

1 Evaluating Metagenome Assembly on a Simple  
2 Defined Community with Many Strain Variants

3 Sherine Awad<sup>1</sup>, Luiz Irber<sup>1</sup>, C. Titus Brown<sup>1\*</sup>  
<sup>1</sup>**Department of Population Health and Reproduction**  
University of California, Davis  
Davis, CA 95616 USA  
\* E-mail: ctbrown@ucdavis.edu

4 July 3, 2017

5 **Abstract**

6 We evaluate the performance of three metagenome assemblers, IDBA,  
7 MetaSPAdes, and MEGAHIT, on short-read sequencing of a defined  
8 “mock” community containing 64 genomes (Shakya et al. (2013)). We  
9 update the reference metagenome for this mock community and detect  
10 several additional genomes in the read data set. We show that strain  
11 confusion results in significant loss in assembly of reference genomes  
12 that are otherwise completely present in the read data set. In agree-  
13 ment with previous studies, we find that MEGAHIT performs best  
14 computationally; we also show that MEGAHIT tends to recover larger  
15 portions of the strain variants than the other assemblers.

## 16 Introduction

17 Metagenomics refers to sequencing of DNA from a mixture of organisms,  
18 often from an environmental or uncultured sample. Unlike whole genome  
19 sequencing, metagenomics targets a mixture of genomes, which introduces  
20 metagenome-specific challenges in analysis [1]. Most approaches to analyz-  
21 ing metagenomic data rely on mapping or comparing sequencing reads to  
22 reference sequence collections. However, reference databases contain only  
23 a small subset of microbial diversity [2], and much of the remaining diver-  
24 sity is evolutionarily distant and reference-based search techniques may not  
25 recover it [3].

26 As sequencing capacity increases and sequence data is generated from  
27 many more environmental samples, metagenomics is increasingly using *de*  
28 *novo* assembly techniques to generate new reference genomes and metagenomes  
29 [4]. There are a number of metagenome assemblers that are widely used -  
30 see [5] for an overview of the available software, and [1] for a review of the  
31 different assembler methodologies. However, evaluating the results of these  
32 assemblers is challenging due to the general lack of good quality reference  
33 metagenomes.

34 Moya et al. in [6] evaluated metagenome assembly using two simulated  
35 454 viral metagenome and six assemblers. The assemblies were evaluated  
36 based on several metrics including N50, percentages of reads assembled,  
37 accuracy when compared to the reference genome. In addition to these met-  
38 rics, the authors evaluated chimeras per contigs and the effect of assembly  
39 on taxonomic and functional annotations.

40 Mavromatis et al. in [7] provided a benchmark study to evaluate the  
41 fidelity of metagenome processing methods. The study used simulated  
42 metagenomic data sets constructed at different complexity levels. The datasets  
43 were assembled using Phrap v3.57, Arachne v.2 [8] and JAZZ [9]. This study  
44 evaluates assembly, gene prediction, and binning methods. However, the  
45 study did not evaluate the assembly quality against a reference genome.

46 Rangwala et al. in [10] presented an evaluation study of metagenome  
47 assembly. The study used a de Bruijn graph based assembler ABYSS [11] to  
48 assemble simulated metagenome reads of 36 bp. The data set is classified at  
49 different complexity levels. The study compared the quality of the assembly  
50 of the data sets in terms of contig length and assembly accuracy. The  
51 study also took into consideration the effect of kmer size and the degree of  
52 chimericity. However, the study evaluated the assembly based on only one  
53 assembler. Also, these previous studies used simulated data, which may lack

54 confounders of assembly such as sequencing artifacts and GC bias.

55 In a landmark study, Shakya et al. (2013) constructed a synthetic com-  
56 munity of organisms by mixing DNA isolated from individual cultures of 64  
57 bacteria and archaea, including a variety of strains across a range of average  
58 nucleotide distances [12]. In addition to performing 16s amplicon analy-  
59 sis and doing 454 sequencing, the authors shotgun-sequenced the mixture  
60 with Illumina. While the authors concluded that this metagenomic sequenc-  
61 ing generally outperformed amplicon sequencing, they did not conduct an  
62 assembly based analysis. This data set was also used in several other eval-  
63 uation studies, including gbttools for binning [13] and benchmarking of the  
64 MEGAHIT assembler [14].

65 More recently, several benchmark studies systematically evaluated metagenome  
66 assembly of short reads. The Critical Assessment of Metagenome Interpre-  
67 tation (CAMI) collaboration benchmarked a number of metagenome assem-  
68 blers on several data sets of varying complexity, evaluating recovery of novel  
69 genomes and multiple strain variants [3]. Notably, CAMI concluded that  
70 “The resolution of strain-level diversity represents a substantial challenge  
71 to all evaluated programs.” Another recent study evaluated eight assem-  
72 blers on nine environmental metagenomes and three simulated data sets  
73 and provided a workflow for choosing a metagenome assembler based on  
74 the biological goal and computational resources available [15]. [5] explored  
75 metagenome assembler performance on a pair of real data sets, again con-  
76 cluding that the biological goal and computational resources defined the  
77 choice of assembler. Also see [16] for an analysis of a previously generated  
78 HMP benchmark data set; however, the Illumina reads used for this study  
79 are much shorter than current sequencing and are arguably not relevant to  
80 future studies.

81 In this study, we extend previous work by delving into questions of  
82 chimeric misassembly and strain recovery in the Shakya et al. (2013) data  
83 set. First, we update the list of reference genomes for Shakya et al. to in-  
84 clude the latest GenBank assemblies along with plasmids. We then compare  
85 IDBA [17], MetaSPAdes [18], and MEGAHIT [19] performance on assem-  
86 bling this short-read data set, and explore concordance in recovery between  
87 the three assemblers. We describe the effects of “strain confusion” between  
88 multiple strains. We also detect and analyze several previously unreported  
89 strains and genomes in the Shakya et al. data set. We find that in the ab-  
90 sence of closely related genomes, all three metagenome assemblers recover  
91 95% or more of known reference genomes. However, in the presence of  
92 closely related genomes, these three metagenome assemblers vary widely in

93 their performance and, in extreme cases, can fail to recover the majority of  
94 some genomes even when they are completely present in the reads. Our re-  
95 port provides strong guidance on choice of assemblers and extends previous  
96 analyses of this low-complexity metagenome benchmarking data set.

## 97 **Datasets**

98 We used a diverse mock community data set constructed by pooling DNA  
99 from 64 species of bacteria and archaea and sequencing them with Illumina  
100 HiSeq. The raw data set consisted of 109,629,496 reads from Illumina HiSeq  
101 101 bp paired-end sequencing (2x101) with an untrimmed total length of  
102 11.07 Gbp and an estimated fragment size of 380 bp [12].

103 The original reads are available through the NCBI Sequence Read Archive  
104 at Accession SRX200676. We updated the 64 reference genomes sets from  
105 NCBI GenBank using the latest available assemblies with plasmid content  
106 (June 2017); the accession numbers are available as `accession-list-ref.txt`  
107 in the Zenodo repository, DOI: 10.5281/zenodo.821919. For convenience, the  
108 updated reference genome collection is available for download at the archival  
109 URL <https://osf.io/vbhy5/>.

## 110 **Methods**

111 The analysis code and run scripts for this paper are written in Python and  
112 bash, and are available at [https://github.com/dib-lab/2016-metagenome-](https://github.com/dib-lab/2016-metagenome-assembly-eval/)  
113 `assembly-eval/` (archived at Zenodo DOI: 10.5281/zenodo.821919). The  
114 scripts and overall pipeline were examined by the first and senior authors for  
115 correctness. In addition, the bespoke reference-based analysis scripts were  
116 tested by running them on a single-colony *E. coli* MG1655 data set with a  
117 high quality reference genome [20].

## 118 **Quality Filtering**

119 We removed adapters with Trimmomatic v0.30 in paired-end mode with  
120 the TruSeq adapters [21], using light quality score trimming (`LEADING:2`  
121 `TRAILING:2 SLIDINGWINDOW:4:2 MINLEN:25`) as recommended in MacManes,  
122 2014 [22].

## 123 Reference Coverage Profile

124 To evaluate how much of the reference metagenome was contained in the  
125 read data, we used `bwa aln` (v0.7.7.r441) to map reads to the reference  
126 genome [23]. We then calculated how many reference bases were covered by  
127 mapped reads (custom script `coverage-profile.py`).

## 128 Measuring k-mer inclusion and Jaccard similarity

129 We used MinHashing as implemented in `sourmash` to estimate k-mer inclu-  
130 sion and Jaccard similarity between data sets [24]. MinHash signatures were  
131 prepared with `sourmash compute` using `--scaled 10000`. K-mer inclusion  
132 was computed by taking the ratio of the number of intersecting hashes with  
133 the query over the total number of hashes in the subject MinHash. Jac-  
134 card similarity was computed as in [25] by taking the ratio of the number  
135 of intersecting hashes between the query and subject over the number of  
136 hashes in the union. K-mer sizes for comparison were chosen at 21, 31, or  
137 51, depending on the level of taxonomic specificity desired - genus, species,  
138 or strain, respectively, as described in [26].

139 Where specified, high-abundance k-mers were selected for counting by  
140 using the script `trim-low-abund.py` script with `-C 5` from `khmer` v2 [27,  
141 28].

## 142 Assemblers

143 We assembled the quality-filtered reads using three different assemblers:  
144 IDBA-UD [17], MetaSPAdes [18], and MEGAHIT [19]. For IDBA-UD v1.1.3  
145 [17], we used `--pre_correction` to perform pre-correction before assembly  
146 and `-r` for the pe files. IDBA could not ingest orphan sequences so singleton  
147 reads were omitted from this assembly.

148 For MetaSPAdes v3.10.1 [18], we used `--meta --pe1-12 --pe1-s` where  
149 `--meta` is used for metagenomic data sets, `--pe1-12` specifies the interlaced  
150 reads for the first paired-end library, and `--pe1-s` provides the orphan reads  
151 remaining from quality trimming.

152 For MEGAHIT v1.1.1-2-g02102e1 [19], we used `-l 101 -m 3e9 --cpu-only`  
153 where `-l` is for maximum read length, `-m` is for max memory in bytes to  
154 be used in constructing the graph, and `--cpu-only` uses only the CPU  
155 and no GPUs. We also used `--presets meta-large` for large and complex  
156 metagenomes, and `--12` and `-r` to specify the interleaved-paired-end and

single-end files respectively. MEGAHIT allows the specification of a memory limit and we used `-M 1e+10` for 10 GB.

All three assemblies were executed on the same XSEDE Jetstream instance (S1.Xxlarge) at Indiana University, running Ubuntu 16.04 (install 6/21/17, Ubuntu 16.04 LTS Development + GUI support + Docker; based on Ubuntu cloud image for 16.04 LTS with basic dev tools, GUI/Xfce added). Assemblers were limited to 16 threads. We recorded RAM and CPU time for each assembly using `/usr/bin/time -v`. Install and execute details as well as output timings and logs are available in the `pipeline/runstats` directory of the Zenodo archive.

Unless otherwise mentioned, we eliminated all contigs less than 500 bp from each assembly prior to further analysis.

## Mapping

We aligned all quality-filtered reads to the reference metagenome with `bwa aln` (v0.7.7.r441) [23]. We aligned paired-end and orphaned reads separately. We then used `samtools` (v0.1.19) [29] to convert SAM files to BAM files for both paired-end and orphaned reads. To count the unaligned reads, we included only those records with the “4” flag in the SAM files [29].

## Assembly analysis using NUCmer

We used the NUCmer tool from MUMmer3.23 [30] to align assemblies to the reference genome with options `-coords -p`. Then we parsed the generated “.coords” file using a custom script `analyze_assembly.py`, and calculated several analysis metrics across all three assemblies at a 99% alignment identity.

## Reference-based analysis of the assemblies

We conducted reference-based analysis of the assemblies under two conditions. “Loose” alignment conditions used all available alignments, including redundant and overlapping alignments. “Strict” alignment conditions took only the longest alignment for any given contig, eliminating all other alignments.

The script `summarize-coords2.py` was used to calculate aligned coverage from the loose alignment conditions: each base in the reference was marked as “covered” if it was included in at least one alignment. The script

190 `analyze_ng50.py` was used to calculate NGA 50 for each individual refer-  
191 ence genome.

## 192 Analysis of chimeric misassemblies

193 We analyzed each assembly for chimeric misassemblies by counting the num-  
194 ber of contigs that contained matches to two distinct reference genomes. In  
195 order to remove secondary alignments from consideration, we included only  
196 the longest non-overlapping NUCmer alignments for each contig at a mini-  
197 mum alignment identity of 99%. We then used the script `analyze_chimeric2.py`  
198 to find individual contigs that matched more than one distinct reference  
199 genome. As a negative control on our analysis, we verified that this ap-  
200 proach yielded no positive results when applied to the alignments of the  
201 reference metagenome against itself.

## 202 Analysis of unmapped reads

203 We conducted assembly and analysis of unmapped reads with MEGAHIT,  
204 NUCmer, and sourmash as above. The new GenBank genomes are listed in  
205 the Zenodo archive at the file `accession-list-unmapped.txt` and for con-  
206 venience are available for download at the archival URL <https://osf.io/34ef8/>.

## 207 Results

### 208 The raw data is high quality.

209 The reads contain 11,072,579,096 bp (11.07 Gbp) in 109,629,496 reads with  
210 101.0 average length (2x101bp Illumina HiSeq).

211 Trimming removed 686,735 reads (0.63%). After trimming, we retained  
212 108,422,358 paired reads containing 10.94 Gbp with an average length of  
213 100.9 bases. A total of 46.56 Mbp remained in 520,403 orphan reads with  
214 an average length of 89.5 bases. In total, the quality trimmed data contained  
215 10.98 Gbp in 108,942,761 reads. This quality trimmed (“QC”) data set was  
216 used as the basis for all further analyses.

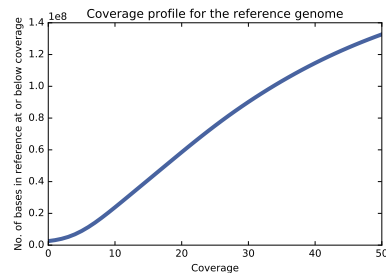


Figure 1: Cumulative coverage profile for the reference metagenome, based on read mapping.

Table 1: Jaccard containment of the reference in the reads

k-mer size	% reference in reads
21	96.8%
31	95.9%
41	94.9%
51	94.1%

217 **The reference metagenome is not completely present in the**  
218 **reads.**

219 We next evaluated the fraction of the reference genome covered by at least  
220 one read (see Methods for details). Quality filtered reads cover 203,058,414  
221 (98.76%) bases of the reference metagenome (205,603,715 bp total size). Fig-  
222 ure 1 shows the cumulative coverage profile of the reference metagenome,  
223 and the percentage of bases with that coverage. Most of the reference  
224 metagenome was covered at least minimally; only 3.33% of the reference  
225 metagenome had mapping coverage  $<5$ , and 1.24% of the bases in the ref-  
226 erence were not covered by any reads in the QC data set.

227 In order to evaluate reconstructability with De Bruijn graph assemblers,  
228 we next examined k-mer containment of the reference in the reads for  $k$  of  
229 21, 31, 41, and 51 (Table 1). The k-mer overlap decreases from 96.8% to  
230 94.1% as the k-mer size increases. This could be caused by low coverage of  
231 some portions of the reference and/or variation between the reads and the  
232 reference.



233 **Some individual reference genomes are poorly represented in**  
 234 **the reads.**

Table 2: Top uncovered genomes

Genome	Read coverage
<i>Desulfovibrio vulgaris</i> DP4	93.2%
<i>Thermus thermophilus</i> HB27	91.1%
<i>Enterococcus faecalis</i> V583	74.6%
<i>Fusobacterium nucleatum</i>	47.6%

235 To see if specific reference genomes exhibited low coverage, we analyzed  
 236 read mapping coverage for individual genomes. Of the 64 reference genomes  
 237 used in the metagenome, 60 had a per-base mapping coverage above 95%.  
 238 The remaining four varied significantly (Table 2), with *F. nucleatum* the  
 239 lowest – only 47.6% of the bases in the reference genome are covered by one  
 240 or more mapped reads.

241 We next did a 51-mer containment analysis of each reference genome in  
 242 the reads; k=51 was chosen so as to be specific to strain content [26]. 99%  
 243 or more of the constituent 51-mers for 51 of the 64 reference genomes were  
 244 present in the reads, suggesting that each of the 51 genomes was entirely  
 245 present at some minimal coverage.

246 We excluded the remaining 13 genomes (see Table 3) from any fur-  
 247 ther reference-based analysis because interpreting recovery and misassembly  
 248 statistics for these genomes would be confounding; also see the discussion of  
 249 strain variants, below.

## 250 **MEGAHIT is the fastest and lowest-memory assembler eval-** 251 **uated**

252 We ran three commonly used metagenome assemblers on the QC data set:  
 253 IDBA-UD, MetaSPAdes, and MEGAHIT. We recorded the time and mem-  
 254 ory usage of each (Table 4). In computational requirements, MEGAHIT  
 255 outperformed both MetaSPAdes and IDBA-UD, , producing an assembly in  
 256 1.5 hours (“wall time”) – 1.6 times faster than IDBA and 2.6 times faster  
 257 than MetaSPAdes. MEGAHIT used only 10 GB of RAM as requested –  
 258 about 60% of the memory used by IDBA and a third of the memory used by  
 259 MetaSPAdes. CPU time measurements (which include processing on multi-  
 260 ple CPU cores) show that all three assemblers use multiple cores effectively.

Table 3: Genomes removed from reference for low 51-mer presence

51-mers in reads	Genome
98.7	<i>Leptothrix cholodnii</i>
98.7	<i>Haloferax volcanii</i> DS2
98.6	<i>Salinispora tropica</i> CNB-440
97.4	<i>Deinococcus radiodurans</i>
97.2	<i>Zymomonas mobilis</i>
97.1	<i>Ruegeria pomeroyi</i>
96.8	<i>Shewanella baltica</i> OS223
95.5	<i>B. bronchiseptica</i> D989
94.5	<i>Burkholderia xenovorans</i>
72.0	<i>Desulfovibrio vulgaris</i> DP4
65.0	<i>Thermus thermophilus</i> HB27
53.4	<i>Enterococcus faecalis</i>
4.7	<i>Fusobacterium nucleatum</i> ATCC 25586

Table 4: Running Time and Memory Utilization

Assembler	CPU time	Wall time	RAM (Max RSS)
MEGAHIT	1191m	1h 33m	10 GB
IDBA-UD	1904m	2h 27m	17 GB
MetaSPAdes	2554m	4h 7m	28 GB

261 The assemblies contain most of the raw data

Table 5: Read and high-abundance (> 5) k-mer exclusion from assemblies

Assembly	Unmapped Reads	51-mers omitted
IDBA	3,328,674 (3.05%)	2.4%
MetaSPAdes	3,844,123 (3.52%)	3.2%
MEGAHIT	2,737,640 (2.51%)	2.8%

262 We assessed read inclusion in assemblies by mapping the QC reads to  
 263 the length-filtered assemblies and counting the remaining unmapped reads.  
 264 Depending on the assembly, between 2.7 million and 3.9 million reads (2.5-  
 265 3.5%) did not map to the assemblies (Table 5). All of the assemblies included  
 266 the large majority of high-abundance 51-mers (more than 96.8% in all cases).

267 **Much of the reference is covered by the assemblies.**

Table 6: Contig coverage of reference with loose alignment conditions.

Assembly	bases aligned	duplication	51-mers
MEGAHIT	94.8%	1.0%	96.7%
MetaSPAdes	93.1%	1.1%	96.2%
IDBA	93.6%	0.98%	97.2%

268 We next evaluated the extent to which the assembled contigs recovered  
 269 the “known/true” metagenome sequence by aligning each assembly to the  
 270 adjusted reference (Table 6). Each of the three assemblers generates contigs  
 271 that cover more than 93.1% of the reference metagenome at high identity  
 272 (99%) with little duplication (approximately 1%). All three assemblies con-  
 273 tain between 96.2% and 97.2% of the 51-mers in the reference.

274 At 99% identity with the loose mapping approach, approximately 2.5% of  
 275 the reference is missed by all three assemblers, while 1.7% is uniquely covered  
 276 by MEGAHIT, 0.74% is uniquely covered by MetaSPAdes, and 0.64% is  
 277 uniquely covered by IDBA.

278 **The generated contigs are broadly accurate.**

Table 7: Contig accuracy measured by reference coverage with strict alignment.

Assembly	% covered
MEGAHIT	89.3%
IDBA	87.7%
MetaSPAdes	83.4%

279 When counting only the best (longest) alignment per contig at a 99%  
 280 identity threshold, each of the three assemblies recovers more than 87.3% of  
 281 the reference, with MEGAHIT recovering the most – 89.3% of the reference  
 282 (Table 7).

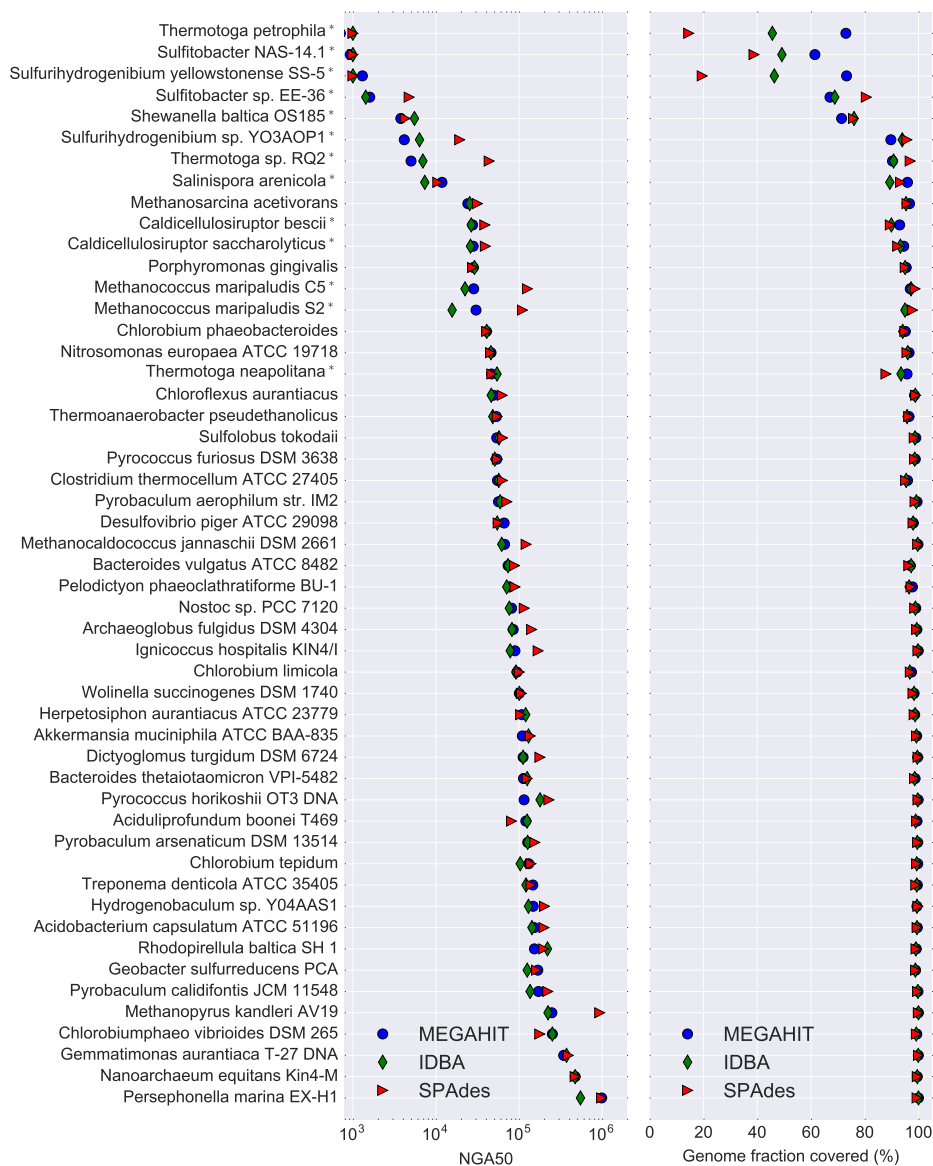


Figure 2: NGA50 and genome fraction covered, by genome and assembler. A '\*' after the name indicates the presence of at least one other genome with > 2% Jaccard similarity at k=31 in the community. Where NGA50 cannot be calculated due to poor coverage, a marker is placed at 1kb.

283 **Individual genome statistics vary widely in the assemblies.**

284 We computed the NGA50 for each individual genome and assembly in order  
 285 to compare assembler performance on genome recovery (see left panel of Fig-  
 286 ure 2). The NGA50 statistics for individual genomes vary widely, but there  
 287 are consistent assembler-specific trends: IDBA yields the lowest NGA50 for  
 288 28 of the 51 genomes, while MetaSPAdes yields the highest NGA50 for 32  
 289 of the 51 genomes.

290 We also evaluated aligned coverage per genome for each of the three  
 291 assemblies (right panel, Figure 2). We found that 13 of the 51 genomes were  
 292 missing 5% or more of bases in at least one assembly, despite all 51 genomes  
 293 having 99% or higher read- and 51-mer coverage.

294 There are 12 genomes with k=31 Jaccard similarity greater than 2%  
 295 to other genomes in the community, and these (denoted by '\*' after the  
 296 name) typically had lower NGA50 and aligned coverage numbers than other  
 297 genomes. In particular, these constituted 12 of the 13 genomes missing 5%  
 298 or more of their content, and the lowest eight NGA50 numbers.

299 **Longer contigs are less likely to be chimeric.**

Table 8: Chimeric contigs by contig length.

Assembly	> 50kb	> 5kb	> 500 bp
IDBA	0	1	7 (0.06%)
MEGAHIT	1	4	14 (0.13%)
MetaSPAdes	0	3	30 (0.48%)

300 Chimerism is the formation of contigs that include sequence from multi-  
 301 ple genomes. We evaluated the rate of chimerism in contigs at three different  
 302 contig length cutoffs: 500bp, 5kb, and 50kb (Table 8). We found that the  
 303 percentage of contigs that match to the genomes of two or more different  
 304 species drop as the minimum contig size increases, to the point where only  
 305 the MEGAHIT assembly had a single chimeric contig longer than 50kb.  
 306 Overall, chimeric misassemblies were rare, with no assembler generating  
 307 more than 30 chimeric contigs out of thousands of total contigs.

308 **The unmapped reads contain strain variants of reference genomes.**

309 Approximately 4.8 million reads (4.4%) from the QC data set did not map  
 310 anywhere in the reference provided by the authors of [12]. We extracted

Table 9: GenBank genomes detected in assembly of unmapped reads

match	GenBank genome
44.1%	<i>Fusobacterium</i> sp. OBRC1
23.0%	<i>P. ruminis</i> strain ML2
18.2%	<i>Thermus thermophilus</i> HB8
7.7%	<i>P. ruminis</i> strain CGMCC
8.2%	<i>Enterococcus faecalis</i> M7
7.3%	<i>F. nucleatum</i> 13_3C
3.7%	<i>F. nucleatum</i> subsp. <i>polymorphum</i>
2.9%	<i>Fusobacterium hwasookii</i>
1.0%	<i>E. coli</i> isolate YS
1.7%	<i>F. nucleatum</i> subsp. <i>polymorphum</i> , alt.
1.9%	<i>F. nucleatum</i> subsp. <i>vincentii</i>

and assembled these reads in isolation using MEGAHIT, yielding 6.5 Mbp of assembly in 1711 contigs > 500bp in length. We then did a k-mer inclusion analysis of this assembly against all of the GenBank genomes at k=31, and estimated the fraction of the k-mers that belonged to different species (Table 9). We find that 51.1% of the k-mer content of these contigs positively match to a genome present in GenBank but not in the reference metagenome.

To verify these assignments, we aligned the MEGAHIT assembly of unmapped reads to the GenBank genomes in Table 9 with NUCmer using “loose” alignment criteria. We found that 1.78 Mbp of the contigs aligned at 99% identity or better to these GenBank genomes. We also confirmed that, as expected, there are no matches in this assembly to the full updated reference metagenome.

We note that all but the two *P. ruminis* matches and the *E. coli* isolate YS are strain variants of species that are part of the defined community but are not completely present in the reads (see Table 2). For *Proteiniclasticum ruminis*, there is no closely related species in the mock community design, and very little of the MEGAHIT assembly aligns to known *P. ruminis* genomes at 99%. However, there are many alignments to *P. ruminis* at 94% or higher, for approximately 2.73 Mbp total. This suggests that the unmapped reads contain at least some data from a novel species of *Proteiniclasticum*; this matches the observation in [12] of a contaminating genome from an unknown *Clostridium* spp., as at the time there was no *P. ruminis* genome.

## 335 Discussion

### 336 Assembly recovers basic content sensitively and accurately.

337 All three assemblers performed well in assembling contigs from the con-  
338 tent that was fully present in reads and k-mers. After length filtering,  
339 all three assemblies contained more than 95% of the reference (Table 6);  
340 even with removal of secondary alignments, more than 87% was recovered  
341 by each assembler (Table 7). About half the constituent genomes had an  
342 NGA50 of 50kb or higher (Figure 2), which, while low for current Illumina  
343 single-genome sequencing, is sufficient to recover operon-level relationships  
344 for many genes.

### 345 The presence of multiple closely related genomes confounds 346 assembly.

347 In agreement with CAMI, we also find that the presence of closely related  
348 genomes in the metagenome causes loss of assembly [3]. This is clearly shown  
349 by Figure 2, where 12 of the bottom 14 genomes by NGA50 (left panel)  
350 also exhibit poor genome recovery by assembly (right panel). Interestingly,  
351 different assemblers handle this quite differently, with e.g. MetaSPAdes  
352 failing to recover essentially any of *Thermotoga petrophila*, while MEGAHIT  
353 recovers 73%. The presence of nearby genomes is an almost perfect predictor  
354 that one or more assembler will fail to recover 5% or more - of the 13/51  
355 genomes for which less than 95% is recovered, 12 of them have close genomes  
356 in the community. Interestingly, very little similarity is needed - all genomes  
357 with Jaccard similarity of 2% or higher at k=31 exhibit these problems.

358 The *Shewanella baltica* OS185 genome is a good example: there are two  
359 strain variants, OS185 and OS223, present in the defined community. Both  
360 are present at more than 99% in the reads, and more than 98% in 51-mers,  
361 but only 75% of *S. baltica* OS185 and 50% of *S. baltica* OS223 are recovered  
362 by assemblers. This is a clear case of “strain confusion” where the assemblers  
363 simply fail to output contigs for a substantial portion of the two genomes.

364 Another interest of this study was to examine cross-species chimeric as-  
365 sembly, in which a single contig is formed from multiple genomes. In Table 8,  
366 we show that there is relatively little cross-species chimerism. Surprisingly,  
367 what little is present is length-dependent: longer contigs are less likely to  
368 be chimeric. This might well be due to the same “strain confusion” effect  
369 as above, where contigs that share paths in the assembly graphs are broken  
370 in twain.

371 **MEGAHIT performs best by several metrics.**

372 MEGAHIT is clearly the most efficient computationally, outperforming both  
373 MetaSPAdes and IDBA in memory and time (Table 4). The MEGAHIT  
374 assembly also included more of the reads than either IDBA or MetaSPAdes,  
375 and omitted only 0.4% more of the unique 51-mers from the reads than  
376 IDBA. MEGAHIT covered more of the reference genome with both loose  
377 and strict alignments (Table 6 and Table 7), with little duplication. This is  
378 clearly because of MEGAHIT’s generally superior performance in recovering  
379 the genomes of closely related strains (Figure 2, right panel). The sum  
380 “fraction of genome recovered” is arguably the most important measure of  
381 a metagenome assembler (see [5] in particular) and here MEGAHIT excels  
382 for individual genomes even in the presence of strain variation.

383 In general other studies have found that MEGAHIT excels in recovery of  
384 sequence through assembly [3, 16] and is considerably more computationally  
385 efficient than most other assemblers [3, 15]. However, studies have also  
386 shown that MEGAHIT produces more misassemblies than other assemblers  
387 [3] and performs poorly on high coverage portions of the data set [5] Thus  
388 while we can recommend MEGAHIT as a good first assembler, we can also  
389 not unambiguously recommend it as the only assembler to use.

390 When comparing details of sequence recovery between the assemblers,  
391 the assembly content differs by only a small amount when loose alignments  
392 are allowed: all three assemblers miss more content (approximately 2.5% of  
393 the reference) than they generate uniquely (1.7% or less). In addition to  
394 preferring no one assembler over any other, this suggests that combining as-  
395 semblies may have little value in terms of recovering additional metagenome  
396 content. The genome alignment statistics in Figure 2 suggest that much of  
397 this differential assembly content is due to the impact of strains.

398 **The missing reference may be present in strain variants of the**  
399 **intended species.**

400 Several individual genomes are missing in measurable portion from the QC  
401 reads (Table 2), and many QC reads (4.4% of 108m) did not map to the full  
402 reference metagenome. These appear to be related issues: upon analysis of  
403 the unmapped reads against GenBank, we find that many of the contigs as-  
404 sembled from the unmapped reads can be assigned to strain variants of the  
405 species in the mock community (Table 9) and align closely to the identified  
406 genomes. This suggests that the constructors of the mock community may  
407 have unintentionally included strain variants of *Fusobacterium nucleatum*,



408 *Thermus thermophilus* HB27, and *Enterococcus faecalis*; note that the mi-  
409 crobes used were sourced from the community rather than the ATCC (M.  
410 Podar, pers. communication). In addition, we detect what may be por-  
411 tions of a novel member of the *Proteiniclasticum* genus in the assembly of  
412 these reads - this is likely the *Clostridium* spp. detected through amplicon  
413 sequencing in [12].

414 Without returning to the original DNA samples, it is impossible to con-  
415 clusively confirm that unintended strains were used in the construction of the  
416 mock community. In particular, our analysis is dependent on the genomes in  
417 GenBank: the genomes we detect in the contigs are clearly closely related to  
418 GenBank genomes not in the reference metagenome, based on k-mer anal-  
419 ysis and contig alignment. However, GenBank is unlikely to contain the  
420 exact genomes of the actually included strain variants, rendering conclusive  
421 identification impossible.

## 422 Conclusions

423 Overall, assembly of this mock community works well, with good recovery  
424 of known genomic sequence for the majority of genomes. All three assem-  
425 blers that we evaluated recover similar amounts of most genomic sequence,  
426 but (recapitulating several other studies [3, 5, 15]) MEGAHIT is compu-  
427 tationally the most efficient of the three. We note that assembly resolves  
428 substantial portions of several previously undetected strain variants, as well  
429 as recovering a substantial portion of a novel *Proteiniclasticum* spp. that  
430 was detected via amplicon analysis in [12], suggesting that assembly is a  
431 useful complement to amplicon or reference-based analyses.

432 The presence of closely related strains is a major confounder of metagenome  
433 assembly, and causes assemblers to drop considerable portions of genomes  
434 that (based on read mapping and k-mer inclusion) are clearly present. In this  
435 relatively simple community, this strain confusion is present but does not  
436 dominate the assembly. However, real microbial communities are likely to  
437 have many closely related strains and any resulting loss of assembly would  
438 be hard to detect in the absence of good reference genomes. While high  
439 polymorphism rates in e.g. animal genomes are known to cause duplication  
440 or loss of assembly, some solutions have emerged that make use of assump-  
441 tions of uniform coverage and diploidy [31]. These solutions cannot however  
442 be transferred directly to metagenomes, which have unknown abundance  
443 distributions and strain content.

444 An additional concern is that metagenome assemblies are often per-  
445 formed after pooling data sets to increase coverage (e.g. [4, 32]); this pooled  
446 data is more likely to contain multiple strains, which would then in turn  
447 adversely affect assembly of strains. This may not be resolvable within the  
448 current paradigm of assembly, which focuses on outputting linear assem-  
449 blies that cannot properly represent strain variation. The human genomics  
450 community is moving towards using *reference graphs*, which can represent  
451 multiple incompatible variants in a single data structure [33]; this approach,  
452 however, requires high-quality isolate reference genomes, which are generally  
453 unavailable for environmental microbes.

454 Long read sequencing (and related technologies) will undoubtedly help  
455 resolve strain variation in the future, but even with highly accurate long-  
456 read sequencing, current sequencing depth is still too low to resolve deep  
457 environmental metagenomes [34, 35]. It is unclear how well long error-  
458 prone reads (such as those output by Pacific Biosciences SMRT [36] and  
459 Oxford Nanopore instruments [37]) will perform on complex metagenomes:  
460 with high error rates, deep coverage of each individual genome is required  
461 to achieve accurate assembly, and this may not be easily obtainable for  
462 complex communities. Single-molecule barcoding (e.g. 10X Genomics [38])  
463 and HiC approaches [39] show promise but these remain untested on well-  
464 defined complex communities and are still challenged by the complexity of  
465 complex environmental metagenomes; see [40, 41, 42].

466 Much of our analysis above depends on having a high-quality “mock”  
467 metagenome. While computationally constructed synthetic communities  
468 and computational “spike-ins” to real data sets can provide valuable controls  
469 (e.g. see [15] and [43]) we strongly believe that standardized communities  
470 constructed *in vitro* and sequenced with the latest technologies are critical  
471 to the evaluation of both canonical and emerging tools, e.g. efforts such as  
472 [44]. From the perspective of tool evaluation, we disagree somewhat with  
473 Vollmers et al. [5]: good metagenome tool evaluation necessarily depends  
474 on mock communities that are as realistic as we can make them. Likewise,  
475 from the perspective of bench biologists, actually sequencing real DNA is  
476 critical because it can evaluate confounding effects such as kit contamina-  
477 tion [45]. Large-scale studies of computational approaches systematically  
478 applied to mock communities such as CAMI [3] can then provide fair com-  
479 parisons of entire toolchains (wet and dry combined) applied to these mock  
480 communities.

481 We omitted two important questions in this study: binning and choice  
482 of parameters. We chose not to evaluate genome binning because most bin-

ning strategies either operate post-assembly (see e.g. [46]), in which case the challenges with assembly discussed above will apply; or require multiple samples (e.g. [47]), which we do not have. We also chose to use only default parameters with all three assemblers, for two reasons. First, we are not aware of any effective automated approaches for determining the “best” set of parameters or evaluating the output for metagenome assemblers, other than those integrated into the assemblers themselves (e.g. the choice of k-mer sizes by MEGAHIT and MetaSPAdes), and absent such guidance we do not feel comfortable blessing any particular set of parameters; here the choice of default parameters is parsimonious (and also see [48] for the dangers of poorly chosen objective functions). Second, any parameter exploration pipeline would not only need to be automated but would need to run multiple assemblies, whose time and resource usage should be measured; in this case, any comparison based on runtime of the parameter choice pipeline should naturally favor MEGAHIT because of its advantage in computational efficiency.

## **Author contributions**

SA, LI and CTB developed, tested, and executed the analytical pipeline. SA and CTB created the tables and figures and wrote the paper.

## **Competing interests**

No competing interest to our knowledge.

## **Grant information**

This work is funded by Gordon and Betty Moore Foundation Grant GBMF4551 and NIH NHGRI R01 grant HG007513-03, both to CTB.

## **Acknowledgments**

We thank Michael R. Crusoe and Phillip T. Brooks for input on analysis and pipeline development. We thank Migun Shakya, Mircea Podar, Jiarong Guo, Harald R. Gruber-Vodicka, Juliane Wippler, Krista Ternus, and Stephen Turner for valuable comments on drafts of this manuscript.

## 512 References

- 513 [1] Jay Ghurye, Victoria Cepeda-Espinoza, and Mihai Pop. Metagenomic assem-  
514 bly: Overview, challenges and applications. *The Yale Journal of Biology and*  
515 *Medicine*, 89(3):353–362, 2016.
- 516 [2] Nikos C. Kyrpides, Philip Hugenholtz, Jonathan A. Eisen, Tanja Woyke,  
517 Markus Göker, Charles T. Parker, Rudolf Amann, Brian J. Beck, Patrick S. G.  
518 Chain, Jongsik Chun, Rita R. Colwell, Antoine Danchin, Peter Dawyndt, Tom  
519 Dedeurwaerdere, Edward F. DeLong, John C. Detter, Paul De Vos, Timothy J.  
520 Donohue, Xiu-Zhu Dong, Dusko S. Ehrlich, Claire Fraser, Richard Gibbs, Jack  
521 Gilbert, Paul Gilna, Frank Oliver Glöckner, Janet K. Jansson, Jay D. Keasling,  
522 Rob Knight, David Labeda, Alla Lapidus, Jung-Sook Lee, Wen-Jun Li, Juncai  
523 MA, Victor Markowitz, Edward R. B. Moore, Mark Morrison, Folker Meyer,  
524 Karen E. Nelson, Moriya Ohkuma, Christos A. Ouzounis, Norman Pace, Julian  
525 Parkhill, Nan Qin, Ramon Rossello-Mora, Johannes Sikorski, David Smith,  
526 Mitch Sogin, Rick Stevens, Uli Stingl, Ken ichiro Suzuki, Dorothea Taylor,  
527 Jim M. Tiedje, Brian Tindall, Michael Wagner, George Weinstock, Jean Weis-  
528 senbach, Owen White, Jun Wang, Lixin Zhang, Yu-Guang Zhou, Dawn Field,  
529 William B. Whitman, George M. Garrity, and Hans-Peter Klenk. Genomic  
530 encyclopedia of bacteria and archaea: Sequencing a myriad of type strains.  
531 *PLoS Biology*, 12(8):e1001920, aug 2014. doi: 10.1371/journal.pbio.1001920.  
532 URL <https://doi.org/10.1371/journal.pbio.1001920>.
- 533 [3] Alexander Sczyrba, Peter Hofmann, Peter Belmann, David Koslicki, Stefan  
534 Janssen, Johannes Droege, Ivan Gregor, Stephan Majda, Jessika Fiedler,  
535 Eik Dahms, Andreas Bremges, Adrian Fritz, Ruben Garrido-Oter, Tue  
536 Sparholt Jorgensen, Nicole Shapiro, Philip D Blood, Alexey Gurevich, Yang  
537 Bai, Dmitrij Turaev, Matthew Z DeMaere, Rayan Chikhi, Niranjana Nagara-  
538 jan, Christopher Quince, Lars Hestbjerg Hansen, Soren J Sorensen, Burton  
539 K H Chia, Bertrand Denis, Jeff L Froula, Zhong Wang, Robert Egan, Dong-  
540 wan Don Kang, Jeffrey J Cook, Charles Deltel, Michael Beckstette, Claire  
541 Lemaitre, Pierre Peterlongo, Guillaume Rizk, Dominique Lavenier, Yu-Wei  
542 Wu, Steven W Singer, Chirag Jain, Marc Strous, Heiner Klingenberg, Peter  
543 Meinicke, Michael Barton, Thomas Lingner, Hsin-Hung Lin, Yu-Chieh Liao,  
544 Genivaldo Gueiros Z. Silva, Daniel A Cuevas, Robert A Edwards, Surya Saha,  
545 Vitor C Piro, Bernhard Y Renard, Mihai Pop, Hans-Peter Klenk, Markus  
546 Goeker, Nikos Kyrpides, Tanja Woyke, Julia A Vorholt, Paul Schulze-Lefert,  
547 Edward M Rubin, Aaron E Darling, Thomas Rattei, and Alice C McHardy.  
548 Critical assessment of metagenome interpretation - a benchmark of compu-  
549 tational metagenomics software. *bioRxiv*, 2017. doi: 10.1101/099127. URL  
550 <http://biorxiv.org/content/early/2017/01/09/099127>.
- 551 [4] I. Sharon, M. J. Morowitz, B. C. Thomas, E. K. Costello, D. A. Relman,  
552 and J. F. Banfield. Time series community genomics analysis reveals rapid  
553 shifts in bacterial species, strains, and phage during infant gut colonization.

- 554 *Genome Research*, 23(1):111–120, aug 2012. doi: 10.1101/gr.142315.112. URL  
555 <https://doi.org/10.1101/gr.142315.112>.
- 556 [5] John Vollmers, Sandra Wiegand, and Anne-Kristin Kaster. Comparing and evaluating metagenome assembly tools from a microbiologist’s perspective - not only size matters! *PLOS ONE*, 12  
557 (1):e0169662, jan 2017. doi: 10.1371/journal.pone.0169662. URL  
558 <https://doi.org/10.1371/journal.pone.0169662>.  
559
- 561 [6] Jorge F Vázquez-Castellanos, Rodrigo García-López, Vicente Pérez-Brocal, Miguel Pignatelli, and Andrés Moya. Comparison of different assembly and  
562 annotation tools on analysis of simulated viral metagenomic communities in  
563 the gut. *BMC genomics*, 15(1):1, 2014.  
564
- 565 [7] Konstantinos Mavromatis, Natalia Ivanova, Kerrie Barry, Harris Shapiro, Eugene Goltsman, Alice C McHardy, Isidore Rigoutsos, Asaf Salamov, Frank  
566 Korzeniewski, Miriam Land, et al. Use of simulated data sets to evaluate the  
567 fidelity of metagenomic processing methods. *Nature methods*, 4(6):495–500,  
568 2007.  
569
- 570 [8] David B Jaffe, Jonathan Butler, Sante Gnerre, Evan Mauceli, Kerstin  
571 Lindblad-Toh, Jill P Mesirov, Michael C Zody, and Eric S Lander. Whole-genome sequence assembly for mammalian genomes: Arachne 2. *Genome  
572 research*, 13(1):91–96, 2003.  
573
- 574 [9] Samuel Aparicio, Jarrod Chapman, Elia Stupka, Nik Putnam, Jer-ming Chia, Paramvir Dehal, Alan Christoffels, Sam Rash, Shawn Hoon, Arian Smit, et al. Whole-genome shotgun assembly and analysis of the genome of *fugu rubripes*.  
575 *Science*, 297(5585):1301–1310, 2002.  
576  
577
- 578 [10] Anveshi Charuvaka and Huzefa Rangwala. Evaluation of short read metagenomic assembly. *BMC genomics*, 12(2):1, 2011.  
579
- 580 [11] Jared T Simpson, Kim Wong, Shaun D Jackman, Jacqueline E Schein, Steven JM Jones, and Inanç Birol. Abyss: a parallel assembler for short read  
581 sequence data. *Genome research*, 19(6):1117–1123, 2009.  
582
- 583 [12] Shakya Migun, Christopher Quince, James Campbell, Zamin Yang, Christopher Schadt, and Mircea Podar. Comparative metagenomic and rRNA microbial  
584 diversity characterization using archaeal and bacterial synthetic communities. *Environmental Microbiology*, 15(6):1882–1899, 2013.  
585  
586
- 587 [13] Brandon K. B. Seah and Harald R. Gruber-Vodicka. gbtools: Interactive visualization of metagenome bins in R. *Frontiers in Microbiology*,  
588 6, dec 2015. doi: 10.3389/fmicb.2015.01451. URL  
589 <https://doi.org/10.3389/fmicb.2015.01451>.  
590

- [14] Dinghua Li, Ruibang Luo, Chi-Man Liu, Chi-Ming Leung, Hing-Fung Ting, Kunihiko Sadakane, Hiroshi Yamashita, and Tak-Wah Lam. MEGAHIT v1.0: A fast and scalable metagenome assembler driven by advanced methodologies and community practices. *Methods*, 102:3–11, jun 2016. doi: 10.1016/j.ymeth.2016.02.020. URL <https://doi.org/10.1016/j.ymeth.2016.02.020>.
- [15] Andries Johannes van der Walt, Marc Warwick Van Goethem, Jean-Baptiste Ramond, Thulani Peter Makhalanyane, Oleg Reva, and Don Arthur Cowan. Assembling metagenomes, one community at a time. *bioRxiv*, 2017. doi: 10.1101/120154. URL <http://biorxiv.org/content/early/2017/06/06/120154>.
- [16] William W. Greenwald, Niels Klitgord, Victor Seguritan, Shibu Yooseph, J. Craig Venter, Chad Garner, Karen E. Nelson, and Weizhong Li. Utilization of defined microbial communities enables effective evaluation of meta-genomic assemblies. *BMC Genomics*, 18(1), apr 2017. doi: 10.1186/s12864-017-3679-5. URL <https://doi.org/10.1186/s12864-017-3679-5>.
- [17] Yu Peng, Henry C.M. Leung, S.M. Yiu, and Francis Y.L. Chin. Idba-ud: a de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth. *Bioinformatics*, 28:1420–1428, 2012.
- [18] Sergey Nurk, Dmitry Meleshko, Anton Korobeynikov, and Pavel A. Pevzner. metaSPAdes: a new versatile metagenomic assembler. *Genome Research*, 27(5):824–834, mar 2017. doi: 10.1101/gr.213959.116. URL <https://doi.org/10.1101/gr.213959.116>.
- [19] Dinghua Li, Ruibang Luo, Chi-Man Liu, Chi-Ming Leung, Hing-Fung Ting, Kunihiko Sadakane, Hiroshi Yamashita, and Tak-Wah Lam. Megahit v1. 0: A fast and scalable metagenome assembler driven by advanced methodologies and community practices. *Methods*, 102:3–11, 2016.
- [20] H Chitsaz, JL Yee-Greenbaum, G Tesler, MJ Lombardo, CL Dupont, JH Badger, M Novotny, DB Rusch, LJ Fraser, NA Gormley, O Schulz-Trieglaff, GP Smith, DJ Evers, PA Pevzner, and RS Lasken. Efficient de novo assembly of single-cell bacterial genomes from short-read data sets. *Nat Biotechnol*, 29(10):915–21, 2011.
- [21] Anthony M. Bolger, Marc Lohse, and Bjoern Usadel. Trimmomatic: A flexible trimmer for illumina sequence data. *Bioinformatics*, 30(15):2114–2120, 2014.
- [22] Matthew D MacManes. On the optimal trimming of high-throughput mrna sequence data. *Frontiers in genetics*, 5:13, 2014.
- [23] Heng Li and Richard Durbin. Fast and accurate short read alignment with burrows–wheeler transform. *Bioinformatics*, 25(14):1754–1760, 2009.

- 629 [24] C. Titus Brown and Luiz Irber. sourmash: a library for MinHash sketch-  
630 ing of DNA. *The Journal of Open Source Software*, 1(5), sep 2016. doi:  
631 10.21105/joss.00027. URL <https://doi.org/10.21105/joss.00027>.
- 632 [25] Brian D. Ondov, Todd J. Treangen, Páll Melsted, Adam B. Mal-  
633 lonee, Nicholas H. Bergman, Sergey Koren, and Adam M. Phillippy.  
634 Mash: fast genome and metagenome distance estimation using MinHash.  
635 *Genome Biology*, 17(1), jun 2016. doi: 10.1186/s13059-016-0997-x. URL  
636 <https://doi.org/10.1186/s13059-016-0997-x>.
- 637 [26] David Koslicki and Daniel Falush. Metapalette: a k-mer painting approach  
638 for metagenomic taxonomic profiling and quantification of novel strain vari-  
639 ation. *mSystems*, 1(3), 2016. doi: 10.1128/mSystems.00020-16. URL  
640 <http://msystems.asm.org/content/1/3/e00020-16>.
- 641 [27] Zhang Qingpeng, Awad Sherine, and Brown Titus. Crossing the streams:  
642 a framework for streaming analysis of short dna sequencing reads. *PeerJ*  
643 *PrePrints 3:e1100* <https://dx.doi.org/10.7287/peerj.preprints.890v1>, 2015.
- 644 [28] MR Crusoe, HF Alameldin, S Awad, E Boucher, A Caldwell, R Cartwright,  
645 A Charbonneau, B Constantinides, G Edvenson, S Fay, J Fenton, T Fenzl,  
646 J Fish, L Garcia-Gutierrez, P Garland, J Gluck, I Gonzlez, S Guermond,  
647 J Guo, A Gupta, JR Herr, A Howe, A Hyer, A Hrpfer, L Irber, R Kidd,  
648 D Lin, J Lippi, T Mansour, P McA’Nulty, E McDonald, J Mizzi, KD Mur-  
649 ray, JR Nahum, K Nanlohy, AJ Nederbragt, H Ortiz-Zuazaga, J Ory, J Pell,  
650 C Pepe-Ranne, ZN Russ, E Schwarz, C Scott, J Seaman, S Sievert, J Simp-  
651 son, CT Skenner, J Spencer, R Srinivasan, D Standage, JA Stapleton,  
652 SR Steinman, J Stein, B Taylor, W Trimble, HL Wiencko, M Wright,  
653 B Wyss, Q Zhang, e zyme, and CT Brown. The khmer software pack-  
654 age: enabling efficient nucleotide sequence analysis [version 1; referees: 2 ap-  
655 proved, 1 approved with reservations]. *F1000Research*, 4(900), 2015. doi:  
656 10.12688/f1000research.6924.1.
- 657 [29] Heng Li, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer,  
658 Gabor Marth, Goncalo Abecasis, Richard Durbin, et al. The sequence align-  
659 ment/map format and samtools. *Bioinformatics*, 25(16):2078–2079, 2009.
- 660 [30] Stefan Kurtz, Adam Phillippy, Arthur L Delcher, Michael Smoot, Martin  
661 Shumway, Corina Antonescu, and Steven L Salzberg. Versatile and open soft-  
662 ware for comparing large genomes. *Genome biology*, 5(2):1, 2004.
- 663 [31] J. H. Kim, M. S. Waterman, and L. M. Li. Diploid genome reconstruc-  
664 tion of *ciona intestinalis* and comparative analysis with *ciona savignyi*.  
665 *Genome Research*, 17(7):1101–1110, jun 2007. doi: 10.1101/gr.5894107. URL  
666 <https://doi.org/10.1101/gr.5894107>.

- [32] Ping Hu, Lauren Tom, Andrea Singh, Brian C. Thomas, Brett J. Baker, Yvette M. Piceno, Gary L. Andersen, and Jillian F. Banfield. Genome-resolved metagenomic analysis reveals roles for candidate phyla and other microbial community members in biogeochemical transformations in oil reservoirs. *mBio*, 7(1):e01669–15, jan 2016. doi: 10.1128/mbio.01669-15. URL <https://doi.org/10.1128/mbio.01669-15>.
- [33] Benedict Paten, Adam M Novak, Jordan M Eizenga, and Erik Garrison. Genome graphs and the evolution of genome inference. *Genome research*, 27(5):665–676, 2017.
- [34] Itai Sharon, Michael Kertesz, Laura A. Hug, Dmitry Pushkarev, Timothy A. Blauwkamp, Cindy J. Castelle, Mojgan Amirebrahimi, Brian C. Thomas, David Burstein, Susannah G. Tringe, Kenneth H. Williams, and Jillian F. Banfield. Accurate, multi-kb reads resolve complex populations and detect rare microorganisms. *Genome Research*, 25(4):534–543, feb 2015. doi: 10.1101/gr.183012.114. URL <https://doi.org/10.1101/gr.183012.114>.
- [35] Richard Allen White, Eric M. Bottos, Taniya Roy Chowdhury, Jeremy D. Zucker, Colin J. Brislawn, Carrie D. Nicora, Sarah J. Fansler, Kurt R. Glaesemann, Kevin Glass, and Janet K. Jansson. Molecule long-read sequencing facilitates assembly and genomic binning from complex soil metagenomes. *mSystems*, 1(3):e00045–16, jun 2016. doi: 10.1128/msystems.00045-16. URL <https://doi.org/10.1128/msystems.00045-16>.
- [36] J. Eid, A. Fehr, J. Gray, K. Luong, J. Lyle, G. Otto, P. Peluso, D. Rank, P. Baybayan, B. Bettman, A. Bibillo, K. Bjornson, B. Chaudhuri, F. Christians, R. Cicero, S. Clark, R. Dalal, A. deWinter, J. Dixon, M. Foquet, A. Gaertner, P. Hardenbol, C. Heiner, K. Hester, D. Holden, G. Kearns, X. Kong, R. Kuse, Y. Lacroix, S. Lin, P. Lundquist, C. Ma, P. Marks, M. Maxham, D. Murphy, I. Park, T. Pham, M. Phillips, J. Roy, R. Sebra, G. Shen, J. Sorenson, A. Tomaney, K. Travers, M. Trulson, J. Vieceli, J. Wegener, D. Wu, A. Yang, D. Zaccarin, P. Zhao, F. Zhong, J. Korlach, and S. Turner. Real-time DNA sequencing from single polymerase molecules. *Science*, 323(5910):133–138, jan 2009. doi: 10.1126/science.1162986. URL <https://doi.org/10.1126/science.1162986>.
- [37] Gerald M Cherf, Kate R Lieberman, Hytham Rashid, Christopher E Lam, Kevin Karplus, and Mark Akeson. Automated forward and reverse ratcheting of DNA in a nanopore at 5-AA precision. *Nature Biotechnology*, 30(4):344–348, feb 2012. doi: 10.1038/nbt.2147. URL <https://doi.org/10.1038/nbt.2147>.
- [38] Eli Moss, Alex Bishara, Ekaterina Tkachenko, Joyce B Kang, Tessa M Andermann, Christina Wood, Christine Handy, Hanlee Ji, Serafim Batzoglou, and Ami S Bhatt. De novo assembly of microbial genomes from human gut metagenomes using barcoded



- 707 short read sequences. *bioRxiv*, 2017. doi: 10.1101/125211. URL  
708 <http://biorxiv.org/content/early/2017/04/07/125211>.
- 709 [39] Caiti Smukowski Heil, Joshua N. Burton, Ivan Liachko, Anne Friedrich,  
710 Noah A. Hanson, Cody L. Morris, Joseph Schacherer, Jay Shendure,  
711 James H. Thomas, and Maitreya J. Dunham. Identification of a  
712 novel interspecific hybrid yeast from a metagenomic open fermentation  
713 sample using hi-c. *bioRxiv*, 2017. doi: 10.1101/150722. URL  
714 <http://biorxiv.org/content/early/2017/06/15/150722>.
- 715 [40] Joshua N. Burton, Ivan Liachko, Maitreya J. Dunham, and Jay Shendure.  
716 Species-level deconvolution of metagenome assemblies with hi-c-based contact  
717 probability maps. *G3*, 4(7):1339–1346, may 2014. doi: 10.1534/g3.114.011825.  
718 URL <https://doi.org/10.1534/g3.114.011825>.
- 719 [41] Martial Marbouty, Axel Cournac, Jean-François Flot, Hervé Marie-Nelly,  
720 Julien Mozziconacci, and Romain Koszul. Metagenomic chromosome con-  
721 formation capture (meta3c) unveils the diversity of chromosome organiza-  
722 tion in microorganisms. *eLife*, 3, dec 2014. doi: 10.7554/elife.03318. URL  
723 <https://doi.org/10.7554/elife.03318>.
- 724 [42] Christopher W. Beitel, Lutz Froenicke, Jenna M. Lang, Ian F. Korf,  
725 Richard W. Michelmore, Jonathan A. Eisen, and Aaron E. Darling. Strain- and  
726 plasmid-level deconvolution of a synthetic metagenome by sequencing proxim-  
727 ity ligation products. *PeerJ*, 2:e415, may 2014. doi: 10.7717/peerj.415. URL  
728 <https://doi.org/10.7717/peerj.415>.
- 729 [43] Adina Chuang Howe, Janet K Jansson, Stephanie A Malfatti, Susannah G  
730 Tringe, James M Tiedje, and C Titus Brown. Tackling soil diversity with the  
731 assembly of large, complex metagenomes. *Proceedings of the National Academy*  
732 *of Sciences*, 111(13):4904–4909, 2014.
- 733 [44] Bonnie L. Brown, Mick Watson, Samuel S. Minot, Maria C.  
734 Rivera, and Rima B. Franklin. MinION™ nanopore sequenc-  
735 ing of environmental metagenomes: a synthetic approach. *Giga-*  
736 *Science*, 6(3):1–10, feb 2017. doi: 10.1093/gigascience/gix007. URL  
737 <https://doi.org/10.1093/gigascience/gix007>.
- 738 [45] Susannah J Salter, Michael J Cox, Elena M Turek, Szymon T Calus,  
739 William O Cookson, Miriam F Moffatt, Paul Turner, Julian Parkhill,  
740 Nicholas J Loman, and Alan W Walker. Reagent and laboratory  
741 contamination can critically impact sequence-based microbiome analyses.  
742 *BMC Biology*, 12(1), nov 2014. doi: 10.1186/s12915-014-0087-z. URL  
743 <https://doi.org/10.1186/s12915-014-0087-z>.
- 744 [46] Cedric C Laczny, Christina Kiefer, Valentina Galata, Tobias Fehlmann,  
745 Christina Backes, and Andreas Keller. Busybee web: metagenomic data anal-

- 746 ysis by bootstrapped supervised binning and annotation. *Nucleic Acids Re-*  
747 *search*, page gkx348, 2017.
- 748 [47] Brian Cleary, Ilana Lauren Brito, Katherine Huang, Dirk Gevers, Terrance  
749 Shea, Sarah Young, and Eric J Alm. Detection of low-abundance bacte-  
750 rial strains in metagenomic datasets by eigengenome partitioning. *Nature*  
751 *Biotechnology*, 33(10):1053–1060, sep 2015. doi: 10.1038/nbt.3329. URL  
752 <https://doi.org/10.1038/nbt.3329>.
- 753 [48] Bastian Greshake, Simonida Zehr, Francesco Dal Grande, Anjuli Meiser,  
754 Imke Schmitt, and Ingo Ebersberger. Potential and pitfalls of eukaryotic  
755 metagenome skimming: a test case for lichens. *Molecular Ecology Re-*  
756 *sources*, 16(2):511–523, sep 2015. doi: 10.1111/1755-0998.12463. URL  
757 <https://doi.org/10.1111/1755-0998.12463>.