DE NOVO PEPTIDE SEQUENCING USING GENERAL-PURPOSE COMPUTING ON A GRAPHICS PROCESSING UNIT

Sankua Chao[1], James R. Green[1], Jeffrey C. Smith[2]
[1] *Department of Systems and Computer Engineering, Carleton University*
[2] *Department of Chemistry, Carleton University*

INTRODUCTION

Tandem mass spectrometry (MS/MS) can be used to identify peptides present in a biological sample containing unknown proteins. Prior to analyzing a protein using MS/MS, it is often cleaved enzymatically into peptides, where each peptide is composed of a chain of amino acids. Each of the 20 naturally occurring amino acids has unique properties, such as mass. Two principle approaches for computationally determining the sequence of a peptide using MS/MS spectral data are: (i) "spectral cross-correlation" between observed spectra and theoretical spectra of known peptides; and (ii) direct determination of the protein sequence from the spectral data (the *de novo* approach) [1]. Unlike spectral cross-correlation methods, the *de novo* approach does not require a complete database of all possible proteins that may be present in a biological sample [2]. Since it is often impossible to determine the complete peptide sequence from imperfect spectral data, a hybrid approach is often used, where *de novo* sequencing determines a "sequence tag" (or relatively short sub-string of a peptide sequence) which is then used to search a database of known peptides [3].

Each cycle of a typical mass spectrometry experiment includes a survey scan followed by a small number of MS/MS scans. The survey scan identifies the mass-to-charge (m/z) ratio of all peptides currently in the mass spectrometer instrument. In each MS/MS scan, one of these peptides is isolated and fragmented by the instrument, and the mass of each fragment is determined. Peptide fragmentation often occurs at the peptide bond between two amino acids, resulting in two fragments: a b-ion and a y-ion (left and right of the fragmentation point respectively). Other ion types are possible if the location of fragmentation is elsewhere within the amino acid molecule. Due to the nature of the peptide fragmentation process, the resulting MS/MS spectrum contains a series of peaks, often representing a series of fragments differing in length by a single amino acid. Ideally, the mass difference between a pair of such fragments would equal the mass of a single amino acid. This observation is straightforward when the fragments are the same ion type; however, an MS/MS spectrum may contain fragments of multiple ion types, and the ion types of individual peaks are not known *a priori*. Computational analysis of MS/MS spectra may be further complicated by the presence of noise artifacts, as the sample preparation process and instrument may introduce small amounts of low-mass, non-peptide molecules into the biological sample.

A number of groups are currently working towards achieving real-time information-driven MS/MS (id-MS/MS), where incremental peptide analysis and hypothesis generation occurs simultaneously with data collection [4][5]. While one framework for directed data acquisition and real-time peptide database searching has been described [5], it is limited to specific types of instruments. Another real-time peptide search engine batch-processes multiple MS/MS spectra [6], but its runtime does not scale well to the realistic operation of a real-time system which may process a single MS/MS spectrum at a time. Furthermore, both of these systems are based on spectral cross-correlation techniques, and could not leverage recent advances in rapid exact peptide sequence search algorithms being developed by our group [7]. Existing *de novo* peptide sequencing algorithms [1][3] generally were not designed for real-time id-MS/MS. An essential requirement for real-time id-MS/MS is

runtime of less than one second for the entire computational protein identification pipeline, which includes *de novo* peptide sequencing. In this work, a *de novo* peptide sequencing algorithm was implemented using general-purpose computing techniques on a graphics processing unit (GPU), in order to reduce the runtime of the algorithm to meet the strict runtime requirements for id-MS/MS.

METHODOLOGY

Depending on the instrument being used, particular ion types are expected to be present in the spectral data. In this work, each peak in a MS/MS spectrum can be assumed to arise from a b-ion, a y-ion, or noise. Since the actual ion type of each peak cannot be known, an "interpretation" represents an ordered set of possible ion type assignments for the peaks in a MS/MS spectrum. Given a list containing L peaks, there are 3^L possible interpretations. An interpretation with T non-noise (i.e. b-ion or y-ion) peaks may lead to a sequence tag of length (T-1). An interpretation containing L peaks can be compactly represented by an "interpretation bit vector" containing (2*L) bits, where each peak type is represented by two bits: 00 indicates noise; 10 indicates b-ion; 11 indicates y-ion. The number of interpretations was reduced by examining only those values of T that may result in sequence tags with an appropriate length (4 to 9 amino acids) for further peptide database search. The number of interpretations was further reduced by realizing each interpretation has a complementary interpretation (arising from the dual relationship between b-ions and y-ions) which would result in the same sequence tag. For each {L,T} combination of interest, interpretation bit vectors were pre-computed and stored as hex values in a text file.

When an interpretation is applied to a given MS/MS spectrum, peaks are adjusted: noise peaks are removed, b-ion peaks are unchanged, and y-ion peaks are replaced with their b-ion counterparts (i.e. the mass of the complete peptide minus the observed y-ion mass). Using these adjusted masses, the mass differences between pairs of adjacent non-noise peaks are then compared with known amino acid masses, and the interpretation is scored based on the relative agreement of this comparison. The *de novo* peptide sequencing algorithm described in this work makes use of the GPU's multiple-core SIMD parallel processing architecture, which allows the score computation to be applied to multiple interpretations in parallel. Three input parameters are required: the name of an MGF (Mascot generic data format) file containing the de-isotoped and centroided MS/MS spectrum; the number of peaks to include in the interpretation (L); a range of values for number of non-noise peaks (T). A ranked list of top-scoring interpretations and corresponding sequence tags are output in a text file.

When there are more than L peaks in the input MS/MS spectrum, a subset of L peaks is selected, with the dual aims of biasing selection towards peaks with relatively high intensity and with mass-to-charge ratio above that of the complete (parent) peptide, since such peaks are less likely to arise from noise. These aims are achieved by doubling the intensity of peaks above the mass-to-charge ratio of the peptide, then selecting the L highest-intensity peaks. The masses of the selected peaks are retained (assuming a charge of 1); however, intensity data are not used any further. The unselected peaks are discarded.

A "lookup table" and a "lower-triangular matrix" are generated, and are commonly used in the computation of each interpretation score. The lookup table (or "LT"), of length (2*L), relates the order of peaks when sorted by original mass and when sorted by adjusted mass (i.e. after application of the interpretation bit vector). The LT is used to obtain indices of elements in the flattened lower-triangular matrix (or "LTM"). Each element in the LTM represents the difference between the adjusted masses of a source peak and a destination peak, where the adjusted mass of the source peak is less than that of the destination peak. The LTM contains difference values between all possible pairs of peaks, including both b-ion and y-ion assignments of each peak. Each element in the LTM is then associated with the amino acid whose mass is closest to the element's difference value. An "edge weight" is calculated (adapted from Vonode [8]) for each element, indicating the agreement between the element's difference value and the mass of the associated amino acid. The difference values in

the LTM are replaced by the edge weight values. Both the LT and LTM are stored in GPU memory.

For a particular {L,T} combination, the total number of interpretations and the interpretation bit vectors are loaded into GPU memory. GPU parameters are specified to maximize the number of GPU threads to be executed in parallel, where each GPU thread processes a single interpretation. For a given interpretation, the unique thread ID is used to retrieve the corresponding interpretation bit vector. The interpretation bit vector determines which peaks are included as which ion types in the interpretation. The elements in the LT corresponding to the non-noise peaks and applied ion types are accessed, to obtain the appropriate indices into the flattened LTM. The interpretation score is calculated as the sum of the edge weight values (from the LTM) for each adjacent pair of non-noise peaks. The interpretation's ID and score are stored to GPU memory. After all interpretations for a particular value of T have been scored, the interpretations are ranked based on score, and the 512 top-ranked interpretations are retained. Each interpretation score is divided by the sequence tag length (T-1), resulting in a normalized interpretation score between 0.0 to 1.0. Using the amino acids associated with each element in the LTM, the sequence tag for each interpretation is constructed.

After all possible T for the given L have been processed, all retained interpretations (i.e. 512 top-ranked per T) are ranked based on normalized interpretation score, and output with corresponding sequence tags in a text file.

RESULTS

The GPU-based *de novo* peptide sequencing algorithm described in this work was implemented using NVIDIA Compute Unified Device Architecture (CUDA) 3.2, in 64-bit Windows 7 Professional SP1, running on a 2.67GHz Intel Core i7 CPU 920, with a NVIDIA GeForce GTX 260 graphics card (192 cores, split evenly among 24 multiprocessors).

The algorithm was tested on a dataset of 64 de-isotoped and centroided MS/MS spectra, each containing at least 17 peaks (Jeffrey C. Smith, unpublished data). The spectra were acquired from an ESI-TRAP (QTRAP) instrument, using two biological samples: bovine serum albumin, and Protmix (a standard sample containing 14 known proteins). The peptide sequence corresponding to each spectrum was validated using high-confidence peptide sequence assignments from the Mascot MS/MS Ions search engine.

Limited {L,T} combinations were possible, due to limited GPU memory. As the values of L and T increase, the number of interpretations increases, requiring more GPU memory to store interpretation bit vectors. The number of interpretations N(L,T) for a given {L,T} combination is calculated using Equation (1), and ranges up to 47297536 for {L,T}={19,10}. Possible {L,T} combinations included: {17-19, 5-10}, {20, 5-9}, {21-22, 5-8}, {23-26, 5-7}, {27-32, 5-6}.

$$N(L,T) = 2^{(T-1)} \times \binom{L}{T} \qquad (1)$$

The performance of the algorithm was assessed in terms of sequence tag accuracy and runtime. For a spectrum, a sequence tag is considered an "exact-match" if the entire sequence tag exactly matches a subsequence of the actual peptide. To be effective for further peptide database searching, at least one exact-match sequence tag should be ranked highly for each spectrum. For each L, the percentage of spectra in the dataset with at least one exact-match sequence tag in the top 1, 50, 100, 200, and 500 ranked sequence tags are shown in Figure 1. Spectra which had fewer than L peaks were excluded from analysis.

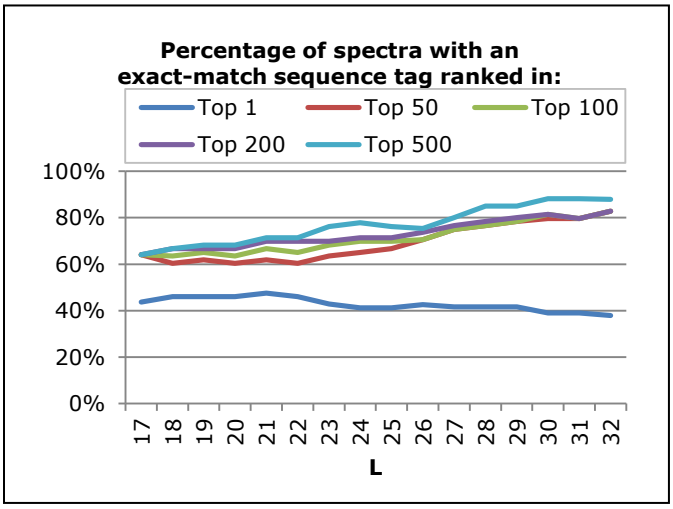

Figure 1: Sequence tag accuracy for dataset.

In general, higher L values (i.e. where more peaks from the original MS/MS spectrum are retained) resulted in greater sequence tag accuracy across the dataset. As seen in Figure 1, for L=32, over 80% of the spectra in the dataset contained an exact-match sequence tag in the top 50 ranked sequence tags. In an actual system, only the largest L permitted by GPU memory should be used.

As seen in Figure 2, inclusion of higher T values for each L caused an increase in runtime, sometimes exceeding the real-time requirement of runtime below one second. In an actual system, values of T between 5 to 10 which do not cause runtime to exceed one second should be used. Excluding higher values of T may be a valid strategy to reduce runtime while maintaining sufficient sequence tag accuracy, since the highest-ranked exact-match sequence tags originated mostly from lower values of T. For example, for both L=17 and L=32, out of the spectra with an exact-match sequence tag in the top 500, over 70% of such spectra had a highest-ranked exact-match sequence tag originating from T=5, despite L=17 having a wider range of possible T than L=32.

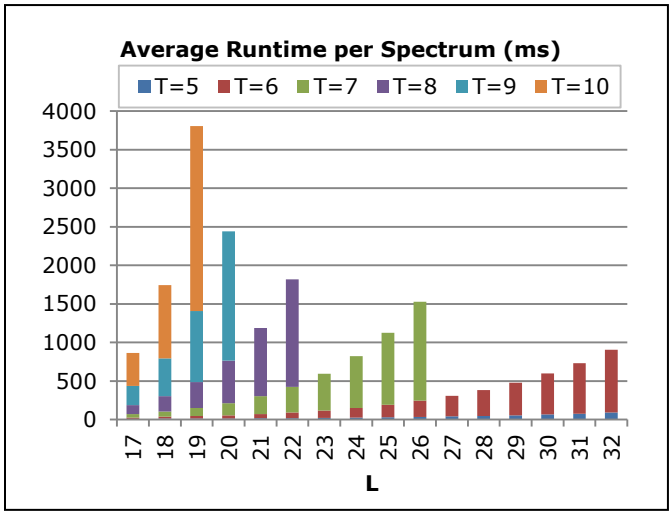

Figure 2: Average runtime per spectrum, for possible {L,T} combinations.

CONCLUSIONS

In this work, a preliminary implementation of a *de novo* peptide sequencing algorithm using general-purpose computing techniques on a GPU was presented, and performance was assessed. Processing a single MS/MS spectrum at a time, the algorithm filters a list of peaks based on relative intensity, generates every possible sequence tag between 4 to 9 amino acids in length, and ranks the sequence tags based on a normalized score. By limiting the length of sequence tags to be generated, the algorithm can meet the real-time performance requirement of runtime below one second.

Future work includes using a sparse vector representation for interpretations (instead of interpretation bit vectors), which will allow a greater number of peaks per interpretation, and increase sequence tag accuracy. Using a newer GPU with more cores and memory will allow more interpretations to be processed in parallel, and reduce runtime. The interpretation score computation could be improved to ensure exact-match sequence tags are ranked highly. Finally, the algorithm could be extended to potentially enable prediction of post-translational modifications.

REFERENCES

[1] A. Nesvizhskii, O. Vitek, and R. Aebersold, "Analysis and validation of proteomic data generated by tandem mass spectrometry," *Nat. Methods*, vol. 4, no. 10, pp. 787-797, 2007.

[2] 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," *J. Proteome Res.*, vol. 10, pp. 2882-2888, 2011.

[3] B. Ma and R. Johnson, "*De novo* sequencing and homology searching," *Mol. Cell. Proteomics*, vol. 11, no. 2, pp. 1-16, 2012.

[4] A. Zerck, *et al.*, "An iterative strategy for precursor ion selection for LC-MS/MS based shotgun proteomics," *J. Proteome Res.*, vol. 8, pp. 3239-3251, 2009.

[5] J. Graumann, R. A. Scheltema, Y. Zhang, J. Cox, and M. Mann, "A framework for intelligent data acquisition and real-time database searching for shotgun proteomics," *Mol. Cell. Proteomics*, vol. 11, no. 3, pp. 1-11, 2012.

[6] P. McQueen, *et al.*, "Information-dependent LC-MS/MS acquisition with exclusion lists potentially generated on-the-fly: case study using a whole cell digest of *Clostridium thermocellum*," *Proteomics*, vol. 12, pp. 1160-1169, 2012.

[7] R. J. Peace, H. A. Mahmoud, and J. R. Green, "Exact string matching for MS/MS protein identification using the Cell broadband engine," *J. Med. Biol. Eng.*, vol. 31, no. 2, pp. 99-104, 2011.

[8] C. Pan, *et al.*, "A high-throughput *de novo* sequencing approach for shotgun proteomics using high-resolution tandem mass spectrometry," *BMC Bioinformatics*, vol. 11, no. 118, 2010.