Centrifuge: rapid and sensitive classification of metagenomic sequences

Daehwan Kim,[1,4] Li Song,[1,2,4] Florian P. Breitwieser,[1,4] and Steven L. Salzberg[1,2,3]

[1] *Center for Computational Biology, McKusick-Nathans Institute of Genetic Medicine, Johns Hopkins University School of Medicine, Baltimore, Maryland 21205, USA;* [2] *Department of Computer Science, Johns Hopkins University, Baltimore, Maryland 21218, USA;* [3] *Departments of Biomedical Engineering and Biostatistics, Johns Hopkins University, Baltimore, Maryland 21205, USA*

Centrifuge is a novel microbial classification engine that enables rapid, accurate, and sensitive labeling of reads and quantification of species on desktop computers. The system uses an indexing scheme based on the Burrows-Wheeler transform (BWT) and the Ferragina-Manzini (FM) index, optimized specifically for the metagenomic classification problem. Centrifuge requires a relatively small index (4.2 GB for 4078 bacterial and 200 archaeal genomes) and classifies sequences at very high speed, allowing it to process the millions of reads from a typical high-throughput DNA sequencing run within a few minutes. Together, these advances enable timely and accurate analysis of large metagenomics data sets on conventional desktop computers. Because of its space-optimized indexing schemes, Centrifuge also makes it possible to index the entire NCBI nonredundant nucleotide sequence database (a total of 109 billion bases) with an index size of 69 GB, in contrast to *k*-mer-based indexing schemes, which require far more extensive space.

[Supplemental material is available for this article.]

Microbes such as archaea and bacteria are found virtually everywhere on earth, from soils and oceans to hot springs and deep mines (Keller and Zengler 2004). They are also abundant on and inside living creatures, including a variety of niches on the human body such as the skin and the intestinal tract (Human Microbiome Project Consortium 2012). These invisible life forms perform a vast range of biological functions; they are indispensable for the survival of many species; and they maintain the ecological balance of the planet. Many millions of prokaryotic species exist (Schloss and Handelsman 2004), although only a small fraction of them (<1% in soil and even fewer in the ocean) can be isolated and cultivated (Amann et al. 1995). High-throughput sequencing of microbial communities, known as metagenomic sequencing, does not require cultivation and therefore has the potential to provide countless insights into the biological functions of microbial species and their effects on the visible world.

In 2004, the RefSeq database contained 179 complete prokaryotic genomes, a number that grew to 954 genomes by 2009 and to 4278 by December 2015. Together with advances in sequencing throughput, this ever-increasing number of genomes presents a challenge for computational methods that compare DNA sequences to the full database of microbial genomes. Analysis of metagenomics samples, which contain millions of reads from complex mixtures of species, necessitates a compact and scalable indexing scheme for classifying these sequences quickly and accurately. Most of the current metagenomics classification programs either suffer from slow classification speed, a large index size, or both. For example, machine-learning-based approaches such as the Naive Bayes Classifier (NBC) (Rosen et al. 2008) and PhymmBL (Brady and Salzberg 2009, 2011) classify <100 reads per minute, which is too slow for data sets that contain millions of reads. In contrast, the pseudoalignment approach employed in Kraken (Wood and Salzberg 2014) processes reads far more quickly, more than 1 million reads per minute, but its exact *k*-mer matching algorithm requires a large index. For example, Kraken's 31-mer database requires 93 GB of memory (RAM) for 4278 prokaryotic genomes, considerably more memory than today's desktop computers contain.

Fortunately, modern read-mapping algorithms such as Bowtie (Langmead et al. 2009; Langmead and Salzberg 2012) and BWA (Li and Durbin 2009, 2010) have developed a data structure that provides very fast alignment with a relatively small memory footprint. We have adapted this data structure, which is based on the Burrows-Wheeler transform (Burrows and Wheeler 1994) and the Ferragina-Manzini (FM) index (Ferragina and Manzini 2000), to create a metagenomics classifier, Centrifuge, that can efficiently store large numbers of genome sequences, taxonomical mappings of the sequences, and the taxonomical tree.

Methods

Database sequence compression

We implemented memory-efficient indexing schemes for the classification of microbial sequences based on the FM-index, which also permits very fast search operations. We further reduced the size of the index by compressing genomic sequences and building a modified version of the FM-index for those compressed genomes, as follows. First, we observed that for some bacterial species, large numbers of closely related strains and isolates have been sequenced, usually because they represent significant human pathogens. Such genomes include *Salmonella enterica* with 138 genomes, *Escherichia coli* with 131 genomes, and *Helicobacter pylori* with 73 genomes available (these figures represent the contents of RefSeq as of December 2015). As expected, the genomic sequences of strains within the same species are likely to be highly

[4] **These authors contributed equally to this work.**
Corresponding author: infphilo@gmail.com

Article published online before print. Article, supplemental material, and publication date are at http://www.genome.org/cgi/doi/10.1101/gr.210641.116. Freely available online through the *Genome Research* Open Access option.

© 2016 Kim et al. This article, published in *Genome Research*, is available under a Creative Commons License (Attribution 4.0 International), as described at http://creativecommons.org/licenses/by/4.0/.

similar to one another. We leveraged this fact to remove such redundant genomic sequences, so that the storage size of our index can remain compact even as the number of sequenced isolates for these species increases.

Figure 1 illustrates how we compress multiple genomes of the same species by storing near-identical sequences only once. First, we choose the two genomes (G_1 and G_2 in the figure) that are most similar among all genomes. We define the two most similar genomes as those that share the greatest number of *k*-mers (using $k = 53$ for this study) after *k*-mers are randomly sampled at a rate of 1% from the genomes of the same species. In order to facilitate this selection process, we used Jellyfish (Marcais and Kingsford 2011) to build a table indicating which *k*-mers belong to which genomes. Using the two most similar genomes allows for better compression as they tend to share larger chunks of genomic sequences than two randomly selected genomes. We then compared the two most similar genomes using nucmer (Kurtz et al. 2004), which outputs a list of the nearly or completely identical regions in both genomes. When combining the two genomes, we discard those sequences of G_2 with ≥99% identity to G_1 and retain the remaining sequences to use in our index. We then find the genome that is most similar to the combined sequences from G_1 and G_2 and combine this in the same manner as just described. This process is repeated for the rest of the genomes.

As a result of this concatenation procedure, we obtained dramatic space reductions for many species; e.g., the total sequence was reduced from 661 to 74 Mbp (11% of the original sequence size) in *S. enterica* and from 655 to 107 Mbp (16%) in *E. coli* (see Table 1). Overall, the number of base pairs from ~4300 bacterial and archaeal genomes was reduced from 15 to 9.1 billion base pairs

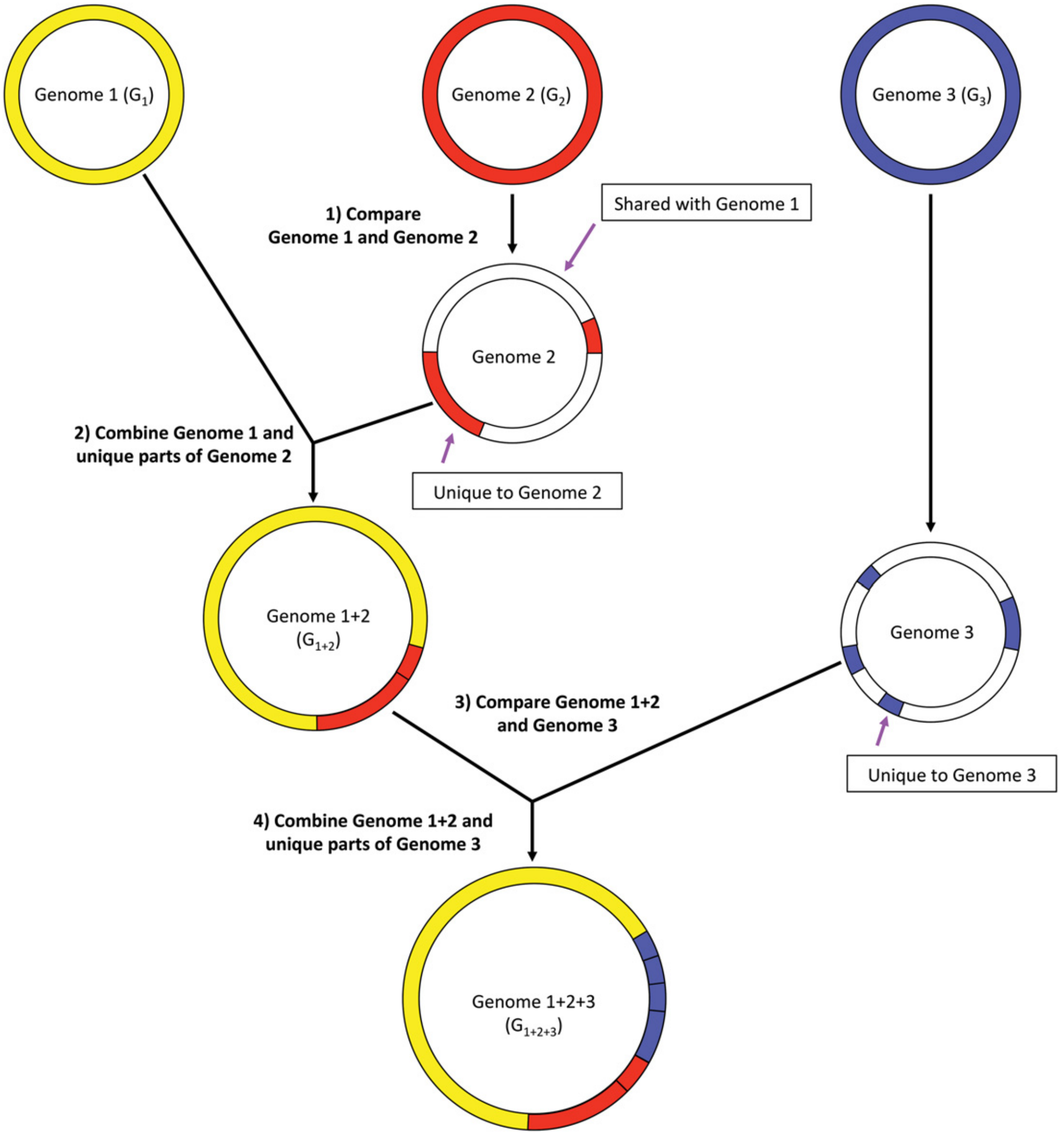
Genome 1 (G_1)
Genome 2 (G_2)
Genome 3 (G_3)
1) Compare Genome 1 and Genome 2
Shared with Genome 1
Genome 2
2) Combine Genome 1 and unique parts of Genome 2
Unique to Genome 2
Genome 1+2 (G_{1+2})
Genome 3
3) Compare Genome 1+2 and Genome 3
Unique to Genome 3
4) Combine Genome 1+2 and unique parts of Genome 3
Genome 1+2+3 (G_{1+2+3})

Figure 1. Compression of genome sequences before building the Centrifuge index. All genomes are compared and similarities are computed based on shared 53-mers. In the figure, genomes G_1 and G_2 are the most similar pair. Sequences of G_2 that are ≥99% identical to G_1 are discarded, and the remaining "unique" sequences from G_2 are added to genome G_1, creating a merged genome, G_{1+2}. Similarity between all genomes is recomputed using the merged genomes. Sequences <99% identical in genome G_3 are then added to the merged genome, creating genome G_{1+2+3}. This process repeats for the entire Centrifuge database until each merged genome has no sequences ≥99% identical to any other genome.

Table 1. Compression ratios for 10 bacterial species that have multiple genomes fully sequenced and available in RefSeq

Species name	Number of genomes	Total size (Mbp)	Total size after compression (Mbp)	Compression ratio
Salmonella enterica	138	661	74	8.9
Escherichia coli	131	655	107	6.1
Staphylococcus aureus	77	220	31	7.1
Helicobacter pylori	73	119	78	1.5
Chlamydia trachomatis	66	69	4	17.3
Listeria monocytogenes	54	160	25	6.5
Burkholderia pseudomallei	47	341	46	7.4
Klebsiella pneumoniae	41	230	47	4.9
Streptococcus pyogenes	39	72	13	5.5
Campylobacter jejuni	37	62	13	4.8

(Gbp). The FM-index for these compressed sequences occupies 4.2 GB of memory, which is small enough to fit into the main memory (RAM) on a conventional desktop computer. As we demonstrate in the Supplemental Methods and Supplemental Table S1, this compression operation has only a negligible impact on classification sensitivity and accuracy.

Classification based on the FM-index

The FM-index provides several advantages over *k*-mer-based indexing schemes that store all *k*-mers in the target genomes. First, the size of the *k*-mer table is usually large; for example, Kraken's *k*-mer table for storing all 31-mers in ~4300 prokaryotic genomes occupies ~100 GB of disk space. Second, using a fixed value for *k* incurs a tradeoff between sensitivity and precision: Classification based on exact matches of large *k*-mers (e.g., 31 bp) provides higher precision but at the expense of lower sensitivity, especially when the data being analyzed originate from divergent species. To achieve higher sensitivity, smaller *k*-mer matches (e.g., 20–25 bp) can be used; however, this results in more false-positive matches. The FM-index provides a means to exploit both large and small *k*-mer matches by enabling rapid search of *k*-mers of any length, at speeds comparable to those of *k*-mer table indexing algorithms (see Results).

Using this FM-index, Centrifuge classifies DNA sequences as follows. Suppose we are given a 100-bp read (note the Centrifuge can just as easily process very long reads, assembled contigs from a draft genome, or even entire chromosomes). We search both the read (forward) and its reverse complement from right to left (3′ to 5′) as illustrated in Figure 2A. Centrifuge begins with a short exact match (16-bp minimum) and extends the match as far as possible. In the example shown in Figure 2A, the first 40 bp match exactly, with a mismatch at the 41st base from the right. The rightmost 40-bp segment of the read is found in six species (A, B, C, D, E, and F) that had been stored in the Centrifuge database. The algorithm then resumes the search beginning at the 42nd base and stops at the next mismatch, which occurs at the 68th base. The 26-bp segment in the middle of the read is found in species G and H. We then continue to search for mappings in the rest of the read, identifying a 32-bp segment that matches species G. Note that only exact matches are considered throughout this process, which is a key factor in the speed of the algorithm. We perform the same procedure for the reverse complement of the read which, in this example, produces more mappings with smaller lengths (17, 16, 28, 18, and 17) compared to the forward strand.

Based on the exact matches found in the read and its reverse complement, Centrifuge then classifies the read using only those mappings with at least one 22-bp match. Figure 2A shows three segment mappings on the forward strand read and one on the read's reverse complement that meet this length threshold. Centrifuge then scores each species using the following formula, which assigns greater weight to the longer segments:

$$\text{Score}(\text{Species } X) = \sum_{\text{hit} \in \text{Species } X} (\text{length}(\text{hit}) - 15)^2.$$

After assessing a variety of formulas, we empirically found that the sum of squared lengths of segments provides the best classification precision. Because almost all sequences of 15 bp or shorter occur in the database by chance, we subtract 15 from the match length. Other values such as 0 and 7 bp work almost as well, while higher values such as 21 bp result in slightly lower precision and sensitivity. For the example in Figure 2, species A, B, C, D, E, and F are assigned the highest score (625), based on the relatively long 40-bp exact match. Species G and H get lower scores because they have considerably shorter matches, even though each has two distinct matches. Note that H has mappings on both the read and its reverse complement, and in this case, Centrifuge chooses the strand that gives the maximum score, rather than using the summed score on both strands, which might bias it toward palindromic sequences.

Centrifuge can assign a sequence to multiple taxonomic categories; by default, it allows up to five labels per sequence. (Note that this strategy differs from Kraken, which always chooses a single taxonomic category, using the lowest common ancestor of all matching species.) In Figure 2, six different species match the read equally well. In order to reduce the number of assignments, Centrifuge traverses up the taxonomic tree. First, it considers the genus that includes the largest number of species, which, in this example (Fig. 2B), is genus I, which covers species A, B, and C. It then replaces these three species with the genus, thereby reducing the number of assignments to four (genus I plus species D, E, and F). If more than five taxonomic labels had remained, Centrifuge would repeat this process for other genera and subsequently for higher taxonomic units until it reduced the number of labels to five or fewer.

The user can easily change the default threshold of five labels per sequence; for example, if this threshold is set to one, then Centrifuge will report only the lowest common ancestor as the taxonomic label, mimicking the behavior of Kraken. In the example shown in Figure 2, this label would be at the family level, which would lose some of the more specific information about which genera and species the reads matched best.

If the size of the index is not a constraint, then the user can also use Centrifuge with uncompressed indexes, which classify reads using the same algorithm. Although considerably larger, the uncompressed indexes allow Centrifuge to classify reads at the strain or genome level; e.g., as *E. coli* K12 rather than just *E. coli*.

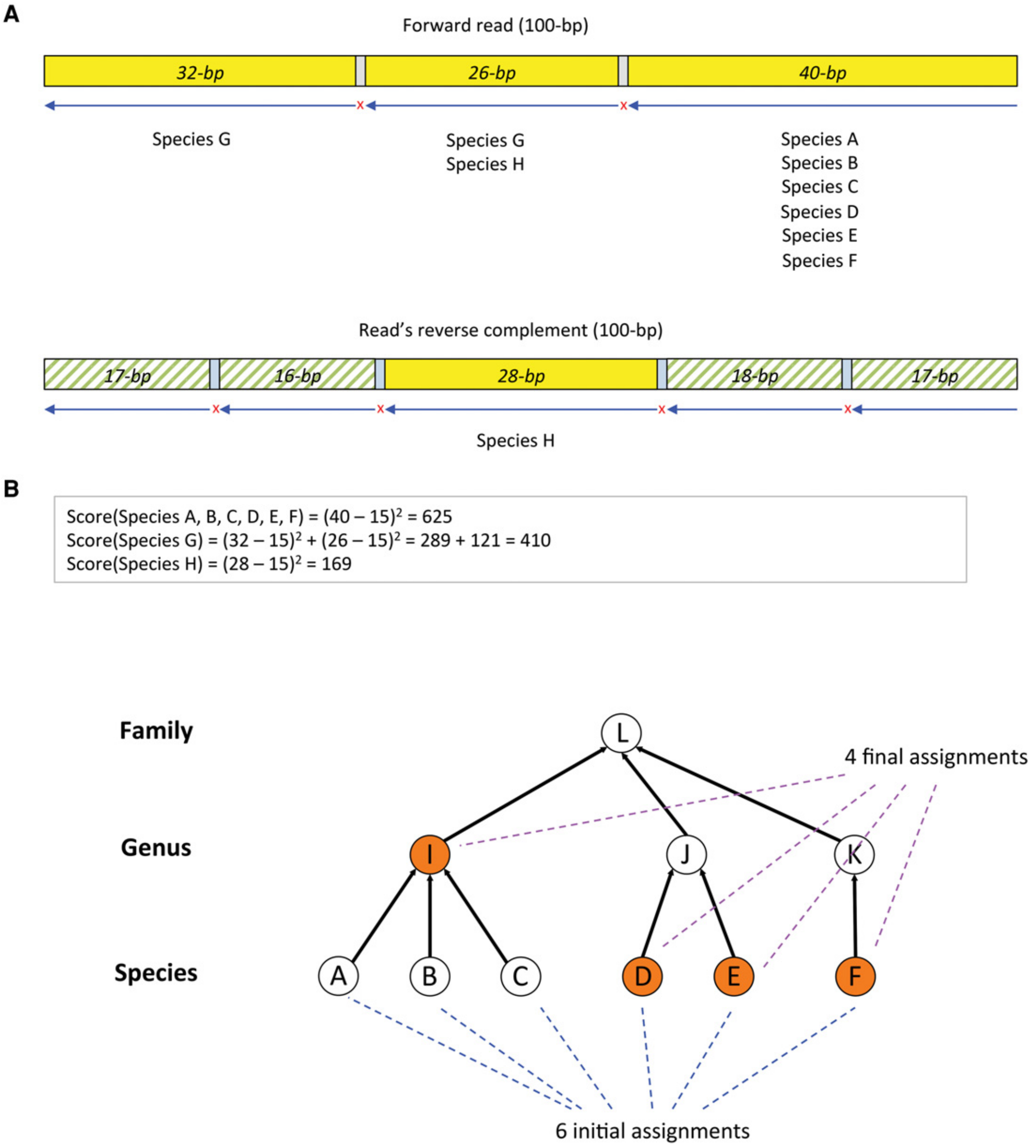

Figure 2. Classification of reads. (*A*) The figure shows how the score for a candidate at the species level is calculated. Given a 100-bp read, both the read (forward) and its reverse complement from *right* to *left* are searched. Centrifuge first identifies a short exact match, then continues until reaching a mismatch: The first 40-bp segment exactly matches six species (A, B, C, D, E, F), followed by a mismatch at the 41st base; the second 26-bp segment matches two species (G and H), followed by a mismatch at the 68th base; and the third 32-bp segment matches only species G. This procedure is repeated for the reverse complement of the read. Centrifuge assigns the highest score (625) to species A, B, C, D, E, and F. (*B*) Centrifuge then traverses up the taxonomic tree to reduce the number of assignments, first by considering the genus that includes the largest number of species, genus I, which covers species A, B, and C, and then replacing these three species with the genus. This procedure results in reducing the number of assignments to four (genus I plus species D, E, and F).

Abundance analysis

In addition to per-read classification, Centrifuge performs abundance analysis at any taxonomic rank (e.g., strain, species, genus). Because many genomes share near-identical segments of DNA with other species, reads originating from those segments will be classified as multiple species. Simply counting the number of the reads that are uniquely classified as a given genome (ignoring those that match other genomes) will therefore give poor estimates of that species' abundance. To address this problem, we define the following statistical model and use it to find maximum likelihood estimates of abundance through an Expectation-Maximization (EM) algorithm. Detailed EM solutions to the model have been previously described and implemented in the Cufflinks (Trapnell et al. 2010) and Sailfish (Patro et al. 2014) software packages.

Similar to how Cufflinks calculates gene/transcript expressions, the likelihood for a specific configuration of species abundance α, given the read assignments *C*, is defined as follows:

$$L(\alpha|C) = \prod_{i=1}^{R} \sum_{j=1}^{S} \frac{\alpha_j l_j}{\sum_k^S \alpha_k l_k} C_{ij},$$

where *R* is the number of the reads, *S* is the number of species, α_j is the abundance of species j, summing up to 1 over all S species, l_j is the average length of the genomes of species j, and C_{ij} is 1 if read i is classified to species j and 0 otherwise.

To find the abundances α that maximize the likelihood function $L(\alpha|C)$, Centrifuge repeats the following EM procedure as also implemented in Cufflinks until the difference between the previous estimate of abundances and the current estimate, $\sum_{j=1}^{S} |\alpha_j - a'_j|$, is less than 10^{-10}.

Expectation (E-step):

$$n_j = \sum_{i=1}^{R} \frac{\alpha_j C_{ij}}{\sum_{k=1}^{S} \alpha_k C_{ik}},$$

where n_j is the estimated number of reads assigned to species j.

Maximization (M-step):

$$\alpha'_j = \frac{n_j/l_j}{\sum_{k=1}^{S} n_k/l_k},$$

where α'_j is the updated estimate of species j's abundance. α′ is then used in the next iteration as α.

Results

We demonstrated the performance of Centrifuge in four different settings involving both real and simulated reads and using several databases with different sizes, specifically one consisting of ∼4300 prokaryotic genomes (index name: *p*, index size: 4.2 GB), another with ∼4300 prokaryotic genomes plus human and viral genomes ($p + h + v$, 6.9 GB), and a third comprised of NCBI nucleotide sequences (*nt*, 69 GB). We compared the sensitivity and speed of Centrifuge to one of the leading classification programs, Kraken (v0.10.5-beta) (Wood and Salzberg 2014). We also included MegaBLAST (Zhang et al. 2000) in our assessment, as it is a very widely used program that is often used for classification. In terms of both sensitivity and precision of classification, Centrifuge demonstrated similar accuracy to the other programs we tested. Centrifuge's principal advantage is that it provides a combination of fast classification speed and low memory requirements, making it possible to perform large metagenomics analyses on a desktop computer using *p* or $p + h + v$ index. For example, Centrifuge took only 47 min on a standard desktop computer to analyze 130 paired-end RNA sequencing runs (a total of 26 GB) from patients infected with Ebola virus (Baize et al. 2014; Gire et al. 2014; Park et al. 2015) as described below. Centrifuge's efficient indexing scheme makes it possible to index the NCBI nucleotide collection (*nt*) database, which is a comprehensive set of sequences (>36 million nonredundant sequences, ∼110 billion bp) collected from viruses, archaea, bacteria, and eukaryotes, and enables rapid and accurate classification of metagenomic samples.

Comparison of Centrifuge, Kraken, and MegaBLAST on simulated reads from 4278 prokaryotic genomes

We created a simulated read data set from the 4278 complete prokaryotic genomes in RefSeq (Pruitt et al. 2014) that were used to build the database, *p*. From these genomes, we generated 10 million 100-bp reads with a per-base error rate of 3% using the Mason simulator, v0.1.2 (Luke et al. 2005). We used an error rate higher than found in Illumina reads (≤0.5%) in order to model the high mutation rates of prokaryotes. Reads were generated randomly from the entire data set; thus, longer genomes had proportionally more reads. The full set of genomes is provided in Supplemental Table S2. We built indexes for each of the respective programs. Kraken and MegaBLAST require 100 GB and 25 GB of space (respectively) for their indexes. In contrast, Centrifuge requires only 4.2 GB to store and index the same genomes. The run-time memory footprint of MegaBLAST is small (Table 2) because it does not read the entire database into memory, in contrast to Kraken and Centrifuge. We classified the reads with Centrifuge, Kraken, and MegaBLAST and calculated sensitivity and precision at the genus and species levels for each program (Table 2). Centrifuge and MegaBLAST often report multiple assignments for a given read, while Kraken instead reports the lowest common ancestor. To make our evaluation consistent across the programs, we only considered uniquely classified reads. Here, we define sensitivity as the number of reads that are correctly classified divided by the total number of reads. Precision (also called positive predictive value) is defined as the number of correctly classified reads divided by the number of predictions made (i.e., reads that have no match and are not classified at a given taxonomic rank or below are not counted).

At the species level, MegaBLAST provides the highest sensitivity at 78.8%, followed by Centrifuge (76.9%) and then Kraken (73.9%). Overall sensitivity is relatively low because many reads are assigned to multiple species and considered as unclassified in our evaluation. MegaBLAST provides the highest precision, 99.4%, followed closely by Kraken at 99%, then Centrifuge at 98.4%. At the genus level, MegaBLAST provides the highest sensitivity at 93.4%, followed by Centrifuge (93.1%) and then by Kraken (90.4%) (Supplemental Table S2). All three programs had near-perfect precision at the genus level, from 99.6% to 99.9%. Kraken was the fastest program on these data, classifying about 1,062,000 reads per minute (rpm), followed by Centrifuge, which was approximately one-half as fast at 563,000 rpm. MegaBLAST is far slower, processing only 327 rpm.

As a side note, fast alignment programs such as Bowtie 2 (Langmead and Salzberg 2012) and BWA (Li and Durbin 2009) can be used for classifying reads, though they were not designed for that purpose. To explore such repurposing, we built a Bowtie 2 index and used Bowtie 2 on the simulated reads. Bowtie 2 is

Table 2. Classification sensitivity and precision for Centrifuge, Kraken, and MegaBLAST using simulated reads

	Genus		Species			
Classifier	Sensitivity	Precision	Sensitivity	Precision	Speed (reads/min)	Memory usage (GB of RAM)
Centrifuge	93.1	99.6	76.9	98.4	563,380	4.2
Kraken	90.4	99.8	73.9	99.0	1,061,947	93.0
MegaBLAST	93.4	99.9	78.8	99.4	327	9.9

In Centrifuge, we used only uniquely classified reads to compute accuracy. To measure speed, we used 10 million reads for Centrifuge and Kraken and 100,000 reads for MegaBLAST. We ran all programs on a Linux system with 1 TB of RAM using one CPU (2.1 GHz Intel Xeon).

very fast, processing >56,000 reads/minute, but is still only one-tenth as fast as Centrifuge. Bowtie 2 also requires 21 GB of RAM, five times more than required by Centrifuge. Bowtie 2 has classification sensitivity and precision comparable to Centrifuge (e.g., sensitivity of 96.8% and precision of 99.1% at the genus level).

In addition to the above per-read classification, Centrifuge estimates abundance at various taxonomic ranks. Centrifuge's abundance assessment closely matches the true abundance distribution of genomes in the simulated reads (Supplemental Fig. S1) at the species level (Pearson's correlation coefficient of 0.919) and the genus level (correlation of 0.986).

Comparison of Centrifuge and Kraken performance on real data sets from sequencing reads of bacterial genomes

To test our method on real sequencing data sets, we downloaded 530 DNA sequencing data sets from the Sequence Read Archive (SRA). We selected them according to whether the SRA samples had been assigned a taxonomic identifier that belongs to a genus for which we have at least one genome in the database. All of the data sets were generated by whole-genome shotgun projects using recent Illumina platforms; 225 were sequenced on HiSeq and 305 on MiSeq instruments, with mean read lengths of 100 and 218 bp, respectively. Supplemental Table S3 contains a complete list of the SRA identifiers, taxonomy IDs, number of reads, and classification results. In total, these data contain over 560 million reads, with an average of 1,061,536 reads per sample.

For this experiment, we compared Centrifuge and Kraken but omitted MegaBLAST because it would take far too long to run. Kraken was chosen as the standard for comparison because it demonstrated superior accuracy over multiple other programs in a recent comparison of metagenomic classifiers (Lindgreen et al. 2016).

Figure 3 and Supplemental Figure S2 show the results for classification sensitivity, accuracy, speed, and memory usage using the database *p* of ∼4300 prokaryotic genomes. On average, Centrifuge had slightly higher sensitivity (0.6% higher) than Kraken. Perhaps due to its use of a longer exact match requirement (31 bases), Kraken had slightly higher precision (2%) than Centrifuge. The lower accuracy of both programs on some data sets may be due to: (1) substantial differences between the genome that we have in the database and the strain that was sequenced; (2) numerous contaminating reads from the host or reagents; or (3) a high sequencing error rate for a particular sample. For example, SRR2225903 is labeled as a strain of *Acinetobacter*, but 85% of the reads are assigned to *Escherichia*. SRR1656428 is labeled as a clinical isolate of *Shigella dysenteriae*, but 92% of the reads are classified as *Klebsiella* (note that for this experiment the taxonomy ID has since been updated by NCBI, but the name has not changed). In other instances (such as SRR1656029 and SRR1655687, labeled as clinical isolates of *Ferrimonas balearica* and *Kytococcus sedentarius*, respectively), we could not match taxonomic IDs to a substantial fraction of the reads, even when searching against the *nt* database. The reads might have come from a species that has no close relative in the database or could not be assigned due to poor quality.

Overall accuracy for both programs was very similar. Kraken was slightly faster, with an average run time of 39.3 sec per genome, while Centrifuge required 50.9 sec per genome (both using eight cores).

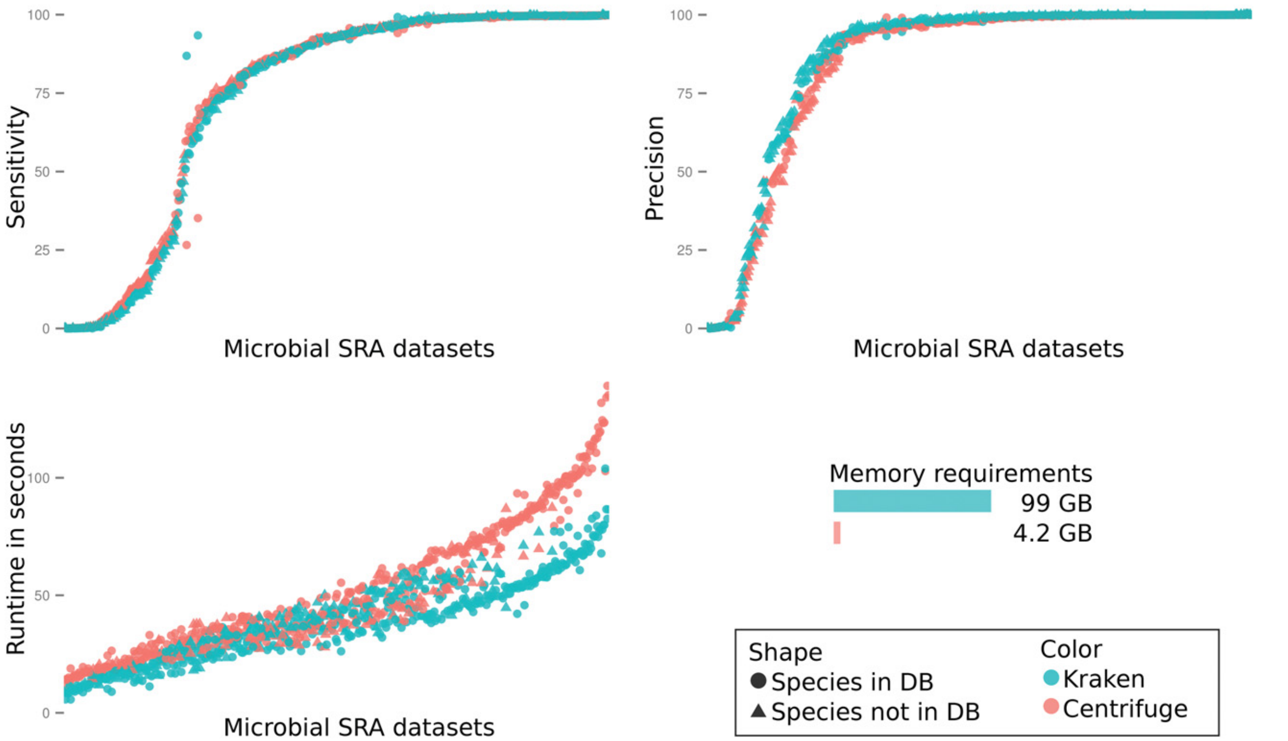

Figure 3. Results on 530 sequencing data sets from bacterial genomes retrieved from the Sequence Read Archive at NCBI. Each dot represents the results for one genome, with Centrifuge shown in orange and Kraken in teal. The *upper left* plot shows sensitivity, computed as the percentage of reads classified as the correct genus. The *upper right* plot shows precision, computed as the percentage of genus-level classifications made by a program that were correct. The *lower left* plot shows runtime measured in seconds.

Application of Centrifuge for analyzing samples with Ebola virus and GB virus C co-infections on a desktop

To demonstrate the speed, sensitivity, and applicability of Centrifuge on a real data set, we used data from the Ebola virus disease (EVD) outbreak. The 2013–2015 EVD outbreak in West Africa cost the lives of over 11,000 people as of August 26, 2015 (WHO Ebola Situation Report, http://apps.who.int/ebola/ebola-situation-reports). In an international effort to research the disease and stop its spread, several groups sequenced the Ebola virus collected from patients' blood samples and released their data sets online (Baize et al. 2014; Gire et al. 2014; Park et al. 2015). The genomic data were used to trace the disease and mutations in the Ebola genome and inform further public health and research efforts. Lauck et al. (2015) reanalyzed one of the data sets (Gire et al. 2014) in order to assess the prevalence and effect of GB virus C co-infection on the outcome of EVD.

We analyzed 130 paired-end sequencing runs from 49 patients reported in Gire et al. (2014) using Centrifuge to look for further co-infections. This data set has a total of 97,097,119 reads (26 GB of FASTA files). The accession IDs of these data are provided in Supplemental Table S4. For this analysis, we used the database $p+h+v$ containing all prokaryotic genomes (compressed), all viral genomes, and the human genome (total index size: 6.9 GB). Running on a desktop computer (quad-core, Intel Core i5-4460 @ 3.2 GHz with 8 GB RAM), Centrifuge completed the analysis of all samples in 47 min with four cores. RNA-sequencing (Mortazavi et al. 2008) requires more steps than DNA-sequencing, including the reverse-transcription of RNA to DNA molecules, which introduces sequencing biases and artifacts. In order to handle these additional sources of errors and remove spurious detections, we filtered the results to include only reads that have a matching length of at least 60 bp on the 2 × 100-bp reads.

Figure 4 shows our classification results for the 49 patients. Centrifuge detects between 3853 and 6,781,684 Ebola virus reads per patient. As reported by Lauck et al., we also detected co-infection of Ebola virus and GB virus C in many of the patients. Centrifuge identified at least one read from this virus in 27 of the 49 patients; nine patient samples had 50 or more reads. Nine patients had between one and 10 reads matching the Hepatitis B virus, and in one sample, over 1000 reads aligned uniquely to this virus. This Hepatitis B co-infection has not been reported previously, demonstrating the inherent advantage of using a metagenomics classification tool, which can also detect off-target species.

Application of Centrifuge for analyzing Oxford Nanopore MinION reads of fruitshake using *nt* database

As a test of Centrifuge's *nt* database, we used it to analyze sequences from a mixture of common fruits and vegetables sequenced using long-read single-molecule technology. The mixture included more than a dozen common foods: grape, blueberry, yam (sweet potato), asparagus, cranberry, lemon, orange, iceberg lettuce, black pepper, wheat (flour), cherry tomato, pear, bread (wheat plus other ingredients), and coffee (beans). The "fruitshake" mixture was blended together, DNA was extracted, and sequencing was run on an Oxford Nanopore MinION. The number of reads generated from the fruitshake sample was 20,809, with lengths ranging from 90 to 13,174 bp and a mean length of 893 bp. Although MinION platforms produce much longer reads than Illumina platforms, MinIONs' high sequencing error rates (estimated at 15%) (Jain et al. 2015) prevent reads from containing long exact matches and increase the chance of noisy and incorrect matches. We initially labeled 8236 reads using Centrifuge. In order to reduce false-positive assignments for these error-prone reads, we filtered out those reads that scored ≤300 and had match lengths ≤50 bp, resulting in 3617 reads ultimately classified. Table 3 shows 14 species to which at least five reads are uniquely assigned, encompassing many of the species included in the sample, such as wheat, tomato, lettuce, grape, barley, and pear. Note that as with any real sample, the true composition of the reads is unknown; we present these results here to illustrate (1) the use of the large *nt* database, and (2) the use of Centrifuge on long, high-error-rate reads. Although apple was not known to be present in the sample, the five reads assigned to apple might have been due to similarity between the apple genome and the pear genome. Twenty-six reads were identified as sheep and eight as cow, which were confirmed separately by BLAST searches. These could represent sample contamination or possibly contaminants in the sheep and cow assemblies. Missing species can be explained either by low abundance in the sample or because their genomes are substantially different from those in the Centrifuge *nt* database.

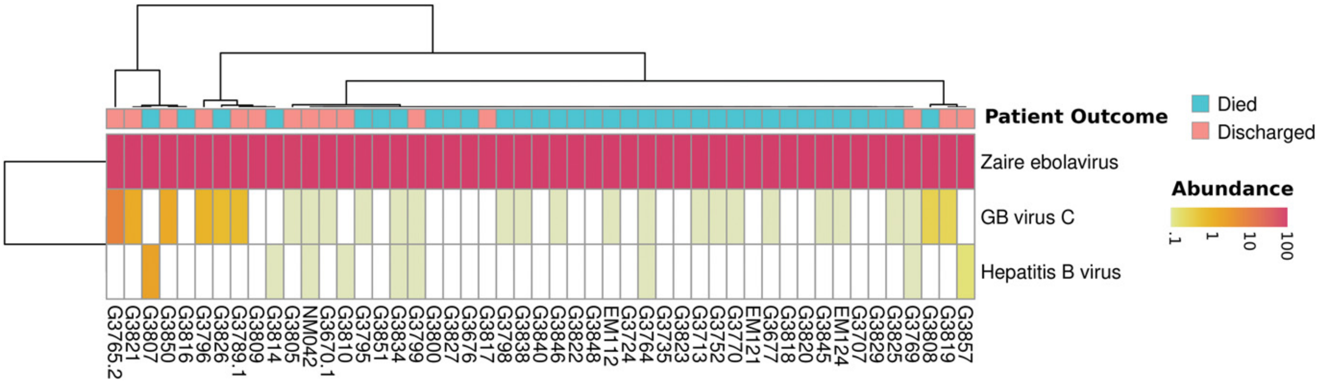

Figure 4. Heat map of the most abundant species in Ebola samples. The color scale encodes species abundance (the number of unique reads normalized by genome size), ranging from yellow (<0.1% of the normalized read count) to red (100%), with white representing an abundance of zero. All species that have a normalized read count over 1% in any of the samples are shown. Zaire ebolavirus dominates the samples; however, there is also a signal for other viruses in some of the patients—namely GB virus C and Hepatitis B virus.

Table 3. Classification of the fruitshake sample using Centrifuge's *nt* database

Scientific name	Common name	Number of uniquely classified reads
Triticum aestivum	**Wheat**	2889
Solanum lycopersicum	**Tomato**	207
Lactuca sativa	**Lettuce**	134
Ovis canadensis canadensis	Sheep	26
Vitis vinifera	**Grape**	16
Aegilops tauschii	**Wheat**	13
Triticum urartu	**Wheat**	12
Solanum pennellii	**Tomato**	12
Triticum turgidum subsp. durum	**Wheat**	8
Triticum monococcum	**Wheat**	8
Bos taurus	Cow	8
Hordeum vulgare subsp. vulgare	**Barley**	6
Malus domestica	Apple	5
Pyrus x bretschneideri	**Pear**	5

The table shows 14 genomes to which at least five reads sequenced from the fruitshake sample were uniquely assigned. Common names in bold represent species known to be present in the mixture.

Discussion

Centrifuge requires a relatively small index for representing and searching ~4300 prokaryotic genomes, only 4.2 GB, lean enough to fit the memory of a personal desktop. These space-optimized indexing schemes also make it possible to index the NCBI nucleotide sequence database that includes a comprehensive set of sequences collected from viruses, archaea, bacteria, and eukaryotes. Identical sequences have been removed to make it nonredundant, but even after this reduction, the database contains over 36.5 million sequences with a total of ~109 billion base pairs (Gbp). This rapidly growing database, called *nt*, enables the classification of sequencing data sets from hundreds of plant, animal, and fungal species as well as thousands of viruses and bacteria and many other eukaryotes. Metagenomics projects often include substantial quantities of eukaryotic DNA, and a prokaryotes-only index cannot identify these species.

The challenge in using a much larger database is the far greater number of unique *k*-mers that must be indexed. For example, using Kraken's default *k*-mer length of 31 bp, the *nt* database contains ~57 billion distinct *k*-mers. Although it employs several elegant techniques to minimize space, Kraken still requires 12 bytes per *k*-mer, which means it would require an index size of 684 GB for the full *nt* database. Reducing the *k*-mer size helps only slightly: With $k = 22$, Kraken would require an index of 520 GB. Either of these indexes would require a specialized computer with very large main memory.

Centrifuge's index is based on the space-efficient Burrows-Wheeler transform, and as a result, it requires only 69 GB for the *nt* database, less than the raw sequence itself. BLAST and MegaBLAST are currently the only alternative methods that can classify sequences against the entire *nt* database; thus, we compared Centrifuge with MegaBLAST using our simulated read data, described above. MegaBLAST uses a larger index, requiring 155 GB on disk, but it does not load the entire index into memory and requires only 16 GB of RAM, while Centrifuge requires 69 GB. However, Centrifuge classified reads at a far higher speed: In our experiments on the Mason simulation data, it processed ~372,000 reads/min, over 3500 times faster than MegaBLAST, which processed only 105 reads/min. Using the much larger *nt* database instead of the prokaryotic database on the Mason simulated reads (Table 2) does not decrease the classification precision and sensitivity of both programs at the genus level, with Centrifuge's sensitivity only marginally decreasing by 3.2%.

As the prokaryotic and *nt* databases continue to rapidly expand and provide more comprehensive coverage, further difficulties arise in analyzing sequencing data. For example, two major challenges remain to be addressed in the statistical estimation of abundance (Lu et al. 2016). First, the RefSeq database includes many genomes nearly identical to one another, which makes it extremely difficult to distinguish those genomes present in the sample from those that are not. For example, many strains of *Chlamydia trachomatis* are almost identical (>99.99%) to one another (e.g., *Chlamydia trachomatis D/UW-3/CX* and *Chlamydia trachomatis strain Ia/CS190/96*). Second, the microbial taxonomy is sometimes not based on genomic sequence similarity and contains taxonomically misnamed or misplaced species (Federhen 2015). Incorrectly positioned species (or strains) can contribute to inaccurate ancestor assignment (e.g., genus or family) in abundance estimations. For example, a genome initially identified as *Anabaena variabilis* ATCC 29413 was reassigned to the genus *Nostoc,* not *Anabaena* (Thiel et al. 2014).

In conclusion, Centrifuge is a rapid and sensitive classifier for microbial sequences with low memory requirements and a speed comparable to the fastest systems. Centrifuge classifies 10 million reads against a database of all complete prokaryotic and viral genomes within 20 min using one CPU core and requiring <8 GB of RAM. Furthermore, Centrifuge can also build an index for NCBI's entire *nt* database of nonredundant sequences from both prokaryotes and eukaryotes. The search requires a computer system with 128 GB of RAM but runs over 3500 times faster than MegaBLAST.

Data access

Centrifuge is available as free, open-source software from https://github.com/infphilo/centrifuge/archive/centrifuge-genome-research.zip and provided in Supplemental Data S1. The fruitshake sequencing data from this study have been submitted to the NCBI BioProject database (http://www.ncbi.nlm.nih.gov/bioproject/) under accession number PRJNA343503.

Acknowledgments

We thank Winston Timp and Rachael Workman for generating the fruitshake data from the MinION sequencing platform. This work was supported in part by the National Institutes of Health (NIH) under grants R01-HG006677 and R01-GM083873, by the US Army Research Office under grant W911NF-1410490, and by the National Science Foundation (NSF) under grant ABI-1356078.

Author contributions: D.K., L.S., F.P.B., and S.L.S. performed the analysis and discussed the results of Centrifuge. D.K., L.S., and F.P.B. implemented Centrifuge. D.K., F.P.B., L.S., and S.L.S. wrote the manuscript. All authors read and approved the final manuscript.

References

Amann RI, Ludwig W, Schleifer KH. 1995. Phylogenetic identification and in situ detection of individual microbial cells without cultivation. *Microbiol Rev* **59:** 143–169.

Baize S, Pannetier D, Oestereich L, Rieger T, Koivogui L, Magassouba N, Soropogui B, Sow MS, Keita S, De Clerck H, et al. 2014. Emergence of Zaire Ebola virus disease in Guinea. *N Engl J Med* **371:** 1418–1425.

Brady A, Salzberg SL. 2009. Phymm and PhymmBL: metagenomic phylogenetic classification with interpolated Markov models. *Nat Methods* **6:** 673–676.

Brady A, Salzberg S. 2011. PhymmBL expanded: confidence scores, custom databases, parallelization and more. *Nat Methods* **8:** 367.

Burrows M, Wheeler DJ. 1994. *A block-sorting lossless data compression algorithm*. Technical Report 124. Digital Equipment Corporation, Palo Alto, CA.

Federhen S. 2015. Type material in the NCBI taxonomy database. *Nucleic Acids Res* **43:** D1086–D1098.

Ferragina P, Manzini G. 2000. Opportunistic data structures with applications. In Proceedings of the 41st IEEE symposium on foundations of computer science, Redondo Beach, CA.

Gire SK, Goba A, Andersen KG, Sealfon RS, Park DJ, Kanneh L, Jalloh S, Momoh M, Fullah M, Dudas G, et al. 2014. Genomic surveillance elucidates Ebola virus origin and transmission during the 2014 outbreak. *Science* **345:** 1369–1372.

Human Microbiome Project Consortium. 2012. A framework for human microbiome research. *Nature* **486:** 215–221.

Jain M, Fiddes IT, Miga KH, Olsen HE, Paten B, Akeson M. 2015. Improved data analysis for the MinION nanopore sequencer. *Nat Methods* **12:** 351–356.

Keller M, Zengler K. 2004. Tapping into microbial diversity. *Nat Rev Microbiol* **2:** 141–150.

Kurtz S, Phillippy A, Delcher AL, Smoot M, Shumway M, Antonescu C, Salzberg SL. 2004. Versatile and open software for comparing large genomes. *Genome Biol* **5:** R12.

Langmead B, Salzberg SL. 2012. Fast gapped-read alignment with Bowtie 2. *Nat Methods* **9:** 357–359.

Langmead B, Trapnell C, Pop M, Salzberg SL. 2009. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol* **10:** R25.

Lauck M, Bailey AL, Andersen KG, Goldberg TL, Sabeti PC, O'Connor DH. 2015. GB virus C coinfections in West African Ebola patients. *J Virol* **89:** 2425–2429.

Li H, Durbin R. 2009. Fast and accurate short read alignment with Burrows–Wheeler transform. *Bioinformatics* **25:** 1754–1760.

Li H, Durbin R. 2010. Fast and accurate long-read alignment with Burrows–Wheeler transform. *Bioinformatics* **26:** 589–595.

Lindgreen S, Adair KL, Gardner PP. 2016. An evaluation of the accuracy and speed of metagenome analysis tools. *Sci Rep* **6:** 19233.

Lu J, Breitwieser FP, Thielen P, Salzberg SL. 2016. Bracken: estimating species abundance in metagenomics data. *bioRxiv* doi: 10.1101/051813.

Luke S, Cioffi-Revilla C, Panait L, Sullivan K, Balan G. 2005. MASON: a multiagent simulation environment. *Simulation* **81:** 517–527.

Marcais G, Kingsford C. 2011. A fast, lock-free approach for efficient parallel counting of occurrences of *k*-mers. *Bioinformatics* **27:** 764–770.

Mortazavi A, Williams BA, McCue K, Schaeffer L, Wold B. 2008. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat Methods* **5:** 621–628.

Park DJ, Dudas G, Wohl S, Goba A, Whitmer SL, Andersen KG, Sealfon RS, Ladner JT, Kugelman JR, Matranga CB, et al. 2015. Ebola virus epidemiology, transmission, and evolution during seven months in Sierra Leone. *Cell* **161:** 1516–1526.

Patro R, Mount SM, Kingsford C. 2014. Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms. *Nat Biotechnol* **32:** 462–464.

Pruitt KD, Brown GR, Hiatt SM, Thibaud-Nissen F, Astashyn A, Ermolaeva O, Farrell CM, Hart J, Landrum MJ, McGarvey KM, et al. 2014. RefSeq: an update on mammalian reference sequences. *Nucleic Acids Res* **42:** D756–D763.

Rosen G, Garbarine E, Caseiro D, Polikar R, Sokhansanj B. 2008. Metagenome fragment classification using *N*-mer frequency profiles. *Adv Bioinformatics* **2008:** 205969.

Schloss PD, Handelsman J. 2004. Status of the microbial census. *Microbiol Mol Biol Rev* **68:** 686–691.

Thiel T, Pratte BS, Zhong J, Goodwin L, Copeland A, Lucas S, Han C, Pitluck S, Land ML, Kyrpides NC, et al. 2014. Complete genome sequence of *Anabaena variabilis* ATCC 29413. *Stand Genomic Sci* **9:** 562–573.

Trapnell C, Williams BA, Pertea G, Mortazavi A, Kwan G, van Baren MJ, Salzberg SL, Wold BJ, Pachter L. 2010. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nat Biotechnol* **28:** 511–515.

Wood DE, Salzberg SL. 2014. Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome Biol* **15:** R46.

Zhang Z, Schwartz S, Wagner L, Miller W. 2000. A greedy algorithm for aligning DNA sequences. *J Comput Biol* **7:** 203–214.

Received May 28, 2016; accepted in revised form October 13, 2016.