

## New system would allow programmers to easily trade computational accuracy for energy savings

October 30 2014, by Larry Hardesty



Credit: Jose-Luis Olivares/MIT

As transistors get smaller, they also grow less reliable. Increasing their operating voltage can help, but that means a corresponding increase in power consumption.

With information technology consuming a steadily growing fraction of



the world's energy supplies, some researchers and hardware manufacturers are exploring the possibility of simply letting chips botch the occasional computation. In many popular applications—video rendering, for instance—users probably wouldn't notice the difference, and it could significantly improve <u>energy efficiency</u>.

At this year's Object-Oriented Programming, Systems, Languages and Applications (OOPSLA) conference, researchers from MIT's Computer Science and Artificial Intelligence Laboratory presented a new system that lets programmers identify sections of their code that can tolerate a little error. The system then determines which program instructions to assign to unreliable hardware components, to maximize <u>energy savings</u> while still meeting the programmers' accuracy requirements.

The system, dubbed Chisel, also features a tool that helps programmers evaluate precisely how much error their programs can tolerate. If 1 percent of the pixels in an image are improperly rendered, will the user notice? How about 2 percent, or 5 percent? Chisel will simulate the execution of the image-rendering algorithm on unreliable hardware as many times as the programmer requests, with as many different error rates. That takes the guesswork out determining accuracy requirements.

The researchers tested their system on a handful of common imageprocessing and financial-analysis algorithms, using a range of unreliablehardware models culled from the research literature. In simulations, the resulting power savings ranged from 9 to 19 percent.

## Accumulating results

The new work builds on a paper presented at last year's OOSPLA, which described a programming language called Rely. Each paper won one of the conference's best-paper awards.



Rely provides the mechanism for specifying the accuracy requirements, and it features an operator—a period, or dot—that indicates that a particular instruction may be executed on unreliable hardware. In the work presented last year, programmers had to insert the dots by hand. Chisel does the insertion automatically—and guarantees that its assignment will maximize energy savings.

"One of the observations from all of our previous research was that usually, the computations we analyzed spent most of their time on one or several functions that were really computationally intensive," says Sasa Misailovic, a graduate student in <u>electrical engineering</u> and computer science and lead author on the new paper. "We call those computations 'kernels,' and we focused on them."

Misailovic is joined on the paper by his advisor, Martin Rinard, a professor in the Department of Electrical Engineering and Computer Science (EECS); by Sara Achour and Zichao Qi, who are also students in Rinard's group; and by Michael Carbin, who did his PhD with Rinard and will join the EECS faculty next year.

In practice, Misailovic says, programs generally have only a few kernels. In principle, Chisel could have been designed to find them automatically. But most developers who work on high-performance code will probably want to maintain a degree of control over what their programs are doing, Rinard says. And generally, they already use tools that make kernel identification easy.

## **Combinatorial explosion**

A single kernel, however, may still consist of 100 or more instructions, any combination of which could be assigned to unreliable hardware. Manually canvassing all possible combinations and evaluating their effects on both computational accuracy and energy savings would still be



a prohibitively time-consuming task.

But the researchers developed three separate mathematical expressions that describe accuracy of computation, reliability of instruction execution, and energy savings as functions of the individual instructions. These expressions constrain the search that the system has to perform to determine which instructions to assign to unreliable hardware. That simpler—though still complex—problem is one that off-the-shelf software can handle.

"I think it's brilliant work," says Luis Ceze, an associate professor of <u>computer science</u> and engineering at the University of Washington. "All the trends point to future hardware being unreliable, because that's one way of making it more energy-efficient and faster."

Ceze points out, however, that the hardware model the MIT researchers used requires specification of the reliability with which individual operations are executed. "That limits the savings that you can get from approximation," Ceze says. "I believe in approximation that is much coarser-grained. You do a whole bunch of operations in one shot, approximately. That can take you from tens of percent to tens of [multiples] in terms of performance and energy efficiency improvement."

As for whether the MIT researchers' work can be adapted to accommodate such a model, "It could be done," Ceze says. "In fact, my group has done similar work in a different context. So it's definitely viable."

**More information:** The paper is available online: people.csail.mit.edu/misailo/papers/chisel14.pdf



This story is republished courtesy of MIT News (web.mit.edu/newsoffice/), a popular site that covers news about MIT research, innovation and teaching.

## Provided by Massachusetts Institute of Technology

Citation: New system would allow programmers to easily trade computational accuracy for energy savings (2014, October 30) retrieved 28 April 2024 from <a href="https://phys.org/news/2014-10-programmers-easily-accuracy-energy.html">https://phys.org/news/2014-10-programmers-easily-accuracy-energy.html</a>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.