DOE award #: DE-FC02-10ER26015/DE-SC0005340
Name of the recipient: Northwestern University
Project title: Scalable and Power Efficient Data Analytics for Hybrid Exascale Systems
PI: Alok Choudhary, Northwestern University;
Co-PIs: Nagiza Samatova, North Carolina State University & Oak Ridge National Laboratory; and Kesheng Wu, Lawrence Berkeley National Laboratory; Wei-keng Liao, Northwestern Unviersity
Date of the report: March 19, 2015

Project Final Report/Accomplishments

Project description:

This project developed a generic and optimized set of core data analytics functions. These functions organically consolidate a broad constellation of high performance analytical pipelines. As the architectures of emerging HPC systems become inherently heterogeneous, there is a need to design algorithms for data analysis kernels accelerated on hybrid multi-node, multi-core HPC architectures comprised of a mix of CPUs, GPUs, and SSDs. Furthermore, the power-aware trend drives the advances in our performance-energy tradeoff analysis framework which enables our data analysis kernels algorithms and software to be parameterized so that users can choose the right power-performance optimizations.

Technical Progress and Accomplishments:

1. Parallel Single-Linkage Hierarchical Clustering*:* Hierarchical clustering is the problem of discovering the large-scale cluster structure of a dataset by forming a dendrogram that attempts to capture the clustering behavior in the dataset – from the most general cluster that encompasses the entire dataset, all the way to the most stringent clusters that only include a single data points each. Hierarchical clustering algorithms can be split into two main types: a) agglomerative, in which data points begin in separate clusters which are progressively merged until all points are in a single cluster; or b) divisive, in which all points begin in a single cluster, and clusters are split until each data point resides in a separate cluster. There are also several metrics for deciding which clusters to merge or divide at each level of the hierarchy, including single-linkage, complete-linkage, average-linkage, centroid, and Ward's method. Here, we focused on single-linkage agglomerative clustering, in which the clusters with the two closest data points are merged at each step. The single-linkage hierarchical clustering problem is equivalent to the problem of calculating a minimum spanning tree (MST) on a complete weighted graph, where the edge weights are the distance between the corresponding data points in the dataset. Hierarchical clustering is a challenging problem to parallelize, as other parallel techniques rely on calculating the distance matrix, which is infeasible to store for large datasets.

To parallelize single-linkage hierarchical clustering, we split a large problem instance into several smaller, overlapping problem instances that can be solved independently and then combined to find the solution to the original problem. To do this, we define the sub-problems by splitting the original dataset into k equal parts, and we assign a processor to compute the MST for each pair of these parts, using $k(k-1)/2$ processors. One of the main advantages of this technique is that it can compute solutions for each dataset independently, and we only need to communicate between processes afterwards, even though we may compute the distance between some pairs of points several times (up to twice as many total distance calculations). This allows us to achieve good scalability on large-scale distributed memory system. Once we have computed the MST for each subgraph, we combine them using a modified version of Kruskal's algorithm. First, we sort the edges of each MST in order of increasing weight, and

then we combine MSTs two at a time, using a Disjoint Set (or Union-Find) data structure to eliminate edges that form cycles (connect vertices in the same component). By combining these MSTs in a binary fashion, we can eliminate redundant MST edges quickly and perform the combination in parallel. Additionally, by combining the MSTs of datasets that overlap first, we can eliminate duplicate edges quickly, reducing the total communication cost of the algorithm.

The software was developed in C with MPI for interprocess communication. The code was compiled and run on the Edison machine at NERSC. Edison is a Cray XC30 supercomputer at National Energy Research Scientific Computing Center (NERSC). It has 664 compute nodes with 64 GB memory per node. Each node consists of two 8-core Intel Sandy Bridge processors at 2.6 GHz.

We evaluated parallel SLINK on 12 synthetic datasets, as well as 4 sampled cosmology datasets. We generated two types of synthetic data, clustered and uniform random data, across both 10 and 20 dimensions in three different data sizes: 100k, 500k, and 1M data points. For the synthetic clustered data, we used 8 random cluster centers with equal-size Gaussian distributions around each. The largest datasets we tested were the sampled cosmology datasets, with 3–5 million data points. With the increased amount of computation, we saw super-linear speedups on all four datasets up to 6050 processes (Figure 1), even after I/O and communication costs were included, most likely due to caching effects. This work was published in LDAV 2013 [LDAV13].

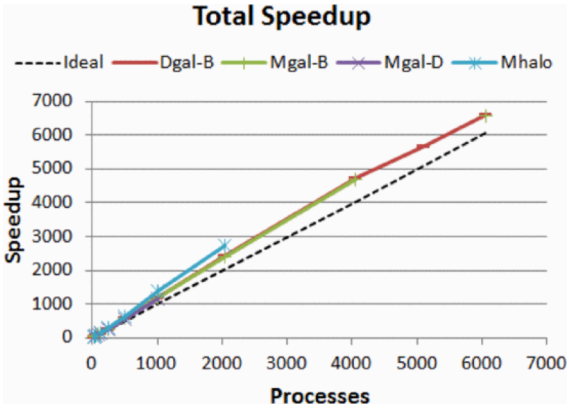

Figure 1: Empirical results for running parallel SLINK on the four sampled cosmology datasets.

2. Parallel DBSCAN*:* DBSCAN is a clustering algorithm that relies on a density-based notion of clusters. The algorithm is capable of discovering arbitrary shaped clusters and also able to handle noise or outliers effectively. The basic concept of the algorithm is that for each data point in a cluster, the neighborhood within a given radius (*eps*) has to contain at least a minimum number of points (*minpts*), i.e. the density of the neighborhood has to exceed some threshold. Although there have been several prior efforts to parallelize DBSCAN, many of them adopt the master-slave model. The data is equally partitioned and distributed among the slaves, each of which computes the clusters locally and sends back the results to the master, which sequentially merges the partial cluster results to obtain the final result. This strategy incurs a high communication overhead between the master and slaves, and a low parallel efficiency for the merging process.

We redesign the DBSCAN algorithm using disjoint-set data structure to make it amenable to parallelization. Our new DBSCAN algorithm exploits the similarities between region growing and identifying connected components in a graph. The algorithm initially creates a singleton tree for each data point. It then keeps merging those trees that belong to the same cluster until all the clusters are discovered. The algorithm eventually produces multiple trees, each one representing a different cluster. To break the inherent data access order and to perform merging efficiently, disjoint-set data structure is used.

To perform experiments, we use Hopper, a Cray XE6 distributed memory parallel computer where each node has two twelve-core AMD 'MagnyCours' 2.1-GHz processors and shares 32 GB of memory. Each core has its own 64 KB L1 and 512 KB L2 caches. Each six cores on the MagnyCours processor share one 6 MB of L3 cache. The algorithms have been implemented in C/C++ using the MPI message-passing library.

We again used synthetic and real datasets to test the scalability of parallel DBSCAN. On a astrophysics dataset *Halo* of 72 million points, we recorded a speedup of 5,765 using 8,192 process cores (Figure 2). This work was published in Supercomputing conference 2012 [SC2012].

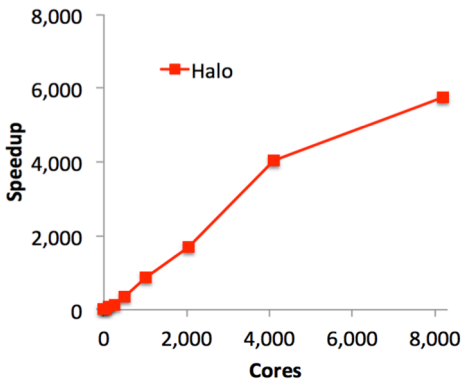

Figure 2: Empirical results for running parallel DBSCAN on a cosmology dataset.

***3. Parallel OPTICS**:* OPTICS (Ordering Points To Identify the Clustering Structure) is a hierarchical density-based clustering algorithm. DBSCAN is a special case of OPTICS. OPTICS addresses DBSCAN's major limitation: the problem of detecting meaningful clusters in data of varying density. OPTICS provides an overview of the cluster structure of a dataset with respect to density and contains information about every cluster level of the dataset. In order to do so, OPTICS generates a linear order of points where spatially closest points become neighbors. Additionally, for each point, a spatial distance (known as reachability distance) is computed which represents the density. Once the order and the reachability distances are computed using *eps* and *minpts*, we can query for the clusters that a particular value of *eps'* (known as clustering distance) would give where *eps'* ≤ *eps*. The query is answered in linear time. One example application of OPTICS, which requires high performance computing, is finding halos and subhalos (clusters) from massive cosmology data in astrophysics. OPTICS is challenging to parallelize though as its data access pattern is inherently sequential.

To overcome this challenge, we develop a scalable parallel OPTICS algorithm (POPTICS) using graph algorithmic concepts. POPTICS exploits the similarities between OPTICS and PRIM's Minimum Spanning Tree (MST) algorithm to break the sequential access of data points in the classical OPTICS algorithm. The main idea is that two points should be assigned to the same cluster if they are sufficiently close (if at least one of them has sufficiently many neighbors). This relationship is transitive so a connected component of points should also be in the same cluster. If the distance bound is set sufficiently high, all vertices will be in the same cluster. As this bound is lowered, the cluster will eventually break apart forming sub-clusters. This is modeled by calculating a minimum spanning tree on the graph using an initial (high) distance bound (eps). Then to query the dataset for the clusters that an *eps'* ≤ *eps* would give, one has only to remove edges from the MST of weight more than *eps'* and the remaining connected components will give the clusters. The idea of our POPTICS algorithm is as follows. Each processor computes a MST on its local dataset without incurring any communication. We then merge the local MSTs to obtain a global MST. Both steps are performed in parallel. Additionally, we extract the clusters directly from the global MST (without a linear order of the points) for any clustering distance, *eps',* by simply traversing the edges of the MST once in an arbitrary order, thus also enabling the cluster generation in parallel using the parallel disjoint-set data structure. POPTICS shows higher concurrency for data access while maintaining a comparable time complexity and quality with the classical OPTICS algorithm.

POPTICS is parallelized using both OpenMP and MPI to run on shared-memory machines and distributed-memory machines, respectively. The same Hopper machine was used for

experiments with MPI implementation. Our performance evaluation used a rich set of high dimensional data consisting of instances from real-world and synthetic datasets containing up to a billion floating point numbers. The speedups obtained on a shared-memory machine show scalable performance, achieving a speedup of up to 27.5 on 40 cores. Similar scalability results were observed for the MPI implementation on a distributed-memory machine with a speedup of 3,008 using 4,096 processors (Figure 3). In our experiments, we found that while achieving the scalability, POPTICS produces clustering results with comparable quality to the classical OPTICS algorithm. This work was published in Supercomputing conference 2013 [SC2013].

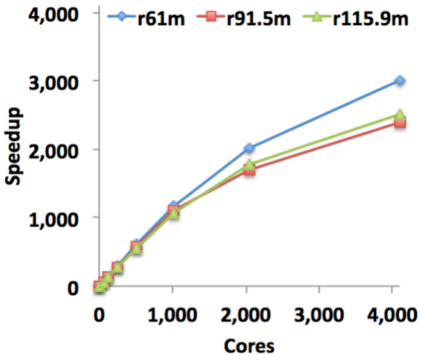

Figure 3: Empirical results for running parallel OPTICS.

4. Biological Sequence Analysis: Millions of DNA sequences (reads) are being generated every day by Next Generation Sequencing machines. In order to fulfill the dream of personalized medicine, high performance algorithms for mapping these sequences to a reference genome are needed to identify single nucleotide polymorphisms or rare transcripts. We have developed a high-throughput parallel sequence-mapping program named pFANGS. pFANGS is designed to find all the matches of a query sequence in the reference genome tolerating a large number of mismatches or insertions/deletions. Our design strategy for pFANGS is to partition the workload to reduce inter-node communication costs for the construction of the global hash table, while repeating some local computations. Using up to 512 processes, we are able to map approximately 31 million 454/Roche queries of length 500 to a reference human genome each hour, allowing up to 5 mismatches. We have produced both OpenMP and MPI-OpenMP hybrid implementations [HiPC11].

We also developed another bioinformatics kernel that estimates the pairwise statistical significance (PSSE) for local sequence alignment. As the algorithm is both data- and compute-intensive, we implemented it on GPUs to accelerate the calculation. After carefully studying the algorithm's data access characteristics, we adopted a tile-based scheme that can produce contiguous data accesses in the GPU global memory and sustain a large number of threads to achieve high GPU occupancy. We also extended our implementation to take advantage of dual GPUs. We observe end-to-end speedups of nearly 250 (370) times using (dual) Tesla C2050 GPU(s) over the CPU implementation on an Intel Core i7 CPU 920 processor [BMC12] (see Figure 4).

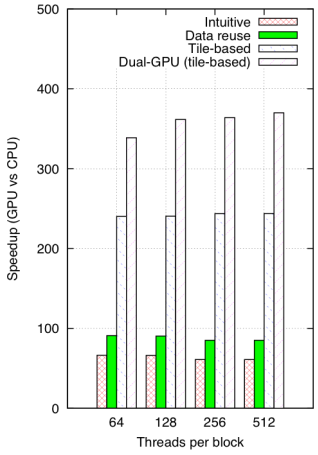

Figure 4: Performance of three strategies for multi-pair PSSE.

5. Query-driven In situ Data Analytics: The data generation process of space-time simulation proceeds from one time step to the next and requires the context of only two time steps, while storing data for only one time step on the disk. In contrast, visual data analytics often requires the full context of the available data, not just a single time step. In fact, simulations that are driven by local space-time relationships are largely performed with the purpose of discovering or explaining non-local and large-scale space-time relationships through interactive "what-if" data exploration. Thus, the fundamental differences in data context and heterogeneity of access

patterns demand for analytics-driven data management solutions. This necessitates making data analytics and data reduction the first class citizens of the data management design and information query processing. To support analytics-driven efficient query processing, ISABELLA code offers a transformative shift from the traditional indexing of data to the indexing of information about data compression and hierarchical data layout in storage. Compared to state-of-the-art scientific data management systems (FastBit, SciDB, MonetDB), ISABELLA offers upto 10-fold data storage reduction and more than 10-fold speed-up of per-core processing, and scalable multi-node, multi-core, and multi-GPU performance. Because of its light-weight storage footprint and embarrassingly execution model, ISABELLA's query engine offers an 8-fold improvement in energy efficiency compared to state-of-the-art technologies (Figure 5). ISABELLA enables in-situ, query-driven data analytics over compressed data. It supports both precision- and multi-resolution level of detail (LoD).

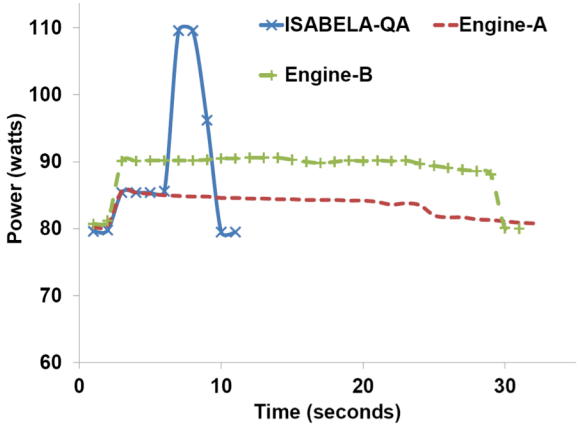

Figure 5: Comparison of energy consumption of ISABELA's query analytics (QA) engine with the state-of-art open source query engines.

6. Index-based Data Analysis: In this project, our work on this topic concentrates on two tasks: FastQuery software development and algorithm for detecting atmospheric rivers.

FastQuery software is an extension of FastBit to provide integration with common scientific file format libraries such as netCDF and pNetCDF. It enables the use to directly use the indexing and searching capability with out converting data to the specific format used by FastBit indexing software. In addition, FastQuery software also provides automatic parallelization of indexing and query answering operations, which further improve the response time of user data analysis [CHA+11, CWP11].

An illustration of the high-level architecture of FastQuery is showing in Figure 6. Under the XAnalytics project, the key tasks are to implement functions for handling netCDF files and pNetCDF files [CWP11].

Extreme precipitation events on the western coast of North America are often traced to an unusual weather phenomenon known as atmospheric rivers (ARs). These events refer to filamentary structures in atmosphere that transport significant amounts of water over a long distance in narrow bands. In one of the earliest studies on this phenomenon, it was determined that such a structure could carry more water than the great river

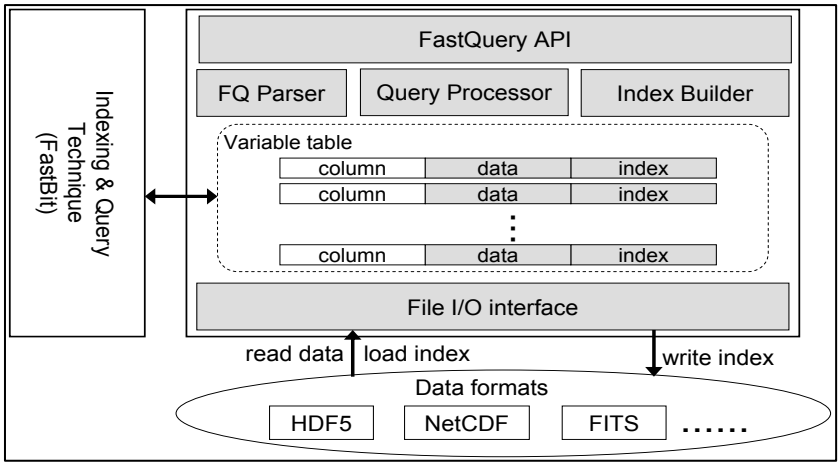

Figure 6: Organization of FastQuery components.

Amazon. Figure 7 shows an example of an atmospheric river that deposited record amounts of rainfall on California over the course of several days in December 2010. For regions such as the west coast of the United States, atmospheric rivers bring more than half of the annual total precipitation and can occur in as few as five days. Their intensity creates a possibility of flooding and wind damage, yet at the same time they provide a significant amount of the fresh water needed.

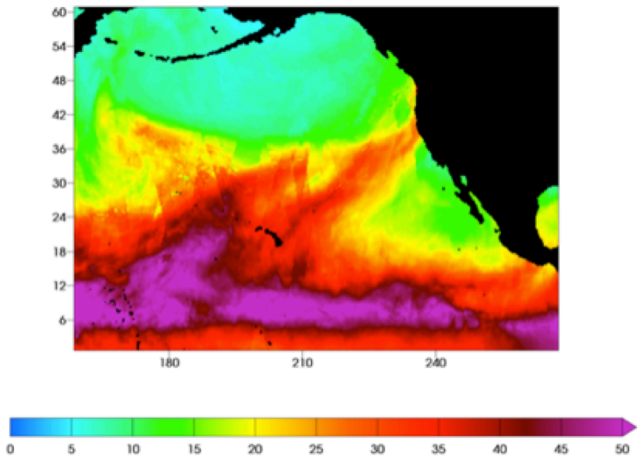

Figure 7: A sample of observed water vapor content in atmosphere. The high concentration of water reaching west coast United States is an Atmospheric River event in December 2010 (Data source rss.com).

Based on earlier study of Atmospheric Rivers, our algorithm works with a 2-D mesh defined over the globe. The key detection step is to find regions of mesh where the integrated water vapor (IWV) density is high. After the mesh points are identified, we group the points into connected regions on the globe. After which, we can measure the length and width of the regions. Those connected regions sufficiently long and narrow are declared as Atmospheric Rivers. In this work, we make use of an efficient connected component labeling algorithm developed earlier to produce an effective algorithm that runs quickly on massively parallel machines [BPW+11]. In a test run, we were able to process a 30-year model output in 3 seconds with 10,000 CPU cores.

7. High Performance Data Mining Using R on Heterogeneous Platforms: The exponential increase in the generation and collection of data has led us in a new era of data analysis and information extraction. Conventional systems based on general-purpose processors are unable to keep pace with the heavy computational requirements of data mining techniques. High performance coprocessors like GPUs and FPGAs have the potential to handle large computational workloads. As part of this project, we developed a scalable framework (Figure 8) aimed at providing a platform for developing and using high performance data mining applications on heterogeneous platforms. The framework incorporates a software infrastructure and a library of high performance kernels. Furthermore, it includes a variety of optimizations which increase the throughput of applications. The framework spans multiple technologies including R, GPUs, multi-core CPUs, MPI, and parallel-netCDF harnessing their capabilities for high-performance computations. This work also introduces the concept of interleaving GPU kernels from multiple applications providing significant performance gain. Thus, in comparison to other tools available for data mining, our framework

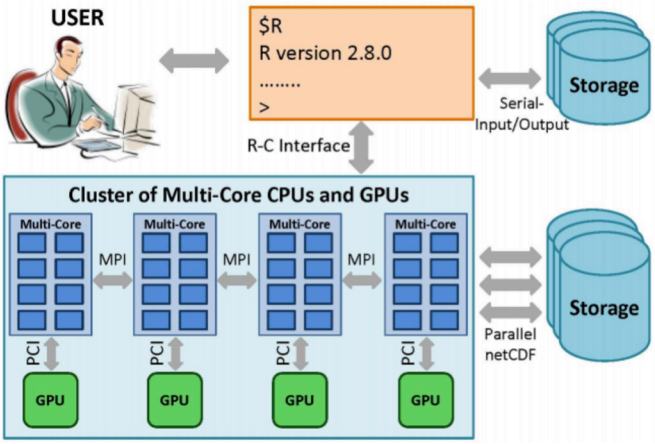

Figure 8: Overview of the framework

provides an easy-to-use and scalable environment both for application development and execution. The framework is available as a software package that can be easily integrated in the R programming environment. This work was published in an IPDPS workshop in 2011 [IPDPS2011].

Project Webpage: http://cucis.ece.northwestern.edu/projects/XANALYTICS/

Software Tools and Libraries

- Parallel Data Clustering Algorithms – parallel SLINK, DBSCAN, and OPTICS, along with source code and data (http://cucis.ece.northwestern.edu/projects/Clustering/index.html)
- Fast Max-Clique Finder (http://cucis.ece.northwestern.edu/projects/MAXCLIQUE/index.html)
- Sequence Mapping Software (http://cucis.ece.northwestern.edu/projects/NGS/agile.html)
- Accelerated pairwise statistical significance estimation using MPI/OpenMP/GPUs (http://cucis.ece.northwestern.edu/projects/PSSE/index.html)
- Network Instance-Based Biased Subgraph Search (NIBBS) is capable of comparing hundreds of genome-scale metabolic networks to identify metabolic subsystems that are statistically biased toward phenotype-expressing organisms (http://freescience.org/cs/NIBBS/).
- The (alpha,beta)-motif finder allows for identification of protein functional modules that, in addition to metabolic subsystems, could include their regulators, sensors, transporters, and even uncharacterized proteins that are predicted to be related to the target phenotype (http://freescience.org/cs/ABClique/).
- MCE-parallel is a scalable, parallel algorithm for the NP-hard clique enumeration problem (http://freescience.org/cs/parallelclique/).
- Bi-clustering approach allows for identification of phenotype-related modules that are analyzed to identify potential phenotype-related cross-talking pathways (http://freescience.org/cs/bi-clustering/).
- Dense ENriched Subgraph Enumeration (DENSE) algorithm allows for incorporating partial prior knowledge about the proteins involved in a phenotype-related process and enriches that knowledge with newly identified sets of functionally associated proteins present in individual phenotype expressing organisms (http://freescience.org/cs/DENSE/).
- System Phenotype-related Interplaying Components Enumerator (SPICE) iteratively enumerates statistically significant and phenotype-relevant cellular subsystems and can be applied to both network data and gene-expression data (http://freescience.org/cs/SPICE).
- Metabolic pathway alignment method for evolutionary analysis of metabolic pathways via multiple metabolic pathway alignment (software available upon request).
- Protein network alignment algorithm for evolutionary analysis of protein functional association networks via alignment of multiple protein functional association networks (software available upon request).
- Hierarchical Modularity Score (HMS) for functional annotation and functional enrichment analysis of hierarchically organized a complementary method to analyze birotein functional modules by taking into account an inherent design principle of biological networks, hierarchical modularity (http://freescience.org/cs/HMS/).
- ISABELA-QA is a memory and storage light-weight parallel query processing engine over ISABELA compressed scientific data capable of multi-core, multi-node, GPU executions (software available upon request).
- ALACRITY: Fast and memory light-weight query processing (and lossless compression) engine for scientific floating point data that is optimized for heterogeneous access pattern (software available upon request).

- Prm_causality a methodology for data-driven, semi-automatic inference of plausible phenomenological models (http://freescience.org/cs/prm_causality/).
- Forecast Error Detection and Correction (DETECTOR), a hierarchical algorithm for detecting and correcting prediction errors in extreme event forecasts (software available upon request).
- Anomalous Community Generator, detects predictive and phase-biased communities in contrasting groups of networks (software available upon request).
- Forecast Oriented Feature Elimination-based Classification of Adverse Spatio-Temporal Extremes (FORECASTER), constructs a forecast-oriented feature elimination-based ensemble of classifiers for robust forecasting of adverse spatio-temporal extremes (software available upon request).
- FastQuery, https://codeforge.lbl.gov/projects/fastquery/

Presentations:

1. Alok Choudhary, "Big Data + Big Compute = An Extreme Scale Marriage for Smarter Science?" Plenary Talk at the Supercomputing Conference, Nov 21, 2013.
2. Alok Choudhary, "Discovering Knowledge from Massive Networks and Science Data - Next Frontier for HPC", keynote at the Department of Energy Computational Science Graduate Fellowship Annual Conference, July 26, 2012.
3. Alok Choudhary, "Discovering Knowledge from Massive Social Networks and Science Data – Next Frontier for HPC", Keynote in the 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, May 2012.
4. Alok Choudhary, Wei-keng Liao, Saba Sehrish, Seung Son, Quincey Koziol, Ben Clifford, Rob Ross, Rob Latham, Tim Tautges, Nagiza Samatova, Drew Bayouka, Sriram Lakshminarasimhan, Xiaocheng Zou, and Zhenhuan Gomg, “Damsel: A Data Model Storage Library for Exascale Science”, Poster and a short technical paper at the DoE ASCR meeting, April 2012.
5. Alok Choudhary, "Discovering Knowledge from Massive Social Networks and Science Data - Next Frontier for HPC", keynote at the International Conference on High Performance Computing, Bangalore, India December 2011.

Publications:

1. M. Patwary, D. Palsetia, A. Agrawal, W.-k. Liao, F. Manne, and A. Choudhary, “A new scalable parallel DBSCAN algorithm using the disjoint-set data structure,” in ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC), 2012, pp. 1–11. Article No. 62. [SC2012]
2. W. Hendrix, D. Palsetia, M. M. A. Patwary, A. Agrawal, W.-k. Liao, and A. Choudhary, “A Scalable Algorithm for Single-Linkage Hierarchical Clustering on Distributed Memory Architectures,” in Proceedings of 3rd IEEE Symposium on Large-Scale Data Analysis and Visualization (LDAV 2013), Atlanta GA, USA, Oct. 2013, 2013, pp. 7–13. [LDAV13]
3. M. Patwary, D. Palsetia, A. Agrawal, W.-k. Liao, F. Manne, and A. Choudhary, “Scalable Parallel OPTICS Data Clustering Using Graph Algorithmic Techniques,” in Proceedings of 25th International Conference on High Performance Computing, Networking, Storage and Analysis (Supercomputing, SC’13), 2013, pp. 1–12. Article No. 49. [SC2013]
4. Yuhong Zhang, Md. Mostofa Ali Patwary, Sanchit Misra, Ankit Agrawal, Wei-keng Liao, and Alok Choudhary. Enhancing Parallelism of Pairwise Statistical Significance Estimation for Local Sequence Alignment. In *the Workshop on Hybrid Multicore Computing, held in*

conjunction with the IEEE International Conference on High Performance Computing, December 2011 [HiPC11].

5. Yuhong Zhang, Sanchit Misra, Ankit Agrawal, Md. Mostofa Ali Patwary, Wei-keng Liao, and Alok Choudhary, "*Accelerating pairwise statistical significance estimation for local alignment by harvesting GPU's power*", BMC Bioinformatics, Volume 13 (Suppl 5):S3, 2012. [BMC12]
6. Lakshminarasimhan, S., Kumar, P., Liao, W., Choudhary, A., Kumar, V., and Samatova, N.F., ``On the Path to Sustainable, Scalable, and Energy-efficient Data Analytics: Challenges, Promises, and Future Directions.'' 2012 International Green Computing Conference (IGCC), San Jose, CA, U. S. A, June 2012. [IGCC'12]
7. John Jenkins, Isha Arkatkar, Sriram Lakshminarasimhan, Neil Shah, Eric R. Schendel, Stephane Ethier, CS Chang, Jackie Chen, Hemanth Kolla, Scott Klasky, Robert Ross, and Nagiza F. Samatova. "Analytics-driven Lossless Data Compression for Rapid In-situ Indexing, Storing, and Querying". Proceedings of the 23rd International Conference on Database and Expert Systems Applications (DEXA). September 2012. [DEXA'12]
8. Lakshminarasimhan,S., Jenkins, J., Arkatkar, I., Gong, Z., Kolla, H., Ku, S-H., Ethier, S., Chen, J., Chang, C.S., Klasky,S., Latham, R., Ross, R., and Samatova, N.F., ``ISABELA-QA:Query-driven Analytics with ISABELA-compressed Extreme-Scale ScientificData.'' Supercomputing (SC 2011), Novemer 12 - 18, 2011, Seattle, Washington. [SC'11]
9. Wilson, K., Rocha, A., Padmanabhan, K., Wang, K., Chen, Z., Jin, Y., Mihelcic, J.R., and Samatova, N.F., ``Detecting pathway cross-talks by analyzing conserved functional modules across multiple phenotype-expressing organisms.'' 2011 IEEE International Conference on Bioinformatics and Biomedicine (BIBM 2011), 12-15 Nov 2011, Atlanta, GA. [BIBM'11]
10. Pendse, S.V., Tetteh, I.K., Semazzi, F., Kumar, V., and Samatova, N.F., ``Data-driven, semi-automatic inference of phenomenological physical models: Application to Eastern Sahel rainfall.'' The 2012 SIAM International Conference on Data Data Mining, April 26-28, 2012, Anaheim, CA, US. [SIAM'12]
11. Pansombut, T., Hendrix, W., Gao, Z., Harrison, B.E., and Samatova, N. F., ``Biclustering-Driven Ensemble of Bayesian Belief Network Classifiers for Underdetermined Problems.'' Proc. of the 22nd Int'l Joint Conference on Artificial Intelligence (IJCAI 2011), July 16-22, 2011, Barcelona, Spain. [IJCA'11a]
12. Sencan, H., Chen, Z., Hendrix, W., Pansombut, T., Semazzi, F., Choudhary, A., Kumar, V., Melechko, A. V., and Samatova, N. F., ``Classification of Emerging Extreme Event Tracks in Multi-Variate Spatio-Temporal Physical Systems Using Dynamic Network Structures: Application to Hurricane Track Prediction.'' Proc. of the 22nd Int'l Joint Conference on Artificial Intelligence (IJCAI 2011), July 16-22, 2011, Barcelona, Spain. [IJCA'11b]
13. Padmanabhan, K., Wilson, K., Rocha, A., Mihelcic, J.R., Samatova, N.F., ``Detecting pathway cross-talks by analyzing conserved functional modules across multiple phenotype-expressing organisms.'' Proteome Science, 2012 (Accepted). [PS'12]
14. Schmidt, M.C., Rocha, A.M., Padmanabhan, K., Shpanskaya, Y., Banfield, J., Scott, K., Mihelcic, J.R., and Samatova, N.F. ``NIBBS-search for fast and accurate prediction of phenotype-biased metabolic systems.'' PLoS Computational Biology, 2012 (DOI: 10.1371/journal.pcbi.1002490). [PCB'12]
15. Chen, Z., Padmanabhan, K., Rocha, A.M., Shpanskaya, Y., Mihelcic, J.R., and Samatova, N.F., ``SPICE: Discovery of phenotype-determining component interplays.'' BMC Systems Biology, 2012, 6(1):40 doi:10.1186/1752-0509-6-40. [BMC'12]
16. Chen, Z., Hendrix, W., and Samatova, N.F., ``Community-based anomaly detection in evolutionary networks.'' Journal of Intelligent Information Systems: Integrating Artificial Intelligence and Database Technologies, 2011 (DOI: 10.1007/s10844-011-0183-2). [JIIS'11]
17. Schmidt, M.C., Rocha, A.M., Padmanabhan, K., Chen, Z., Scott, K., Mihelcic, J.R., and Samatova, N.F., ``Efficient alpha,beta-motif finder for identification of phenotype-related functional modules.'' BMC Bioinformatics, 2011, 12:440. [BMCB'12]

18. Hendrix, W., Rocha, A.M., Padmanabhan, K., Choudhary, A., Scott, K., Mihelcic, J.R., and Samatova, N.F., ``DENSE: efficient and prior knowledge-driven discovery of phenotype-associated protein functional modules.'' BMC Systems Biology, 2011, 5:172. [BMC'12b]
19. Surendra Byna, Prabhat, Michael F. Wehner, and Kesheng Wu. Detecting atmospheric rivers in large climate datasets. In *Proceedings of the 2nd international workshop on Petascal data analytics: challenges and opportunities* (PDAC '11). 2011. ACM, New York, NY, USA, 7-14. DOI=10.1145/2110205.2110208. [BPW+11]
20. Jerry Chou, Mark Howison, Brian Austin, Kesheng Wu, Ji Qiang, E. Wes Bethel, Arie Shoshani, Oliver Rübel, Prabhat, and Rob D. Ryne. Parallel index and query for large scale data analysis. In *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis* (SC '11). 2011. ACM, New York, NY, USA, Article 30. DOI=10.1145/2063384.2063424 [CHA+11]
21. Jerry Chou, Kesheng Wu, and Prabhat. FastQuery: A general indexing and querying system for scientific data. In SSDBM, pages 573–574, 2011. http://dx.doi.org/ 10.1007/978-3-642-22351-8_42. [CWP11]
22. Prabhat Kumar, Berkin Ozisikyilmaz, Wei-keng Liao, Gokhan Memik, and Alok Choudhary. High Performance Data Mining Using R on Heterogeneous Platforms. In Workshop on Multithreaded Architectures and Applications, in conjunction with the International Parallel and Distributed Processing Symposium, May 2011, pp. 1715-1724 [IPDPS2011].