

The First JILP Data Prefetching Championship (DPC-1)

<http://www.jilp.org/dpc>

The tremendous increase in processor frequency over several decades has made memory latency a dominant factor influencing system performance. Even large caches cannot completely bridge the divide between processor and memory speeds. In recent years, there has been great progress in the advancement of prefetching techniques. We are optimistic that even more innovative prefetchers are on the horizon.

The purpose of the First JILP Data Prefetching Championship (DPC-1) is to provide a dedicated venue for computer architecture researchers to put their latest prefetching ideas and implementations to the test. The spirit of competition is intended to provide incentive for students, faculty, and industry to push the envelope of prefetching. Contestants competed for the top honor of best-performing prefetcher, in the context of a common simulation infrastructure, common workloads, a fixed storage budget, and diverse cache and bandwidth scenarios.

Contestants submitted source code and a paper describing their prefetcher. Their prefetchers were tested by the DPC-1 organizers on the common simulation infrastructure and ranked by a summary performance metric. Separately and without knowledge of the performance ranking, the papers were reviewed by the program committee to ensure a quality program and understand the contestants' prefetchers.

In its inaugural year, the competition has been a great success. We received a total of 13 papers and 20 code submissions (contestants could submit up to three different prefetchers or variations of their prefetchers). Eight of the papers were assembled for this workshop proceedings. These papers are listed in Table 1. From what is described in the papers, we broadly classified each prefetcher based on the style of prefetching that it is closely related to. All source code, papers, and detailed results on the common infrastructure are publicly available on the DPC-1 website, so that the computer architecture community can benefit from the results.

Alaa R. Alameldeen, *Intel*
DPC-1 Organizer

Eric Rotenberg, *NC State University*
DPC-1 Organizer and Program Chair

Table 1. Papers in this proceedings (ordered alphabetically by last name of first author) and their prefetcher classification.

Authors	Paper Title	Prefetcher Taxonomy
M. Dimitrov, H. Zhou	Combining Local and Global History for High Performance Data Prefetching	Global/stride, PC/delta
M. Ferdman, S. Somogyi, B. Falsafi	Spatial Memory Streaming with Rotated Patterns	Region
M. Grannaes, M. Jahre, L. Natvig	Storage Efficient Hardware Prefetching using Delta Correlating Prediction Tables	PC/delta
Y. Ishii, M. Inaba, K. Hiraki	Access Map Pattern Matching Prefetch: Optimization Friendly Method	L1: stream, L2: Czone/stride
G. Liu, Z. Huang, J.-K. Peir, X. Shi, L. Peng	Enhancement for Accurate Stream Prefetching	Stream
L. Ramos, J. Briz, P. Ibañez, V. Viñals	Multi-level Adaptive Prefetching based on Performance Gradient Tracking	L1: sequential, L2: PC/delta
A. Sharif, H. Lee	Data Prefetching Mechanism by Exploiting Global and Local Access Patterns	Global/delta, PC/stride
S. Verma, D. Koppelman, L. Peng	A Hybrid Adaptive Feedback Based Prefetcher	PC/stride

About Performance Scores

We provided a common framework for contestants to evaluate the performance of their prefetching techniques. Details about the simulator and the information it provides can be found on the DPC-1 website: <http://www.jilp.org/dpc/framework.html>. Each contestant's code was evaluated on three configurations that differ in their L2 cache sizes and available bandwidth:

1. **Large cache, unlimited bandwidth.** The L2 cache size is 2 MB, the L1 to L2 bandwidth is infinite, and the L2 to memory bandwidth is infinite.
2. **Large cache, limited bandwidth.** The L2 cache size is 2 MB, the L1 to L2 bandwidth is limited to one request per cycle, and the L2 to memory bandwidth is limited to one request every ten cycles.
3. **Small cache, limited bandwidth.** The L2 cache size is 512 KB, the L1 to L2 bandwidth is limited to one request per cycle, and the L2 to memory bandwidth is limited to one request every ten cycles.

We used a subset of the SPEC 2006 benchmarks to evaluate performance for each contestant's code on the three configurations. We used the first reference input, warmed up the benchmarks for 40 billion instructions, and simulated the following 100 million instructions. We computed the speedup over no prefetching for each simulation run across all benchmarks. Each contestant got three scores, one for each configuration, representing the geometric mean of the speedup (over no prefetching) across all benchmarks. The final performance score was the sum of the scores for the three

configurations. As a reference, the “no prefetching” technique has a score of 1.0 for each configuration, for an overall performance score of 3.0. We also distributed a rudimentary PC-based stride prefetcher with the simulator. That simple prefetcher had scores of 1.14, 1.07 and 1.06 for the three configurations, for an overall performance score of 3.27.

Figure 1 shows anonymous performance scores of all 20 code submissions using the common infrastructure. The actual performance rankings will be announced live at the workshop, and detailed results as well as prefetcher codes will be posted on the DPC website after the workshop. The top three performers all had scores above 4.3, which is a significant improvement over no prefetching and over many prefetching implementations.

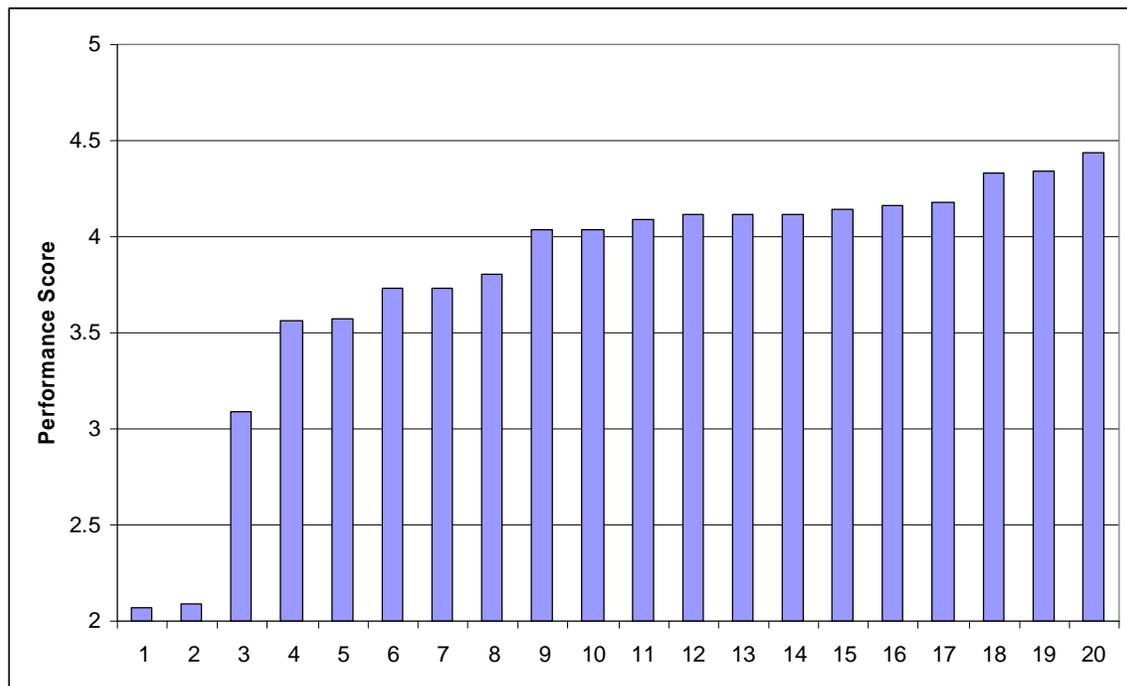


Figure 1. Performance scores for all twenty code submissions. Note that the results are anonymous, and numbers on the X-axis do not correspond to paper numbers.

DPC-1 Committees

Steering Committee

Alaa R. Alameldeen, *Intel*

Eric Rotenberg, *North Carolina State University*

Organizing Committee

Alaa R. Alameldeen, *Intel*

Zeshan Chishti, *Intel*

Aamer Jaleel, *Intel*

Daniel Luchi, *Intel*

Chris Wilkerson, *Intel*

Program Chair

Eric Rotenberg, *North Carolina State University*

Program Committee

Alaa Alameldeen, *Intel*

Yuan Chou, *Sun*

David Kaeli, *Northeastern*

Alvin Lebeck, *Duke*

Kyle Nesbit, *Google*

Suleyman Sair, *Intel*