both the L1 (1.57542-GHz) and L2 (1.2276-GHz) GPS signals.

This program was written by Sung Byun, George Hajj, and Lawrence Young 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 Don Hart of the California Institute of Technology at (818) 393-3425. Refer to NPO-40463.

Parallel Adaptive Mesh Refinement Library

Parallel Adaptive Mesh Refinement Library (PARAMESH) is a package of Fortran 90 subroutines designed to provide a computer programmer with an easy route to extension of (1) a previously written serial code that uses a logically Cartesian structured mesh into (2) a parallel code with adaptive mesh refinement (AMR). Alternatively, in its simplest use, and with minimal effort, PARAMESH can operate as a domain-decomposition tool for users who want to parallelize their serial codes but who do not wish to utilize adaptivity. The package builds a hierarchy of sub-grids to cover the computational domain of a given application program, with spatial resolution varying to satisfy the demands of the application. The sub-grid blocks form the nodes of a tree data structure (a quad-tree in two or an oct-tree in three dimensions). Each grid block has a logically Cartesian mesh. The package supports one-, two- and three-dimensional models.

This program was written by Peter MacNeice of Raytheon/STX and Kevin Olson of George Mason University for **Goddard Space Flight Center**. *For further information, access http://sdcd.gsfc.nasa.gov/RIB/repositories/inhouse_gsfc/Users_manual/amr_tutorial.html. GSC-14626-1*

Space Physics Data Facility Web Services

The Space Physics Data Facility (SPDF) Web services provides a distributed programming interface to a portion of the SPDF software. (A general description of Web services is available at *http://www.w3.org/* and in many current software-engineering texts and articles focused on distributed programming.) The SPDF Web services distributed programming interface enables additional collaboration and integration of the SPDF software system with other software systems, in furtherance of the SPDF mission to lead collaborative efforts in the collection and utilization of space physics data and mathematical models. This programming interface conforms to all applicable Web services specifications of the World Wide Web Consortium. The interface is specified by a Web Services Description Language (WSDL) file. The SPDF Web services software consists of the following components:

- A server program for implementation of the Web services; and
- A software developer's kit that consists of a WSDL file, a less formal description of the interface, a Java class library (which further eases development of Java-based client software), and Java source code for an example client program that illustrates the use of the interface.

This program was written by Robert M. Candey, Bernard T. Harris, and Reine A. Chimiak of **Goddard Space Flight Center**. *For further information, access http://spdf.gsfc.nasa.gov/. GSC-14730-1*

Predicting Noise From Aircraft Turbine-Engine Combustors

COMBUSTOR and CNOISE are computer codes that predict far-field noise that originates in the combustors of modern aircraft turbine engines — especially modern, low-gaseous-emission engines, the combustors of which sometimes generate several decibels more noise than do the combustors of older turbine engines. COMBUSTOR implements an empirical model of combustor noise derived from correlations between engine-noise data and operational and geometric parameters, and was developed from databases of measurements of acoustic emissions of engines. CNOISE implements an analytical and computational model of the propagation of combustor temperature fluctuations (hot spots) through downstream turbine stages. Such hot spots are known to give rise to far-field noise. CNOISE is expected to be helpful in determining why low-emission combustors are sometimes noisier than older ones, to provide guidance for refining the empirical correlation model embodied in the COMBUSTOR code, and to provide insight on how to vary downstream turbine-stage geometry to reduce the contribution of hot spots to far-field noise.

These programs were written by P. Gliebe, R. Mani, S. Salamah, and R. Coffin of General Electric Co. and Jayesh Mehta of Diversitec, Inc., 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-17385-1.

Generating Animated Displays of Spacecraft Orbits

Tool for Interactive Plotting, Sonification, and 3D Orbit Display (TIPSOD) is a computer program for generating interactive, animated, four-dimensional (space and time) displays of spacecraft orbits. TIPSOD utilizes the programming interface of the Satellite Situation Center Web (SSCWeb) services to communicate with the SSC logic and database by use of the open protocols of the Internet. TIPSOD is implemented in Java 3D and effects an extension of the pre-existing SSCWeb two-dimensional static graphical displays of orbits. Orbits can be displayed in any or all of the following seven reference systems: true-of-date (an inertial system), J2000 (another inertial system), geographic, geomagnetic, geocentric solar ecliptic, geocentric solar magnetospheric, and solar magnetic. In addition to orbits, TIPSOD computes and displays Sibeck's magnetopause and Fairfield's bow-shock surfaces. TIPSOD can be used by the scientific community as a means of projection or interpretation. It also has potential as an educational tool. Documentation and links for downloading the software can be found at *http://sscweb.gsfc.gov/tipsod/*.

This program was written by Robert M. Candey, Reine A. Chimiak, and Bernard T. Harris of **Goddard Space Flight Center**. *For more information contact the Goddard Commercial Technology Office at (301) 286-5810. GSC-14732-1*

Diagnosis and Prognosis of Weapon Systems

The Prognostics Framework is a set of software tools with an open architecture that affords a capability to integrate various prognostic software mechanisms and to provide information for operational and battlefield decision-making and logistical planning pertaining to weapon systems. The Prognostics

Framework is also a system-level "health"-management software system that (1) receives data from performance-monitoring and built-in-test sensors and from other prognostic software and (2) processes the received data to derive a diagnosis and a prognosis for a weapon system. This software relates the diagnostic and prognostic information to the overall health of the system, to the ability of the system to perform specific missions, and to needed maintenance actions and maintenance resources. In the development of the Prognostics Framework, effort was focused primarily on extending previously developed model-based diagnostic-reasoning software to add prognostic reasoning capabilities, including capabilities to perform statistical analyses and to utilize information pertaining to deterioration of parts, failure modes, time sensitivity of measured values, mission criticality, historical data, and trends in measurement data. As thus extended, the software offers an overall health-monitoring capability.

This program was written by Mary Nolan, Rebecca Catania, and Gregory deMare of Giordano Automation Corp. for **Marshall Space Flight Center**. *Further information is contained in a TSP (see page 1).*
MFS-31644

Training Software in Artificial-Intelligence Computing Techniques

The Artificial Intelligence (AI) Toolkit is a computer program for training scientists, engineers, and university students in three soft-computing techniques (fuzzy logic, neural networks, and genetic algorithms) used in artificial-intelligence applications. The program promotes an easily understandable tutorial interface, including an interactive graphical component through which the user can gain hands-on experience in soft-computing techniques applied to realistic example problems. The tutorial provides step-by-step instructions on the workings of soft-computing technology, whereas the hands-on examples allow interaction and reinforcement of the techniques explained throughout the tutorial. In the fuzzy-logic example, a user can interact with a robot and an obstacle course to verify how fuzzy logic is used to command a rover traverse from an arbitrary start to the goal location. For the genetic algorithm example, the problem is to determine the minimum-length path for visiting a user-chosen set of planets in the solar system. For the neural-network example, the problem is to decide, on the basis of input data on physical characteristics, whether a person is a man, woman, or child. The AI Toolkit is compatible with the Windows 95,98, ME, NT 4.0, 2000, and XP operating systems. A computer having a processor speed of at least 300 MHz, and random-access memory of at least 56MB is recommended for optimal performance. The program can be run on a slower computer having less memory, but some functions may not be executed properly.

This program was written by Ayanna Howard, Eric Rogstad, and Eugene Chalfant 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 Don Hart of the California Institute of Technology at (818) 393-3425. Refer to NPO-40496.

APGEN Version 5.0

Activity Plan Generator (APGEN), now at version 5.0, is a computer program that assists in generating an integrated plan of activities for a spacecraft mission that does not oversubscribe spacecraft and ground resources. APGEN generates an interactive display, through which the user can easily create or modify the plan. The display summarizes the plan by means of a time line, whereon each activity is represented by a bar stretched between its beginning and ending times. Activities can be added, deleted, and modified via simple mouse and keyboard actions. The use of resources can be viewed on resource graphs. Resource and activity constraints can be checked. Types of activities, resources, and constraints are defined by simple text files, which the user can modify. In one of two modes of operation, APGEN acts as a planning expert assistant, displaying the plan and identifying problems in the plan. The user is in charge of creating and modifying the plan. In the other mode, APGEN automatically creates a plan that does not oversubscribe resources. The user can then manually modify the plan. APGEN is designed to interact with other software that generates sequences of timed commands for implementing details of planned activities.

This program was written by Pierre Maldague, Dennis Page, and Adam Chase 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 Don Hart of the California Institute of Technology at (818) 393-3425. Refer to NPO-40520.

Single-Command Approach and Instrument Placement by a Robot on a Target

AUTOAPPROACH is a computer program that enables a mobile robot to approach a target autonomously, starting from a distance of as much as 10 m, in response to a single command. AUTOAPPROACH is used in conjunction with (1) software that analyzes images acquired by stereoscopic cameras aboard the robot and (2) navigation and path-planning software that utilizes odometer readings along with the output of the image-analysis software. Intended originally for application to an instrumented, wheeled robot (rover) in scientific exploration of Mars, AUTOAPPROACH could be adapted to terrestrial applications, notably including the robotic removal of land mines and other unexploded ordnance. A human operator generates the approach command by selecting the target in images acquired by the robot cameras. The approach path consists of multiple legs. Feature points are derived from images that contain the target and are thereafter tracked to correct odometric errors and iteratively refine estimates of the position and orientation of the robot relative to the target on successive legs. The approach is terminated when the robot attains the position and orientation required for placing a scientific instrument at the target. The workspace of the robot arm is then autonomously checked for self/terrain collisions prior to the deployment of the scientific instrument onto the target.

This program was written by Terrance Huntsberger and Yang Cheng of **NASA's Jet Propulsion Laboratory**. *Further information is contained in a TSP (see page 1).*

This software is available for commercial licensing. Please contact Don Hart of the California Institute of Technology at (818) 393-3425. Refer to NPO-30529

Three-Dimensional Audio Client Library

The Three-Dimensional Audio Client Library (3DAudio library) is a group of software routines written to facilitate development of both stand-alone (audio only) and immersive virtual-reality application pro-