Darwin-WGA: A Co-processor Provides Increased Sensitivity in Whole Genome Alignments with High Speedup

Yatish Turakhia1, Sneha D. Goenka, Gill Bejerano1, William J. Dally1,2
1 Stanford University, 2 NVIDIA Research

Whole Genome Alignment (WGA)

WGA is the computational process of finding pairwise local alignments in the entire genomes of two or more species. It helps:
1. infer the evolutionary relationship between species
2. identify and predict functional elements, such as genes and regulatory sequences

Motivation for Acceleration
- Several projects look at sequencing and characterizing the whole of life – Earth BioGenome, Earth Microbiome, B10K, G10K
- Thousands of species will be sequenced over the next decade => millions of pairwise WGA
- Overwhelming computational cost - single human-mouse WGA requires ~1,000 CPU hours

Motivation for Gapped filtering
- Current software whole genome aligners do not capture several biologically-meaningful, high-scoring alignment regions [1]
- All software aligners are based on the popular seed-and-extend paradigm (motivated by BLAST)

1. Seeding – find a pattern of matches between sequences
   - Seed: 11101001100101111 GTAGCGGCACACTCCTT ||| X ||| X ||| X ||| X ||| GACTGCCTAGACCTT
2. Ungapped filtering
3. Gapped extension through dynamic programming

- Distantly related species have more frequent indels, which increase the need for gapped extension step which slows down the current software WGA by 100x

Darwin – WGA Algorithm Overview

Step 1 – Seeding and D-SOFT
- Reference bin and Query chunk determines diagonal band
- Each seed hit falls in a single diagonal band
- At most 1 seed hit per diagonal band is extended

Step 2 – Gapped extension filtering
- The seed is extended using banded Smith-Waterman
- If the maximum score is greater than a threshold, the maximum score position is the anchor or the starting point for alignment extension using GACT-X (see below)

Step 3 – GACT-X extension
- Anchor is extended in the left and the right directions
- Extension occurs in tiled fashion (like GACT)
- Traceback pointer is used or discarded depending on the overlap region
- Each new tile starts from the position the previous traceback completed or overlap boundary

Hardware Architecture (Step 2 and 3)

- PEs compute the cell scores and global maximum in 1 cycle in a systolic fashion – diagonal wavefront, in the region with score > (max - y_drop)
- Direction pointers stored in BRAM sequentially, start/stop position pointers in another BRAM
- This novel architecture considering Y-drop uses small constant memory to align arbitrarily long sequences
- Banded Smith-Waterman does not use traceback memory

Results

Ordered local alignments in both species chained together using AXTCHAIN [3].

False Positive Rate Analysis
- To measure false positive rate, we simulated a random genome with same chromosome sizes and nucleotide composition as cb4 and aligned and chained to the ce11 genome
- False positive rate is measured as G_f/G_t, where G_f is the number of matching base-pairs in chain generated on the simulated random genome and G_t is the equivalent for the cb4 genome
- We find G_f = 0bp and G_t = 140Mbp, implying a negligibly small false positive rate
- With the gapped threshold lowered to 3000 (same as LASTZ), the false positive rate goes up to 0.1%

Architectural Setup
- FPGA Architecture: Xilinx Virtex UltraScale+
- Banded Smith-Waterman: 16PEs/array, 48 arrays
- GACT-X: 16 PE/Array, 1MB traceback memory per array, 8 arrays
- D-SOFT implemented in software

Performance
- 20X speedup comparison to single-thread LASTZ
- 2000X speedup as compared to gapped filtering software

Motivation for New Algorithms
- Whole genome alignments contain several long gaps - current constant memory hardware acceleration algorithms (GACT) [2] require very large on-chip memory (~4-10MB) for optimal results
- Co-design Hardware with algorithm to improve sensitivity at high speed and small constant on-chip memory requirement (1MB)