On comparison of R performance with hardware enabled parallelism

Hui Zhang
Pervasive Technology Institute
Indiana University
huizhang@iu.edu

Weijia Xu
TACC
University of Texas at Austin
xwj@tacc.utexas.edu

Huian Li
Pervasive Technology Institute
Indiana University
huili@iu.edu

Yaakoub El Khamra
TACC
University of Texas at Austin
yaakoub@tacc.utexas.edu

Niall Gaffney
TACC
University of Texas at Austin
ngaffney@tacc.utexas.edu

Eric Wernert
Pervasive Technology Institute
Indiana University
ewernert@iu.edu

ABSTRACT

R has been adopted as a popular data analysis and mining tool in many domain fields over the past decade. With recent hardware and software developments, it is possible to enable massive parallelism with existing R solutions with little to no modification. In this poster, we evaluated approaches to speed up R computations with the utilization of the multiple cores, Intel Xeon Phi SE10P Co-processor and general purpose graphic processing unit using a popular R benchmark.

we used the R-25 benchmark script for testing performance of different approaches[1]. The testing script includes fifteen common computational tasks grouped into three categories: *Matrix Calculation*, *Matrix functions* and *Programmation*. The Stampede cluster at Texas Advanced computing Center was used for our testing environment. Stampede supports several latest hardware technologies for improved computational performance including using Xeon Phi accelerators and/or NVIDIA Kepler 20 GPUs for large matrix calculations. Our experiment focused on the benefit of R programs exploiting Xeion Phi co-processor with *MKL*, multicore computing, and GPU technologies with *HiPLAR*. Based on our previous observation of significant performance improvement of benchmark version of R computation using MKL and offload model, we test R25 benchmark script by choosing work-sharing at the 30% host (16 threads) 70% coprocessor (240 threads) sweet spot (see e.g., [1]). We then tested R25 with multi-core and GPU acceleration using HiPLAR package. Figure 1 indicates the obtained speed-ups from the three strategies we tried. We consistently achieve significant speed-ups over various matrices size and matrix based functions. These methods/packages strive to retain the optimization within the Matrix package in R. The user will notice no difference between using the offload model, the functions of HiPLAR packages or indeed, results. This feature enables programmers easily access to the latest computational architectures though the linked linear algebra libraries.

[1]http://r.research.att.com/benchmarks/

XSEDE'14, July 13 - 18 2014, Atlanta, GA, USA

http://dx.doi.org/10.1145/2616498.2616557.

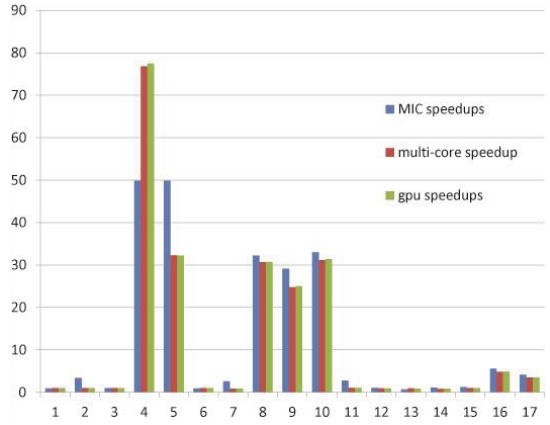

Figure 1: Basic vectorized and matrixed operations can obtain significant speed-ups by using offload model with MKL and MIC, multi-core, and GPU technologies.

Categories and Subject Descriptors

D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Performance Evaluation, Intel Xeon Phi, GPGPU

Keywords

High Performance Computing, R

1. REFERENCES

[1] Y. El-Khamra, N. Gaffney, D. Walling, E. Wernert, W. Xu, and H. Zhang. Performance evaluation of r with intel xeon phi coprocessor. In *Big Data, 2013 IEEE International Conference on*, pages 23–30. IEEE, 2013.