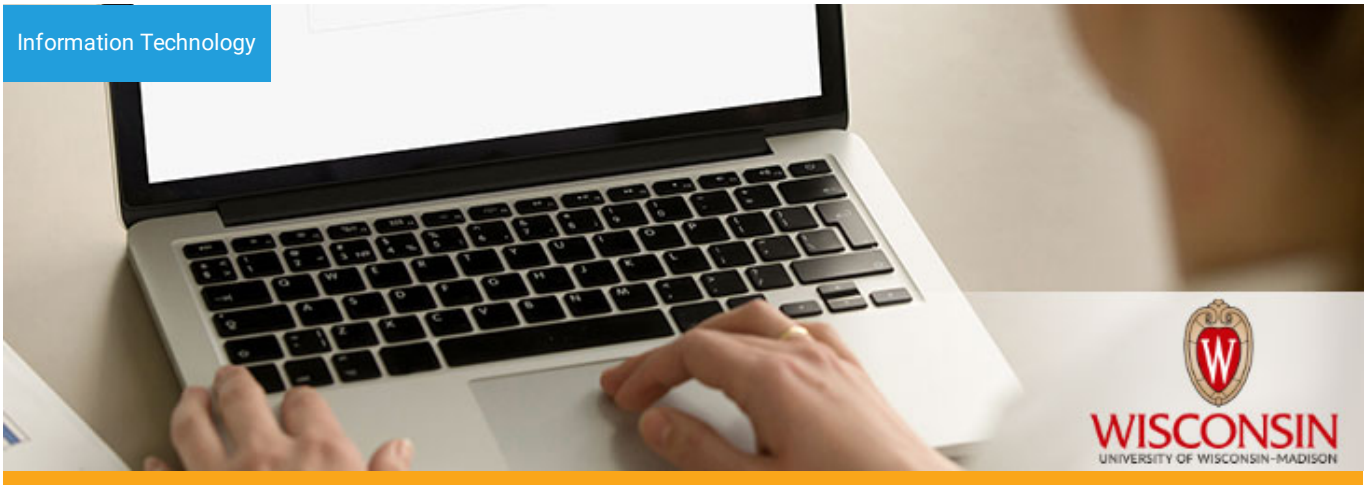
Information Technology



Computer Architecture Enables Simplified Recovery from Speculative Execution Errors without Checkpoints

View U.S. Patent No. 9,244,772 in PDF format.

WARF: P110133US01

Inventors: Karthikeyan Sankaralingam, Marc De Kruijf, Chen-Han Ho

The Wisconsin Alumni Research Foundation (WARF) is seeking commercial partners interested in developing a computer architecture that enables simplified recovery after mis-speculation without the need to restore complex state information from checkpoints.

Overview

The effective speed of computer processors may be increased by speculative execution in which computer instructions are executed before the data necessary for that execution are available. Speculation is particularly useful in pipeline architectures, which largely require that later instructions be inserted into the pipeline before earlier instructions have exited, even though the later instructions depend on the earlier instructions.

As long as the prediction inherent in speculative execution is reasonably accurate, idle time of the processor is reduced and the instruction execution rate is increased. But if the speculation is in error, the speculatively executed instructions must be undone and the execution recovered. This squashing process is done by returning the processor to the same state it had before execution of the speculative instructions. The process relies on a checkpoint created at the beginning of the speculation, which accurately records the state of the processor at that time.

However, the circuitry needed to create and maintain checkpoints is complex and consumes substantial energy. Some processors that have large numbers of cores such as graphic processor units do not employ speculation because of the circuitry overhead.

The Invention

UW–Madison researchers have developed a simplified processor that can recover from mis-speculation or execution of other erroneous instructions without maintaining or reloading a conventional checkpoint. The processor simply re-executes from the start of a block of instructions that includes the erroneous instructions. This is possible by constructing programs in terms of consecutive idempotent regions, which produce the same results even when executed repeatedly, and limiting speculation to occur in those regions. The ability to recognize (at compile-time) and exploit idempotent regions eliminates much of the circuitry and energy consumption needed to recover from mis-speculation or hardware failure.

Applications

- Computer processing

Key Benefits

- Improves energy efficiency of the processor
- Lowers design complexity and therefore cost

- Requires no speculative state in the hardware
- Commonly performs within two to 16 percent of conventional out-of-order processors that use hardware checkpoints
- Compatible with existing processor designs
- Enables recovery from instruction errors without complex checkpoint circuitry
- Permits speculation in processors like graphic processors that normally would not support speculation circuitry
- Preserves the ability of memory dependence to be correctly resolved simply by re-executing the idempotent region
- Compatible with out-of-order execution
- Capable of handling hardware execution errors caused by intermittent thermal or electrical problems
- Allows compiler-based identification of idempotent regions

Additional Information

For More Information About the Inventors

- Karthikeyan Sankaralingam

Tech Fields

- Information Technology : Computing methods, software & machine learning
- Information Technology : Hardware

For current licensing status, please contact Jeanine Burmania at jeanine@warf.org or 608-960-9846