



Scalable DNA similarity detection: A case study of OpenMP vs. CUDA

George L. Coulouris <coulouri@ncbi.nlm.nih.gov>

National Center for Biotechnology Information

Bethesda, MD 20894



Introduction

DNA sequence alignment, ubiquitous in computational biology, is an example of a needle-in-a-haystack problem. Fortunately, heuristics exist that can exploit the structure of the problem to prune the search space. Unfortunately, however, employing such heuristics can result in irregular data structures, random memory access patterns, and branchy execution paths, which hinder scalability on parallel architectures. We describe the design of a DNA sequence similarity detection algorithm that exhibits regular memory access patterns and high memory bandwidth utilization. We then compare OpenMP and CUDA implementations of the algorithm.

Algorithm

Given a word size k and block size n , the algorithm proceeds as follows. Divide a nucleotide query sequence Q into x non-overlapping subsequences of length n base pairs, yielding $\{Q_1, Q_2, \dots, Q_x\}$. Similarly, divide a nucleotide subject sequence S into y non-overlapping subsequences of length n base pairs, yielding $\{S_1, S_2, \dots, S_y\}$. Let q_i be the set of unique k -mers that occur in Q_i , $1 \leq i \leq x$. Similarly, let s_j be the set of unique k -mers that occur in S_j , $1 \leq j \leq y$. Finally, let $P_{ij} = |q_i \cap s_j|$ be the number of unique k -mers in common between Q_i and S_j .

Properties

The set of unique k -mers corresponding to a subsequence can be represented compactly by a bit vector. When q_i and s_j are implemented in this fashion, computing P_{ij} reduces to counting the number of bits set in $q_i \wedge s_j$. Since contemporary computer architectures offer hardware support for counting the number of bits set, this quantity can be computed efficiently. Furthermore, each P_{ij} depends only upon q_i and s_j , so all xy such computations can proceed in parallel. Before proceeding, however, appropriate values for the parameters k and n must be chosen.

Parameter selection

Since the goal is to detect regions of similarity, k and n were chosen to maximize the correlation between P_{ij} and the Smith-Waterman score for Q_i and S_j . The optimal parameters of $k = 9$ and $n = 2254$ were obtained with discrete SPSA using Monte Carlo random sequences in the MATLAB Bioinformatics Toolbox. The results were then validated with real sequences.

Implementation

For ease of implementation, the near-optimal values of $k = 8$ and $n = 2048$ were ultimately chosen as algorithm parameters. To test scalability, the algorithm was implemented for CPU and GPU architectures. In both cases, random bit vectors were generated with $x = y = 512$. Since $n=2048$, this corresponds to artificial sequences of 1048576 base pairs each.

Methodology

Since the size of the problem is fixed and the size of the bit vectors is known, it is possible to compute the effective memory bandwidth consumed by the algorithm.

CPU tests were conducted on a dual-quadcore Intel Xeon X5550 and were parallelized using OpenMP on icc 11. The maximum memory bandwidth of the X5550 is 32 GB/s.

GPU tests were conducted on a variety of Nvidia GPU hardware and were parallelized using CUDA. For ease of comparison, the number of stream processors (SPs) for each GPU was normalized. The maximum memory bandwidth of the Tesla C1060 is 102 GB/s.

Discussion

Hardware population count instructions were used to count the number of bits set on both CPU and GPU architectures. In both cases, the algorithm scaled well and was able to effectively utilize the available memory bandwidth.

Acknowledgements

Thanks to Kenneth C. Dyke, Jason Papadopoulos, and Sumit Gupta for testing on various GPU configurations. This research was supported in part by the Intramural Research Program of the NIH.

Results

CPU cores	Memory bandwidth (GB/s)	Bandwidth ratio
1	7.00	1.00
2	13.94	1.99
4	27.31	3.90
8	28.01	4.00

GPU	# SPs	# SPs (normalized)	Memory bandwidth (GB/s)	Bandwidth ratio
GeForce 9400M	16	1	4.54	1.00
GeForce 9600M GT	32	2	8.23	1.812
GeForce 9800GT	112	7	28.36	6.25
Tesla C1060	240	15	67.45	14.86

References

- S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Res*, 25(17):3389–3402, September 1997.
- R. C. Edgar. Local homology recognition and distance measures in linear time using compressed amino acid alphabets. *Nucleic Acids Res*, 32(1):380–385, 2004.
- Z. Vágó L Gerencsér, S. D. Hill. Optimization over discrete sets via SPSA. In *Winter Simulation Conference*, pages 466–470, 1999.
- T. F. Smith and M.S. Waterman. Identification of common molecular subsequences. *J Mol Biol*, 147(1):195–197, 1981.
- J. C. Spall. *Introduction to Stochastic Search and Optimization*. John Wiley & Sons, Inc., New York, NY, USA, 2003.
- D. Wagner and S. M. Bellovin. *A programmable plaintext recognizer*, 1994