Scalable and Power Efficient Data Analytics for Hybrid Exascale Systems

Alok Choudhary (PI), Wei-keng Liao, Northwestern University
Nagiza Samatova, North Carolina State University and Oak Ridge National Laboratory
John Wu, Lawrence Berkeley National Laboratory

Overview

Recent DOE Workshops on Exascale Computing have articulated emerging trends related to data, hardware, and energy issues that necessitate the next-generation algorithms and software libraries for data analysis and mining. They recognized that the increasing gap between the opportunities created by these trends and the current data analytics capabilities will soon become a major bottleneck on our path to exascale. Given the data trend—scientific data grow not only in the size but also in the complexity—the demands for more sophisticated analyses increase. The execution of many data analysis algorithms is dominated by a small number of kernels.

This project is to provide a generic and highly optimized set of cores, or kernel, analytics functions, so that a broad constellation of high performance analytical pipelines could be organically consolidated. The expected outcome is a comprehensive library of such exascale data analysis and mining kernels. In the long term, this project will bring the development of analytics algorithms to the next level: the impact maybe akin to that of the ScaLAPACK linear algebra library in scientific computing. Furthermore, as the architectures of emerging HPC systems are becoming inherently heterogeneous, this project will design algorithms for data analysis kernels accelerated on hybrid multi-node, multi-core HPC architectures comprised of a mix of CPUs, GPUs, and SSDs. Finally, the power-aware trend drives the proposed advances in our performance-energy tradeoff analysis framework that would enable our data analysis kernels algorithms and software to be parameterized so that users can choose the right power-performance optimizations.

Impact

The apex of this proposal is a library of functions and software to accelerate data analytics, mining, knowledge discovery for large-scale scientific applications, thereby, increasing productivity of both scientists and the systems.

Current Activities

R Statistical Functions in CUDA—We are developing a set of statistical functions from the R software package that can be run on the GPUs. Being open-source and incorporating a large pool of statistical functions, R is very popular among the research community of data analysis and mining. As an illustration, Table 1 lists important statistical kernels for representative data mining algorithms in classification, association

Table 1. The frequency of kernel operations in illustrative data mining algorithms and applications.

Application	Top 3 Kernels (%)			Sum
	Kernel 1 (%)	Kernel 2 (%)	Kernel 3 (%)	%
K-means	Distance (68)	Center (21)	minDist (10)	99
Fuzzy K-means	Center (58)	Distance (39)	fuzzySum (1)	98
BIRCH	Distance (54)	Variance (22)	redist.(10)	86
HOP	Density (39)	Search (30)	Gather (23)	92
Naïve Bayesian	probCal (49)	Variance (38)	dataRead (10)	97
ScalParC	Classify (37)	giniCalc (36)	Compare (24)	97
Apriori	Subset (58)	dataRead (14)	Increment (8)	80
Eclat	Intersect (39)	addClass (23)	invertC (10)	72

rule mining, and clustering. Note that the fraction of execution by the top three kernels usually exceeds 90% and reaches up to 99% for this set of algorithms. If these kernels were effectively executed, the overall applications could be significantly accelerated. Table 2 lists the progress on implementing R statistical functions and data mining kernels. Other functions (not listed here) are also in progress.

Table 2. Development status.

Functions in R	Description	CUDA
max, mean, median, min	primitive statistical functions	√
sd	Standard Deviation	√
hist	Histogram	√
crossprod	Cross Product	√
prod	Product of Vector Elements	√
var	Covariance matrices	√
cor	Correlation and covariance matrices	√
diff	Lagged differences	√
fivenum	Turkey five-number summaries	√
cum (sum, min, max prod)	Cumulative sum, min, max, product	√
sort	Sort a vector	o
svd	Singular value decomposition	o
DBSCAN	Density-based spatial clustering	o
Bisecting K-Means	Modified Kmeans clustering	√
Taper	Associate rule mining	√

√: completed o: in progress

Backtracking Paradigm on GPU—The backtracking paradigm, a depth-first search method that finds solutions in a memory efficient manner, is ubiquitous in computational sciences. Examples include constraint satisfaction in AI, frequent itemset mining, maximal clique enumeration in graph mining, k-d tree traversal for ray tracing in graphics, and logic programming languages. We explored the backtracking paradigm with properties seen as sub-optimal for GPU architectures, using as a case study the maximal clique enumeration problem, and found that the presence of these properties limited GPU performance to approximately 1.4–2.25 times a single CPU core. The GPU performance "lessons" we found critical to providing this performance included a coarse-and-fine-grain parallelization of the search space, a low-overhead load-balanced distribution of work, global memory latency hiding through coalescence, saturation, and shared memory utilization, and the use of GPU output buffering as a solution to irregular workloads and a large solution domain. We also found a strong reliance on an efficient global problem structure representation that bounds any efficiencies gained from these lessons, and highlighted the potential for backtracking problems with GPU-friendly properties, such as k-d tree construction and traversal for GPU ray-tracing, to be successfully accelerated on the GPU. This study was performed in collaboration with Dr. John D. Owens of U.C. Davis.

Index-Based Data Analysis and Mining Kernels—We are designing index-based analytics algorithms for both performance and power improvements. Two complementary strategies are under development: (a) selective data mining kernels with FastBit and (b) index-based perturbation analysis kernels for noisy and uncertain data. These approaches perform data analysis tasks on the indexed data rather than on the raw data. Not only do they offer an order of magnitude faster solution but reduce memory requirements by bitmap data compression and minimize data movement required for parallel processing. In addition, they hold a substantial promise for saving power. The approaches for measuring power and memory access efficiency are also under development. . As the first step, we have implemented a system for indexing and querying array-based data such as those in NetCDF and HDF5. This system exports the indexing capability of FastBit to new types of data files and significantly improves the query performance.

Contact

For more information, please contact Alok Choudhary <choudhar@eecs.northwestern.edu>.

U.S. DEPARTMENT OF ENERGY NORTHWESTERN UNIVERSITY NC STATE UNIVERSITY OAK RIDGE National Laboratory BERKELEY LAB