tum and classical aspects, and therefore, it will be useful for implementation of quantum computing.

Recent advances in quantum information theory have inspired an explosion of interest in new quantum algorithms for solving hard computational problems. Three basic "non-classical" properties of quantum mechanics — superposition, entanglement, and direct tensor-product decomposability — were main reasons for optimism about capabilities of quantum computers and quantum communications as well as for a new approach to cryptography. However, one major problem is keeping the components of a quantum computer in a coherent state, as the slightest interaction with the external world would cause the system to decohere. Another problem is measurement: by the laws of quantum mechanics, a measurement yields a random and incomplete answer, and it destroys the stored state.

This proposed reinterpretation of quantum formalism opens up new advantages of quantum computers: if the Madelung equations are implemented on a classical scale (using, for instance, electrical circuits or optical devices), all the quantum effects important for computations would be preserved; at the same time, the problems associated with decoherence and measurement would be removed.

This work was done by Michail Zak of Caltech for NASA's Jet Propulsion Laboratory. For more information, contact iaoffice@jpl.nasa.gov. NPO-46731

Optimal Padding for the Two-Dimensional Fast Fourier Transform

Appending data to an optimum length decreases computing runtime.

Goddard Space Flight Center, Greenbelt, Maryland

One-dimensional Fast Fourier Transform (FFT) operations work fastest on grids whose size is divisible by a power of two. Because of this, padding grids (that are not already sized to a power of two) so that their size is the next highest power of two can speed up operations. While this works well for one-dimensional grids, it does not work well for two-dimensional grids.

For a two-dimensional grid, there are certain pad sizes that work better than others. Therefore, the need exists to generalize a strategy for determining optimal pad sizes. There are three steps in the FFT algorithm. The first is to perform a one-dimensional transform on each row in the grid. The second step is to transpose the resulting matrix. The third step is to perform a one-dimensional transform on each row in the resulting grid. Steps one and three both benefit from padding the row to the next highest power of two, but the second step needs a novel approach.

An algorithm was developed that struck a balance between optimizing the grid pad size with prime factors that are small (which are optimal for one-dimensional operations), and with prime factors that are large (which are optimal for two-dimensional operations). This algorithm optimizes based on average run times, and is not fine-tuned for any specific application. It increases the amount of times that processor-requested data is found in the set-associative processor cache. Cache retrievals are 4–10 times faster than conventional memory retrievals.

The tested implementation of the algorithm resulted in faster execution times on all platforms tested, but with varying sized grids. This is because various computer architectures process commands differently. The test grid was 512×512. Using a 540×540 grid on a Pentium V processor, the code ran 30 percent faster. On a PowerPC, a 256×256 grid worked best. A Core2Duo computer preferred either a 1040×1040 (15 percent faster) or a 1008×1008 (30 percent faster) grid.

There are many industries that can benefit from this algorithm, including optics, image-processing, signal-processing, and engineering applications.

This work was done by Bruce H. Dean, David L. Aronstein, and Jeffery S. Smith of Goddard Space Flight Center. Further information is contained in a TSP (see page 1).

This invention is owned by NASA, and a patent application has been filed. Inquiries concerning nonexclusive or exclusive license for its commercial development should be addressed to the Patent Counsel, Goddard Space Flight Center, (301) 286-7351. Refer to GSC-15678-1.

Spatial Query for Planetary Data

This technology is extensible to Earth science and satellite monitoring and surveillance.

NASA's Jet Propulsion Laboratory, Pasadena, California

Science investigators need to quickly and effectively assess past observations of specific locations on a planetary surface. This innovation involves a location-based search technology that was adapted and applied to planetary science data to support a spatial query capability for mission operations software.

Conventional databases of planetary datasets are indexed and searchable by various metadata, such as acquisition time, phase of mission, and target. Searching these datasets will produce enormous datasets that are difficult, or impractical, to browse through to identify observations of very specific targets. For queries at specific locations, it is fundamentally more efficient to specify the location as the target of the query; and to have the database search based on the location of the data rather than metadata that is only indirectly or tangentially related to location.

High-performance location-based searching requires the use of spatial data structures for database organization. Spatial data structures are designed to organize datasets based on their coordinates in a way that is optimized for location-based retrieval. The particular spatial data structure that was adapted for planetary data search is the R+ tree. The R+ tree arranges data as a set of nodes that represents bounding rectangles. Every leaf