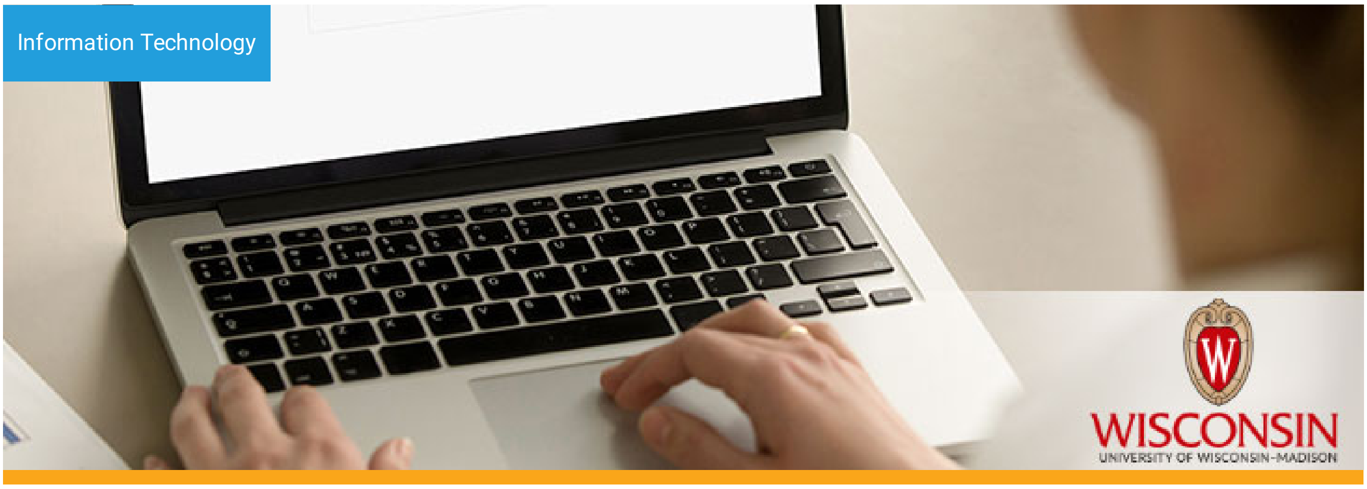

Architecture for Speculative Parallel Execution Improves Performance, Simplifies Programming

View U.S. Patent No. 10,209,997 in PDF format.

WARF: P140373US01

Inventors: Gurindar Sohi, Gagan Gupta

The Wisconsin Alumni Research Foundation (WARF) is seeking commercial partners interested in developing a system that allows for speculative execution of program tasks before data dependencies for those tasks are resolved.

Overview

Parallel processing enables simultaneous data processing tasks to increase computational speed. To leverage this capability with multiprocessors, first an ordered multiprocessor program defines the order for its constituent computations. The system may execute the ordered program's computations concurrently, but may still maintain the program's ordered semantics. In comparison to parallel programs, ordered semantics simplifies programming and use of multiprocessors. Ordered semantics is achieved by establishing the data dependencies between the computations – prior to their execution – based on the data shared between the computations and the relative order in the program.

Computations are scheduled for execution such that the data dependencies are honored. However, problems arise when data dependencies cannot be determined before the computations execute. This is the case when determining the data accessed by a computation which also is a step within the computation itself (called a data-dependent data set). Such computations can limit the exploitable parallelism in the program and limit performance.

The Invention

UW–Madison researchers have developed a system that permits speculative execution of program tasks prior to determining data dependency. Before commitment of the tasks in a sequential execution order, data dependencies are resolved through a token system that tracks read and write access to data elements accessed by the program portions.

Eliminating the need to wait until late in the program execution to detect or resolve dependencies helps improve processor utilization. Advancing the execution of tasks that ultimately do not experience data dependency problems may have a ripple-through effect, reducing later data dependencies as well.

Applications

- Implementation of high performance programs for multiprocessors

Key Benefits

- Simplifies programming and use of multiprocessors
- Provides simple mechanism for detecting data dependency problems
- Can be readily implemented

- Improves processor utilization
- Flexibly accommodates different mechanisms for parallel execution

Stage of Development

A run-time prototype of the system has been developed. Sample ordered multiprocessors have been coded to establish a proof of the concept. Results indicate that the new system outperforms alternative parallel programming methods.

Additional Information

For More Information About the Inventors

- Gurindar Sohi

Related Technologies

- For information about complementary parallel processing innovations developed by the researcher, see WARF reference numbers:
- P100343US01
- P130018US01

Tech Fields

- Information Technology : Computing methods, software & machine learning

For current licensing status, please contact Emily Bauer at emily@warf.org or 608-960-9842