Fall 2015

Big Data Proteogenomics and High Performance Computing: Challenges and Opportunities

Fahad Saeed
Western Michigan University, fahad.saeed@wmich.edu

Big Data Proteogenomics and High Performance Computing: Challenges and Opportunities

Fahad Saeed
Department of Electrical and Computer Engineering
Department of Computer Science
Western Michigan University
Kalamazoo, MI 49008–5466
Email: fahad.saeed@wmich.edu

***Abstract*—Proteogenomics is an emerging field of systems biology research at the intersection of proteomics and genomics. Two high-throughput technologies, Mass Spectrometry (MS) for proteomics and Next Generation Sequencing (NGS) machines for genomics are required to conduct proteogenomics studies. Independently both MS and NGS technologies are inflicted with data deluge which creates problems of storage, transfer, analysis and visualization. Integrating these big data sets (NGS+MS) for proteogenomics studies compounds all of the associated computational problems. Existing sequential algorithms for these proteogenomics datasets analysis are inadequate for big data and high performance computing (HPC) solutions are almost non-existent. The purpose of this paper is to introduce the big data problem of proteogenomics and the associated challenges in analyzing, storing and transferring these data sets. Further, opportunities for high performance computing research community are identified and possible future directions are discussed.**

I. Introduction

Proteogenomics is a new and emerging field of biological research which is at the intersection of proteomics and genomics. Proteogenomics has a wide range of applications of crucial importance such as environmental microbiology [1], bacteriology and virology [2], gene annotation [3], human neurology [4], cancer-biology [5] and countering bio-terrorism [6]. Data from two high-throughput technologies need to be combined and integrated for proteogenomics studies: Mass Spectrometry (MS) data (billions of spectra [7]) for proteomics and Next Generation Sequencing (NGS) data (billions of short DNA reads) for genomics [8], [9]. Independently both of these technologies are inflicted with data deluge which creates problems of storage [10], transfer [11], analysis [12] and visualization [13]. Integrating these big peta-byte level data sets for proteogenomics studies compounds all of the associated computational problems. Tools that can analyze these datasets in a reasonable amount of time are almost non-existent and do not scale well (in time, memory or resources) with large eukaryotic genomes [14], [15].

The state of the art is even bleaker for HPC algorithms for proteogenomics. At the time of writing this paper, the only known HPC algorithm for proteogenomics exploits embarrassingly parallel techniques on large clusters [16]. We are not aware of any fine-grained parallelism approaches using ubiquotous architectures such as graphical processing units (GPU's) and Intel Phi's. Consequently, only the most resourceful experimental labs have been able to do large-scale proteogenomics studies using high-performance techniques. Most of the previous proteogenomics studies have been accomplished using serial versions of the scarce tools without a comprehensive framework for analysis [9]. Limited number of available tools with most of them exhibiting poor scalability and the enormous volume of the proteogenomics data is the primary motivator for the need of algorithms that can exploit ubiquitous high-performance architectures.

In this paper we will identify analytic, storage and transmission problems associated with these complex data sets. We will discuss our progress in developing high-performance solutions to storage, analysis and transmission of these proteogenomics data sets. We will further illustrate the possible course that can be taken by the parallel processing research community to solve these big data problems that are likely to have a broad impact.

II. Background Information

The understanding of the gene has evolved over time from the classical definition (Mendel's work) being "unit of heredity" to "..a union of genomic sequences encoding a coherent set of potentially overlapping functional products" [17], especially after the ENCODE project. In other words the definition of "gene that translates one protein which functions" have evolved into becoming "a set of genes from different fragments of the genomes that are translated into proteins that function". One metaphor popular in describing genes in computational terms is to think of them in terms of various subroutines in a big operating system (OS) [17]. The nucleotides of the genomes are grouped together into a code that is executed through the process of transcription and translation; the genome in this case can be thought of as OS for the living organism. Genes are the individual subroutines that are repeatedly called in the process of transcription. The new ENCODE project gives a different perspective to this and does not fit with the metaphor that a gene is a simple callable routine for the OS. In the new perspective one can enter into a gene subroutine in many different ways and functions. One can still understand the current view if one considers gene transcriptions in terms of parallel threads of execution where the threads do no follow modular subroutines but instead a poorly constructed computer program code with many GOTO statements coming in and out of loops. This new view of genes and proteins have implications for both the genomic and proteomics research.

Traditionally the role of genomics and proteomics communities have been defined. Genomics community was suppose to identify genes and the corresponding proteins. Proteomics communities were more concerned with the function of the proteins and their expression under different conditions, tissues and cells. Since the definitions of the gene itself is not clear, the proteomics and genomics community must work together to elucidate gene structures and corresponding proteins [9] [17]. This gives rise to the field of proteogenomics. The most effective and high-throughput tools for studying genomics and proteomics are next generation sequencing machines (NGS) [18] and mass spectrometers (MS) [19], respectively. Proteogenomics requires integration and analysis of data from both of these high-throughput technologies. The combination of these two high-throughput data sets gives rise to big data proteogenomics.

The peptides identified by proteogenomics framework have unique information about the gene annotation such as confirming translation, excluding pseudo genes, determining frames for the gene and quality of the protein (not up for degradation), specifying the translation begin- and end- sites, prediction of novel genes and verification of splice variants in large scale studies [14] and distant evolutionary relationships [20]. Below we will list problems in each of the datatypes and the challenges that lie ahead to integrate the framework.

A. *Big NGS data and computational challenges*

New next generation sequencing (NGS) technologies are used for whole-genome/exosome sequencing, transcriptome profiling (RNA-Seq), DNA-protein interactions (ChIP-sequencing). These machines produce short fragments of DNA or RNA sequences called reads. The sheer volume of data from these machines (3 billion DNA/RNA reads and 0.6TB per run [21]) needs efficient and high-performance computational tools [22] [18]. In order to process the genomic data it is usually mapped to the reference genome. However, if one is looking at novel transcripts (usually in the case of proteogenomics studies) then those genomic regions would not be present in the reference genome to begin with!. Hence complex transcriptome reconstruction algorithms are required. Due to enormous volume of the data, transcriptome assembly is complex and requires a lot of computational time and resources e.g. only 10's of GB of data can take days to compute a transcriptome assembly [23] and can easily reach peta-byte level [24]. These NGS datasets have the inherent problems of storage and transmission due to their large volume and velocity.

B. *Big Mass Spectrometry data and computational challenges*

A typical mass spectrometer produces hundreds of thousands of complex stochastic spectra within hours which are a combination of mass-to-charge (m/z) ratio and intensity of the peaks and require complex algorithms for further processing [25] [26]. There are two parts of MS data that leads to computational bottlenecks: First is the volume of the data which can easily reach peta-byte level for millions of spectra (Thermo Orbitrap Fusion, SWATH-type data generation) and processing of billions of spectra will take unreasonably long amount of time. The other is the number of peaks per spectra that needs processing e.g. each spectra on average has 4000 peaks [27] and for 60k human proteins the distinct peaks that need to be compared is 240 million. This number is just for a single human proteome and multiple datasets are required for proteogenomics studies. The usual first step after production of the MS data is to search the raw spectra against a *protein database* to deduce peptides. The peptide identification is error-prone due to high noise, high dimensionality, and compounded isometry. The algorithms used [28] to search the spectra against a known database is one of the single most time consuming steps in the MS data analysis e.g. 1.5 days to search 26172 MS2 spectra (233MB mzXML file) and 33MB protein database file [29]. Processing millions of spectra will take impractically long time.

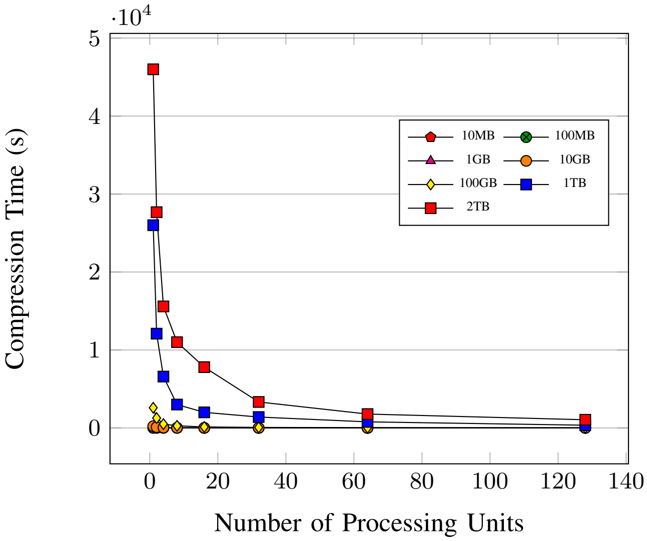

Fig. 1. Compression time (s) vs. number of processing units for paraDSRC [8]

III.Proteogenomics data and computational challenges

In case of proteogenomics datasets the problem is compounded since, apart from millions of spectra, the databases against which search is done are also very large (especially for mammalian species) e.g. human genome has six billion residues in it and can lead to 600-fold increase in the size of the proteome database [14]. In contrast the current protein Uniprot database only needs 180MB for 250 organisms [30]. A major challenge for spectra-to-peptide match algorithms is that they are designed for small number of spectra which are to be matched to a small database which leads to poorly scaling matching algorithms. Further, high-confidence matches obtained using spectra-to-peptide match algorithms is close to 30% and this shortcoming is well documented [26]. Therefore the conventional techniques used for proteomics will prove to be useless due to poor scalability and lack of sensitivity.

At the very *least* a proteogenomics experiment has one genomic NGS data set which has to be reconstructed, one big proteome database created using six-frame conversion from genomic data and, one big mass spectrometry data sets. Generally at least two data sets are required to complete an experiment (1 control and 1 vehicle) and biologists perform a single experiment 3 times to confirm their observations. This makes 6 sets of NGS+MS data sets for a single experimental result.

In order to take full advantage of integrating two high-throughput technologies one would need a plethora of computational tools. Broadly speaking the integration and analysis presents two main challenges: 1) The enormous amount of

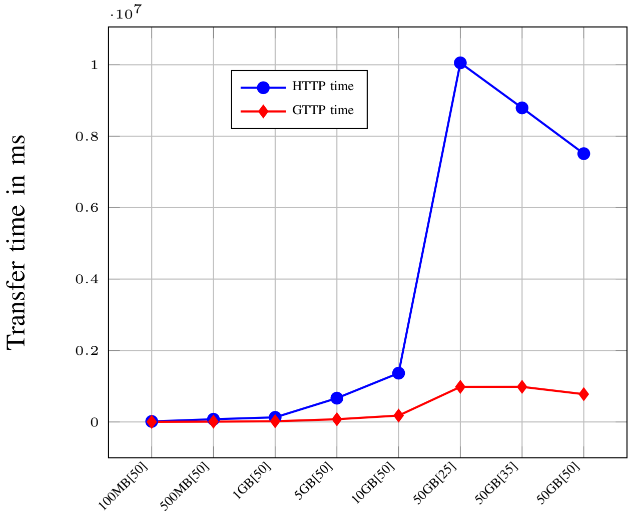

Fig. 2. Transfer time of simulated genomic datasets using both HTTP and Genomic transfer Protocol (GTTP) [35]

data from both NGS (10 TB/run and billions of short reads) and mass spectrometry (TB's/run with millions of spectra more complex than NGS data) machines both in terms of speed and volume 2) The current algorithms are inefficient as they are not designed for big data and lack sensitivity/accuracy [9]. Therefore, from a HPC point of view the problem is both compute- and data-intensive. The large volume of data also creates storage and transmission issues. Below we will discuss storage, transmission and analysis of these data sets in detail and our progress in all of these areas.

A. HPC storage solutions for big data proteogenomics

Massive amounts of data from proteogenomics data sets requires that one has efficient storage solutions. For the current discussion we will assume an NGS data sets which can be easily extended for MS data sets. General propose storage solutions are not very efficient and specialized compression algorithms have been proposed [31]–[34]. However, these specialized compression algorithms suffer from poor scalability and HPC solutions are required [8].

To this end, we presented a parallelization strategy using distributed-memory architecture for compression of big NGS datasets [8]. To our knowledge, this is the first attempt to investigate domain decomposition strategy implemented on a memory-distributed architecture for compression of big Next Generation Sequencing datasets. The proposed strategy allowed us to devise a highly scalable parallel algorithm, which exhibited linear-speedups for most datasets and gave a minimalist communication footprint. The results with increasing number of processing units is shown in Fig. 1. The next steps of this research would be studying the scalability of the proposed parallelization technique on larger clusters. We will also focus on HPC solutions for decompression of data.
Future Research Directions: The proposed storage solution should have at least the following properties: 1) compresses large amount of data in a short time using HPC architectures 2) decompress the data in an efficient manner 3) is specific to NGS/MS data sets and has excellent compression ratio 4) allow analysis on the compressed-form of the data without the need to decompression. Ultimately we will like to reach the forth kind of compression solutions for both NGS and MS data sets.

B. HPC transmission solutions for big data proteogenomics

Transmission of large datasets is accomplished using protocols such as FTP and HTTP. However, these protocols are designed for general purpose data transfer. Only two *data-oblivious* protocols are reported in literature for efficient transmission of NGS data [36], [37]. Both methods use FTP/HTTP protocols and multiple machines to increase throughput. We are not aware of any data-aware protocol for big genomic data.
Future Research Directions: We assert that if the data is known a-priori (as in the case of proteogenomics) then we should be able to make the protocols more data-aware. Again for the sake of discussion we will assume that we are dealing with NGS data sets but the arguments are extendible for MS data sets. To this end, we have presented a data-aware HTTP-based protocol which allows efficient encoding of DNA nucleotides using limited number of bits [35]. This efficient encoding then leads to faster transmission of the data using same bandwidth and traffic congestions. Transmission timing results are shown in Fig. 2. We are currently working on developing data-aware protocols which can encode data in an efficient way and can dynamically change the encoding scheme depending on the data being transmitted. Such a system, if successful, will allow us to transfer data on the fly using reconfigurable hardware such as FPGA's.

C. HPC analytic for big data proteogenomics

For the analytic part we will only discuss the peptide deduction problem using large databases in this paper. In case of protoegenomics studies, the database size increases many folds due to six-frame translation of genome into proteome [9] which makes the current serial version of the tools not scalable. HPC solutions have been proposed [38], [38]–[42] but most of them report incremental speedups with increasing number of processing units. These HPC algorithms assume a compute intensive parallel computing model and do not consider the data-intensive aspect. Hence most of them exhibit poor scalability with increasing number of processing units. Probably the most serious drawback is that the results from these parallel algorithms are different than the results of the serial version of the same algorithm [41]. Absence of architecture-aware algorithms and non-existent techniques to deal with big data (such as sampling, sketching or streaming) severely degrades the usefulness of these parallel systems which consequently leads to sub-optimal speedups.
Future Research Directions: Big data is general is inflicted with the problem of spurious correlations. For HPC solutions to be useful and scalable for large volume of proteogenomics data, we will have to come up with useful similarity metrics that does not suffer from spurious correlations [9]. To this end, we have recently presented a novel metric, called F-set, for comparison of similarity of spectra [43]. F-set metric is based on the observation that consecutive peaks in succession are much more accurate metric than single peak metrics. The probability that consecutive sets of peaks would be common between spectra that are not related is very small and it has the potential to be a better similarity metric than individual peak counting techniques. Although F-set has been introduced as a metric to cluster spectra, it can be used as a scoring scheme

for spectra-to-peptide match (SPM) algorithms [26]. A F-set metric will be a good starting point for creating metrics for SPM algorithms for proteogenomics since large number of sequences in the database leads to less sensitive matchings. Once we are able to deduce the peptides from proteogenomics data in a more confident way, we can move towards creating HPC solutions to SPM algorithms.

For future research, both compute-intensive and data-intensive nature of the problem should be kept into view. Assuming that the spectra are emitted at a high speed from the mass spectrometers. The proposed HPC algorithm should be able to sketch the spectra. This sketch is a rough estimate of the spectra which can be used in the future for deductions. The sketch serves two purposes: It allows us to get a rough estimate of the incoming spectra. It further reduces the number of comparisons that have to be made to the big database. Hence such an algorithm would be scalable for large number of spectra and/or databases. Another direction that we would like to take is introduction of data-reduction for MS data. The basic idea is that most of the peaks in the MS data are not useful for deduction of the peptide. Therefore, if one can eliminate most of the peaks using a low computational cost method, it would have tremendous effect on the computational capability of existing tools. To this end, we have presented our first preliminary study of using random sampling to eliminate large chunk of peaks from MS2 spectra while keeping the integrity of the data i.e. even after reducing the data by 50% we were able to deduce the correct spectra using standard search tools [7]. Although random sampling is low-complexity procedure and offers peptide deduction for a wide range of spectra; more accurate data-reduction methods will be investigated to improve the accuracy. As an initial point of investigation we will perform empirical studies using regular sampling, stratified sampling and accidental sampling. Further, MS data-reduction at higher-dimensions would be investigated. Note that the objective of data-reduction is not *only* to eliminate peaks that are noisy but also eliminate peaks that do not contribute to peptide deduction but are a computational burden in the spectra without any gainful conclusions.

IV. Discussion and Conclusions

Proteogenomics is a new and emerging area in systems biology and have the potential for transforming medicine with the introduction of personal genomics and proteomics in predictive and precision treatment for humans. However, the amount of data and the time it takes to analyze these data sets for useful information is a serious bottleneck for scientists as well as for clinical diagnostics. If useful research has to proceed, there must be plethora of efficient computational tools necessary to analyze these big data sets. Proteogenomics studies requires the scientists to generate and integrate data from two high-throughput technologies namely, next generation sequencing (NGS) technologies, and mass spectrometers (MS). This integration of peta-bytes of data requires high-performance computing solutions for analysis, storage as well as transfer of these data sets. In this paper we have identified three broad areas where high performance computing can have a seminal effect in the area of big data proteogenomics.

One way to deal with big data is to compress these data sets which will lead to saving in compute time, I/O and bandwidth. High performance solution that can take advantage of ubiquitous architectures such as multicore and GPU's are essential for quick compression of the data. Current general purpose compression tools that require decompression consume valuable time and resources for large data sets. Therefore, frameworks which allow scientists to access the data without decompression is vital for useful and efficient analysis of both NGS and MS data sets. The research will enable us to compress and compute on data sets which will save resource, time and I/O bandwidth required for cloud infrastructures used by most of the system biologists for NGS/MS data analysis.

Another area which is crucial for proteogenomics data is the transmission of these data sets. The current general purpose protocols used for transmission are not specific to NGS or MS data sets. In order to transfer data over communication networks, data-aware transmission protocols will be needed. These transmission protocols must be able to encode the omics data in a more efficient manner and hence must exhibit high-performance transmission of data. Such transmission protocols will be of immense advantage to systems biologists and clinicians.

High performance analytics is essential part of any big data problem. For the analysis of proteogenomics data sets HPC algorithms would be required. However, as is being witnessed, tools developed using conventional parallel computing models are likely to fail due to large volume of the data. The new generation of HPC algorithms should be designed keeping in mind both the data-intensive and compute-intensive nature of the proteogenomics problems. Therefore, novel sampling, sketching & streaming, data and dimensionality reduction techniques will be required along with efficient design of these techniques that can run on HPC architectures. Techniques that can exploit ubiquitous architectures such as multicore, manycore and graphical processing units (GPU's) would be most beneficial to system biology and clinical labs who cannot host large clusters.

Acknowledgements

This work was supported by grant NSF CCF-1464268.

References

[1] J. Armengaud, E. Marie Hartmann, and C. Bland, "Proteogenomics for environmental microbiology," *Proteomics*, vol. 13, no. 18-19, pp. 2731–2742, 2013.

[2] H. Sun, C. Chen, B. Lian, M. Zhang, X. Wang, B. Zhang, Y. Li, P. Yang, and L. Xie, "Identification of hpv integration and gene mutation in hela cell line by integrated analysis of rna-seq and ms/ms data," *Journal of proteome research*, vol. 14, no. 4, pp. 1678–1686, 2015.

[3] N. E. Castellana, Z. Shen, Y. He, J. W. Walley, S. P. Briggs, V. Bafna, *et al.*, "An automated proteogenomic method uses mass spectrometry to reveal novel genes in zea mays," *Molecular & Cellular Proteomics*, vol. 13, no. 1, pp. 157–167, 2014.

[4] M.-G. Kang, K. Byun, J. H. Kim, N. H. Park, H. Heinsen, R. Ravid, H. W. Steinbusch, B. Lee, and Y. M. Park, "Proteogenomics of the human hippocampus: The road ahead," *Biochimica et Biophysica Acta (BBA)-Proteins and Proteomics*, 2015.

[5] J. A. Alfaro, A. Sinha, T. Kislinger, and P. C. Boutros, "Onco-proteogenomics: cancer proteomics joins forces with genomics," *Nature methods*, vol. 11, no. 11, pp. 1107–1113, 2014.

[6] E. M. Hartmann and J. Armengaud, "Proteogenomics for the enhanced discovery of bacterial biomarkers," in *Detection of Chemical, Biological, Radiological and Nuclear Agents for the Prevention of Terrorism*, pp. 169–177, Springer, 2014.

[7] M. G. Awan and F. Saeed, "On the sampling of big mass spectrometry data," in *Bioinformatics and Computational Biology (BICoB) Conference*, ISCA, 2015.

[8] S. V. Perez and F. Saeed, "A parallel algorithm for compression of big next-generation sequencing (ngs) datasets," in *Proceedings of IEEE International Symposium on Parallel and Distributed Processing with Applications (IEEE ISPA-15)*, IEEE, August 2015.

[9] A. I. Nesvizhskii, "Proteogenomics: concepts, applications and computational strategies," *Nature methods*, vol. 11, no. 11, pp. 1114–1125, 2014.

[10] Y. Zhang, L. Li, J. Xiao, Y. Yang, and Z. Zhu, "Fqzip: Lossless reference-based compression of next generation sequencing data in fastq format," in *Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems-Volume 2*, pp. 127–135, Springer, 2015.

[11] T. Kwon, W. G. Yoo, W.-J. Lee, W. Kim, and D.-W. Kim, "Next-generation sequencing data analysis on cloud computing," *Genes & Genomics*, pp. 1–13, 2015.

[12] B. Calabrese and M. Cannataro, "Cloud computing in healthcare and biomedicine," *Scalable Computing: Practice and Experience*, vol. 16, no. 1, 2015.

[13] Y. Zhang, R. Bhamber, I. Riba-Garcia, H. Liao, R. D. Unwin, and A. W. Dowsey, "Streaming visualisation of quantitative mass spectrometry data based on a novel raw signal decomposition method," *Proteomics*, vol. 15, no. 8, pp. 1419–1427, 2015.

[14] B. A. Risk, W. J. Spitzer, and M. C. Giddings, "Peppy: proteogenomic search software," *Journal of proteome research*, vol. 12, no. 6, pp. 3019–3025, 2013.

[15] S. H. Nagaraj, N. Waddell, A. K. Madugundu, S. Wood, A. Jones, R. A. Mandyam, K. Nones, J. V. Pearson, and S. M. Grimmond, "Pgtools: a software suite for proteogenomics data analysis and visualization," *Journal of proteome research*, 2015.

[16] A. Tovchigrechko, P. Venepally, and S. H. Payne, "Pgp: parallel prokaryotic proteogenomics pipeline for mpi clusters, high-throughput batch clusters and multicore workstations," *Bioinformatics*, p. btu051, 2014.

[17] M. B. Gerstein, C. Bruce, J. S. Rozowsky, D. Zheng, J. Du, J. O. Korbel, O. Emanuelsson, Z. D. Zhang, S. Weissman, and M. Snyder, "What is a gene, post-encode? history and updated definition," *Genome research*, vol. 17, no. 6, pp. 669–681, 2007.

[18] F. Saeed, A. Perez-Rathke, J. Gwarnicki, T. Berger-Wolf, and A. Khokhar, "A high performance multiple sequence alignment system for pyrosequencing reads from multiple reference genomes," *Journal of parallel and distributed computing*, vol. 72, no. 1, pp. 83–93, 2012.

[19] V. G. Keshamouni, G. Michailidis, C. S. Grasso, S. Anthwal, J. R. Strahler, A. Walker, D. A. Arenberg, R. C. Reddy, S. Akulapalli, V. J. Thannickal, *et al.*, "Differential protein expression profiling by itraq-2dlc-ms/ms of lung cancer cells undergoing epithelial-mesenchymal transition reveals a migratory/invasive phenotype," *Journal of proteome research*, vol. 5, no. 5, pp. 1143–1154, 2006.

[20] W. Gish, D. J. States, *et al.*, "Identification of protein coding regions by database similarity search," *Nature genetics*, vol. 3, no. 3, pp. 266–272, 1993.

[21] H. Buermans and J. den Dunnen, "Next generation sequencing technology: advances and applications," *Biochimica et Biophysica Acta (BBA)-Molecular Basis of Disease*, vol. 1842, no. 10, pp. 1932–1941, 2014.

[22] B. G. Jackson, M. Regennitter, X. Yang, P. S. Schnable, and S. Aluru, "Parallel de novo assembly of large genomes from high-throughput short reads," in *Parallel & Distributed Processing (IPDPS), 2010 IEEE International Symposium on*, pp. 1–10, IEEE, 2010.

[23] V. Sachdeva, C. Kim, K. Jordan, and M. Winn, "Parallelization of the trinity pipeline for de novo transcriptome assembly," in *Parallel & Distributed Processing Symposium Workshops (IPDPSW), 2014 IEEE International*, pp. 566–575, IEEE, 2014.

[24] G. Cochrane, R. Akhtar, J. Bonfield, L. Bower, F. Demiralp, N. Faruque, R. Gibson, G. Hoad, T. Hubbard, C. Hunter, *et al.*, "Petabyte-scale innovations at the european nucleotide archive," *Nucleic acids research*, vol. 37, no. suppl 1, pp. D19–D25, 2009.

[25] F. Saeed, T. Pisitkun, J. D. Hoffert, G. Wang, M. Gucek, and M. A. Knepper, "An efficient dynamic programming algorithm for phosphorylation site assignment of large-scale mass spectrometry data," in *Bioinformatics and Biomedicine Workshops (BIBMW), 2012 IEEE International Conference on*, pp. 618–625, IEEE, 2012.

[26] F. Saeed, J. Hoffert, and M. Knepper, "Cams-rs: clustering algorithm for large-scale mass spectrometry data using restricted search space and intelligent random sampling," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 11, no. 1, pp. 128–141, 2014.

[27] O. Vorst, C. De Vos, A. Lommen, R. Staps, R. Visser, R. Bino, and R. Hall, "A non-directed approach to the differential analysis of multiple lc–ms-derived metabolic profiles," *Metabolomics*, vol. 1, no. 2, pp. 169–180, 2005.

[28] J. K. Eng, B. Fischer, J. Grossmann, and M. J. Maccoss, "A Fast SEQUEST Cross Correlation Algorithm," *J. Proteome Res.*, September 2008.

[29] B. Pratt, J. J. Howbert, N. I. Tasman, and E. J. Nilsson, "Mr-tandem: parallel x! tandem using hadoop mapreduce on amazon web services," *Bioinformatics*, vol. 28, no. 1, pp. 136–137, 2012.

[30] U. Consortium *et al.*, "The universal protein resource (uniprot) in 2010," *Nucleic acids research*, vol. 38, no. suppl 1, pp. D142–D148, 2010.

[31] D. C. Jones, W. L. Ruzzo, X. Peng, and M. G. Katze, "Compression of next-generation sequencing reads aided by highly efficient de novo assembly," *Nucleic acids research*, p. gks754, 2012.

[32] W. Tembe, J. Lowey, and E. Suh, "G-sqz: compact encoding of genomic sequence and quality data," *Bioinformatics*, vol. 26, no. 17, pp. 2192–2194, 2010.

[33] S. Deorowicz and S. Grabowski, "Compression of dna sequence reads in fastq format," *Bioinformatics*, vol. 27, no. 6, pp. 860–862, 2011.

[34] E. Grassi, F. Di Gregorio, and I. Molineris, "Kungfq: A simple and powerful approach to compress fastq files," *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, vol. 9, pp. 1837–1842, Nov. 2012.

[35] M. Aledhari and F. Saeed, "Design and implementation of network transfer protocol for big genomic data," in *Proceedings of IEEE Big Data Congress (IEEE Big Data Congress 2015),*, IEEE, June 2015.

[36] C. Wilks, D. Maltbie, M. Diekhans, and D. Haussler, "Cghub: Kick-starting the worldwide genome web," *Proceedings of the Asia-Pacific Advanced Network*, vol. 35, pp. 1–13, 2013.

[37] J. Bresnahan, M. Link, G. Khanna, Z. Imani, R. Kettimuthu, and I. Foster, "Globus gridftp: what's new in 2007," in *Proceedings of the first international conference on Networks for grid applications*, p. 19, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007.

[38] D. T. Duncan, R. Craig, and A. J. Link, "Parallel tandem: A program for parallel processing of tandem mass spectra using pvm or mpi and x!tandem," *Journal of Proteome Research*, vol. 4, no. 5, pp. 1842–1847, 2005. PMID: 16212440.

[39] L. A. Baumgardner, A. K. Shanmugam, H. Lam, J. K. Eng, and D. B. Martin, "Fast parallel tandem mass spectral library searching using gpu hardware acceleration," *Journal of Proteome Research*, vol. 10, no. 6, pp. 2882–2888, 2011.

[40] B. Pratt, J. J. Howbert, N. I. Tasman, and E. J. Nilsson, "Mr-tandem: parallel x!tandem using hadoop mapreduce on amazon web services," *Bioinformatics*, vol. 28, no. 1, pp. 136–137, 2012.

[41] R. D. Bjornson, N. J. Carriero, C. Colangelo, M. Shifman, K.-H. Cheung, P. L. Miller, and K. Williams, "X!!tandem, an improved method for running x!tandem in parallel on collections of commodity computers," *Journal of Proteome Research*, vol. 7, no. 1, pp. 293–299, 2008.

[42] G. Kulkarni, A. Kalyanaraman, W. R. Cannon, and D. Baxter, "A scalable parallel approach for peptide identification from large-scale mass spectrometry data," in *Parallel Processing Workshops, 2009. ICPPW'09. International Conference on*, pp. 423–430, IEEE, 2009.

[43] F. Saeed, T. Pisitkun, M. A. Knepper, and J. D. Hoffert, "An efficient algorithm for clustering of large-scale mass spectrometry data," in *Bioinformatics and Biomedicine (BIBM), 2012 IEEE International Conference on*, pp. 1–4, IEEE, 2012.