How to develop and implement efficient high-performance, local sequence alignment algorithms?

Problem

- Sequence alignment: one of the most commonly used tools (i.e., by computational and molecular biologists)
- Fast heuristic algorithms like BLAST do not replace the more rigorous, slower Smith-Waterman algorithm
- The rigorous algorithms are time-limited which is an issue with the sustained exponential genomic data growth
- Unique ASsociative Computing (ASC) model has search capabilities well suited for the Smith-Waterman algorithm

Smith-Waterman: Exact Local Sequence Alignment

Background

- Optimization problem: find best local alignment, i.e., the substrings with the greatest similarity
- Smith-Waterman uses the Dynamic Programming technique, returns only one, highest rated local alignment
- BLAST and FASTA are fast, provide multiple results, but as heuristics the result may not be the best alignment
- SWAMP is the Smith-Waterman algorithm for the parallel Associative MultiProcessor, ASC

ASsociative Computing (ASC) Data-Parallel Multi-Processor: Massive Parallelism

ASC Model

- Based on generalized associative data-parallel computing style, such as Goodyear Aerospace’s STARAN
- 100’s or 1000’s of processing elements (PEs) each with own local memory, a single instruction stream (IS), a broadcast/reduction network, and an optional cell interconnection network (See poster layout for components)
- ASC locates data responders (searches) based on content, not the memory location
- Constant time search/respond operations and constant time global reduction operations like max/min
- The data is configured in a tabular fashion

SWAMP: Smith-Waterman on ASC Multi-Processor

Work

- Reduced running time: original \( rm \) time reduced by a factor of \( m \) with \( r \) PEs where \( m \), \( n \), length of compared sequences
- Future work includes: SWAMP+ to return multiple local, non-overlapping and non-intersecting regions of similarity utilizing ASC’s fast parallel searching capabilities
- Possibly extend the search for multiple regions to identify regulatory regions and responder elements
- Web interface for running SWAMP and SWAMP+ remotely over the Internet

Algorithms and the Computational Model

Conclusions

- Parallel SWAMP algorithm has a greatly improved running time compared to sequential versions
- ASC model supports extension for SWAMP+ to return multiple regions of similarity
- Data-parallel model uses a simple programming paradigm with no explicit synchronization or task management
- Commercial-off-the-shelf (COTS) data-parallel hardware is becoming more prevalent (GPUs / Cell Processor in PS3 / PCI "SIMD on a board" / etc.) that can emulate ASC’s associative functions therefore ASC algorithms

Figure 1: Proteins with three local non-intersecting alignments. SWAMP will find the single best local alignment (i.e., the two segments shown in yellow). SWAMP+ will return at most the top \( k \) non-intersecting, non-overlapping segments, or in this case all three local alignments.

Figure 2: NVIDIA GeForce 8800 GTX Graphics Processing Unit (GPU) with 128 stream processors running at 1.35 GHz

Figure 3a: Cell Broadband Engine running at 3.2 GHz Sony PlayStation 3

Figure 4: WorldScape Dual 64 PCI SIMD board with 50 GFLOPS performance