input(s) to a system and the resultant output(s) in real time or a posteriori, or from software-generated data sets, were presented to the system, which generated outputs. Once a system is learned, the coefficients and constants can be frozen and the algorithm embedded in an application.

This work was done by Michael J. Krasowski and Norman F. Prokop of 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, Innovative Partnerships Office, Attn: Steven Fedor, Mail Stop 4–8, 21000 Brookpark Road, Cleveland, Ohio 44135. Refer to LEW-18887-1.

Kalman Filter Input Processor for Boresight Calibration

The new software brings this technology to the industrial level.

NASA's Jet Propulsion Laboratory, Pasadena, California

Ka-band ranging provides the phase center (PC) to phase center range, which needs to be converted to the center of mass (CM) to center of mass range. Nominally, both PC and CM lie on the line connecting the spacecraft GRAIL A and GRAIL B. In this case, the conversion should be done simply by adding the CM-to-PC distance L to the measured range for both spacecraft. However, due to various technical reasons, such as displacement of the true CM from its nominal position in the SRF, or spacecraft attitude fluctuations, the PC and CM define a unit vector that may be different from the nominal line of sight. The objectives of the software are to determine the actual line of sight direction for each spacecraft and correct the previously recorded range data, and to provide instructions for how to maneuver each spacecraft to make necessary attitude corrections.

While elements of this approach have been used for the boresight calibration in the GRACE project, the new software brings this technology to the industrial level. It is now fully documented and can be used by people other than its developers. This innovation provides graphic outputs and log files that are critical for quick analysis and troubleshooting. In addition to the line of sight direction, the software allows one to evaluate the CM-PC base length, which is important when the PM location is subject to variations (e.g., due to fuel depletion).

This software is implemented in Python and offers excellent cross-platform porting possibilities. It is very versatile, and may be applied under various circumstances and for other related purposes. This innovation is capable of combining the input data from several calibration maneuvers, evaluating individual range biases, and compressing the time stamps. It uses Lagrange interpolation for the orbit data, and a unique quaternion-interpolating algorithm for interpolating the attitude data. As a result, data files with different data rates and independent time stamps can be handled together.

This work was done by Dmitry V. Strekalov, Gerhard L. Kruizinga, Meegyeong Paik, Dah-Ning Yuan, and Sami W. Asmar of Caltech for NASA's Jet Propulsion Laboratory. For more information, please contact Brian Morrison at Brian.A.Morrison@jpl.nasa.gov.

This software is available for commercial licensing. Please contact Dan Broderick at Daniel.F.Broderick@jpl.nasa.gov. Refer to NPO-48479.

Organizing Compression of Hyperspectral Imagery to Allow Efficient Parallel Decompression

Higher compression factors can be attained.

NASA's Jet Propulsion Laboratory, Pasadena, California

A family of schemes has been devised for organizing the output of an algorithm for predictive data compression of hyperspectral imagery so as to allow efficient parallelization in both the compressor and decompressor. In these schemes, the compressor performs a number of iterations, during each of which a portion of the data is compressed via parallel threads operating on independent portions of the data. The general idea is that for each iteration it is predetermined how much compressed data will be produced from each thread.

A simple version of this technique is applicable when the image is divided into "pieces" that are compressed independently. As an example, for a compressor that does not make use of interband correlation, a piece could be defined to be an individual spectral band, or a fixed number of bands. In the technique, the compressed output for a piece is comprised of multiple "chunks." The concatenated chunks for a given piece form the compressed output for the piece. Most of the compressed image is produced in multiple iterations, where during a given iteration, one chunk is produced for each piece. Prior to the start of an iteration, chunk sizes are calculated for each piece. The chunks can be produced or decompressed in parallel. It is noted that it is not specified how much of the image data will go into a chunk, and in fact a chunk may contain incomplete portions of encoded samples (at the chunk's start or end). The compressor iterates the process of deciding on chunk sizes and producing chunks for each piece of the requested size, until compression of each piece is almost finished. At that point, the remainder of the pieces is compressed serially without a target chunk size.

Typically, the chunk size calculation should seek to balance the progress through each piece, i.e., to leave equal numbers of samples remaining in each piece; a suggested procedure has this aim. A key requirement on the chunk