UCL DEPARTMENT OF
COMPUTER SCIENCE

Research Note

RN/13/03

Which is faster: Bowtie2^{GP} > Bowtie > Bowtie2 > BWA

21 January 2013

W. B. Langdon

Abstract

We have recently used genetic programming to automatically generate an improved version of Langmead's DNA read alignment tool Bowtie2 [Langdon and Harman, 2012, Sect. 5.3]. We find it runs more than four times faster than the Bioinformatics sequencing tool (BWA) currently used with short next generation paired end DNA sequences by the Cancer Institute, takes less memory and yet finds similar matches in the human genome.

Keywords: double-ended DNA sequence, Solexa nextgen NGS, sequence query, Smith-Waterman, Bowtie2GP, fuzzy string matching, Homo sapiens genome reference consortium GRCh37.p5 h_sapiens_37.5_asm, IP29.

1 Introduction

As part of the Gismo project we have used search based software engineering to automatically tailor a version of the DNA look up tool Bowtie2 [Langmead and Salzberg, 2012] which runs considerably faster than the original released code on "single ended" short (36 bp) DNA sequences produced by the Broad Institute's Illumina Genome Analyzer II Solexa scanner. The multi-objective goals of Bowtie2^{GP} were to find matches in the human genome faster without unduly sacrificing the quality of the matches[1]. On out-of-sample Solexa sequences on average it runs more than 70 times faster than the original release of Bowtie2 and finds very slightly better matches [Langdon and Harman, 2012].

While we would normally advocate re-optimising the Bowtie2 C++ code for new circumstances, in order to ease the wide spread up take of Bowtie2^{GP}, we show the original optimised version can also process DNA sequences from other sources by applying it to "double ended" short DNA sequence used by the Cancer Institute for human blood studies.

Although the program is identical, "double ended" sequences require Bowtie2^{GP} to combine the results of looking up two DNA sequences (one from each end of the sequence). Naturally this combination code was not optimised when using the Broad Institute's "single ended" data. Nevertheless Bowtie2^{GP} is able to find high quality matches and retains some speed advantage over the original released version of Bowtie2. Indeed Bowtie2^{GP} on an ACER aspire 5742 laptop is able to beat BWA [Li and Durbin, 2010] on our 3 GHz 32 GB server.

There are many Bioinformatics computer based sequencing tools. In January 2013, Wikipedia alone listed more than 140. [Fonseca *et al.*, 2012] considered 60 of them. Bowtie is one of the most widely used and cited (on average 485 citations per annum[2]). Langmead rewrote it in C++ to give Bowtie2 (first released 16^{th} October 2011). However BWA is also well respected (108 cites pa) and is used by the Cancer Institute. We compare these three human written DNA sequence tools with Bowtie2^{GP} specifically for the Cancer Institute's own data. For completeness we would have liked to compare against BLAST [Altschul *et al.*, 1997] (44 454 cites), which is often taken as the "gold standard" for Bioinformatics sequence matching, however it cannot deal with paired end data and, as we shall see in the next section, even treating each end of each DNA sequence pair separately, it is far too slow for normal use with nextGen sequences.

2 Method

We selected uniformly at random one million pairs from the 38 722 867 produced by the scanner. (All the pairs have a 36 DNA base sequence at each end.) We then ran each program (with default parameters to generate SAM output) on the sample three times on our 32 gigabyte Linux server. To allow ease of comparison only a single server CPU core was used. To check for variability this whole procedure was also repeated three times.

In a similar way we have also tested BLAST (version blastn 2.2.25+) by running it on a random sample of 1000 DNA sequences. However it was timed out by a 10 minute CPU limit that we imposed. (The modern alignment tools can process more than 100 times as many sequences within ten minutes. See Table 1.) Hence Tables 1 and 2 refer only to normal paired end runs with BWA, Bowtie, Bowtie2 and Bowtie2^{GP}.

3 Results

BWA finds more matches than the other three tools (Table 1, column "% pairs matched"). However the difference between BWA and Bowtie2 is only 0.2% and BWA takes more than three times as long. The fastest program is Bowtie but it is almost the same speed as Bowtie2^{GP} and find 5-6% fewer matches than the other tools. Bowtie2^{GP} and Bowtie2 produce very similar matches but Bowtie2^{GP} is 26% faster.

[1] The ^{GP} suffix denotes Bowtie2 was optimised by genetic programming [Poli *et al.*, 2008].

[2] Citation counts from Google Scholar 14 January 2013

Table 1: Mean CPU time taken to process a million paired-end reads randomly chosen from the 38 722 867 supplied against the human genome (NCBI release 37 patch p5). ($\pm$ is shows the observed standard deviation over the 3×3 runs.) The fourth column is the percentage of DNA sequences where the tool reported a suitable match for both ends. The next pair of columns were calculated by randomly taking 1000 of each of the three large samples of paired end reads and where the tool reports a match calculating the Smith-Waterman score for both ends. This is normalised by summing the scores and dividing through by the maximum possible score (72) and expressing this as a percentage. (With the usual parameters, i.e. $\mu = 0.33$ and $\delta = 1.33$, a single mismatch at one end corresponds to a normalised score of 98.2).

Tool	CPU secs	% pairs matched	Normalised Smith-Waterman score	RAM memory
BWA	2140 ± 55	83.1 ± 0.01	98.4 ± 3.3	5.3 GBytes
Bowtie	490 ± 12	77.2 ± 0.01	98.7 ± 1.9	2.9 GBytes
Bowtie2	630 ± 17	82.9 ± 0.02	98.4 ± 2.6	2.2 GBytes
Bowtie2^{GP}	500 ± 17	82.1 ± 0.02	98.5 ± 2.5	2.2 GBytes

Table 2: Results of statistical comparisons on a random sample of 3000 paired end DNA sequences ($p = 0.05$, sign test, $\cdot$ indicates difference is not significant). Left more or better matches. Right comparison of match quality where both tools report a match. BWA finds more or better matches. Whilst Bowtie finds fewer matches but they are of the same quality as those also reported by Bowtie2 or Bowtie2^{GP}.

more matches	Bowtie	Bowtie2	Bowtie2^{GP}
BWA	Yes	Yes	Yes
Bowtie		No	No
Bowtie2			$\cdot$

better matches	Bowtie	Bowtie2	Bowtie2^{GP}
BWA	Yes	Yes	Yes
Bowtie		$\cdot$	$\cdot$
Bowtie2			$\cdot$

4 Discussion

Although we do not see the fabulous speed up we get when our own variant of Bowtie2, Bowtie2^{GP}, is used in the way it was optimised for, it does performs well on paired end DNA sequence data. Although Bowtie2^{GP} found marginally fewer matches but higher quality matches than Bowtie2, the differences were not significant in a sample of 3000 paired end reads (see Table 2).

5 Conclusions

BWA is currently in use by UCL's Cancer Institute. However on typical data it is **more than four times slower** than Bowtie2^{GP} and yields only 1% more valid matches, see Table 1.

Bowtie2^{GP} is effectively the same speed as Bowtie and yet finds matches in the human genome in 5% more cases. That is, although Bowtie2 was written to give additional functionality over Bowtie at the expense of run time, by optimising Bowtie2 to give Bowtie2^{GP}, we have recovered the lost speed and retained the additional functionality. (Bowtie/Bowtie2^{GP} are the fastest of the five tools tried. BLAST is by the far slowest, data not shown.) On the Cancer Institute's paired end DNA sequence data Bowtie2^{GP} is 26% faster than Bowtie2 from which it was derived.

Acknowledgements

I would like to thank Gareth Wilson of the Cancer Institute.

Funded by EPSRC grant EP/I033688/1.

References

[Altschul *et al.*, 1997] Stephen F. Altschul, Thomas L. Madden, Alejandro A. Schaffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J. Lipman. Gapped BLAST and PSI-BLAST a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402, 1997.

[Fonseca *et al.*, 2012] Nuno A. Fonseca, Johan Rung, Alvis Brazma, and John C. Marioni. Tools for mapping high-throughput sequencing data. *Bioinformatics*, 28(24):3169–3177, 2012.

[Langdon and Harman, 2012] William B. Langdon and Mark Harman. Genetically improving 50000 lines of C++. Research Note RN/12/09, Department of Computer Science, University College London, Gower Street, London WC1E 6BT, UK, 19 September 2012.

[Langmead and Salzberg, 2012] Ben Langmead and Steven L Salzberg. Fast gapped-read alignment with Bowtie 2. *Nature Methods*, 9(4):357–359, 4 March 2012.

[Li and Durbin, 2010] Heng Li and Richard Durbin. Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics*, 26(5):589–595, 2010.

[Poli *et al.*, 2008] Riccardo Poli, William B. Langdon, and Nicholas Freitag McPhee. *A field guide to genetic programming*. Published via `http://lulu.com` and freely available at `http://www.gp-field-guide.org.uk`, 2008. (With contributions by J. R. Koza).

A Software Versions Used

- BWA 0.6.2-r131
- Bowtie 0.12.7
- Bowtie 2 2.0.0-beta2
- Bowtie2^{GP} 2.0.0-beta2 updated by 7 line patch as described in technical report [Langdon and Harman, 2012]. Available via FTP.

B Seven Line Change to Bowtie2 (2.0.0-beta2)

Source file	line	Original Code	New Code
bt2_io.cpp	622	`i < offsLenSampled`	`i < this->_nPat`
sa_rescomb.cpp	50	`i < satup_->offs.size()`	`0`
sa_rescomb.cpp	69	`j < satup_->offs.size()`	
aligner_swsse_ee_u8.cpp	707	`vh = _mm_max_epu8(vh, vf);`	`vmax = vlo;`
aligner_swsse_ee_u8.cpp	766	`pvFStore += 4;`	
aligner_swsse_ee_u8.cpp	772	`_mm_store_si128(pvHStore, vh);`	`vh = _mm_max_epu8(vh, vf);`
aligner_swsse_ee_u8.cpp	778	`ve = _mm_max_epu8(ve, vh);`	

Adapted from [Langdon and Harman, 2012, Figure 16].