Sudipta Bandyopadhyay
MCDB 452a: Genomics and Bioinformatics
Final Project
December 13, 2006

Synteny Mapping: Not Just Another Alignment

As more and more genomes are being sequenced, the field of *comparative genomics* is emerging as a pivotal way to make sense of and bring order to the ever-growing data set. Comparisons between genomes can help elucidate form, function, and evolutionary mechanisms, whether the comparisons are on a small scale (i.e., a sequence alignment between two homologous genes on different chromosomes in humans) or large scale (i.e., a synteny map of the human and *D. melanogaster* genomes).

While the fundamental principles of the field date back several decades, many newer applications are still being perfected. Indeed, only the most basic technique of sequence alignment has been available since 1970 (Needleman and Wunsch, 1970), and it was not until the mid-1980s that a search feature finally became available. Since then, many more specialized algorithms have been developed, yet these newer procedures—many of which focus on broader aspects of comparative genomics—recapitulate on the classical themes of older methods, as I shall demonstrate here in the case of synteny mapping.

Sequence Alignments

The primary method of analysis in comparative genomics has long been the Needleman-Wunsch algorithm for global alignment (Needleman and Wunsch, 1970). Published in 1970, the algorithm optimally aligns two sequences A and B using a two-

dimensional matrix, with rows and columns sequentially representing residues[1] in A and B. This layout concisely captures every possible alignment between the two sequences as a different pathway through the matrix. Each cell is then given, by default, a value of one for a match or zero for a mismatch; however, the possibility of more elaborate point values as per the PAM and BLOSUM substitution matrices was predicted in the original paper by Needleman and Wunsch. The matrix is summed from bottom right to top left, in such a way that matches contribute positively to an alignment while mismatches and gaps penalize the alignment. The summed matrix allows for easy tracing of the optimal alignment and, moreover, returns an absolute score measuring the strength of this optimal alignment.

In 1981, the Needleman-Wunsch algorithm was slightly modified to create the Smith-Waterman algorithm for finding local alignments (Smith and Waterman, 1981). The differences between the two algorithms are merely in the details of execution and of the input parameters; otherwise, Smith-Waterman functions in the same way as Needleman-Wunsch. As such, the common ground between these two techniques set some striking thematic precedents that have stuck around in most techniques to come thereafter. First, both methods return a score measuring the strength of the sequence alignment. Moreover, as computing power increased, these absolute scores were turned into expectations (or *P* values) based on the value of random simulations. It was found that these *P* values fit an extreme value distribution. Secondly, every possible alignment between the two sequences is represented as a different pathway through the matrix; the algorithm is thus unbiased, in that it is performed upon a platform which gives every

[1] Though the paper explained the methodology in terms of protein alignments, the same tactic is clearly applicable for DNA or RNA alignments; as such, "residue" may apply equally to either an amino acid in a protein or a nucleotide in a nucleic acid.

possibility equal weight, and so the optimal alignment of sequences A and B is also an unbiased alignment. Third, the algorithm is computationally intensive, on the order of $O(mn)$, where m and n are the lengths of sequences A and B.

Neither the Needleman-Wunsch nor the Smith-Waterman algorithm, however, is amenable to searching genomic databases for high-scoring alignments. While such a search may be construed simply as a local alignment between the query sequence (a nice, manageable size m) and a sequence representing a composite of the entire genomic database (of exorbitant size n), the latter yields a prohibitively high value for the computational complexity $O(mn)$. As such, separate techniques had to be pioneered to allow searches for sequences yielding high-scoring alignments.

Alignment Searches

In 1985, FASTP was introduced as a means for performing such searches among protein sequences (Lipman and Pearson, 1985). The method was soon upgraded to FASTA, which included the ability to search for nucleic acid sequences as well (Lipman and Pearson, 1988). FASTA utilizes a lookup table to narrow down the possibilities very quickly in a rather cheap operation, on the order of $O(n + m)$, although the sensitivity and computational complexity of the search depends partly on the value of the *ktup* parameter. FASTA then proceeds through a series of steps modeled off of the Smith-Waterman alignment algorithm to score and rank the search results.

FASTA is faster than the Smith-Waterman algorithm, but it is not the fastest search tool available. Among popular bioinformatics tools, BLAST is faster (Altschul *et al.*, 1990). BLAST scans the query sequence and divides it into seeds of a given "word

size" (set to 11 by default). BLAST then looks up these seeds in pre-defined hash tables to very quickly identify potential matches. Along these matches, BLAST extends the alignment in both directions without gaps. If a high-scoring alignment is found, BLAST runs a Smith-Waterman-like gapped alignment to calculate alignment scores, which are then used to rank the results.

Though BLAST and FASTA differ from each other more than Needleman-Wunsch and Smith-Waterman do, the common ground between the two search methods are nonetheless especially relevant as thematic precedents. In particular, both approaches split the query into smaller words and take advantage of hash tables or lookup tables to quickly search for potential matches on the basis of short stretches of identity (i.e., shared words) between the sequences. This list of potential hits is then further pared down while avoiding the computationally-expensive sequence alignment as long as possible. Finally, in order to score and rank the hits, a modified sequence alignment is performed.

Synteny as an Alignment Implementation

These aforementioned bioinformatics tools were well tested and their precedents well established by the time that enough genomes had been physically mapped to form synteny maps between two different genomes. On the surface, creating a synteny map is hardly a problem; after all, it is merely a local alignment of segments of chromosomes between two genomes. Unfortunately, the complexity of this operation is prohibitive: at $O(mn)$, with m and n on the order of whole genomes, the Smith-Waterman algorithm would simply not work for the same reason that it did not work for searches. Indeed, this

is why bioinformaticians seeking efficient synteny maps eventually turned to methods that recapitulated the themes seen in the search mechanisms discussed earlier.

Nonetheless, the first synteny maps between mouse and human were made by an alignment analysis of human and mouse clone maps using 51,486 homology crosslinks (Gregory *et al*., 2002). However, this work is on a prohibitive scale if it is to be done between every two genomes of interest. Indeed, the paper features 86 authors from 7 different institutions, but even then the synteny map was but an estimate. Quite appropriately, in a paper of such epic proportions, we find the quip: "Ultimately, the alignment of finished sequences of the two genomes will be required to define the exact number and boundaries of every conserved segment." The scientific value of defining the exact number and boundaries of every conserved segment is debatable, but there is little doubt that reaching such high-resolution conclusions is near impossible due to evolutionary noise, even if a perfect and large-scale implementation of the Smith-Waterman algorithm could be executed. Instead, a more meaningful contribution to science would be the completion of lower-resolution synteny maps of as many pairings of different genomes as possible. With the bar for the desired resolution thus lowered, it is advisable to approach the problem of computing synteny maps as a search implementation.

Synteny as a Search Implementation

In 2004, a method for synteny mapping based on "UniMarkers" was developed (Liao *et al.*, 2004). The hallmark of this approach is the UniMarker (UM), which even the authors of the paper point out is in the tradition of hash tables and suffix-trees, and

can thus "significantly speed up the computation time required for sequence mapping." A UM is a 15-mer sequence that appears exactly once in a genome, and thus lends itself naturally for use as a marker or seed like the words in a hash table. UMs are surprisingly abundant; earlier work found that, in a draft sequence of the human genome, there were 162,253,846 UMs, representing more than 15% of all potential 15-mer sequences (Chen *et al.*, 2002). Indexing the UMs in a genome is a process of $O(n)$ complexity in the genome size. This algorithm for synteny mapping relies on UM pairs between the two organisms, which is of $O(m + n)$ complexity to produce.

Once the UM pairs (UMps) are established between genomes A and B, each chromosome of B is divided into a set of minimally overlapping fragments, each containing approximately 300,000 UMps. For every chromosomal fragment in B, genome A is scanned with a 50kb window moving with a step size of 10kb. The ratio M_{ij}, the number of UMps common to both the i-th window of genome A and the j-th chromosomal fragment of genome B (N_{ij}) to the total number of UMps found in the i-th window of genome A (N_i) (i.e. $M_{ij} = N_{ij}/N_i$), is computed. Appropriate cut-offs and recursive iterations on alternating genomes are used to identify orthologous regions, or "anchoring islands", between genomes A and B. Each anchoring island maps uniquely from genome A to genome B. False positives are reduced by repeating the maneuvers in the opposite direction as well, and taking the overlap in both sets of anchoring islands. The remaining anchoring islands are merged into conserved segments, which are further merged into syntenic blocks, thereby creating a full synteny map.

The synteny map was notable for several reasons. First, it agreed very favorably with previously constructed mouse/human synteny maps. Second, a few unique results

were later confirmed via BLASTZ, thus demonstrating that this method can be used to find syntenic blocks that seemingly higher-resolution tactics cannot. Third, this algorithm was extremely fast: it finished in one day of computing time on a single Pentium IV personal computer. However, this is not any sort of golden standard, and even the authors themselves confess that "novel methods may rival or ever better [this] method" in terms of computational speed. In particular, they acknowledge the inherent superiority of the program PatternHunter, which has significantly cut down on processing time by dropping BLAST's requirement that the seeds of default word size 11 consist of consecutive letters (Ma *et al.*, 2002).

My proposal

Liao *et al.*'s UM-based approach for synteny mapping was an insightful innovation *per se*, and ought not be dwarfed by PatternHunter's innovation of non-consecutive seeds. I propose that the two innovations be combined, in that UMs of non-consecutive letters be considered. The developers of PatternHunter arrived at optimal models for given seed weights (Ma *et al.*, 2002); similar analysis might reveal optimal models for UMs, which are the thematic equivalent to the seeds in BLAST and PatternHunter. If such a model increased the sensitivity of the UMps, higher values of the ratio M_{ij} would be realized in the statistically significant cases, thereby letting less syntenic blocks, particularly those of smaller size, slip through the cracks. Different models may also result in different chromosomal coverage densities; currently, some parts of the genomes are very sparsely spanned by UMps, and as such introduce bias into the computation. A more uniform distribution might iron out this particular bias.

Other variables associated with the UMs, such as length, should also be optimized. A length of 15 was used for UMs in synteny mapping mainly as a relic from prior projects, where a length of 15 proved beneficial. Longer UM lengths yield greater numbers of distinct UMs in a genome—in fact, quite expectedly, there were approximately four times as many 16-mer UMs in the draft human genome as there were 15-mer UMs (Chen *et al.*, 2002). Taken together, 16-mer UMs following a non-consecutive model could allow more efficient segregation of statistically significant matches when drawing up the anchoring islands. This would enable higher resolution mapping at little additional computation time.

I further propose that a scoring mechanism be developed, since widespread use of synteny mappings as a cornerstone of comparative genomics demands that the degree of syntenic homology be, in some way, quantified. It may not be possible to return a simple Smith-Waterman type alignment score, since an alignment cannot be run on scales this big (although the computational power of PatternHunter might be able to change that). Fortunately, there is precedent for the properties of such a score: its expectation (*P* value) would ideally follow an extreme value distribution (Levitt and Gerstein, 1998).

Once these proposals are enacted, synteny mapping would become less of an art and more of a powerful bioinformatics tool available to researchers. These potential innovations also immediately suggest further directions for the field of comparative genetics, including multiple maps (the syntenic equivalent to multiple alignments) and other advanced applications to better compare genomes.

Sources

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., Lipman, D. J. "Basic Local Alignment Search Tool." *J. Mol. Biol.* (1990) 215: 403-410.

Chen, L. Y. Y., Lu, S. H., Shih, E. S. C., Hwang M. J. "Single Nucleotide Polymorphism Mapping Using Genome-Wide Unique Sequences." *Genome Res.* (2002) 12: 1106-1111.

Gregory, S. G., Sekhon, M., Schein, J., Zhao, S., Osoegawa, K., Scott, C. E., Evans, R. S., Burridge, P. W., Cox, T. V., Fox, C. A., *et al*. "A physical map of the mouse genome." *Nature* (2002) 418: 743-750.

Levitt, M. and Gerstein, M. "A unified statistical framework for sequence comparison and structure comparison." *Proc. Natl. Acad. Sci.* (1998) 95: 5913-5920.

Liao, B. Y., Chang, Y. J., Ho, J. M., and Hwang, M. J. "The UniMarker (UM) method for synteny mapping of large genomes." *Bioinformatics* (2004) 20: 3156-3165.

Lipman, D. J. and Pearson, W. R. "Improved tools for biological sequence comparison." *Proc. Natl. Acad. Sci.* (1988) 85: 2444-2448.

Lipman, D. J. and Pearson, W. R. "Rapid and Sensitive Protein Similarity Searches." *Science* (1985) 227: 1435-1441.

Ma, B., Tromp, J., and Li, M. "PatternHunter: faster and more sensitive homology search." *Bioinformatics* (2002) 18: 440-445.

Needleman, S. B. and Wunsch, C. D. "A general method applicable to the search for similarities in the amino acid sequence of two proteins." *J. Mol. Biol.* (1970) 48: 443-453.

Smith, T. F. and Waterman, M. S. "Identification of Common Molecular Subsequences." *J. Mol. Biol.* (1981) 147: 195-197.