Supplementary Information for "Vargas: heuristic-free alignment for assessing linear and graph read aligners"

Charlotte A. Darby, Ravi Gaddipati,
Michael C. Schatz, and Ben Langmead

Datasets used

100bp reads from the 1000 Genomes Project sample NA18505, SRA accession ERR239486 are available at `ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/phase3/data/NA18505/sequence\_read/ERR239486\_1.filt.fastq.gz`. We use the first 100,000 reads.

250bp reads from the 1000 Genomes Project sample NA19017, SRA accession SRR1295544. We used the first 100,000 reads accessed using sratoolkit fastq-dump (and analyzed Read 1 only since this is a paired-end library).

Vargas memory consumption

The only sizable in-memory data structure used by Vargas is the reference genome. For the linear reference genome GRCh38, the representation consists of the string itself with one node per contig (199 total). On disk it is 2.9 Gb and alignment uses maximum 3.16 Gb. A graph genome requires a more complex representation, including the nodes and edges that represent various choices of alleles. The graph for GRCh38 plus all 1000 Genomes variants is 230 million nodes and 17 Gb on disk. When the entire graph is loaded into memory, alignment uses maximum 101.3 Gb, although this could be reduced by changing the implementation to process the graph by chromosome or smaller chunks.

Trendline calculation

On all plots, the solid line shown for visualization purposes is fitted to the scatterplot using R's `geom_smooth` function with Loess smoothing and smoothing parameter 0.5. Note that the trendline is **not weighted** by the number of reads in each mapping quality or alignment score bin.

Vargas alignment parameters

Bowtie 2 semiglobal and HISAT2 `--ete --ma 0 --mp 2,6 --np 1 --rdg 5,3 --rfg 5,3`

Bowtie 2 local `--ma 2 --mp 2,6 --np 1 --rdg 5,3 --rfg 5,3`

BWA-MEM and vg `--ma 1 --mp 4,4 --np 1 --rdg 6,1 --rfg 6,1`

BWA aln `--ete --ma 0 --mp 3,3 --np 3 --rdg 11,4 --rfg 11,4`

				All reads			Highest-scoring 95%			Lowest-scoring 5%		
Tool	Mode	Time (s)	Mem. (GB)	% U	% I	% C	% U	% I	% C	% U	% I	% C
Bowtie 2 [5] semiglobal	very-sensitive	100.89	3.38	1.03	0.63	98.35	0.01	0.15	99.84	20.59	12.09	67.32
	sensitive (default)	48.93	3.37	1.17	0.71	98.12	0.01	0.21	99.78	23.38	13.24	63.38
	fast	43.35	3.37	1.37	0.84	97.80	0.05	0.27	99.69	26.75	15.78	57.47
	very-fast	34.81	3.37	1.74	0.89	97.37	0.14	0.34	99.52	32.59	16.40	51.01
Bowtie 2 [5] local	very-sensitive	85.42	3.38	0.32	1.08	98.60	0.00	0.18	99.82	6.51	19.64	73.85
	sensitive (default)	64.28	3.38	0.33	1.26	98.41	0.00	0.26	99.74	6.71	21.84	71.44
	fast	38.32	3.37	0.38	1.52	98.11	0.00	0.32	99.68	7.65	26.51	65.84
	very-fast	32.74	3.37	0.49	1.85	97.66	0.01	0.48	99.52	9.84	31.29	58.87
BWA-MEM [6]	k16 r1.2	87.51	5.51	0.35	0.43	99.23	0.00	0.04	99.96	7.05	8.52	84.43
	k19 r1.5 (default)	59.73	5.49	0.35	0.46	99.19	0.00	0.03	99.97	7.11	9.27	83.63
	k22 r3	42.81	5.47	0.36	0.60	99.04	0.00	0.04	99.96	7.31	12.17	80.52
	k25 r4	39.40	5.46	0.38	0.67	98.95	0.00	0.05	99.95	7.61	13.50	78.88
BWA aln [7]	o5 n15	245.67	4.25	1.38	0.42	98.21	0.01	0.03	99.97	28.24	11.11	60.65
	o3 n10	253.87	3.87	1.63	0.33	98.03	0.01	0.02	99.97	33.48	9.50	57.01
	o1 n5 (default)	62.69	3.19	2.70	0.15	97.15	0.02	0.02	99.97	55.42	6.21	38.37
HISAT2 [4] linear	very-sensitive	44.09	4.47	0.64	1.38	97.98	0.69	1.48	97.83	12.16	24.82	63.01
	sensitive	25.43	4.47	1.56	0.96	97.48	1.56	1.04	97.40	29.66	17.55	52.79
	fast (default)	17.50	4.47	3.66	0.41	95.93	3.46	0.51	96.03	67.99	3.47	28.53
HISAT2 [4] graph	very-sensitive	52.73	6.73	0.69	1.48	97.83	0.08	0.30	99.62	12.44	27.47	60.09
	sensitive	58.06	6.73	1.56	1.04	97.40	0.12	0.32	99.56	29.34	20.59	50.07
	fast (default)	32.97	6.72	3.46	0.51	96.03	0.24	0.34	99.41	65.35	9.88	24.77
vg [2]	linear	397.20	24.16	0.19	0.56	99.25	0.00	0.04	99.96	3.81	11.01	85.18
	graph	381.63	26.58	0.18	0.52	99.29	0.00	0.03	99.97	4.19	11.59	84.22

Table S1: Alignment and correctness for the 100,000 100bp reads. Reported runtime is the median of three consecutive trials, and reported memory usage is the maximum memory footprint during alignment. U = unaligned; I = incorrect-by-score; C = correct-by-score. Time for bwa aln is reported for 'aln' only, not 'samse' which converts the intermediate output into SAM format. Related to Section 3.2.

				All reads			Highest-scoring 95%			Lowest-scoring 5%		
Tool	Mode	Time (s)	Mem. (GB)	% U	% I	% C	% U	% I	% C	% U	% I	% C
Bowtie 2 [5] semiglobal	very-sensitive	193.59	3.38	5.01	1.18	93.81	1.22	1.09	97.69	77.13	3.01	19.86
	sensitive (default)	97.23	3.37	6.11	1.39	92.50	2.25	1.29	96.46	79.56	3.47	16.97
	fast	69.15	3.37	7.42	1.89	90.70	3.50	1.79	94.71	81.92	3.73	14.35
	very-fast	47.98	3.37	8.38	2.45	89.17	4.40	2.40	93.20	84.17	3.49	12.35
Bowtie 2 [5] local	very-sensitive	237.80	3.40	2.14	2.26	95.60	0.54	1.15	98.31	32.93	23.71	43.35
	sensitive (default)	177.94	3.39	2.44	2.60	94.96	0.73	1.44	97.83	35.45	24.87	39.69
	fast	126.44	3.38	3.83	3.03	93.14	1.72	1.98	96.30	44.66	23.25	32.09
	very-fast	109.29	3.38	4.79	3.84	91.37	2.46	2.83	94.71	49.62	23.29	27.08
BWA-MEM [6]	k16 r1.2	237.04	5.55	1.67	0.72	97.61	0.00	0.41	99.59	33.73	6.84	59.43
	k19 r1.5 (default)	119.74	5.45	1.84	0.86	97.30	0.01	0.46	99.53	36.92	8.64	54.44
	k22 r3	97.70	5.43	2.27	1.11	96.63	0.60	0.60	98.80	43.88	10.82	45.30
	k25 r4	89.74	5.43	2.78	1.23	95.99	0.26	0.70	99.04	51.12	11.32	37.56
HISAT2 [4] linear	very-sensitive	124.43	4.48	3.46	3.53	93.01	1.36	2.46	96.18	43.33	23.99	32.69
	sensitive	61.76	4.48	6.95	2.82	90.23	2.37	2.92	94.71	94.05	0.88	5.07
	fast (default)	20.98	4.47	17.40	1.52	81.08	13.06	1.60	85.34	100.00	0.00	0.00
HISAT2 [4] graph	very-sensitive	171.59	6.74	3.47	4.20	92.33	1.36	3.11	95.53	43.66	25.06	31.28
	sensitive	96.25	6.73	6.96	3.34	89.70	2.38	3.47	94.15	94.14	0.94	4.92
	fast (default)	35.03	6.72	17.21	1.79	81.01	12.87	1.88	85.25	100.00	0.00	0.00
vg [2]	linear	633.29	24.15	1.31	1.29	97.41	0.06	0.44	99.50	25.34	17.47	57.19
	graph	692.07	26.55	1.29	1.29	97.42	0.06	0.44	99.51	24.79	17.60	57.61

Table S2: Alignment and correctness for the 100,000 250bp reads. Reported runtime is the median of three consecutive trials, and reported memory usage is the maximum memory footprint during alignment. U = unaligned; I = incorrect-by-score; C = correct-by-score. Related to Section 3.2.

Tool	Reference	Availability	Type	Graph?
Vargas	this work	`https://github.com/langmead-lab/vargas`	Software	DAG
PaSGAL	[3]	`https://github.com/ParBLiSS/PaSGAL`	Software	DAG
vg align	[2]	`https://github.com/vgteam/vg`	Software	DAG
GraphAligner	[9]	`https://github.com/maickrau/GraphAligner`	Software	Any graph
SSW	[10]	`https://github.com/mengyao/Complete-Striped-Smith-Waterman-Library`	Library	no
GSSW		`https://github.com/vgteam/gssw`	Library	DAG
seqan	[8]	`https://github.com/seqan/seqan3`	Library	no
Parasail	[1]	`https://github.com/jeffdaily/parasail`	Library	no

Tool	SIMD instructions	Vectorization strategy	Local?	Semiglobal?	Scoring function
Vargas	SSE, AVX2, AVX512BW	query-parallel	yes	yes	match/baseq-scaled mismatch affine insertion/affine deletion
PaSGAL	AVX512BW	query-parallel	yes	no	match/mismatch/insertion/deletion
vg align	SSE	striped	yes	no	match/mismatch/affine gap
GraphAligner	no	bit-parallel	no	yes	unit cost
SSW	SSE	striped	yes	no	match/mismatch/affine gap
GSSW	SSE	striped	yes	no	match/mismatch/affine gap
seqan	SSE, AVX2, AVX512BW	query-parallel	yes	yes	match/mismatch/affine gap
Parasail	SSE, AVX2	diagonal, blocked, striped, prefix-scan	yes	no	match/mismatch/affine gap

Table S3: Summary of exact dynamic programming pairwise alignment algorithms available in the literature and their features. Related to Section 3.1.

References

[1] Daily, J. (2016). Parasail: SIMD C library for global, semi-global, and local pairwise sequence alignments. *BMC Bioinformatics*, **17**(1), 81.

[2] Garrison, E. *et al.* (2018). Variation graph toolkit improves read mapping by representing genetic variation in the reference. *Nature Biotechnology*, **36**(9), 875–879.

[3] Jain, C. *et al.* (2019). Accelerating Sequence Alignment to Graphs. In *2019 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 451–461. IEEE.

[4] Kim, D. *et al.* (2019). Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype. *Nature Biotechnology*, **37**(8), 907–915.

[5] Langmead, B. and Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. *Nature Methods*, **9**(4), 357–359.

[6] Li, H. (2013). Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM.

[7] Li, H. *et al.* (2008). Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome research*, **18**(11), 1851–8.

[8] Rahn, R. *et al.* (2018). Generic accelerated sequence alignment in SeqAn using vectorization and multithreading. *Bioinformatics*, **34**(20), 3437–3445.

[9] Rautiainen, M. *et al.* (2019). Bit-parallel sequence-to-graph alignment. *Bioinformatics*.

[10] Zhao, M. *et al.* (2013). SSW Library: An SIMD Smith-Waterman C/C++ Library for Use in Genomic Applications. *PLoS ONE*, **8**(12), e82138.

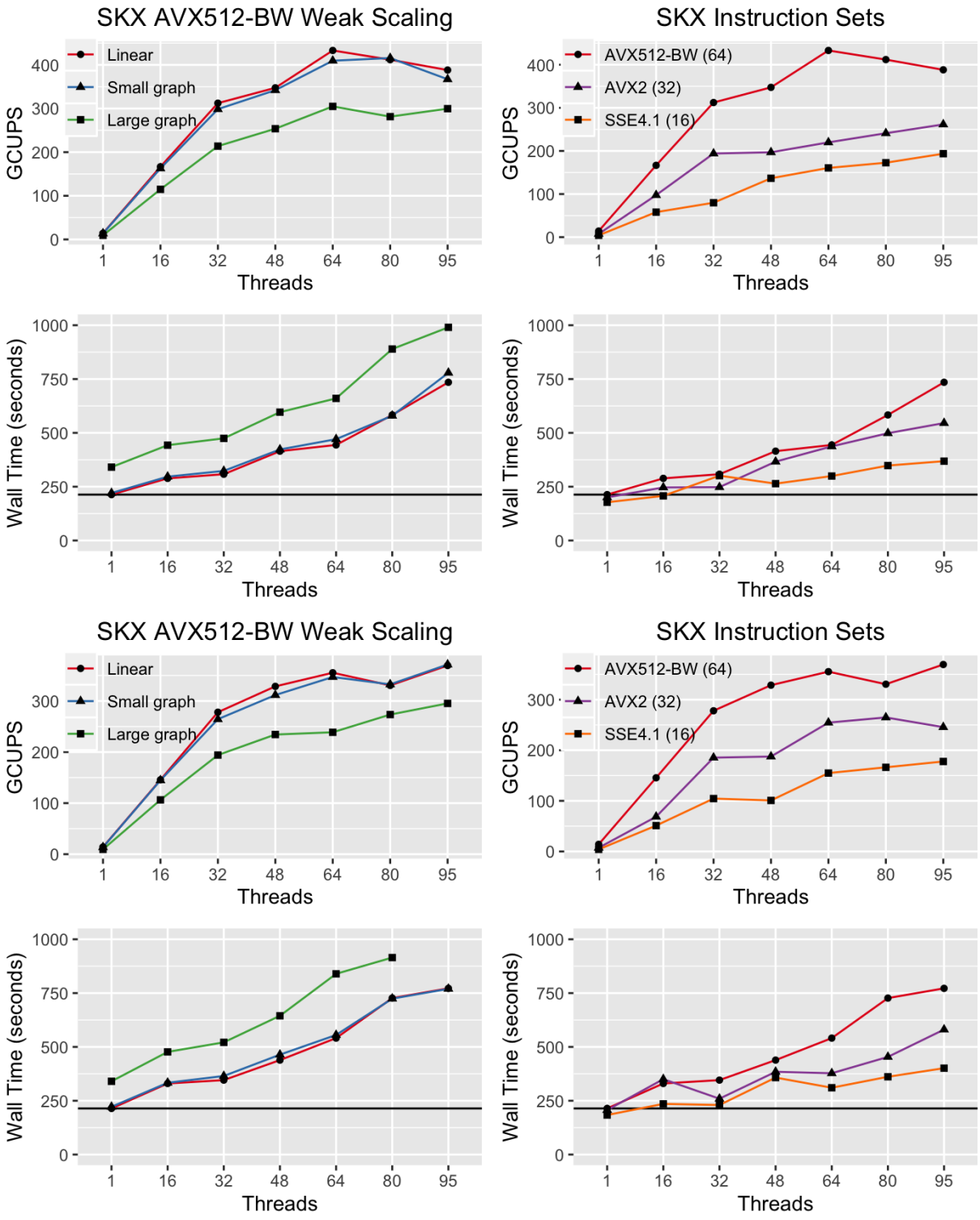

Figure S1: Weak scaling for Skylake (SKX), semiglobal alignment (top four) and local alignment (bottom four). Related to Figure 2.

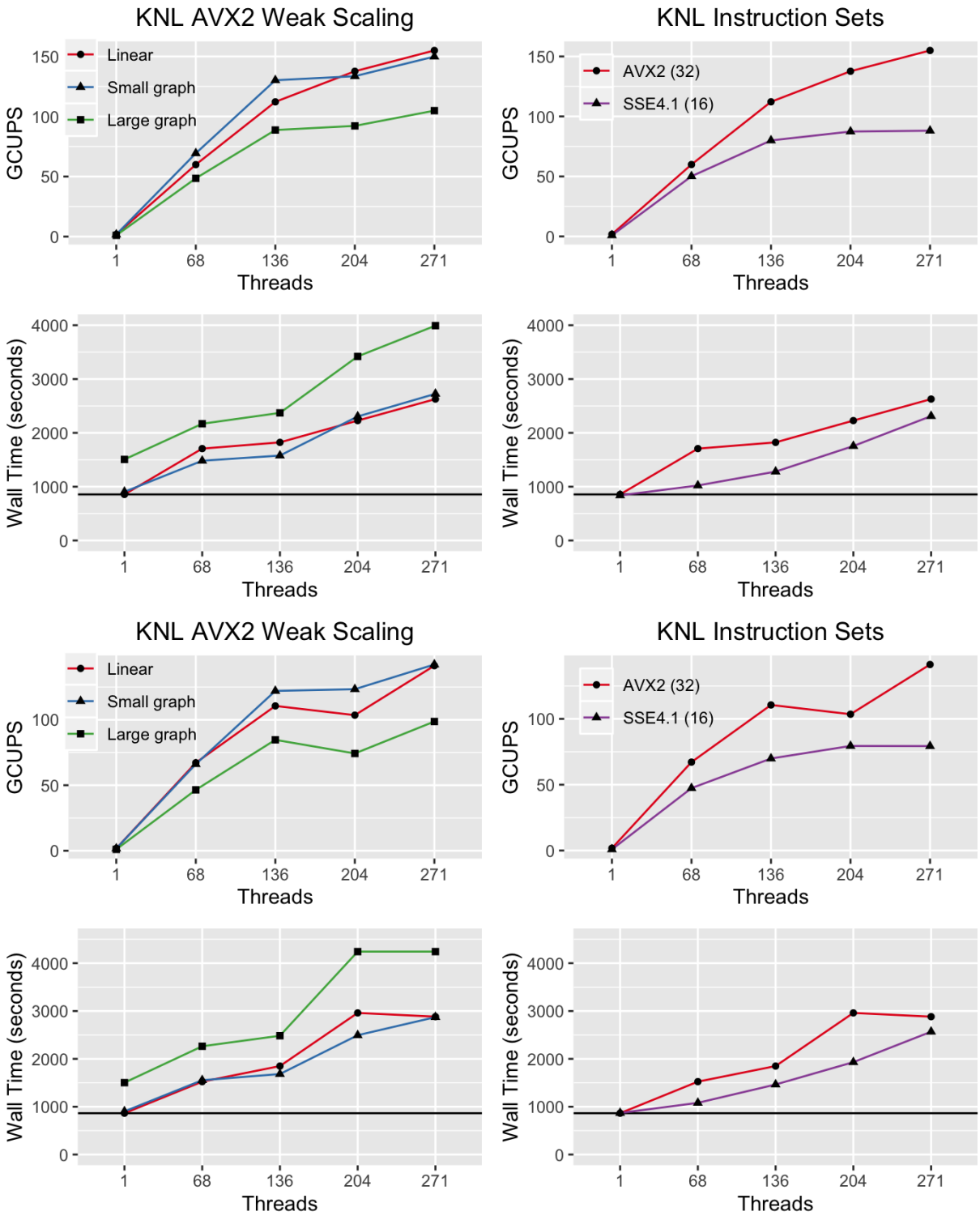

Figure S2: Weak scaling for Knight's Landing (KNL), semiglobal alignment (top four) and local alignment (bottom four). Related to Figure 2.

Figure S3: Correct-by-score plots for all aligners tested on the 100bp read dataset. Points are more transparent when representing fewer reads with a given optimal alignment score. Related to Figure 3.

Figure S4: For the 100bp read dataset, comparing the correct-by-score and correct-by-location measurements, with different buffer sizes (0, 5, 30) for location of the left alignment coordinate. Since Vargas indicates whether the optimal alignment is unique within one read-length, the right column considers only reads that have a unique optimal alignment. Points are more transparent when representing fewer reads with a given optimal alignment score. Related to Figure 3.