Clustering Jets at the Exascale

Steven Ko, Salvatore Rappoccio, Lukasz Ziarek

Overview : In the past, particle physics has relied upon improvements in processing speed of single computing cores in order to improve data acquisition rates. This feature will be critical to the proposed "High Luminosity Large Hadron Collider" (HL-LHC) and other colliders, where the computational challenges move from the petascale to the exascale. Unfortunately, since the mid-2000's, this improvement in single-core processing speed has hit a limit, and improvements in processing time must now come from concurrent processing. However, the current reconstruction techniques are not enabled for concurrent processing. In order to continue improving the processing capabilities of particle physics experiments in the HL-LHC era, it will be necessary to explore cutting-edge techniques in parallel processing.

There are several general classes of problems in particle physics event reconstruction that could be modified in order to achieve concurrent processing. One such opportunity that has not yet been explored is in "jet clustering," a nearest-neighbor type of algorithm used to cluster hadronically-fragmented jets into a single object.

Intellectual Merit : This proposal focuses on parallelizing the existing jet clustering algorithms in use at the LHC experiments. The proposed improvements will be to use this as a test case for deployment of cutting-edge parallelization techniques such as lightweight concurrency extraction, speculative computing, and smarter distribution. Some recent experience shows that the nearest-neighbor type of algorithm used by the jet clustering is amenable to such improvements.

Broader Impacts : The benefits of this proposal are twofold : firstly, there will be an immediate improvement of the jet clustering algorithms themselves that will lead to higher data acquisition rates at the LHC. Secondly, the computing techniques developed could be used in other applications, inside of particle physics and elsewhere. Since nearest-neighbor algorithms are ubiquitous in scientific computing, it is expected that techniques developed to parallelize this particular problem will be applicable to a wide variety of others in academia and industry.

In addition, these core developments can train students in the newest computing techniques, giving them cutting-edge experience that is highly relevant in academia and private industry.

Clustering Jets at the Exascale

Steven Ko, Salvatore Rappoccio, Lukasz Ziarek

1 Introduction

The research proposed here will focus on deploying advanced techniques in parallelization to improve algorithms involved in particle physics research at the Large Hadron Collider (LHC) and beyond. As a specific test case, this research focuses on immediate improvements to one particular algorithm, jet clustering, although the principles developed could be deployed in other areas of particle physics reconstruction and elsewhere.

With the discovery of the Higgs boson by the Large Hadron Collider (LHC) experiments ATLAS and CMS [1, 2], the standard model (SM) of particle physics is now complete. This model unifies the electromagnetic force (carried by the *photon*) with the weak force, responsible for radioactive decay (carried by the *W and Z bosons*). At long last, physicists now understand that via interactions with the Higgs field, the W and Z bosons acquire a mass, but the photon does not. This is referred to as "electroweak symmetry breaking".

A new phase of particle physics has therefore begun. The questions have shifted from the cause of electroweak symmetry breaking, to the study of the Higgs boson and its interactions in detail. To understand the larger picture of the fundamental forces in nature, it will be imperative that high-luminosity colliders be built to study the Higgs in detail, requiring exascale computing tools (in speed and throughput) to reach the goals.

One of the major technical challenges that lies ahead in exascale-level computing for these high-luminosity colliders is the continuation of the scaling of computational power year by year, known colloquially as "Moore's Law". To set the scale, at the CMS experiment with the LHC collision flux ("luminosity") reaching $7 \times 10^{33}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$, the processing time to reconstruct each collision event by CMS was approximately 20 seconds per event. However, as the luminosity is increased, the computational time currently scales quadratically. The luminosity at the HL-LHC is expected to reach as high as $> 1 \times 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$, which would correspond (naively) to enormous processing times, on the order of many minutes to several hours per event! Clearly, it is necessary for the computing power to scale in order to compensate for this dramatic increase in CPU time with increasing luminosity.

Unfortunately, the expected end of the historic scaling of single-core processing capability [3] can adversely affect the long-term processing capability for particle physics experiments. Without improvements in single-core computational speed, the only avenues left are algorithmic improvements, and parallelization. Algorithm development is being undertaken at the LHC experiments to reduce the computational time of the software for data acquisition and reconstruction, using both single-core and multi-core approaches. However, long-term improvements will require fundamental improvements in processing capabilities using parallelization.

The currently proposed exploratory research will deploy new and innovative computer science techniques to particle physics. One algorithm that is particularly amenable to improvement using these techniques is the clustering of final-state particles produced in collisions into groups called "jets". Jets are produced via quantum chromodynamic (QCD) interactions of quarks and gluons that hadronize. These "jet clustering" algorithms group the particles into jets based on nearest-neighbor clustering (NN). This algorithm can become computationally expensive as the number of particles that are produced in a collision grows, scaling as $N^2 \log N$ or $N \log N$. Techniques that

leverage and combine speculative computing, lightweight concurrency, and smarter distribution can strongly impact these types of NN algorithms [4, 5, 6, 7, 8]. Thus, this research will serve as a real-world test case for the deployment of these algorithms, with both immediate and long-term benefits.

2 Prior Work

2.1 Results from Prior Support

2.1.1 PI Rappoccio

PI Rappoccio does not have NSF support yet as he has recently started his faculty career.

2.1.2 Co-PI Ziarek

Co-PI Ziarek does not have NSF support yet as he has recently started his faculty career.

2.1.3 Co-PI Ko

Co-PI Ko has been awarded "CI-ADDO-NEW: PhoneLab: A Programmable Participatory Smartphone Testbed" on 06/01/12. The award number is CNS-1205656 and the amount is $1,358,510.00. The duration is for three years. The results include the following.

Results Related to Intellectual Merit: the PhoneLab team has distributed 300 smartphones to the faculty, students, and staff of UB who are using the smartphones as their primary phone. The team has developed the testbed infrastructure where they can monitor the usage of each phone. The team has also developed a front-end website where experimenters and participants can use for various testbed-related functions. The team has opened the PhoneLab testbed for public experimentation on 10/31/13. There are two public experiments running on the testbed and 7 experiments are waiting for approval.

Results Related to Broader Impact: the PhoneLab team is in the process of recruiting 4 undergraduate students as part of their outreach program. Two undergraduate students have been working with the team for one month now. The team is actively interviewing undergraduate students to hire. In addition, the team is in talks with Buffalo Academy of Science Charter School in order to explore the possibility of establishing a program where the team teaches high school students with smartphone programming skills.

Publications: the PhoneLab team has published one paper so far, "PhoneLab: A Large Programmable Smartphone Testbed" in the First International Workshop on Sensing and Big Data Mining, 2013.

Evidence of Research Products: the PhoneLab testbed is currently in use; there are roughly 300 participants and several experiments are either running or waiting for approval. Experimenters can submit their experiments through the website: `http://www.phone-lab.org`.

2.2 Overview of Activities

The investigators of this proposal have a widely-varied and applicable skill set to accomplish the goals of extending LHC computing to the exascale.

Salvatore Rappoccio (tenure-track Assistant Professor) joined the Faculty at the University at Buffalo, SUNY (UB) in 2012. He has 15 years of experience programming in a high-energy physics environment, as well as other numerical software design for the private sector.

Since 2007, Rappoccio has been a member of the Compact Muon Solenoid (CMS) experiment at CERN. From 2008-2010, he was the co-leader of the Analysis Tools group of the CMS Software Project. He was responsible for deployment of jet reconstruction algorithms as well as other tools for data analysis of LHC collision data during the startup phase of the LHC. His primary responsibilities included managing deployment of highly-performing software (including visualization) tools in a team consisting of $\sim$15 people. He has been instrumental in improving the single-core computational speed for jet clustering at CMS since 2008.

His research interests include utilizing jet clustering algorithms in new and innovative ways to search for signals of physics beyond the standard model of particle physics. His pioneering efforts resulted in the first measurements and searches for new phenomena with advanced jet clustering algorithms at CMS, outlined in Refs. [9, 10, 11]. These techniques have become hugely popular in the LHC experiments and in the theoretical community. Prof. Rappoccio is a leader in the jet reconstruction community, contributing to and editing seminal reports on the subject in Refs. [12, 13, 14]. In addition to this academic work, he has also been involved in numerical software design for MIT's Lincoln Laboratories (the details of which are classified).

Currently, Rappoccio is mentoring two students (one graduate student, Jaba Chelidze, and one undergraduate student, Jonathan Goodrum) in achieving parallelization of the jet clustering algorithms. The latter is working under the "Collegiate Science and Technology Entry Program" (CSTEP) [15]. As described by the program, CSTEP exists "to support talented underrepresented students pursuing science, technology, engineering and mathematics".

Lukasz Ziarek has 9 years of experience in language, compiler, and runtime design targeted at improving multicore performance. He has worked on 5 compilers and 3 Java VMs. He is an expert at speculative and transactional computation focusing on the extraction of parallelism and lightweight concurrency.

Steven Ko has 10 years of experience in distributed systems. His recent focus has been large-scale data processing in the cloud using MapReduce and other technologies built on top of it. He also has 5 years of experience in large-scale storage and data management in data centers.

3 Proposed Program of Research

The "jet clustering" technique is employed by many different particle physics experiments and theorists worldwide, and is implemented in a common software framework called `fastjet` [16]. The mathematical problem is analogous to the "K-nearest neighbors algorithm" [17] (kNN).

The specific jet-clustering algorithms that have become enormously popular in the particle physics community are based on sequential clustering, similar to nearest-neighbor clustering [18]. In these sequential-clustering algorithms, a list of the four-momentum of particles are input. These algorithms combine four-vectors of input pairs of particles until certain criteria are satisfied and jets are formed. For the jet algorithms considered in this paper, for each pair of particles i and j, a "distance" metric between the two particles (d_{ij}), and the so-called "beam distance" for each particle (d_{iB}), are computed:

$$d_{ij} = \min({p_{\mathrm{T}}}_i^{2n}, {p_{\mathrm{T}}}_j^{2n})\Delta R_{ij}^2/R^2 \quad (1)$$

$$d_{iB} = {p_{\mathrm{T}}}_i^{2n}, \quad (2)$$

where $p_{\mathrm{T}i}$ and $p_{\mathrm{T}j}$ are the transverse momenta of particles i and j, respectively, "min" refers to the lesser of the two p_{T} values, the integer n depends on the specific jet algorithm, $\Delta R_{ij} = \sqrt{(\Delta y_{ij})^2 + (\Delta\phi_{ij})^2}$ is the distance between i and j in rapidity ($y = \frac{1}{2}\ln(E + p_z)/(E - p_z)$) and azimuth ($\phi$), and R is the "size" parameter of order unity [19], with all angles expressed in radians. The particle pair (i, j) with smallest d_{ij} is combined into a single object. All distances are recalculated using the new object, and the procedure is repeated until, for a given object i, all the d_{ij} are greater than d_{iB}. Object i is then classified as a jet and not considered further in the algorithm. The process is repeated until all input particles are clustered into jets.

The value for n in Eqs. (1) and (2) governs the topological properties of the jets. For $n = 1$ the procedure is referred to as the k_{T} algorithm (KT). The KT jets tend to have irregular shapes and are especially useful for reconstructing jets of lower momentum [19]. For $n = -1$, the procedure is called the anti-k_{T} (AK) algorithm, with features close to an idealized cone algorithm. The AK algorithm is used extensively in LHC experiments and by the theoretical community for finding well-separated jets [19]. For $n = 0$, the procedure is called the Cambridge–Aachen (CA) algorithm. This relies only on angular information, and, like the k_{T} algorithm, provides irregularly-shaped jets in (y, ϕ). The CA algorithm is useful in identifying jet substructure [20, 21].

The single-core optimization of jet clustering is outlined in Ref. [18]. In a single core, the computational time scales as $O(N^2)$ or $O(N \ln N)$, where N is the number of inputs to the algorithm, which scales linearly with luminosity. Since the luminosity of future colliders is expected to drastically increase over existing machines, it will be critical to develop parallelization strategies to maintain scalability of the jet clustering algorithms that exist to future machines.

Since the particles are clustered pairwise, there are numerous opportunities to separately compute portions of the event and then combine them at later stages. We now discuss specific strategies that can be developed to optimize concurrent performance in this algorithm.

3.1 Technical Challenges

To achieve the necessary improvements in performance required for scalability of jet clustering, we propose to examine parallelization opportunities across the entire software stack, including three specific areas : (1) the use of lightweight concurrency extraction to mask high-latency computations or I/O actions, (2) extraction of parallelization from the computation itself in the form of optimistic speculation and specialized transform, and (3) new methods for distributing the computation to maximize parallelization on each node.

Lightweight Concurrency for Latency Masking

Many mathematical kernels contain opportunities for extracting "micro parallelism," usually on the order of tens of instructions, from their computational components. Unfortunately, it is very difficult to parallelize this computation profitably as the overhead of thread creation, scheduling, synchronization, and migration outweigh the gains in parallelism. Instead of extracting explicit parallelism from such computations, we propose to explore methods of lightweight asynchrony to allow for computation to proceed while waiting on high latency I/O operations to complete or the results of other computations. Since the creation of threads and associated schedule and synchronization costs are typically prohibitive, we will explore new threading models that allow for

logically-distinct computations to execute within a given construct. The PIs previous research has indicated that such schemes can profitably boost overall performance in the context of ML code [22, 23]. The salient research challenges in applying this strategy are as follows: 1) identifying what computation can be executed safely during high latency operations at compile time, 2) providing a lightweight threading runtime and programming model in the context of an imperative language, 3) specializing the approach to numeric kernels, and 4) building support for computation in a distributed setting.

Speculative Computation

In addition to exploring explicit parallelization of the numeric kernels in jet clustering, we propose to explore extraction of parallelism via speculative computation. At its core, speculative computation breaks apart sequential or parallel tasks into smaller tasks to be run in parallel. Once the speculation has completed, the runtime system validates the computation. If the computation is incorrect (*i.e.* a "datarace" is detected, the computation cannot be serialized, *etc.*), the incorrect computation is re-executed in a non-speculative manner. If the rate of mis-speculation is low, such techniques can be leveraged to extract additional parallelism. There have been many different proposals, including large efforts on transactional memory [], lock elision [], thread level speculation and speculative multithreading [], for integrating speculative computation into programming languages and their associated runtimes []. The PIs have extensive experience with transactional memory [24], lightweight rollback methods [25], leveraging memoization to reduce re-computation costs [26, 27], and deterministic speculation [28]. We propose to explore a specialized speculation framework leveraging different speculation strategies, including speculation extracted by the programmer via programming language primitives, library level speculation, and compiler extracted speculation. The salient research challenges in applying this strategy are as follows: 1) identification the appropriate speculation model and discovering speculation points at compile time, 2) providing a speculative runtime specialized for jet clustering and capable of realizing user, library, and compiler injected speculation, and 3) exploring new and specialized lightweight validation and re-execution mechanisms, including validation across multiple speculation strategies.

Smart Distribution

In order to increase parallelism, we will explore the use of the MapReduce execution framework [29, 30]. MapReduce is a runtime system recently developed for large-scale parallel data processing. It enables programmers to easily deploy their applications on a cluster of machines. Programmers only need to write two functions, Map and Reduce, and submit these two functions as a job to the system. Then the MapReduce framework takes care of all the aspects of the execution of the job. For example, the framework packages and distributes the two functions over the cluster so that the whole cluster can be utilized to execute the job; it also takes care of fault-tolerance by monitoring the cluster during the execution of the job and redistributes the job if some machine fails.

Due to this simplicity and power, it is quickly gaining popularity in industry for large-scale data processing. Many applications in scientific computing have not yet explored the use of MapReduce in depth, however previous research has explored implementing similar kNN-style algorithms with MapReduce [7, 8]. We intend to explore this question in the context of jet clustering for the LHC.

3.2 Research Plan

Preliminary studies of naive parallelization of a few specific algorithms in the `fastjet` package reduced processing times by factors of 2-3 by moving to concurrent processing with 8 cores, with no

increase in memory cost. By further deploying the advanced strategies described above, this factor of 2-3 improvement is expected to grow significantly. These preliminary studies were performed by Rappoccio and Mr. Goodrum (undergrad student) over the summer of 2013. Having a postdoctoral fellow to work on this project will greatly increase the likelihood of success.

The postdoctoral fellow requested in this proposal is envisioned to have a computer science background. There will be two graduate students, one with a physics background (but with strong computational skills), and one with a computer science background. Furthermore, it is expected that Mr. Goodrum will continue to work with the group under his CSTEP Fellowship. This team, under the guidance of Rappoccio, Ziarek, and Ko, will perform the implementation and study of these algorithms in the real-world `fastjet` software environment, testing the improvements at the extensive cluster at the **Center for Computational Research** here at UB.

During the first year, it is expected that naive improvements using established techniques could be deployed into the `fastjet` package. Deployment of the improvements achieved by Mr. Goodrum and Rappoccio should take several months to a year of full deployment time.

In the meantime, (**timeline from CS improvements goes here.**)

4 Broader impacts

The broader impacts of this research are many fold. The jet-clustering algorithm is very similar to the "kNN" algorithm, which is widely applicable throughout research and industry. The improvements that are developed here at the cutting edge of scientific inquiry will possibly be adaptable to other real-life applications.

In addition to the core physics developments enabled by improved scalability of the `fastjet` software environment, this award will also serve as a validation mechanism for core computer science research in programming languages and distributed systems. The development of the system will require core systems research in expanding and applying the technological advances made previously by the PIs.

5 Outreach and Education

While it is critical to pursue a rigorous research program, a large part of the responsibility of scientists is to educate the next generation effectively. There is already extensive work being done to educate high school-level students and teachers about particle physics via the *QuarkNet* program at UB, however there is very little in the way of educating the general public. In addition to participating in the existing *QuarkNet* activities, the plan outlined in this proposal will extend the coverage of the outreach program at UB to engage the broader public in discussions of major results in particle physics, as well as to enliven particle physics for young students. This will be implemented based on similar events as the "HiggsFest" [31] that Prof. Rappoccio organized at UB.

5.1 Higgsfest and other public events

The "Higgsfest" that was organized here at UB in 2012 is highlighted in Ref. [31]. The aim of the event was to invite the general public for "plain English" summaries and hands-on demonstrations that were geared for a multitude of age and knowledge levels. This was attended by over 100 people, including children, high-school students, physics and non-physics undergraduates, and interested members of the community.

Some of the hands-on demonstrations included building models of Feynman diagrams from craft material (for young children), a fully-functional four-layer coincidental muon scintillator detector, a cloud chamber made out of tupperware, felt, and dry ice, and the actual Higgs events from the CMS collaboration in an interactive event display. The event was covered by the "UB Reporter" here at UB [32].

Two more such events are proposed, the first to coincide with the LHC turn-on sometime in 2015, and the second to coincide with the newest results from the LHC after data-taking commences. These are events that should generate high media coverage, and will be a good opportunity to capitalize on public interest in this field. In the event of a major new discovery at the LHC during Run 2, the public interest will be very high, so having the experience of what works and what does not work in such events is extremely valuable to maximize the public impact.

In addition, this removes the stigma associated with science and technology fields at an early age. When young children can attend an event with their parents and take something away from it, this shows them that science is an integral part of life, and nothing to be particularly nervous about pursuing. It may even convince younger people to pursue a scientific career.

One of the major points learned during the last "Higgsfest" is that it is often difficult to have economically-disadvantaged students attend the lectures because of a lack of transportation possibilities. This is something to rectify for future projects along these lines. Therefore, in addition to holding the event directly at the UB North Campus (which is difficult for inner-city Buffalo schools to reach), a duplicated event is also proposed closer to the inner city that is easier to attend, or possibly to visit these schools directly. Some possible locations are the UB South Campus, or at "Babeville" [33], where the UB Physics Department routinely organizes the "Science and Art Cabaret" [34]. Both locations provide the infrastructure needed for the event, and access for disadvantaged schools and students in the inner city of Buffalo.

5.2 Undergraduate research and Diversity

As discussed above, Mr. Goodrum is participating in the "Collegiate Science and Technology Entry Program" (CSTEP) here at UB while working on this project. This program focuses on disadvantaged or minority students who would otherwise have a difficult time pursuing STEM-related fields. Prof. Rappoccio is a strong believer in the principles and practice of this program, and intends to continue this work in the future. There are also numerous opportunities for Independent Study and Honors' Theses for undergraduates in the Physics department.

Co-PI Ko is actively involved in mentoring undergraduates. Co-PI Ko is involved in several formal programs such as the McNair Program and UB Honors College Program. Through these programs, two undergraduate students, Edward Poon and Mitchell Nguyen, are currently conducting research with him. Edward Poon, a junior, participates in the McNair Program, which is "designed to provide encouragement and services to low-income and first generation college students, and increase participation from underrepresented groups in pursuing doctoral study." [1] Co-PI Ko is currently listed as a McNair mentor. The other student, Mitchell Nguyen, is a freshman Honors College student. Mitch was involved in the development of PhoneLab and hopes to continue research involvement throughout his college career.

Co-PI Ko also has a track record of recruiting women students. Co-PI Ko has successfully recruited two female PhD students, Sonali Batra and Anudipa Maiti. Co-PIs Ko Ziarek have also been working with a female Master's student, Namita Vishnubhotla, and have successfully recruited her as a PhD student; she will start as a PhD student from Spring, 2014. All PIs will actively continue to recruit female and minority students in their research programs.

[1] `http://cads.buffalo.edu/mcnair`

5.3 Summary

In summary, the problem of expanding LHC computing to the exascale is a difficult, but tractable one. This proposal investigates the possibility of applying cutting-edge parallelization techniques such as lightweight concurrency extraction, speculative computation, and smarter distribution, to the real-world application of LHC data processing. The overall goal is to reduce the computational time for k-nearest-neighbor-like numerical kernels used for jet clustering. The investigators of this proposal have extensive experience in the various aspects of the problem, and the synergistic application of this experience is expected to attain considerable improvements in this area, which are absolutely critical to the success of the future LHC physics program.

References

[1] Serguei Chatrchyan et al. Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC. *Phys. Lett. B*, 2012.

[2] Georges Aad et al. Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC. *Phys. Lett. B*, 2012.

[3] Samuel H. Fuller and Lynette I. Millett. *The Future of Computing Performance: Game Over or Next Level?* The National Academies Press, 2011. Committee on Sustaining Growth in Computing Performance; National Research Council.

[4] Vincent Garcia, Eric Debreuve, Frank Nielsen, and Michel Barlaud. k-nearest neighbor search: fast GPU-based implementations and application to high-dimensional feature matching. In *IEEE International Conference on Image Processing (ICIP)*, Hong Kong, China, September 2010.

[5] V. Garcia, E. Debreuve, and M. Barlaud. Fast k nearest neighbor search using gpu. In *CVPR Workshop on Computer Vision on GPU*, Anchorage, Alaska, USA, June 2008.

[6] Vincent Garcia. *Suivi d'objets d'intrt dans une squence d'images : des points saillants aux mesures statistiques.* PhD thesis, Universit de Nice - Sophia Antipolis, Sophia Antipolis, France, December 2008.

[7] Wei Lu, Yanyan Shen, Su Chen, and Beng Chin Ooi. Efficient processing of k nearest neighbor joins using mapreduce. *Proc. VLDB Endow.*, 5(10):1016–1027, June 2012.

[8] Chi Zhang, Feifei Li, and Jeffrey Jestes. Efficient parallel knn joins for large data in mapreduce. In *Proceedings of the 15th International Conference on Extending Database Technology*, EDBT '12, pages 38–49, New York, NY, USA, 2012. ACM.

[9] Serguei Chatrchyan et al. Search for anomalous t t-bar production in the highly-boosted all-hadronic final state. *JHEP*, 1209:029, 2012.

[10] Serguei Chatrchyan et al. Search for heavy resonances in the W/Z-tagged dijet mass spectrum in pp collisions at 7 TeV. *Phys.Lett.*, B723:280–301, 2013.

[11] Serguei Chatrchyan et al. Studies of jet mass in dijet and W/Z+jet events. *JHEP*, 1305:090, 2013.

[12] A. Abdesselam, E. Bergeaas Kuutmann, U. Bitenc, G. Brooijmans, J. Butterworth, et al. Boosted objects: A Probe of beyond the Standard Model physics. *Eur. Phys. J. C*, 71:1661, 2011.

[13] A. Altheimer, S. Arora, L. Asquith, G. Brooijmans, J. Butterworth, et al. Jet Substructure at the Tevatron and LHC: New results, new tools, new benchmarks. *J. Phys. G*, 39:063001, 2012.

[14] A. Altheimer, A. Arce, L. Asquith, J. Backus Mayes, E. Bergeaas Kuutmann, et al. Boosted objects and jet substructure at the LHC. 2013.

[15] SUNY at Buffalo. Collegiate science and technology entry program (cstep). `http://cpmc.buffalo.edu/cstep/`.

[16] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. FastJet User Manual. *Eur.Phys.J.*, C72:1896, 2012.

[17] T. Cover and P. Hart. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27, 1967.

[18] Matteo Cacciari and Gavin P. Salam. Dispelling the N^3 myth for the k_t jet-finder. *Phys.Lett.*, B641:57–61, 2006.

[19] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. The Anti-k_{T} jet clustering algorithm. *JHEP*, 04:063, 2008.

[20] Yuri L. Dokshitzer, G. D. Leder, S. Moretti, and B. R. Webber. Better Jet Clustering Algorithms. *JHEP*, 08:001, 1997.

[21] M. Wobisch and T. Wengler. Hadronization corrections to jet cross sections in deep- inelastic scattering. 1998.

[22] Lukasz Ziarek, KC Sivaramakrishnan, and Suresh Jagannathan. Composable asynchronous events. In *ACM SIGPLAN Notices*, volume 46, pages 628–639. ACM, 2011.

[23] KC Sivaramakrishnan, Lukasz Ziarek, Raghavendra Prasad, and Suresh Jagannathan. Lightweight asynchrony using parasitic threads. In *Proceedings of the 5th ACM SIGPLAN workshop on Declarative aspects of multicore programming*, pages 63–72. ACM, 2010.

[24] Lukasz Ziarek, Adam Welc, Ali-Reza Adl-Tabatabai, Vijay Menon, Tatiana Shpeisman, and Suresh Jagannathan. A uniform transactional execution environment for java. *ECOOP 2008–Object-Oriented Programming*, pages 129–154, 2008.

[25] Lukasz Ziarek and Suresh Jagannathan. Lightweight checkpointing for concurrent ml. *Journal of Functional Programming*, 20(02):137–173, 2010.

[26] Lukasz Ziarek and Suresh Jagannathan. Memoizing multi-threaded transactions. *Workshop on Declarative Aspects of Multicore Programming*, 2008.

[27] Lukasz Ziarek, KC Sivaramakrishnan, and Suresh Jagannathan. Partial memoization of concurrency and communication. *ACM Sigplan Notices*, 44(9):161–172, 2009.

[28] Lukasz Ziarek, Siddharth Tiwary, and Suresh Jagannathan. Isolating determinism in multi-threaded programs. *Runtime Verification*, pages 63–77, 2012.

[29] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. In *Proceedings of the 6th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2004.

[30] Hadoop MapReduce. `http://hadoop.apache.org/mapreduce`.

[31] SUNY at Buffalo Physics Department. Higgsfest. `http://www.physics.buffalo.edu/HiggsFest`.

[32] Charlotte Hsu. Higgsfest celebrates physics discovery. `http://www.buffalo.edu/ubreporter/archive/2012_11_29/higgs_fest.html`.

[33] Righteous Babe Records. The ninth ward at babeville. `http://www.babevillebuffalo.com`.

[34] Babeville and SUNY at Buffalo. Science and art cabaret. http://www.hallwalls.org/science-art.php.

[35] Gnu general public license, version 2. http://www.gnu.org/licenses/gpl-2.0.html, June 2007. Last retrieved 2012-05-10.

BUDGET JUSTIFICATION

Institution : The State University of New York at Buffalo (UB)
PI : Salvatore Rappoccio
Co-I : Lukasz Ziarek, Steven Ko

Personnel

The requested funds of $951,102 USD (for 3 years starting in 2014) would cover two (2) months of summer salary for Profs. Rappoccio, Ziarek and Ko per year for three (3) years ($172,062), the full salary for one (1) postdoctoral fellow for three (3) years ($154,224), and salary plus tuition for two (2) graduate students, totaling $91,812 in salary and $33,432 in tuition. This will also cover $22,300 of computer fees.

If awarded, efforts for Profs. Rappoccio, Ziarek and Ko will be reduced to be in compliance with NSF 2 month policy.

Fringe

Fringe benefit rates are based on the applicable federally negotiated rates published at http://www.research.buffalo.edu/sps/about/rates.cfm.

Travel

This research will require regular travel to conferences and CERN. Hence, the proposal requests $12,486 in domestic travel funds and $12,486 in foreign travel funds for the three years of activity.

Facilities and Administration Indirect Costs

Indirect cost rates are based on the applicable federally negotiated rates published at http://www.research.buffalo.edu/sps/about/rates.cfm

Facilities, Equipment and Other Resources

Center For Computational Research

UB has a large computational research center, CCR, which is a Linux-based cluster on the Open Science Grid, and also has a large GPU cluster for possible parallel processing developments.

Office

The faculty, postdoctoral fellows and graduate students all have office space at UB.

Postdoctoral Fellow Mentoring

One postdoctoral fellow will be funded on this project. There are extensive postdoctoral fellowship mentoring activities at UB, as well as via the CMS Experiment at CERN. These include guidance in career paths, work/life balance discussions, and technical skill development such as writing grant proposals, etc. Specific elements are highlighted below.

- **University at Buffalo (UB)**
  - The UB Office of Postdoctoral Scholars offers diverse services for postdoctoral fellows, including the *"Postdoc Survival Skills Workshops"*, targeted seminars and symposia for postdoctoral fellows, social functions, and logistical assistance.
  - The UB Physics Department offers several services to our postdoctoral fellows, including a biweekly Journal Club for particle physics and cosmology, weekly seminars and colloquia, and weekly social functions inside the department.
- **CERN**
  - The opportunities for a postdoctoral fellow at CERN are extensive. There are also a plethora of workshops, seminars, conferences, etc, at CERN. There are also smaller weekly avenues for networking possibilities, as well as seminars for postdoctoral fellows to gain visibility for their work.
  - It is also worth pointing out that, because of the world-class nature of CERN, it often attracts very high-level members of the particle physics community on a regular basis. Such opportunities for visibility among the top-tier scientists in the world (including Nobel and Milner Prize winners, etc) are hard to understate.

In all, the postdoctoral fellow that will be supported by this proposal will have ample opportunities for professional advancement and development, as well as a myriad of opportunities for a community of peers in both professional and social settings.

Data Management Plan

The data that are produced from this proposal will be in the form of software algorithms and procedures. The format and content will be as source files.

The algorithms that are developed with this research plan will be integrated into the main `fastjet` software framework for distribution among collaborators (both experimental and theoretical) worldwide [2]. This software is open-source, freely available, and continually maintained by the `fastjet` maintenance team under the GNU Public License V2 [35]. The web pages related to this project are stored on machines at the "Laboratoire de Physique Thorique et Hautes Energies" `http://www.lpthe.jussieu.fr/spip/?lang=en`.

The `fastjet` team has indicated their willingness to integrate improvements into the main `fastjet` package.

Should authors use the `fastjet` package (including possible improvements as a result of this grant), they should cite [18, 16] and any publications that follow from the proposed research.

[2] http://fastjet.fr