

Accelerating Sequence Alignments with the use of Suffix Trees

Katerina Perdikuri,¹ Athanasios Tsakalidis²

Keywords: sequence alignment, suffix trees

1 Introduction.

Sequence alignment is by far the most common task in Bioinformatics. Procedures relying on sequence comparison are diverse and range from database searches to secondary structure prediction.

BLAST and FASTA are two commonly used programs for similarity searching on biological sequences. Both tools employ heuristics to speed up their search. BLAST attempts to optimize a specific similarity measure. It permits a tradeoff between speed and sensitivity, with the setting of the “threshold” parameter T . In its original version [1], BLAST program, seeks short word pairs whose aligned score is at least T . Each such “hit” is then extended, to test whether it is contained within a high-scoring alignment. The extension step consumes most of the processing time (typically accounts for 90% of BLAST’s execution time). It is therefore desirable to reduce the number of extensions performed. PSI-BLAST, introduced the new “two-hit” method, which requires the existence of two non-overlapping word pairs on the same diagonal, and within a distance A , of one another, before an extension is invoked. Moreover PSI-BLAST puts on the ability to generate gapped alignments, using dynamic programming to extend a central pair of aligned residues in both directions. On the other hand, FASTA [2], extends non-overlapping hits that are found within a small distance on the same diagonal. The trade-off between speed and sensitivity is controlled by the *ktup* parameter, which specifies the size of a word. Accelerating a sequence alignment technique could be based upon the observation that a *High Segment Pair* of interest could be derived with the use of Suffix Trees and it is much longer in length than a single word pair, that BLAST or FASTA derives.

2 Sequence Alignment with Suffix Trees.

A Suffix Tree for an m -character string S is the compressed trie of all possible subwords starting within each suffix s_i of S . Suffix trees are useful for solving a wide variety of string based problems. The exact pattern matching problem can be solved in time proportional to the length of the query, once the suffix tree is built on the database string.

Based on this approach, various techniques that transform a DNA or amino acid database such that it becomes easier to search have been proposed in the relative literature. This route was taken by a group from IBM developing the FLASH program [3], which transforms the database into an index for storing the offsets of gapped k -tuples. In their work Heumann and Mewes [4] developed a data structure that allows for efficient mapping to disk to handle the problem that the size of a suffix tree is several times the magnitude of the original data which for a database means that it is unlikely to fit into main memory.

¹Department of Computer Engineering and Informatics, University of Patras, 26500 Patras, Greece. E-mail: perdikur@ceid.upatras.gr

²Research Academic Computer Technology Institute, 61 Riga Feraiou Str., 26221 Patras, Greece. E-mail: tsak@cti.gr

As already described the main goal in accelerating a sequence alignment technique is to reduce the extension time needed to create High Segment Pairs. Building the Suffix Tree for a given sequence s , we extract the common substrings among a query q and s above a predefined length l . Using as a guide those common substrings we extend them until the alignment score for each one aligned subword drops more than a constant c . The methodology looks similar to the original idea of BLAST, but the new idea is to start the extension process using as a basis longer common substrings. Some bioinformatics applications such as MUMmer [6] and OASIS [7] exploit suffix trees to efficiently evaluate queries on biological sequence datasets. Moreover in [5], Gusfield, introduces two different ways to combine suffix trees with dynamic programming to produce a *hybrid dynamic programming* method that is faster than dynamic programming alone. In more detail in [5], 12.4, the *P-against-all problem* and *threshold all-against-all problem* are discussed and the use of suffix trees reduces the computational task of two large-scale *alignment problems*.

3 Future Research.

The growing number of species that have been sequenced, imposes the development of efficient and accurate whole-genome alignment programs. Suffix trees are versatile data structures that can help efficiently querying large string datasets. Although, suffix trees are not widely used because of their high cost of construction, in recent years a large focus has been on new algorithms for fast *disk-based* suffix tree construction [8], [9]. If we are able to efficiently construct a suffix tree on the entire human genome (an input string of 3 billion symbols) we could use a Generalized Suffix Tree on a set of genome sequences to discover the evolutionary relationships and functions of thousands of proteins from hundreds of different species, simply by reporting longest common substrings from the set of input genome sequences.

References

- [1] Altschul, S., Gish, W., Miller, W., Myers, E., Lipman, D. J. 1990. A basic local alignment search tool. *J. Mol. Biol.* 215:403–410, 1990.
- [2] Pearson, W. 2000. Flexible sequence similarity searching with the FASTA3 program package. *Methods Mol. Biol.* 132, 185–219.
- [3] A. Califano, I. Rigoutsos. FLASH: A Fast look-up algorithm for string homology. *In Proceedings of the 1st International Conference on Intelligent Systems in Molecular Biology*, 56–64.
- [4] K. Heumann, H.W. Mewes. The Hashed Position Tree (HPT): a suffix tree variant for large data sets stored on slow mass storage devices. *In the Proc. of the 3rd South American Workshop on String Processing*, Carleton University Press, Ottawa 101–114.
- [5] Gusfield, D. 1997. *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*. Cambridge University Press, New York, 1997.
- [6] Kurtz, S., Phillippy A., Delcher A., Smoot M., Shumway, M., Antonescu, C., and Salzberg, S. 2004. Versatile and Open Software for Computing Large Genomes. *Genome Biology*, 5(R12).
- [7] Meek, C., Patel, J.M., and Kassetly, S., 2003. OASIS: An Online and Accurate Technique for Local-alignment Searches on Biological Sequences. *In VLDB*.
- [8] Hunt, E., Atkinson M.P., Irving, R.W., 2001. A Database Index to Large Biological Sequences. *In Proceedings of the 27th VLDB Conference*.
- [9] Tata, S., Hankins R.A., Patel, J.M., 2004. Practical Suffix Tree Construction. *In Proceedings of the 30th VLDB Conference*.