Scalable and Power Efficient Data Analytics for Hybrid Exascale Systems

Alok Choudhary¹, Wei-keng Liao¹, William Hendrix¹, Prabhat Kumar¹, Nagiza Samatova², John Wu³
¹Northwestern University, ²North Carolina State University, ³Oak Ridge National Laboratory, ⁴Lawrence Berkeley National Laboratory

Project Goals

Data Mining Kernels on Hybrid Architectures

- GPU-based analysis kernels
- Coordinated CPU-GPU programming
- SSDs for out-of-core computation

Energy Efficiency

- Optimizing data access in memory hierarchy
- Developing approximate analytics kernels
- Performance and energy trade-offs

Index-based Query and Analysis

- Data analysis on indexed data based on FastBit
- Parallel query on distributed indexed data
- Perturbation analytics for noisy and uncertain data

Data Analytics Kernels

The frequency of kernel operations in illustrative data mining algorithms

<table>
<thead>
<tr>
<th>Application</th>
<th>Top 3 Kernels</th>
<th>Sum %</th>
</tr>
</thead>
<tbody>
<tr>
<td>K-means</td>
<td>minDist (10)</td>
<td>58</td>
</tr>
<tr>
<td>Fuzzy K-means</td>
<td>Center (21)</td>
<td>58</td>
</tr>
<tr>
<td>BIRCH</td>
<td>variance (22)</td>
<td>86</td>
</tr>
<tr>
<td>HOP</td>
<td>search (30)</td>
<td>92</td>
</tr>
<tr>
<td>Naive Bayesian</td>
<td>probCal (49)</td>
<td>97</td>
</tr>
<tr>
<td>ScalParC</td>
<td>gminCalc (36)</td>
<td>97</td>
</tr>
<tr>
<td>Apriori</td>
<td>dataRead (14)</td>
<td>97</td>
</tr>
<tr>
<td>Eclat</td>
<td>addClass (23)</td>
<td>97</td>
</tr>
<tr>
<td>SVMight</td>
<td>quadGrad (38)</td>
<td>97</td>
</tr>
</tbody>
</table>

- Performance of representative data mining algorithms is dominated by a small number of kernels
- Top 3 kernels usually exceed 90% of execution time
- If these kernels were effectively executed, the overall applications could be significantly accelerated
- Research products
  - A C/Fortran/Cuda library of highly optimized analytical kernels
  - A framework for programmers to combine these kernels
  - Integration to popular analytics/visualization software, such as Matlab, R, VisIt.

An example of data analytics, statistics, and mining functions that will be explored as a part of this project

- Global reduction: sum, min, max, mean
- Time series similarity: TAPER, mutual information
- Distribution: standard deviation, histograms, etc
- Data Preprocessing (e.g., dimensionality reduction): PCA, ABB, LVF
- Clustering: K-means, MAFIA, DBSCAN, Bisecting K-means, SNN
- Anomaly/Outlier Detection: LOF, Outlier Detection
- Change Detection: Change detection in time series
- Predictive modeling, classification: ScalParC, Decision trees, Naive Bayesian, RIPPER, SVM
- Association rule mining: Apriori, FP-growth, MAFIA
- Feature extraction: Edge detection, Blob detection

Harnessing Hybrid Architectures

- Design algorithms for data analysis kernels accelerated on hybrid HPC with a mix of multi-core CPUs, GPUs, and SSDs
- Data access management of memory hierarchy (hard disks / SSDs / host memory / GPU memory)
  - Develop a memory tiling technique
  - Coordinated CPU and GPU analytical computing
  - Out-of-core analytics using SSDs
  - Utilize data locality and parallel I/O techniques

Lessons Learned from Exploring Backtracking Paradigm on the GPU

- Backtracking is a depth-first exploration of a problem space, ubiquitous in memory-intensive data analytics problems.
  - Using exemplar problem of Maximal Clique Enumeration, performance of hard backtracking problems on GPU limited to ~2.25 times a single CPU core (see table for def. of hardness).
  - Optimizations due to efficient output buffering, load-balanced, fine-grain parallelization of search, and memory latency hiding through saturation and efficient memory usage essential to performance.

Utilizing Indexed Data

- Develop index-based data analysis kernels and algorithms for performance and power optimizations
- Query Processing
  - Indexing array-based data
  - FastBit compressed bitmap indexing
  - FastBit indexes can answer queries more than 10X faster than commonly used techniques
  - FastBit indexing has been extended to HDF5 and NetCDF formats

Approximate analytics algorithms

- Convert floating-point operations into integer operations
- Provide better performance and are more energy efficient
- Use multi-level strategies via divide-and-conquer
- Approaches
  - Parallelization and acceleration on GPUs
  - Determining performance and accuracy with respect to original algorithms
  - User power measurement devices to evaluate the actual energy consumption

Project contact: Alok Choudhary <choudhar@eecs.northwestern.edu>

- Perturbation analytics for noisy and uncertain data
- An example of data analytics, statistics, and mining functions that will be explored as a part of this project
- Global reduction: sum, min, max, mean
- Time series similarity: TAPER, mutual information
- Distribution: standard deviation, histograms, etc
- Data Preprocessing (e.g., dimensionality reduction): PCA, ABB, LVF
- Clustering: K-means, MAFIA, DBSCAN, Bisecting K-means, SNN
- Anomaly/Outlier Detection: LOF, Outlier Detection
- Change Detection: Change detection in time series
- Predictive modeling, classification: ScalParC, Decision trees, Naive Bayesian, RIPPER, SVM
- Association rule mining: Apriori, FP-growth, MAFIA
- Feature extraction: Edge detection, Blob detection

An example of data analytics, statistics, and mining functions that will be explored as a part of this project

- Global reduction: sum, min, max, mean
- Time series similarity: TAPER, mutual information
- Distribution: standard deviation, histograms, etc
- Data Preprocessing (e.g., dimensionality reduction): PCA, ABB, LVF
- Clustering: K-means, MAFIA, DBSCAN, Bisecting K-means, SNN
- Anomaly/Outlier Detection: LOF, Outlier Detection
- Change Detection: Change detection in time series
- Predictive modeling, classification: ScalParC, Decision trees, Naive Bayesian, RIPPER, SVM
- Association rule mining: Apriori, FP-growth, MAFIA
- Feature extraction: Edge detection, Blob detection

Functionality | Examples
--- | ---
Global reduction | sum, max, min, mean
Time series similarity | TAPER, mutual information
Distribution | standard deviation, histograms, etc
Data Preprocessing (e.g., dimensionality reduction) | PCA, ABB, LVF
Clustering | K-means, MAFIA, DBSCAN, Bisecting K-means, SNN
Anomaly/Outlier Detection | LOF, Outlier Detection
Change Detection | Change detection in time series
Predictive modeling, classification | ScalParC, Decision trees, Naive Bayesian, RIPPER, SVM
Association rule mining | Apriori, FP-growth, MAFIA
Feature extraction | Edge detection, Blob detection

Problem Instance | Backtracking (worst-case) | GPU optimal
--- | --- | ---
Irregular access pattern (e.g. sparse matrix multiplication) | Regular access with locality (e.g. dense matrix multiplication)
Variable in size and computation (e.g. CSP on large sets) | Constant size, SIMD (e.g. stream processing)
Exponential size, hard to estimate (e.g. subset enumeration) | Polynomial size, apriori (e.g. dense matrix multiplication)
Tree-based, unbalanced (e.g. 8-queens) | Fixed, apriori (if applicable) (e.g. k-d trees)