Published in final edited form as:

J Proteome Res. 2004 ; 3(6): 1292–1295.

GFSWeb: A web tool for genome-based identification of proteins from mass spectrometric samples

Michael S. Wisz, Melissa Kimball Suarez, Mark R. Holmes, and Morgan C. Giddings*

Departments of Microbiology & Immunology and Biomedical Engineering, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599-7290

Abstract

The interpretation of mass spectrometry data for protein identification has become a vital component of proteomics research. However, since most existing software tools rely on protein databases, their success is limited, especially as the pace of annotation efforts fails to keep pace with sequencing. We present a publicly available, web-based version of a software tool that maps peptide mass fingerprint data directly to their genomic origin, allowing for genome-based, annotation-independent protein identification.

Keywords

Mass spectrometry; genome; peptide mass fingerprint; web interface; gene finding

Introduction

Genome-based fingerprint scanning (GFS) allows the identification of unknown proteins by mapping a peptide mass fingerprint (PMF), obtained by mass spectrometry, directly to its genomic locus¹. Unlike other PMF-based protein identification programs that rely on protein sequence databases, GFS searches against raw genomic sequence, making possible the discovery of unannotated proteins and genes. GFS has a unique advantage over these other programs, especially when protein samples from recently sequenced or more complex genomes are studied. For example, although the genomes of *F. tularensis*² and the SARS-associated coronavirus³ have been sequenced, identification of proteins unique to these species is very limited with current protein identification software. Furthermore, even for more relatively well-annotated genomes, reliance on protein databases is still limiting. For example, it has been shown that more than 50% of *C. elegans* genes were predicted incorrectly⁴, the yeast proteome may vary over a range of over 1700 open reading frames (ORFs)⁵, and up to 50% of genes in the human genome remain unknown, according to a large-scale microarray study of human chromosome 22⁶.

The GFS program takes as input an experimentally obtained peptide mass fingerprint, scans a genome sequence of interest, and outputs the most likely regions of the genome from which the mass fingerprint is derived. Prior to searching, the software generates a theoretical mass list by translating the genome of interest in 6 reading frames (3 each on the forward and reverse strands) and digesting the resulting proteins *in silico* according to cleavage rules associated with the specified protease. GFS uses mass lists that are precomputed and stored on disk, eliminating the time-consuming step of generating them for each query and minimizing server startup time. The algorithm then finds matches (within a user-specified instrument tolerance)

Corresponding author e-mail: giddings@unc.edu.

between these theoretical masses and the input experimental masses, using a B-tree indexing algorithm to maximize computational efficiency. The actual B-tree is stored on disk as a binary file, and memory usage is minimized by only loading one tree node at a time, as needed. These matches (or “hits”) are grouped into high-density regions on the genome that can be scored and ranked according to a probability-based scoring scheme that takes into account the fractional mass coverage of the window by theoretical peptide masses. These regions are derived by scanning the genome with a fixed-sized window and then scoring each window according to criteria detailed in the original GFS publication¹. The subset of windows whose scores exceed a window score cutoff is then selected for an additional scan, subject to a separate extension score cutoff, that extends the start and stop positions of the *window* into a larger *region*. The present paper discusses the recently developed web interface to the GFS software, named *GFSWeb*, as well as some notable improvements in the software that have not been previously reported.

User interface

The GFSWeb user interface (Figure 1) includes three main sections: “Input Options”, “Output Options”, and mass list input. The “Input Options” panel allows user selection of parameters controlling selection of the theoretical mass list, the mass matching process, and identification of high-density peptide hit regions. The “Output Options” panel controls the scope and nature of the results presented. A single experimental mass list is input by either entering masses into the text box provided in the interface, or by uploading a text file to the web server. The results are then displayed in the web browser following completion of the request, and the HTML results file is optionally sent to the user’s e-mail address.

The “Input Options” section allows the user to select parameters affecting the analysis, allowing the selection of: the genome to be scanned (currently yeast and three *E. coli* strains, and to be expanded soon); the mass matching tolerance (in parts-per-million); the number of missed cleavages allowed; the size of the genome-scanning window (in nucleotides) from which the size of high-density hit regions are calculated; the mass type used to calculate the theoretical fragment masses (monoisotopic or average); whether or not the input masses are charged; whether methionines are oxidized; and whether iterative mode should be used in calculation of the hit regions. The last option has been added since the original version of GFS, and it stores the highest scoring hit region, removes the theoretical masses belonging to this region, and calculates the remaining hit regions. This process is repeated while masses remain and hit regions score above a cutoff determined by the pre-calculated score distribution of all the regions. In this initial version, there are a few parameters that are not yet user adjustable, such as the protease used; it is currently set to trypsin due to the ubiquitous use of that enzyme in proteomics experiments.

The “Output Options” section allows the user to specify the maximum number of hit regions to display, and the mass type (monoisotopic or average) used for calculation of predicted protein molecular weights that may be associated with a given hit region.

Graphical output

A table near the top of the HTML-formatted results page contains the number of the hit region, the name of the sequence in which the region is found, the maximum window score within the region, the start and stop position of the hit region, and the number of peptide hits in the region. Clicking on the number of the hit region will take the user directly to a detailed view of that particular region (Figure 2), which itself gives the location of peptide matches and potential ORFs found in the target sequence, shown both as a color-coded nucleotide sequence and as textual tables, as well as any annotation information found for the region. The colored nucleotide letters denote the location and frame of a peptide match (red for frame 0, green for

frame 1, blue for frame 2), where the first nucleotide of the match is colored pink and the last is brown. The underlined text follows a similar frame-based color scheme and denotes a potential ORF consistent with these peptide matches. The tables show the same information, but with additional details on the peptides and putative ORF-encoding proteins. A new GFS feature, the annotation-finder, searches .ptt and .faa files from NCBI for potential proteins whose start and stop locations overlap the peptide hit region, and displays anything found. It should be noted here that the current GFSWeb implementation does not handle introns, so all potential annotations found at this stage will be limited to single-exon genes. Since there are no intron-containing genes in *E. coli* and a only small number (~3.5% of the initially identified 6275 ORFs⁷) in yeast, this should not pose a major problem for users, but the ability to handle multiple-exon genes is a key GFSWeb improvement currently in development (see *Conclusions and Future Directions*).

How it works

The core GFS program is a Unix-based command-line program running on Mac OS X. The web interface was created using WebObjects (Apple Computer). The overall scheme, from user submission of an experimental mass list to GFS graphical output, is shown in Figure 3. When the user starts a GFSWeb job, the selected parameters and mass list are processed by the portal node, then passed to an available node on a 42-node Apple Xserve cluster via the Sun Grid Engine (SGE, from Sun Microsystems, Santa Clara, CA). Upon launch, GFS reads from disk all the pre-calculated peptide fragments generated from the *in silico* digest (dependent on the user's selection of organism, number of missed cleavages, and charge and isotopic state of the input masses). When this loading process is complete, the peptide mass matching process commences (dependent on the selected mass matching tolerance and whether oxidized methionines are being considered), followed by the clustering and scoring of peptide matches (constrained by the selected scanning window size and whether iterative mode is activated). After completion, WebObjects is notified by the Sun Grid Engine, then the HTML-formatted results are displayed in the user's browser and sent to their e-mail address. Although the time lag for results to be returned is dependent on the parameters used and the peptide mass list size, approximate benchmarks can be obtained by looking at extreme cases. Thus, a wait time from 30 seconds (for a search against *E. coli* using a 10 peptide mass list and no oxidized methionines) to 3 minutes (for a search against yeast using a 200 peptide mass list and oxidized methionines) can be expected for GFSWeb.

By distributing jobs onto a large computing cluster via SGE, GFSWeb offers two key advantages to the user. The first, most immediately noticeable consequence of this scheme is the ability of the interface to handle a large number of simultaneous jobs. Thus, unless the number of simultaneous users exceeds the number of cluster nodes, there will be no additional waiting for results beyond the actual calculation time. Moreover, even if the number of cluster nodes becomes a limiting factor, SGE is configured on this cluster to allow additional jobs to be farmed out to nodes operating under less than 100% load, thus still keeping the waiting time as low as possible. The second advantage of cluster distribution is not readily seen, but is more significant in its implications. When larger genomes (*e.g.*, *Tetrahymena*, mouse or human) are made available through GFSWeb, it will no longer be computationally efficient to use a single node for a given job, due to the large number of theoretical digest fragments produced by these genomes. For these future cases we have built and successfully used a model in which a given genome's peptide fragments are distributed between multiple nodes (either the entire cluster or a subset); the corresponding GFSWeb job is distributed to these nodes, and the results are combined and sent back to the user.

Conclusions and Future Directions

The web interface presented here represents Version 1.0 of a continually improving piece of software. Future releases will incorporate additional features, some already available at the command line and some currently in development. First, the inclusion of more genomes, both predominantly unspliced (*e.g.*, *Vibrio cholerae*, additional *E. coli* strains, and *Tetrahymena*) and multiple-exon (*e.g.*, human and mouse) should broaden the appeal of the software to a larger group of investigators. Second, the ability to handle new types of input data will be added. The submission of variable charge peptide data (*i.e.*, from an electrospray experiment) will soon become possible, and tandem MS data will be interpreted via a new approach that combines *de novo* sequencing and database searching. Third, an improved probability-based scoring scheme will replace the current heuristic-based scoring scheme. Finally, increased disk caching will increase program startup time and decrease memory usage.

The GFS software has proven a powerful tool for analyzing proteomic data, both for us and for our collaborators. For example, by using a combination of the GFS software and BLAST⁸, we have identified regions of the *F. tularensis* and *Tetrahymena* genomes which likely code for unannotated proteins. Moreover, similar studies by collaborators on the SARS-associated coronavirus are underway. Our hope is that this newly developed web interface will extend the benefits of GFS to the scientific community. Ideally, extensive application of the software to many diverse research projects, some that were previously infeasible without this method, will show our approach to be a viable alternative to protein database searching, as well as lead to the discovery of novel, functionally interesting proteins.

Synopsis

Today's proteomics investigators using mass spectrometric data for protein identification rely on a number of publicly available software tools that all have one feature in common – their reliance on protein databases. The development of GFSWeb (<http://gfs.unc.edu>) provides a genome-based, annotation-independent alternative to these programs, and potentially allows for the identification of unannotated proteins and proteins expressed through alternative mechanisms.

Acknowledgements

The development of GFSWeb was supported by the NIH National Human Genome Research Institute, grant HG000044 to MCG and an F32 NIH Postdoctoral Fellowship to MSW.

References

1. Giddings MC, Shah AA, Gesteland RF, Moore M. Proc Natl Acad Sci USA 2003;100:20–25. [PubMed: 12518051]
2. Prior RG, Klasson L, Larsson P, Williams K, Lindler L, Sjostedt A, Svensson T, Tamas I, Wren BW, Oyston PCF, Andersson SGE, Titball RW. Journal of Applied Microbiology 2001;91:614–620. [PubMed: 11576297]
3. Marra MA, Jones SJM, Astell CR, Holt RA, Brooks-Wilson A, Butterfield YSN, Khattri J, Asano JK, Barber SA, Chan SY, Cloutier A, Coughlin SM, Freeman D, Girn N, Griffin OL, Leach SR, Mayo M, McDonald H, Montgomery SB, Pandoh PK, Petrescu AS, Robertson AG, Schein JE, Siddiqui A, Smailus DE, Stott JE, Yang GS, Plummer F, Andonov A, Artsob H, Bastien N, Bernard K, Booth TF, Bowness D, Czub M, Drebot M, Fernando L, Flick R, Garbutt M, Gray M, Grolla A, Jones S, Feldmann H, Meyers A, Kabani A, Li Y, Normand S, Stroher U, Tipples GA, Tyler S, Vogrig R, Ward D, Watson B, Brunham RC, Krajden M, Petric M, Skowronski DM, Upton C, Roper RL. Science 2003;300:1399–1404. [PubMed: 12730501]
4. Reboul J, Vaglio P, Lamesch P, Martinez M, Armstrong CM, Li S, Jacotot L, Bertin N, Janky R, Moore T, Hudson JR Jr, Hartley J, Brasch MA, Vandenhaute J, Boulton S, Endress GA, Jenna A, Chevet E, Papasotiropoulos V, Tolias PP, Ptacek J, Snyder M, Huang R, Chance MR, Lee H, Doucette-Stamm L, Hill DE, Vidal M. Nature Genetics 2003;34:35–41. [PubMed: 12679813]
5. Harrison PM, Kumar A, Lang N, Snyder M, Gerstein M. Nucleic Acids Res 2002;30:1083–1090. [PubMed: 11861898]
6. Rinn JL, Euskirchen G, Bertone P, Martone R, Luscombe NM, Hartman S, Harrison PM, Nelson FK, Miller P, Gerstein M, Weissman S, Snyder M. Genes and Development 2003;17:529–540. [PubMed: 12600945]
7. Goffeau A, Barrell BG, Bussey H, Davis RW, Dujon B, Feldmann H, Galibert F, Hoheisel JD, Jacq C, Johnston M, Louis EJ, Mewes HW, Murakami Y, Philippsen P, Tettelin H, Oliver SG. Science 1996;274:563–567.
8. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. J Mol Biol 1990;215:403–10. [PubMed: 2231712]

GFS
Genome-based Peptide Fingerprint Scanning

Input Options

Select a **genome**. Yeast

Indicate the **tolerance** in parts per million you will allow for peptide mass comparison. 100 ppm

GFS scans the genome in "**windows**" to locate matches. Select a window size. 1000 nt

Would you like to **iteratively** re-submit lower scoring peptides to GFS? Yes

Select the peptide mass **INPUT type**.
☐ Average
☒ Monoisotopic

Select the **Mass Charge State**.
☒ [M]
☐ [M+H]⁺

Select max number of **Missed Cleavages**. 1

Select any variable modifications. ☐ Oxidized Methionines

Output Options

Select the peptide mass **OUTPUT type**.
☐ Average
☒ Monoisotopic

Enter top number of **matches** to display. 10

E-mail: Results will be displayed, then emailed to your registered e-mail account. gfsadmin@listserv.unc.edu

Peptide Masses
(delimited with any combination of: space, comma, tab, newline)

Upload a text file for **data acquisition**:
 no file selected

----- OR -----

Enter peptide masses below:

Please be patient while this computationally exhaustive program runs on our cluster of [Apple XServes](#). Thank you!

Genome-based fingerprint scanning (GFS) is a recently reported method developed by the Giddings lab (Giddings et al., 2003) that maps peptide mass fingerprint data directly to raw genomic sequence, enabling rapid, low-cost identification of proteins in genomes for which annotation is lacking. The program takes as input an experimentally obtained peptide mass fingerprint, scans a genome sequence of interest, and outputs the most likely regions of the genome from which the mass fingerprint is derived. The software first generates a theoretical mass list by translating the genome of interest in 6 reading frames (3 each on the forward and reverse strands) and digesting the resulting proteins in silico according to cleavage rules associated with the specified protease (trypsin in this case). The algorithm then finds matches (within a given mass tolerance) between these theoretical masses and the input experimental masses. [more...](#)

Figure 1.
The GFSWeb job submission screen (<http://gfs.unc.edu>).

Figure 2.

Example of a detailed hit region description. Summary information is first displayed, along with results from any annotation search, followed by the genomic sequence of the region. Color-coded nucleotides represent the location and frame of the experimental peptide masses mapped to this region, and color-coded underlines similarly represent potential ORFs. In this case, the only ORF predicted corresponds to the rbsB gene found by the annotation search.