Changing the future in computers



Alberto Ros
University of Murcia, Spain

Multicore processors are present in most digital devices, from smartphones to high-performance servers. The increasing computational power of these processors is essential for enabling many important emerging application domains such as big data, media, medical, or scientific modelling.

A fundamental technique to improve computers performance is *speculation*. This technique consists of executing work before it is known if it is actually needed or even correct. If speculation is right, important performance gains are obtained. However, if the work executed speculatively is not useful, then a large penalty may be paid. For example, in hardware, speculation comes with a high cost to track the speculative state of the execution in case of misprediction, significantly increases energy consumption by performing operations that may be later discarded, and brings important security vulnerabilities, as shown by the Meltdown and Spectre attacks at the beginning of 2018. In software (e.g. compilers), speculation is not enabled by default thus limiting optimisations and leading to sub-optimal performance.

The ECHO (Extending Coherence for Hardware-Driven Optimizations in Multicore Architectures) project aims to remove the inefficiencies of speculation at both hardware and software levels, boosting the performance and energy efficiency of future computers. Since performance in current multicore processors is limited by their power budget, it is imperative to make multicore processors as energy-efficient as possible to increase performance even further.

The motivation behind ECHO to improve speculative execution is to change the future in computers, that is, to alter the events that happen in a computer such that they follow the initial prediction. This way, the execution of the work performed ahead of time does not need to be re-started. As Abraham Lincoln once said: "The best way to predict your future is to create it."

A key question in ECHO is: what if we could make it so that speculative execution is always successful? Then the execution would no longer be speculative. Computers would get all the advantages of speculation—without its cost—achieving more efficient run-time execution and enabling compile-time 'speculative' optimisations. Since they will always be correct, they will no longer be speculative.

Analogy: sunny or rainy?

Every morning when getting ready to go to work, Echo predicts that it is going to be a sunny day and decides to take a 20 minute walk to the office. Since Echo lives in the south of Spain, his prediction is correct most of the time. However, some days it starts raining during his walk, and he has to go back home, change his clothes, and take the car to go to the office, losing valuable time.

But what if the rain could be delayed for a few minutes until Echo arrives at the office, such that his decision is always correct? This challenge is the aim of the ECHO project. If one cannot predict the future correctly, why not try to change the future to follow the prediction?

ECHO foundations

In a multicore architecture, the cache coherence protocol is an essential component since its unique but challenging role is to offer a simple and unified view of the memory hierarchy, which eases computers programming. ECHO envisions that extending the role of the coherence protocol allowing simplifications other system components will be the key to overcome the performance and energy limitations of current multicore processors. ECHO proposes to add simple but effective extensions to the cache coherence protocol to: (1) reduce and even eliminate mis-speculations at the processing cores and synchronisation mechanisms; and (2) enable

Adobe Stock © Graphic in Motion

speculative optimisations at compile time. The end goal of this innovative approach is to improve the performance and energy efficiency of future multicore architectures.

As a motivational example, ECHO focuses on techniques that allow increasing the instruction parallelism of computers by reordering instructions, while guaranteeing correct execution, that is, instructions *appear* to be executed in the order they reside in the programs. This is fundamental for ease of programming and debugging.

Let us consider two instructions that read two independent values from the computer memory. In computers with an easy-to-program, strong memory model, instructions will always appear to be executed in the order described in the program, even though the processor may speculatively reorder them for performance reasons. Current processors speculatively allow reordering reads, and cancel them in case they are incorrect (e.g. other processing core writes to the same memory location). With some changes in the coherence protocol, writes can be delayed until it is not possible to observe the read performed in an order contradicting program order. This way, reads will always be correct, as no write can happen to the same location while reads are reordered. Following the rain analogy, this corresponds to delaying the rain until Echo arrives at the office, such that the prediction is correct. Once the read is in order, the write can proceed (e.g. it can start raining). This concept, WritersBlock, was published in the most prestigious conference in computer architecture (Ros et al., 2017), and selected as one of the 12 most relevant papers published in 2017 in computer architecture (Kaxiras et al., 2018). Furthermore, this idea can be leveraged to optimise value-based speculation (Ros and Kaxiras, 2018b).

Different from instructions that read from memory, instructions that write to memory are not, even speculatively, reordered in current multicore processors. The reason being that cancelling writes from memory in case speculation is wrong is a costly operation. However, we have demonstrated that reordering write instructions is also possible in a non-speculative way (Ros and Kaxiras, 2018a). The key is that reordered writes need to have effect atomically in memory. The cache coherence protocol can delay both reads and writes from remote cores such that the writes in the local core perform in an atomic way. This was the first non-speculative solution that allowed reordering of write operations guaranteeing that such reorder is not visible to the programmer. These concepts form the ECHO foundations.

First research outcome: entangling prefetchers

Current computers use several memory (cache) levels to access fast data that the processing units need. Lower levels, closer to the processing units, are smaller and faster than higher levels. When a processor accesses data, the ideal scenario is that data are present in the lowest level, guaranteeing fast access to the information, and therefore high performance.

A key technique to find requested data in the lowest cache level is to bring (prefetch) data predicted to be used in the near future to that cache level. Prefetching is another example of changing the future: an access that would not find the data in the lowest cache level will indeed find it if the memory system places it there in advance to the actual access.

The first outcome of ECHO is a novel prefetching technique, named entangling prefetchers. In an analogy to quantum entanglement, entangling prefetchers consist in pairing two pieces of data such that the access of one by the processor will imply the access of the other in a near future (Figure 1). Entanglement is done by carefully measuring the time data need to be stored in the lowest cache level, and once the first piece of data is accessed by the processor, the second is requested to the memory system too for placing in the lowest cache level.

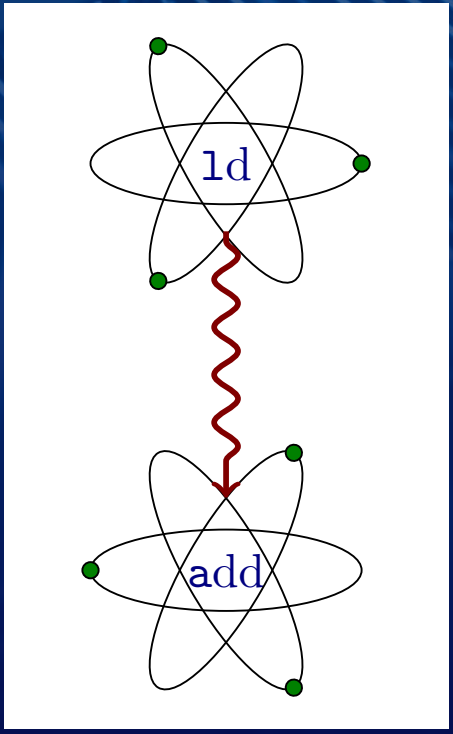


Figure 1: The entangling instruction prefetcher.

An entangled prefetcher for instructions recently won the first Instruction Prefetching Championship[1], which belongs to a series of prestigious championships in computer architecture. The entangling prefetcher obtained 29.5 per cent performance improvements over a computer that did not implement instruction prefetching. An energy- and area- efficient version of the entangling instruction prefetcher obtaining over 28 per cent performance improvements has recently been published in a peer-reviewed journal (Ros and Jimborean, 2020)

Impact

With its innovative concepts, ECHO will be eliminating the costly speculative execution that has been employed for decades and transforming it into a low-cost non-speculative execution. This change brings low-cost solutions for memory reordering, concurrent mutual exclusion, and compile-time speculative optimisations. The overall impact of these low-cost solutions is estimated to be two times faster and 50 per cent energy-consuming multicore processors.

1 https://research.ece.ncsu.edu/ipc/

References

Ros, A., Carlson, T., Alipour, M. and Kaxiras, S. (2017) Non-Speculative Load-Load Reordering in TSO. *44th Annual International Symposium on Computer Architecture* (ISCA '17), IEEE Press, pp. 187–200.

Ros, A. and Kaxiras, S. (2018a) Non-Speculative Store Coalescing in Total Store Order. In *45th Annual International Symposium on Computer Architecture* (ISCA '18), IEEE Press, pp. 221–234.

Ros, A. and Kaxiras, S. (2018b) The Superfluous Load Queue. In *51st International Symposium on Microarchitecture (MICRO)*, IEEE Press, pp. 95–107.

Kaxiras, S., Carlson, T., Alipour, M. and Ros, A. (2018) Non-speculative load reordering in total store ordering. *IEEE Micro*, 38(3), pp. 48-57.

Ros, A. and Jimborean, A. (2020) Entangling Prefetcher for Instructions. *IEEE Computer Architecture Letters*, June 2020.

PROJECT SUMMARY

The ERC Consolidator Grant project "ECHO: Extending Coherence for Hardware-Driven Optimizations in Multicore Architectures", aims to change the events that occur in multiprocessors such that predictions or decisions being made by the processing units will find the best possible outcome in the future. This way, large performance and energy improvements are expected in future multiprocessors leveraging ECHO concepts.

PROJECT LEAD PROFILE

Alberto Ros is an associate professor at the University of Murcia, Spain. Funded by the Spanish government to conduct PhD studies, he received a PhD in computer science from the University of Murcia in 2009. He received an ERC Consolidator Grant in 2018 to improve the performance of multicore architectures. Working on cache coherence, memory consistency, and processor microarchitecture, he has co-authored more than 70 peer-reviewed articles.

PROJECT PARTNERS

The ECHO project is based at the Faculty of Computer Science of the University of Murcia. The Computer Engineering department has a long tradition in computer architecture research. A collaboration with Uppsala University, established in the past, will be reinforced for the duration of the project.

CONTACT DETAILS

Alberto Ros

ECHO: Extending Coherence for Hardware-Driven Optimizations in Multicore Architectures

Facultad de Informática, Campus de Espinardo, 30100

+34 868 888 518

aros@um.es

http://webs.um.es/aros

FUNDING

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement no. 819134.