rests on a model known as the generalized method of cells (GMC) — a continuum-based model of micromechanics that provides closed-form expressions for the macroscopic response of a composite material in terms of the properties, sizes, shapes, and responses of the individual constituents or phases that make up the material. Enhancements in version 4.0 include a capability for modeling thermomechanically and electromagnetically coupled ("smart") materials; a more-accurate (high-fidelity) version of the GMC; a capability to simulate discontinuous plies within a laminate; additional constitutive models of materials; expanded yield-surface-analysis capabilities; and expanded failure-analysis and life-prediction capabilities on both the microscopic and macroscopic scales.

This program was written by S. M. Arnold of **Glenn Research Center** *and B. A. Bednarcyk of Ohio Aerospace Institute. Further information is contained in a TSP (see page 1).*

Inquiries concerning rights for the commercial use of this invention should be addressed to NASA Glenn Research Center, Commercial Technology Office, Attn: Steve Fedor, Mail Stop 4–8, 21000 Brookpark Road, Cleveland, Ohio 44135. Refer to LEW-17495-1.

Component-Based Visualization System

A software system has been developed that gives engineers and operations personnel with no "formal" programming expertise, but who are familiar with the Microsoft Windows operating system, the ability to create visualization displays to monitor the health and performance of aircraft/spacecraft. This software system is currently supporting the X38 V201 spacecraft component/system testing and is intended to give users the ability to create, test, deploy, and certify their subsystem displays in a fraction of the time that it would take to do so using previous software and programming methods. Within the visualization system there are three major components: the developer, the deployer, and the widget set. The developer is a blank canvas with widget menu items that give users the ability to easily create displays. The deployer is an application that allows for the deployment of the displays created using the developer application. The deployer has additional functionality that the developer does not have, such as printing of displays, screen captures to files, windowing of displays, and also serves as the interface into the documentation archive and help system. The third major component is the widget set. The widgets are the visual representation of the items that will make up the display (i.e., meters, dials, buttons, numerical indicators, string indicators, and the like). This software was developed using Visual C++ and

uses COTS (commercial off-the-shelf) software where possible.

*This program was written by Francisco Delgado of **Johnson Space Center**. For further information, contact the Johnson Technology Transfer Office at (281) 483-3809. MSC-23490*

⚙ Software for Engineering Simulations of a Spacecraft

Spacecraft Engineering Simulation II (SES II) is a C-language computer program for simulating diverse aspects of operation of a spacecraft characterized by either three or six degrees of freedom. A functional model in SES can include a trajectory flight plan; a submodel of a flight computer running navigational and flight-control software; and submodels of the environment, the dynamics of the spacecraft, and sensor inputs and outputs. SES II features a modular, object-oriented programming style. SES II supports event-based simulations, which, in turn, create an easily adaptable simulation environment in which many different types of trajectories can be simulated by use of the same software. The simulation output consists largely of flight data. SES II can be used to perform optimization and Monte Carlo dispersion simulations. It can also be used to perform simulations for multiple spacecraft. In addition to its generic simulation capabilities, SES offers special capabilities for space-shuttle simulations: for this purpose, it incorporates submodels of the space-shuttle dynamics and a C-language version of the guidance, navigation, and control components of the space-shuttle flight software.

*This program was written by Kirk Shireman and Gene McSwain of **Johnson Space Center,** and Bernell McCormick and Panayiotis Fardelos of the Boeing Co. For further information, contact the Johnson Technology Transfer Office at (281) 483-3809. MSC-23537*

✳ LabVIEW Interface for PCI-SpaceWire Interface Card

This software provides a LabView interface to the NT drivers for the PCI-SpaceWire card, which is a peripheral component interface (PCI) bus interface that conforms to the IEEE-1355/SpaceWire standard. As SpaceWire grows in popularity, the ability to use SpaceWire links within LabVIEW will be important to electronic ground support equipment vendors. In addition, there is a need for a high-level LabVIEW interface to the low-level device-driver software supplied with the card.

The LabVIEW virtual instrument (VI) provides graphical interfaces to support all (1) SpaceWire link functions, including message handling and routing; (2) monitoring as a passive "tap" using specialized hardware; and (3) low-level access to satellite mission-control subsystem functions. The software is supplied in a zip file that contains LabVIEW VI files, which provide various functions of the PCI-SpaceWire card, as well as higher-link-level functions. The VIs are suitably named according to the matching function names in the driver manual. A number of test programs also are provided to exercise various functions.

*This work was done by James Lux, Frank Loya, and Alex Bachmann 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 (818) 393-2827.Refer to NPO-35207.

✜ Path Following With Slip Compensation for a Mars Rover

A software system for autonomous operation of a Mars rover is composed of several key algorithms that enable the rover to accurately follow a designated path, compensate for slippage of its wheels on terrain, and reach intended goals. The techniques implemented by the algorithms are visual odometry, full vehicle kinematics, a Kalman filter, and path following with slip compensation. The visual-odometry algorithm tracks distinctive scene features in stereo imagery to estimate rover motion between successively acquired stereo image pairs, by use of a maximum-likelihood motion-estimation algorithm. The full-vehicle kinematics algorithm estimates motion, with a no-slip assumption, from measured wheel rates, steering angles, and angles of rockers and bogies in the rover suspension system. The Kalman filter merges data from an inertial measurement unit (IMU) and the visual-odometry algorithm. The merged estimate is then compared to the kinematic estimate to determine whether and how much slippage has occurred. The kinematic estimate is used to complement the Kalman-filter estimate if no statistically significant slippage has occurred. If slippage has occurred, then a slip vector is calculated by subtracting the current Kalman filter estimate from the kinematic estimate. This slip vector is then used, in conjunction with the inverse

kinematics, to determine the wheel velocities and steering angles needed to compensate for slip and follow the desired path.

*This work was done by Daniel Helmick, Yang Cheng, Daniel Clouse, Larry Matthies, and Stergios Roumeliotis 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 (818) 393-2827. Refer to NPO-40703.

➋ International Space Station Electric Power System Performance Code — SPACE

The System Power Analysis for Capability Evaluation (SPACE) software analyzes and predicts the minute-by-minute state of the International Space Station (ISS) electrical power system (EPS) for upcoming missions as well as EPS power generation capacity as a function of ISS configuration and orbital conditions. In order to complete the Certification of Flight Readiness (CoFR) process — in which the mission is certified for flight — each ISS System must thoroughly assess every proposed mission to verify that the system will support the planned mission operations; SPACE is the sole tool used to conduct these assessments for the power system capability. SPACE is an integrated power system model that incorporates a variety of modules tied together with integration routines and graphical output. The modules include orbit mechanics, solar array pointing/shadowing/thermal and electrical, battery performance, and power management and distribution performance. These modules are tightly integrated within a flexible architecture featuring data-file-driven configurations, source- or load-driven operation, and event scripting. SPACE also predicts the amount of power available for a given system configuration, spacecraft orientation, solar-array-pointing conditions, orbit, and the like. In the source-driven mode, the model must assure that energy balance is achieved, meaning that energy removed from the batteries must be restored (or balanced) each and every orbit. This entails an optimization scheme to ensure that energy balance is maintained without violating any other constraints. In the load-driven mode, SPACE determines whether a given distributed, time-varying electrical load profile can be supported by the power system and will determine whether the system stays in energy balance. Load-driven