RAPID COMMUNICATIONS IN MASS SPECTROMETRY
Rapid Commun. Mass Spectrom. 2005; **19**: 833–837
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/rcm.1853

RCM

Hardware-accelerated protein identification for mass spectrometry

Anish T. Alex[1,2], Michel Dumontier[2], Jonathan S. Rose[1] and Christopher W. V. Hogue[2]*

[1]Department of Electrical and Computer Engineering, University of Toronto, 10 King's College Road, Toronto, Ontario, M5S 3G4 Canada
[2]Blueprint Initiative, Samuel Lunenfeld Research Institute, Mount Sinai Hospital, 522 University Avenue, Suite 900, Toronto, Ontario, M5G 1W7 Canada

Received 19 December 2004; Revised 18 January 2005; Accepted 19 January 2005

An ongoing issue in mass spectrometry is the time it takes to search DNA sequences with MS/MS peptide fragments (see, e.g., Choudary *et al.*, *Proteomics* 2001; 1: 651–667.) Search times are far longer than spectra acquisition time, and parallelization of search software on clusters requires doubling the size of a conventional computing cluster to cut the search time in half. Field programmable gate arrays (FPGAs) are used to create hardware-accelerated algorithms that reduce operating costs and improve search speed compared to large clusters. We present a novel hardware design that takes full spectra and computes 6-frame translation word searches on DNA databases at a rate of approximately 3 billion base pairs per second, with queries of up to 10 amino acids in length and arbitrary wildcard positions. Hardware post-processing identifies *in silico* tryptic peptides and scores them using a variety of techniques including mass frequency expected values. With faster FPGAs protein identifications from the human genome can be achieved in less than a second, and this makes it an ideal solution for a number of proteome-scale applications. Copyright © 2005 John Wiley & Sons, Ltd.

Although it is possible to generate the *de novo* sequence of a peptide given its tandem (MS/MS) mass spectra, full length sequencing of proteins remains a significant challenge.[1,2] Common approaches to protein identification use protein databases to either compare experimental with theoretical spectra with cross-correlation measures,[3] or compare experimental peptide masses with theoretical peptide maps.[4] Inherent in both approaches is the assumption that gene predictions are comprehensive and accurate so as to identify all protein open reading frames and splice variants. EST sequencing has increased the coverage of these peptides, but EST sequences are known to contain errors. In contrast, a DNA-based search for protein identification has become more feasible with the recent production of several high-quality drafts of the human genome and of a growing number of organisms, combined with improvements in MS/MS instrumentation leading to higher quality peptide sequencing.[5] By searching through unannotated DNA sequences researchers can identify peptides that are mis-predicted or missed altogether by gene prediction algorithms; such cases are appreciable, even for organisms like *Saccharomyces cerevisiae,* for which a complete set of coding regions is thought to be known.[6] If the protein-coding region could be quickly located with a DNA search, then this sequence and any predicted splice variants can be compared with other MS-derived peptides and can be used for protein identification. In addition, an exclusion list containing peptide masses from an unambiguously identified protein or expected peptide masses that need not be analyzed could be constructed on the fly, allowing the MS operator (or an automated approach) to target novel features, such as alternative splice variations and post-translational modifications (PTMs), while reducing the amount of sample consumed.

With standard MS techniques, sample is rapidly consumed during analysis. Purified samples are hard to acquire and prepare,[7] and thus fast and sensitive analysis is crucial (Sciex. Toronto, ON, Canada). However, search algorithms are limited by available memory and the latency imposed by conventional PC memory architectures. A common solution to this problem is to parallelize search jobs across a cluster, although the associated costs of cluster acquisition and maintenance can be high. Furthermore, the inherent communication delay in a conventional network cluster demands a non-linear increase in cluster size to realize enhanced performance.

Often when a parallelizable task, such as a search, contains simple repeatable operations at its core, custom hardware offers a practical solution. Field-programmable gate arrays (FPGAs) are essentially programmable circuits that can be customized to maximize the memory bandwidth and suffer none of the overhead associated with a sequential microprocessor-based program. The work presented here describes an FPGA-based solution to searching DNA with an initial peptide and mass spectrum.

**Correspondence to*: C. W. V. Hogue, Blueprint Initiative, Samuel Lunenfeld Research Institute, Mount Sinai Hospital, 522 University Avenue, Suite 900, Toronto, Ontario, M5G 1W7 Canada.
E-mail: chogue@blueprint.org

Contract/grant sponsor: The Natural Sciences and Engineering Research Council of Canada.

DESCRIPTION AND METHODS

The hardware solution presented here is divided into three blocks: a *search engine* that takes a query peptide and locates all possible coding sequences within DNA sequences; a *peptide mass calculator* that translates the coding sequence in six frames to a set of peptides; and a *scoring unit* that compares the set of peptides with those detected by MS (Fig. 1).

Upon initialization, the DNA sequence, which may be agglomerated from multiple organisms, is loaded into memory that is directly connected to the FPGAs. The query peptide is translated into the set of all possible DNA sequences in the six reading frames using the appropriate amino acid to nucleic acid codon translation table. Amino acids encoded by redundant codons are represented using wildcards, as shown in Fig. 2. All query translations are sent to the search engine, while the masses detected by MS are sent to the scoring unit.

The search engine reads a set of nucleotides from the DNA sequence (this is called the memory word) and compares it with the query translations. The standard nucleotide encoding scheme uses 3 bits, but this can be extended to 4-bit encoding to accommodate known single-nucleotide polymorphisms (SNPs). In the 3-bit scheme the two least significant bits encode the nucleic acid, while the third bit indicates the presence of a wildcard. In the 4-bit scheme each bit position corresponds to the four possible bases and a wildcard is represented by asserting all 4 bits. In either scheme the hardware is customized to mask wildcard positions in both the query translations and the memory word. This masking signals an automatic match to the search engine at the corresponding position, reducing the amount of computation necessary to process strings with wildcards.

To exploit the advantages of customizable architecture, multiple copies of the query translations are staggered across the word by one nucleotide and compared with every nucleotide in the entire memory word simultaneously (Fig. 2). When a match occurs, a variable-sized nucleotide memory buffer (3000 nucleotides for yeast, 12 000 for human), approximating the size of two genes, is centered on the match and is passed to the *peptide mass calculator*.

The *peptide mass calculator* translates the nucleotide memory buffer in six frames, identifies proteolytic cleavage points (tryptic in our current implementation), and generates a set of possible digested peptides for comparison with the peptide masses detected by MS (Fig. 3). Multiple calculator units process each codon in the memory word, allowing the simultaneous translation of multiple nucleotides. Each codon is translated into its corresponding amino acid residue mass, and the remaining codons in the word are buffered within the calculator and passed to subsequent calculation units. Additional units detect peptide cleavage points and wildcards within the memory word to ensure accurate translation. Although the calculator operates continuously, the resulting masses are stored only after a hit is detected by the search engine.

Masses are stored on the FPGA using the content addressable memory (CAM) style of addressing, so that the memory address of the mass corresponds to its value. In the *scoring unit*, calculated tryptic masses are compared with peptide masses using the four most significant bits (i.e., the 4 leftmost bits) of the mass. For example, a calculated tryptic peptide mass of 71 (stored as 1000111) will match a parent mass of 70 (stored as 1000110) using '1000' as the address. If the difference between these masses remains within a user-defined threshold, a match is signaled. This technique allows

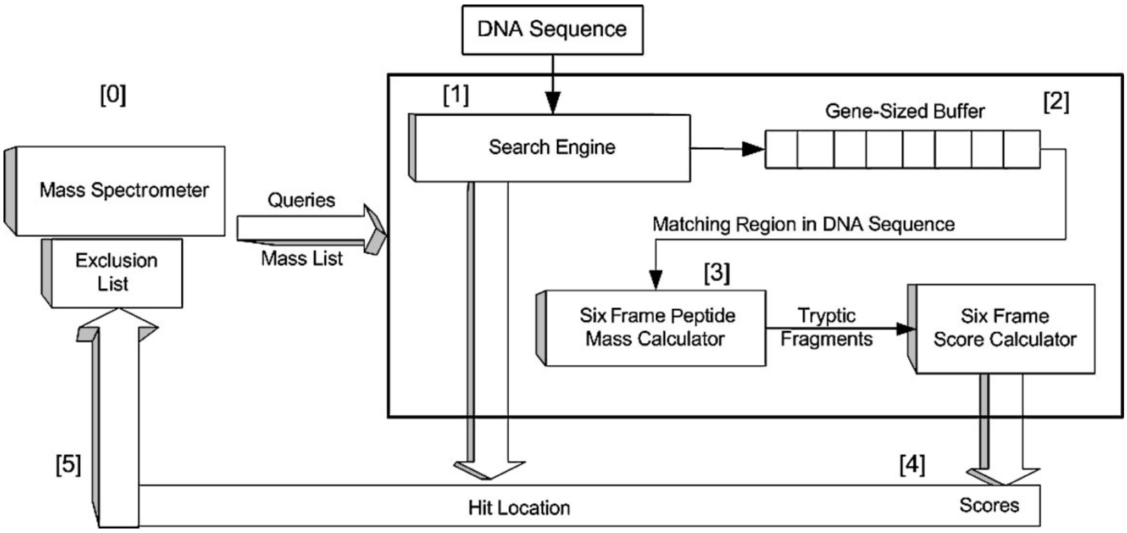

Figure 1. Device architecture: the flow of data through the system occurs as follows. (0) The mass spectrometer analyzes the sample and produces a list of precursor ion masses as well as fragment ion masses for a selected precursor ion. The fragment ion spectrum is translated to a peptide string using *de novo* techniques.[2] This translated peptide is sent as a query to the hardware along with the parent mass list. (1) The search engine reads the DNA sequences from external memory and locates potential coding regions for the query. (2) A sequence of nucleotides around the hit is sent to the calculator for translation. (3) The translated nucleotide sequence is digested, and tryptic peptide (precursor) masses are calculated. (4) The calculated masses are compared with the precursor mass list sent by the mass spectrometer. (5) Peptide masses are selectively added to the exclusion list to reduce analysis time and sample consumption.

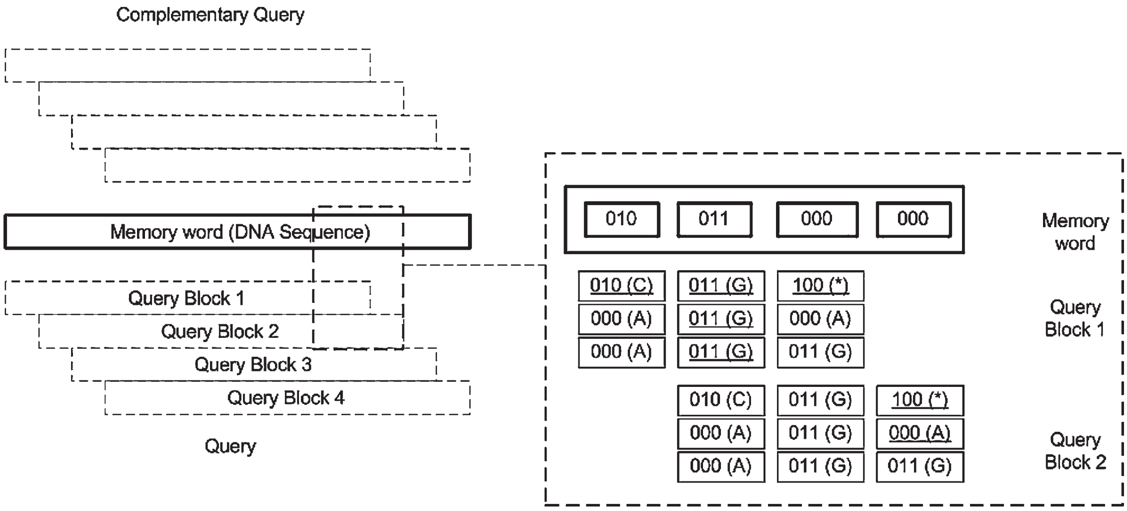

Figure 2. Search engine: the memory word is 63 bits in length and consists of 21 3-bit bases. As a simple example consider searching for the amino acid alanine, which can be coded by any of {CG*, AGA, AGG}. In the single time step illustrated, a match is found in the first row of query block 1, as every nucleic acid in the row matches its corresponding memory word (matches are underlined). Note that inclusion of the wildcard encoding (100) allows compression of the query such that each of the four nucleotides need not be explicitly stored in the hardware.

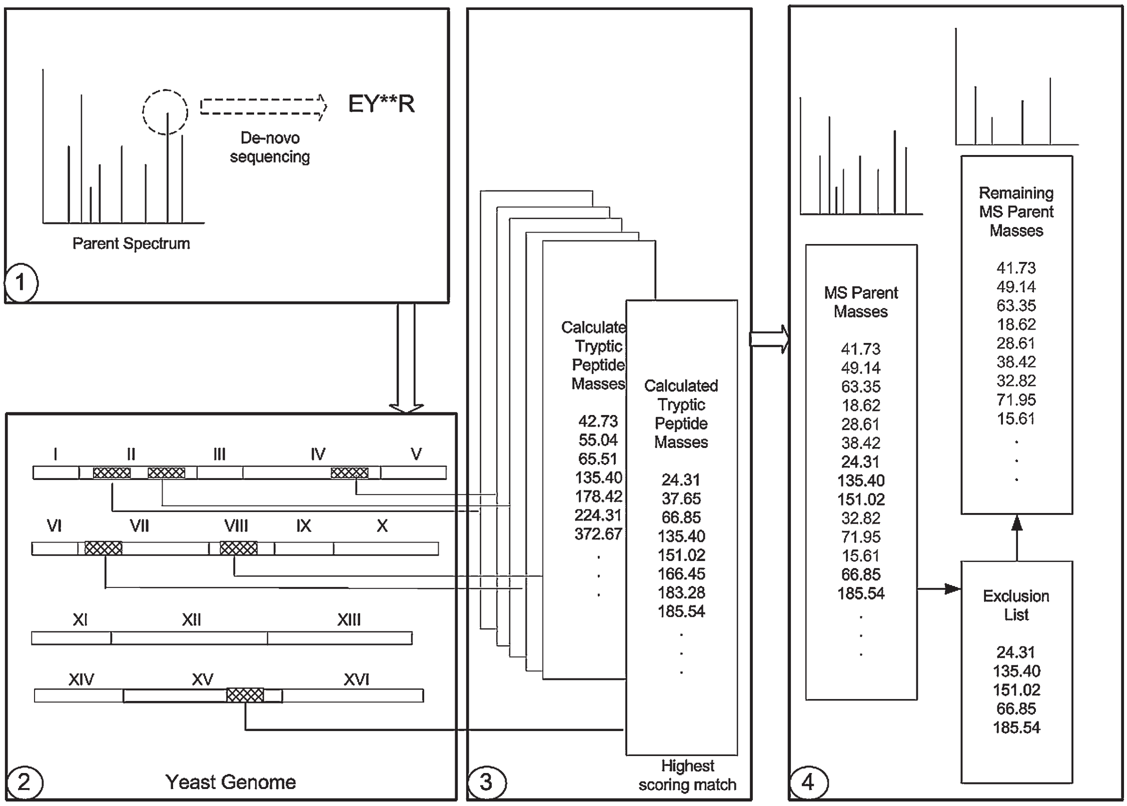

Figure 3. Example search against the yeast genome. (1) A single fragment ion is picked from the MS/MS spectrum of the original precursor ion, and its sequence (including possible wildcards) is obtained by *de novo* methods. (2) The search engine identifies and ranks sets of potential coding locations (thatched boxes) for this query across the 16 chromosomes of the yeast genome. (3) Each potential coding location is translated in six frames to produce sets of tryptic masses. (4) Each calculated tryptic fragment list is then compared with the original list of precursor masses. Matches from the highest scoring region are added to the exclusion list, effectively truncating the list of remaining precursor masses to be analyzed.

the hardware to compare a single mass value with a large list in a single operation.

MS efficiency and sample conservation is often aided by the use of an exclusion list. These lists typically contain peptide masses that are considered to be of little interest, such as those from trypsin or keratin, and are filtered by the mass spectrometer datasystem so as to be eliminated from MS/MS analysis. The hardware-matching units described above take the original list of masses detected by MS and dynamically build an exclusion list consisting of other tryptic peptide masses from a match to a high-scoring query peptide (Fig. 3). This approach not only increases the confidence of locating multiple peptides close to the query hit location, but also ensures that unidentified peptides and novel unexpected peptides will be further analyzed. Thus, the ability to generate dynamic exclusion lists greatly reduces analysis time and the amount of sample required.

We developed two simple scoring methods to rank the list of possible hit locations for the query peptide. The simplest scoring scheme ranks the hits by the number of tryptic matches in the original mass spectra. The second scoring scheme extends the first scoring method by using a hardware-encoded frequency of mass occurrence histogram, based on the MOWSE algorithm,[8] to assign a confidence score to mass matches. A higher score reflects the increased likelihood that the mass matches correspond to true coding regions, and lower scores that fail to meet a threshold can be filtered from the result set. In cases where genes contain large introns and the size of the gene exceeds the scoring window, multiple hits would be treated independently. However, post-processing could cluster the hit locations to known gene locations and increase overall match confidence.

To investigate sensitivity and specificity of naively searching DNA sequences, we determined the number and size of peptide queries required to uniquely identify a protein on both the yeast and human genomes. We found that peptide queries of only 6 amino acids uniquely matched coding locations across the yeast genome, and that 2 peptides of length 5 or 5 peptides of length 4 would be required to identify the true coding region. In comparison, a single query peptide of length 8, two peptides of length 7, or 3 peptides of 6 residues, were required to uniquely identify a coding region on human chromosome 13.

To determine whether the scoring algorithm increases accuracy, we considered a set of 3617 proteins identified from *S. cerevisiae* protein complexes using liquid chromatography/tandem mass spectrometry (LC/MS/MS) and commercial search engines.[9] After *in silico* digestion, we determined the number and length of generated peptides required for unique matches to proteins located on the *S. cerevisiae* genome using our hardware. For example, 4 or 5 length peptide queries from tryptic fragments of either the Rab escort protein [NP_015015] or an SSB2 variant of the HSP70 family of heat shock proteins [NP_014190] resulted in multiple matches across the yeast genome, as expected. In all tested cases, the best scoring hit identified the true coding region without requiring additional queries. Queries consisting of three or fewer residues significantly increased the number of false positives, but again the use of multiple short query peptides from the same protein resulted in a noticeable clustering of hits around the true coding location. In the future, we plan to pursue an improved scoring scheme that would take into account the number of short length matches required to identity the coding location from some predetermined distribution.

After the search is complete, the hardware outputs a list of the hit positions and scores. Our post-processing software maps the hits to genomes, thereby allowing researchers to interpret matches and visualize potential gene matches along chromosomes, and potentially assist in the delineation of intron and exon ranges within a gene. For higher organisms, genes are mapped to the Unigene EST clusters, providing insight into tissue expression. We are pursuing a graphical user interface using a three-dimensional model of the human body capable of highlighting major organs corresponding to the tissue in which a match is expressed. Cross-referencing with sufficiently detailed anatomy ontology with disease dictionaries will provide a future framework for clinical diagnostics.

The hardware was originally designed and implemented using the University of Toronto's Transmogrifier 3A (TM3-A) hardware prototyping board (M. van Ierssel, D. Galloway, P. Chow and J. Rose, Dept. of Electrical and Computer Engineering, University of Toronto). It has since been implemented on the Gidel ProcStar 40-3 board (Gidel Ltd.[10]), with PCI interface and 2 GB of RAM. TM3-A designs were implemented using Synplify 7.2 and Xilinx ISE 5.2.3, while designs for the commercially available Gidel board were synthesized and placed using Altera's Quartus II 3.0.

Our hardware implementation requires 1.6 s to locate a query within the human genome. By comparison, a software implementation of the search algorithm against an unannotated human genomic sequence can process at an average rate of 210 s per query on a 600 MHz Pentium III.[1] Scaling this to current processor speeds, and assuming an optimistic linear increase in speed, such a search would require 52.5 s on a 2.4 GHz processor. Extrapolating this to a cluster of 2.4 GHz processors, a search time of 0.8 s could in theory be achieved with a 64 processor cluster. In contrast, two hardware units could deliver this performance at a cost 40 times lower than that of an equivalently capable software cluster.

The performance of the hardware solution is also influenced by the underlying scoring algorithm. The hardware search engine alone can process 4.1×10^9 bases per second, whereas the simple score search processes 3.2×10^9 bases per second and the improved scoring scheme based on frequency processes 2.025×10^9 bases per second. In addition, the reprogrammable fabric of the FPGA can be easily modified to accommodate a variety of scoring schemes customized to the user's needs.

SUMMARY

The hardware-accelerated protein identification solution presented here addresses a number of important issues in mass spectrometry. In particular, we have devised a high-performance machine that is capable of searching entire genomes in a time-frame comparable to spectra acquisition time. Moreover, our focus on DNA searching provides a framework for the identification of novel or short peptides which cannot be found in common protein databases. The architecture, coupled with comparatively high-performance low-cost custom hardware, makes it an ideal solution for a number of proteome-scale applications.

Acknowledgements

This work was funded by the Natural Sciences and Engineering Research Council of Canada (NSERC) in a grant to CWVH.

REFERENCES

1. Choudary JS, Blackstock WP, Creasy DM, Cottrell JS. *Proteomics* 2001; **1**: 651.
2. Taylor JA, Johnson RS. *Anal. Chem*. 2001; **73**: 2594.
3. Eng J, McCormack AL, Yates JR III. *J. Am. Soc. Mass Spectrom*. 1994; **5**: 976.
4. Pappin DJC, Rahman D, Hansen HF, Bartlet-Jones M, Jeffery W, Bleasby AJ. *Mass Spectrom. Biol. Sci*. 1996; 135.
5. McLuckey SA, Wells JM. *Chem. Rev*. 2001; **101**: 571.
6. Kumar A, Harrison PM, Cheung KH, Lan N, Echols N, Bertone P, Miller P, Gerstein MB, Snyder M. *Nat. Biotechnol*. 2002; **20**: 58.
7. Marshall A. *Nat. Biotechnol*. 2003; **21**: 213.
8. Pappin DJ, Hojrup P, Bleasby AJ. *Curr Biol*. 1993; **3**: 327.
9. Ho Y, Gruhler A, Heilbut A, Bader GD, Moore L, Adams SL, Millar A, Taylor P, Bennett K, Boutilier K, Yang L, Wolting C, Donaldson I, Schandorff S, Shewnarane J, Vo M, Taggart J, Goudreault M, Muskat B, Alfarano C, Dewar D, Lin Z, Michalickova K, Willems AR, Sassi H, Nielsen PA, Rasmussen KJ, Andersen JR, Johansen LE, Hansen LH, Jespersen H, Podtelejnikov A, Nielsen E, Crawford J, Poulsen V, Sorensen BD, Matthiesen J, Hendrickson RC, Gleeson F, Pawson T, Moran MF, Durocher D, Mann M, Hogue CW, Figeys D, Tyers M. *Nature*. 2002; **415**: 180.
10. See: www.gidel.com.