

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

3 Sherine Awad¹, Luiz Irber¹, C. Titus Brown^{1*}
1 Department of Population Health and Reproduction
University of California, Davis
Davis, CA 95616 USA
* E-mail: ctbrown@ucdavis.edu

4 June 24, 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 the much of the remaining
24 diversity is evolutionarily distant and search techniques may not recover it
25 [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 However, evaluating the results of these assemblers is challenging due to the
31 general lack of good quality reference metagenomes.

32 Moya et al. in [5] evaluated metagenome assembly using two simulated
33 454 viral metagenome and six assemblers. The assemblies were evaluated
34 based on several metrics including N50, percentages of reads assembled, ac-
35 curacy when compared to the reference genome. In addition to, chimeras per
36 contigs and the effect of assembly on taxonomic and functional annotations.

37 Mavromatis et al. in [6] provided a benchmark study to evaluate the
38 fidelity of metagenome processing methods. The study used simulated
39 metagenomic data sets constructed at different complexity levels. The datasets
40 were assembled using Phrap v3.57, Arachne v.2 [7] and JAZZ [8]. This study
41 evaluates assembly, gene prediction, and binning methods. However, the
42 study did not evaluate the assembly quality against a reference genome.

43 Rangwala et al. in [9] presented an evaluation study of metagenome
44 assembly. The study used a de Bruijn graph based assembler ABYSS [10] to
45 assemble simulated metagenome reads of 36 bp. The data set is classified at
46 different complexity levels. The study compared the quality of the assembly
47 of the data sets in terms of contig length and assembly accuracy. The
48 study also took into consideration the effect of kmer size and the degree of
49 chimericity. However, the study evaluated the assembly based on only one
50 assembler. Also, both previous studies used simulated data, which may lack
51 confounders of assembly such as sequencing artifacts and GC bias.

52 In a landmark study, Shakya et al. (2013) constructed a synthetic com-
53 munity of organisms by mixing DNA isolated from individual cultures of

54 64 bacteria and archaea, including a variety of strains across a range of
55 nucleotide distances [11]. In addition to performing 16s amplicon analy-
56 sis and doing 454 sequencing, the authors shotgun-sequenced the mixture
57 with Illumina. While the authors concluded that this metagenomic sequenc-
58 ing generally outperformed amplicon sequencing, they did not conduct an
59 assembly based analysis. This data set was also used in several other eval-
60 uation studies, including gbtools for binning [12] and benchmarking of the
61 MEGAHIT assembler [13].

62 More recently, several benchmark studies systematically evaluated metagenome
63 assembly of short reads. The Critical Assessment of Metagenome Interpre-
64 tation (CAMI) collaboration benchmarked a number of metagenome assem-
65 blers on several data sets of varying complexity, evaluating recovery of novel
66 genomes and multiple strain variants [3]. Notably, CAMI concluded that
67 “The resolution of strain-level diversity represents a substantial challenge to
68 all evaluated programs.” Another recent study evaluated eight assemblers
69 on nine environmental metagenomes and three simulated data sets [14] but
70 used no mock. Also see [15].

71 In this study, we extend previous work by delving into questions of
72 chimeric misassembly and strain recovery in the Shakya et al. (2013) data
73 set. First, we update the list of reference genomes for Shakya et al. to in-
74 clude the latest Genbank assemblies along with plasmids. We then compare
75 IDBA [16], MetaSPAdes [17], and MEGAHIT [18] performance on assem-
76 bling this short-read data set, and explore concordance in recovery between
77 the three assemblers. We describe the effects of “strain confusion” between
78 multiple strains. We also detect and analyze several previously unreported
79 strains and genomes in the Shakya et al. data set. We find that in the ab-
80 sence of closely related genomes, all three metagenome assemblers recover
81 95% or more of known reference genomes. However, in the presence of
82 closely related genomes, these three metagenome assemblers vary widely in
83 their performance and, in extreme cases, can fail to recover the majority of
84 some genomes even when they are completely present in the reads. Our re-
85 port provides strong guidance on choice of assemblers and extends previous
86 analyses of this low-complexity metagenome benchmarking data set.

87 Datasets

88 We used a diverse mock community data set constructed by pooling DNA
89 from 64 species of bacteria and archaea and sequencing them with Illumina
90 HiSeq. The raw data set consisted of 109,629,496 reads from Illumina HiSeq

101 bp paired-end sequencing (2x101) with an untrimmed total length of 11.07 Gbp and an estimated fragment size of 380 bp [11].

The original reads are available through the NCBI Sequence Read Archive at Accession SRX200676. We updated the 64 reference genomes sets from NCBI Genbank using the latest available assemblies with plasmid content (June 2017); updated data is available for download at <https://osf.io/8uxj9/>.

Methods

The analysis code and run scripts for this paper are written in Python and bash, and are available at: <https://github.com/dib-lab/2015-metagenome-assembly/>. The scripts and overall pipeline were examined by the first and senior authors for correctness. In addition, the bespoke reference-based analysis scripts were tested by running them on a single-colony *E. coli* MG1655 data set with a high quality reference genome [19].

Quality Filtering

We removed adapters with Trimmomatic v0.30 in paired-end mode with the TruSeq adapters [20], using light quality score trimming (LEADING:2 TRAILING:2 SLIDINGWINDOW:4:2 MINLEN:25) as recommended in MacManes, 2014 [21].

Reference Coverage Profile

To evaluate how much of the reference metagenome was contained in the read data, we used `bwa aln` (v0.7.7.r441) to map reads to the reference genome [22]. We then calculated how many reference bases were covered by mapped reads (custom script `coverage-profile.py`).

Measuring k-mer inclusion and Jaccard similarity

We used MinHashing as implemented in sourmash to estimate k-mer inclusion and Jaccard similarity between data sets [23]. MinHash signatures were prepared with `sourmash compute` using `--scaled 10000`. K-mer inclusion was computed by taking the ratio of the number of intersecting hashes with the query over the total number of hashes in the subject MinHash. Jaccard similarity was computed as in [24] by taking the ratio of the number of intersecting hashes between the query and subject over the number of

122 hashes in the union. K-mer sizes for comparison were chosen at 21, 31, or
123 51, depending on the level of taxonomic specificity desired - genus, species,
124 or strain, respectively, as described in [25].

125 When specified, high-abundance k-mers were selected for counting by
126 using the script `trim-low-abund.py` script with `-C 5` from khmer v2 [26,
127 27].

128 Assemblers

129 We assembled the quality-filtered reads using three different assemblers:
130 IDBA-UD [16], MetaSPAdes [17], and MEGAHIT [18]. For IDBA-UD v1.1.1
131 [16], we used `--pre-correction` to perform pre-correction before assembly
132 and `-r` for the pe files.

133 For MetaSPAdes v3.9.0 [17], we used `--meta --pe1-12 --pe1-s` where
134 `--meta` is used for metagenomic data sets, `--pe1-12` specifies the interlaced
135 reads for the first paired-end library, and `--pe1-s` provides the orphan reads
136 remaining from quality trimming.

137 For MEGAHIT v1.1.1-2-g02102e1 [18], we used `-l 101 -m 3e9 --cpu-only`
138 where `-l` is for maximum read length, `-m` is for max memory in bytes to
139 be used in constructing the graph, and `--cpu-only` to use only the CPU
140 and no GPUs. We also used `--presets meta-large` for large and complex
141 metagenomes, and `--12` and `-r` to specify the interleaved-paired-end and
142 single-end files respectively. MEGAHIT allows the specification of a memory
143 limit and we used `-M 1e+10` for 10 GB.

144 All three assemblies were executed on the same high-memory buy-in
145 node on the Michigan State University High Performance Compute Cluster,
146 and we recorded RAM and CPU time of each assembly job using the `qstat`
147 utility at the end of each run.

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

150 Mapping

151 We aligned all quality-filtered reads to the reference metagenome with `bwa`
152 `aln` (v0.7.7.r441) [22]. We aligned paired-end and orphaned reads separately.
153 We then used `samtools` (v0.1.19) [28] to convert SAM files to BAM files for
154 both paired-end and orphaned reads. To count the unaligned reads, we
155 included only those records with the “4” flag in the SAM files [28].

156 **Assembly analysis using NUCmer**

157 We used the NUCmer tool from MUMmer3.23 [29] to align assemblies to the
158 reference genome with options `-coords -p`. Then we parsed the generated
159 “coords” file using a custom script `analyze_assembly.py`, and calculated
160 several analysis metrics across all three assemblies at a 99% alignment iden-
161 tity.

162 **Reference-based analysis of the assemblies**

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

168 The script `summarize-coords2.py` was used to calculate aligned cov-
169 erage from the loose alignment conditions: each base in the reference was
170 marked as “covered” if it was included in at least one alignment. The script
171 `analyze_ng50.py` was used to calculate NGA 50 for each individual refer-
172 ence genome.

173 **Analysis of chimeric misassemblies**

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

183 **Results**

184 **The raw data is high quality.**

185 The reads contains 11,072,579,096 bp (11.07 Gbp) in 109,629,496 reads with
186 101.0 average length (2x101bp Illumina HiSeq).

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%

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

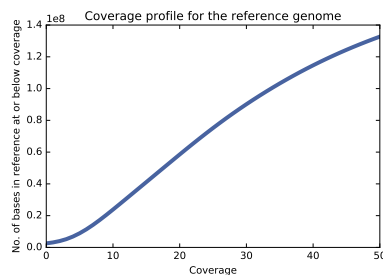


Figure 1: Cumulative coverage profile for the reference metagenome, based on read mapping.

The reference metagenome is not completely present in the reads.

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

In order to evaluate reconstructability with De Bruijn graph assemblers,

we next examined k-mer containment of the reference in the reads for k of 21, 31, 41, and 51 (Table 1). The k-mer overlap decreases from 96.8% to 94.1% as the k-mer size increases. This could be caused by low coverage of some portions of the reference and/or variation between the reads and the reference.

Some individual reference genomes are poorly represented in 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%

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

We next did a 51-mer containment analysis of each reference genome in the reads; $k=51$ was chosen so as to be specific to strain content [25]. 99% or more of the constituent 51-mers for 51 of the 64 reference genomes were present in the reads, suggesting that each of the 51 genomes was entirely present at some minimal coverage.

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

MEGAHIT is the fastest and lowest-memory assembler evaluated

We ran three commonly used metagenome assemblers on the QC data set: IDBA-UD, MetaSPAdes, and MEGAHIT. We recorded the time and memory usage of each (Table 4). In computational requirements, MEGAHIT

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
MEGAHIT	52hr 25m	4 hr 9m	11.4 GB
IDBA-UD	49h	49h	39.8GB
MetaSPAdes	94hr 43m	94hr 44m	100.7 GB

231 outperformed both MetaSPAdes and IDBA-UD considerably, producing an
 232 assembly in four hours (“wall time”) – approximately 12 times faster than
 233 IDBA and 23 times faster than MetaSPAdes. MEGAHIT used only 11.4
 234 GB of RAM – 1/3rd to 1/9th the memory used by IDBA and SPAdes,
 235 respectively.

236 CPU time measurements (which include processing on multiple CPU
 237 cores) show that MEGAHIT and IDBA are competitive in overall process-
 238 ing time, but MEGAHIT’s ability to make use of multiple cores results in
 239 significantly less overall assembly time; this is particularly relevant given
 240 the increasing availability of manycore processors. Despite a variety of con-
 241 figuration attempts, we were unable to get MetaSPAdes to use threading
 242 effectively; however, we note that even with perfectly parallel processing
 243 on 16 cores, MetaSPAdes would take 6 hours and still use approximately 9
 244 times as much RAM as MEGAHIT.

245 **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%

246 We assessed read inclusion in assemblies by mapping the QC reads to
 247 the length-filtered assemblies and counting the remaining unmapped reads.
 248 Depending on the assembly, between 2.7 million and 3.9 million reads (2.5-
 249 3.5%) did not map to the assemblies (Table 5). All of the assemblies included
 250 the large majority of high-abundance 51-mers (more than 96.8% in all cases).

251 **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%

252 We next evaluated the extent to which the assembled contigs recovered
 253 the “known/true” metagenome sequence by aligning each assembly to the
 254 adjusted reference (Table 6). Each of the three assemblers generates contigs
 255 that cover more than 93.1% of the reference metagenome at high identity
 256 (99%) with little duplication (approximately 1%). All three assemblies con-
 257 tain between 96.2% and 97.2% of the 51-mers in the reference.

258 At 99% identity with the loose mapping approach, approximately 2.5% of
 259 the reference is missed by all three assemblers, while 1.7% is uniquely covered
 260 by MEGAHIT, 0.74% is uniquely covered by MetaSPAdes, and 0.64% is
 261 uniquely covered by IDBA.

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

263 When counting only the best (longest) alignment per contig at a 99% identity
 264 threshold, each of the three assemblies recovers more than 87.3% of the

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

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

reference, with MEGAHIT recovering the most – 93.8% of the reference
(Table 7).

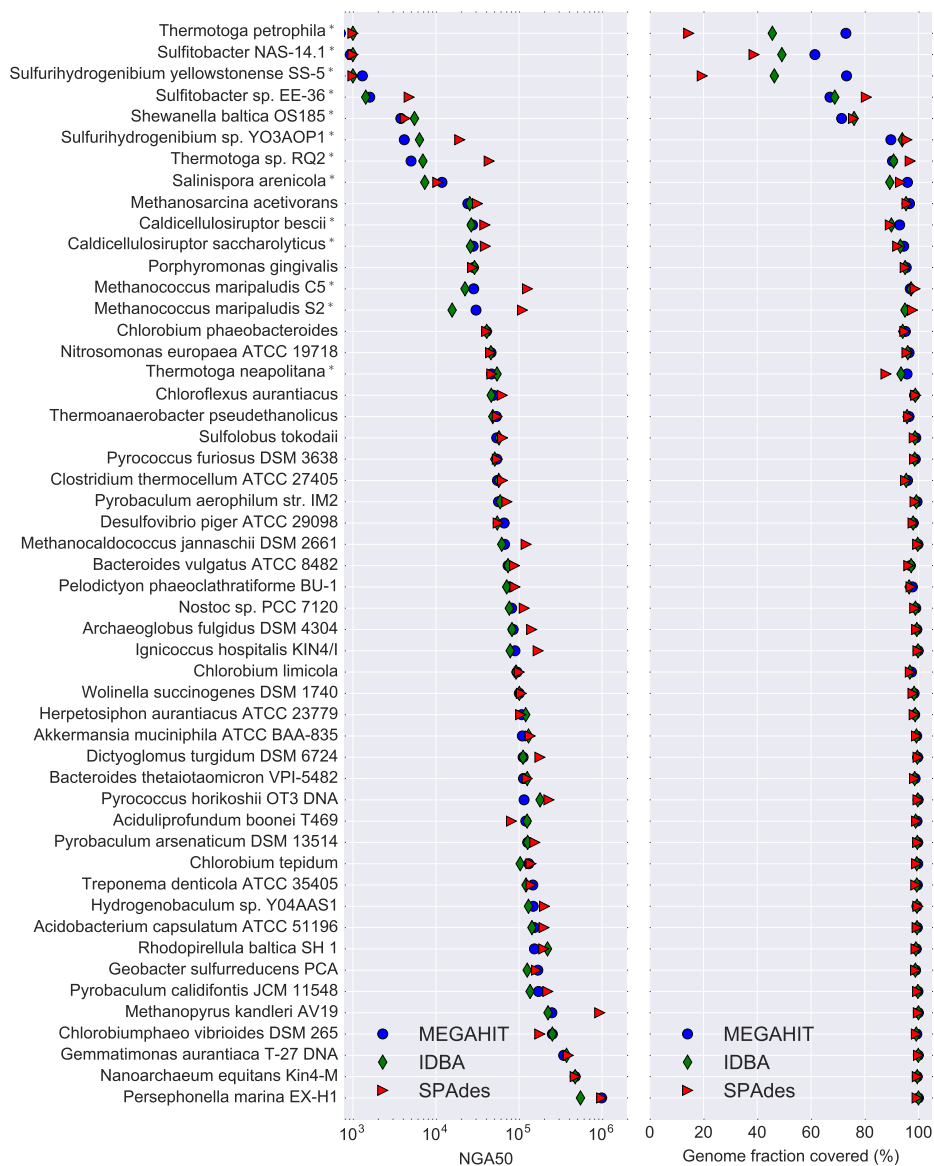


Figure 2: NGA50 by genome and assembler. A '*' after the name indicates the presence of at least one other genome with > 2% Jaccard similarity at k=31 in the community.

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

268 We computed the NGA50 for each individual genome and assembly in order
 269 to compare assembler performance on genome recovery (see left panel of Fig-
 270 ure 2). The NGA50 statistics for individual genomes vary widely, but there
 271 are consistent assembler-specific trends: IDBA yields the lowest NGA50 for
 272 28 of the 51 genomes, while MetaSPAdes yields the highest NGA50 for 32
 273 of the 51 genomes.

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

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

283 **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%)

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

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

293 Approximately 4.8 million reads (4.4%) from the QC data set did not map
 294 anywhere in the reference provided by the authors of [11]. We extracted and

Table 9: Genbank genomes detected in assembly of unmapped reads

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

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

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

307 We note that all but the two *P. ruminis* matches and the *E. coli* isolate
 308 YS are strain variants of species that are part of the defined community
 309 but are not completely present in the reads (see Table 2). For *Proteiniclas-*
 310 *ticum ruminis*, there is no closely related species in the mock community
 311 design, and very little of the MEGAHIT assembly aligns to known *P. ru-*
 312 *minis* genomes at 99%. However, there are many alignments to *P. ruminis*
 313 at 94% or higher, for approximately 2.73 Mbp total. This suggests that the
 314 unmapped reads contain at least some data from a novel species of *Proteini-*
 315 *clasticum*; this matches the observation in [11] of a contaminating genome
 316 from an unknown *Clostridium* spp.

317 Discussion

318 Assembly recovers basic content sensitively and accurately.

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

327 The presence of multiple closely related genomes confounds 328 assembly.

329 In agreement with CAMI, we also find that the presence of closely related
330 genomes in the metagenome causes many assembly problems. This is clearly
331 shown by Figure 2, where 12 of the bottom 14 genomes by NGA50 (left
332 panel) also exhibit poor genome recovery by assembly (right panel). Inter-
333 estingly, different assemblers handle this quite differently, with e.g. MetaS-
334 PAdes failing to recover essentially any of *Thermotoga petrophila*, while
335 MEGAHIT recovers 73%. The presence of nearby genomes is an almost
336 perfect predictor that one or more assembler will fail to recover 5% or more
337 - of the 13/51 genomes for which less than 95% is recovered, 12 of them
338 have close genomes in the community. Interestingly, very little similarity
339 is needed - all genomes with Jaccard similarity of 2% or higher at k=31
340 exhibited these problems.

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

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

353 in twain.

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

355 MEGAHIT is clearly the most efficient computationally, outperforming both
356 MetaSPAdes and IDBA by 5-10x in memory and 17-42x in time (Table 4).
357 The MEGAHIT assembly also included more of the reads than either IDBA
358 or MetaSPAdes, and omitted only 0.4% more of the unique 51-mers from
359 the reads than IDBA. MEGAHIT covered more of the reference genome
360 with both loose and strict alignments (Table 6 and Table 7), with little
361 duplication. This is clearly because of MEGAHIT’s generally superior per-
362 formance in recovering the genomes of closely related strains (Figure 2, right
363 panel). The sum “fraction of genome recovered” is arguably the most im-
364 portant measure of a metagenome assembler (see [?] in particular) and
365 here MEGAHIT excels for individual genomes even in the presence of strain
366 variation.

367 When comparing details of sequence recovery between the assemblers,
368 the assembly content differs by only a small amount when loose alignments
369 are allowed: all three assemblers miss more content (approximately 2.5% of
370 the reference) than they generate uniquely (1.7% or less). In addition to
371 preferring no one assembler over any other, this suggests that combining as-
372 semblies may have little value in terms of recovering additional metagenome
373 content.

374 **The missing reference may be present in strain variants of the** 375 **intended species.**

376 Several individual genomes are missing in measurable portion from the QC
377 reads (Table 2), and many QC reads (4.4% of 108m) did not map to the
378 full reference metagenome. These appear to be related issues: upon anal-
379 ysis of the unmapped reads against Genbank, we find that many of the
380 contigs assembled from the unmapped reads can be assigned to strain vari-
381 ants of the species in the mock community (Table 9). This suggests that
382 the constructors of the mock community may have unintentionally included
383 strain variants of *Fusobacterium nucleatum*, *Thermus thermophilus* HB27,
384 and *Enterococcus faecalis*; note that the microbes used were sourced from
385 the community rather than the ATCC (M. Podar, pers. communication).
386 In addition, we detect what may be portions of a novel member of the *Pro-*
387 *teiniclasticum* genus in the assembly of these reads - likely the *Clostridium*

spp. detected through amplicon sequencing [11].

Without returning to the original DNA samples, it is impossible to conclusively confirm that unintended strains were used in the construction of the mock community. In particular, our analysis is dependent on the genomes in Genbank: the genomes we detect in the contigs are clearly more closely related to Genbank genomes other than the species in the reference metagenome, based on k-mer analysis and contig alignment. However, Genbank is unlikely to contain the exact genomes of the included strain variants, rendering conclusive identification impossible.

Conclusions

Overall, assembly of this mock community works well, with good recovery of known genomic sequence for the majority of genomes. All three assemblers that we evaluated recover similar amounts of most genomic sequence, but (recapitulating several other studies @cite) MEGAHIT is computationally most efficient. We note that assembly resolves substantial portions of several previously undetected strain variants, as well as recovering a substantial portion of a novel *Proteiniclasticum* spp. that was detected via amplicon analysis in [11].

The presence of closely related strains is a major confounder of metagenome assembly, and causes assemblers to drop considerable portions of genomes that (based on read mapping and k-mer inclusion) are clearly present. In this relatively simple community, this strain confusion is present but does not dominate the assembly. However, real microbial communities are likely to have many closely related strains and any resulting loss of assembly will be hard to detect in the absence of good reference genomes. While high polymorphism rates in e.g. animal genomes is known to cause duplication or loss of assembly, some solutions have emerged that make use of assumptions of uniform coverage and diploidy [30]. These solutions cannot however be transferred directly to metagenomes, which have unknown abundance distributions and strain content.

An additional concern is that metagenome assemblies are often performed after pooling data sets to increase coverage (e.g. [4, 31]; this pooled data is more likely to contain multiple strains, which would then in turn adversely affect assembly of strains. This may not be resolvable within the current paradigm of assembly, which focuses on outputting linear assemblies that cannot properly represent strain variation. The human genomics

community is moving towards using *reference graphs*, which can represent multiple incompatible variants in a single data structure [32]; this approach, however, requires high-quality isolate reference genomes, which are generally unavailable for environmental metagenomes.

Long read sequencing (and related technologies) will undoubtedly help resolve strain variation in the future, but even with highly accurate long-read sequencing, current sequencing depth is still too low to resolve deep environmental metagenomes [33, 34]. It is unclear how well long error-prone reads (such as those output by Pacific Biosciences SMRT [35] and Oxford Nanopore instruments [36]) will perform on complex metagenomes: with high error rates, deep coverage of each individual genome is required to achieve accurate assembly, and this may not be easily obtainable for complex communities. Single-molecule barcoding (e.g. 10X Genomics [?]) and HiC approaches [37] show promise but these remain untested on well-defined complex communities and are still challenged by the complexity of complex environmental metagenomes; see [38, 39, 40].

Much of our analysis depended on having a high-quality “mock” metagenome. While computationally constructed synthetic communities and computational “spike-ins” to real data sets can provide valuable controls (e.g. see [14] and [41]) we strongly believe that standardized communities constructed *in vitro* and sequenced with the latest technologies are critical to the evaluation of both canonical and emerging tools, e.g. efforts such as [42]. From the perspective of tool evaluation, we must disagree somewhat with Vollmers et al. [?]: good metagenome tool evaluation necessarily depends on mock communities that are as realistic as we can make them. Likewise, from the perspective of bench biologists, actually sequencing real DNA is critical because it can evaluate confounding effects such as kit contamination [43]. Large-scale studies of computational approaches systematically applied to mock communities such as CAMI [3] can then provide fair comparisons of entire toolchains (wet + dry) applied to these mock communities.

We omitted two important questions in this study: binning and choice of parameters. We chose not to evaluate genome binning because most binning strategies either operate post-assembly (see e.g. [44]), in which case the challenges with assembly discussed above will apply, or require multiple samples (e.g. [45]). We also chose to use only default parameters with all three assemblers, for two reasons. First, we are not aware of any widely used automated approaches for determining the “best” set of parameters or evaluating the output, other than those integrated into the assemblers themselves (e.g. choice of k-mer sizes), and absent such guidance we do not

463 feel comfortable blessing any particular set of parameters; here the choice
464 of default parameters is parsimonious. Second, any parameter exploration
465 pipeline would not only need to be automated but would need to run multiple
466 assemblies; in this case, any comparison based on runtime of the parameter
467 choice pipeline would naturally favor MEGAHIT because of its substantial
468 advantage in computational efficiency.

469 **Author contributions**

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

472 **Competing interests**

473 No competing interest to our knowledge.

474 **Grant information**

475 This work is funded by Moore and NIH.

476 **Acknowledgments**

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

481 **References**

- 482 [1] Jay Ghurye, Victoria Cepeda-Espinoza, and Mihai Pop. Metagenomic assem-
483 bly: Overview, challenges and applications. *The Yale Journal of Biology and*
484 *Medicine*, 89(3):353–362, 2016.
- 485 [2] Nikos C. Kyrpides, Philip Hugenholtz, Jonathan A. Eisen, Tanja Woyke,
486 Markus Göker, Charles T. Parker, Rudolf Amann, Brian J. Beck, Patrick S. G.
487 Chain, Jongsik Chun, Rita R. Colwell, Antoine Danchin, Peter Dawyndt, Tom
488 Dedeurwaerdere, Edward F. DeLong, John C. Detter, Paul De Vos, Timothy J.
489 Donohue, Xiu-Zhu Dong, Dusko S. Ehrlich, Claire Fraser, Richard Gibbs, Jack
490 Gilbert, Paul Gilna, Frank Oliver Glöckner, Janet K. Jansson, Jay D. Keasling,
491 Rob Knight, David Labeda, Alla Lapidus, Jung-Sook Lee, Wen-Jun Li, Juncai
492 MA, Victor Markowitz, Edward R. B. Moore, Mark Morrison, Folker Meyer,

493 Karen E. Nelson, Moriya Ohkuma, Christos A. Ouzounis, Norman Pace, Julian
494 Parkhill, Nan Qin, Ramon Rossello-Mora, Johannes Sikorski, David Smith,
495 Mitch Sogin, Rick Stevens, Uli Stingl, Ken ichiro Suzuki, Dorothea Taylor,
496 Jim M. Tiedje, Brian Tindall, Michael Wagner, George Weinstock, Jean Weis-
497 senbach, Owen White, Jun Wang, Lixin Zhang, Yu-Guang Zhou, Dawn Field,
498 William B. Whitman, George M. Garrity, and Hans-Peter Klenk. Genomic
499 encyclopedia of bacteria and archaea: Sequencing a myriad of type strains.
500 *PLoS Biology*, 12(8):e1001920, aug 2014. doi: 10.1371/journal.pbio.1001920.
501 URL <https://doi.org/10.1371/journal.pbio.1001920>.

502 [3] Alexander Sczyrba, Peter Hofmann, Peter Belmann, David Koslicki, Stefan
503 Janssen, Johannes Droege, Ivan Gregor, Stephan Majda, Jessika Fiedler,
504 Eik Dahms, Andreas Bremges, Adrian Fritz, Ruben Garrido-Oter, Tue
505 Sparholt Jorgensen, Nicole Shapiro, Philip D Blood, Alexey Gurevich, Yang
506 Bai, Dmitriy Turaev, Matthew Z DeMaere, Rayan Chikhi, Niranjana Nagara-
507 jan, Christopher Quince, Lars Hestbjerg Hansen, Soren J Sorensen, Burton
508 K H Chia, Bertrand Denis, Jeff L Froula, Zhong Wang, Robert Egan, Dong-
509 wan Don Kang, Jeffrey J Cook, Charles Deltel, Michael Beckstette, Claire
510 Lemaitre, Pierre Peterlongo, Guillaume Rizk, Dominique Lavenier, Yu-Wei
511 Wu, Steven W Singer, Chirag Jain, Marc Strous, Heiner Klingenberg, Peter
512 Meinicke, Michael Barton, Thomas Lingner, Hsin-Hung Lin, Yu-Chieh Liao,
513 Genivaldo Gueiros Z. Silva, Daniel A Cuevas, Robert A Edwards, Surya Saha,
514 Vitor C Piro, Bernhard Y Renard, Mihai Pop, Hans-Peter Klenk, Markus
515 Goeker, Nikos Kyrpides, Tanja Woyke, Julia A Vorholt, Paul Schulze-Lefert,
516 Edward M Rubin, Aaron E Darling, Thomas Rattei, and Alice C McHardy.
517 Critical assessment of metagenome interpretation - a benchmark of compu-
518 tational metagenomics software. *bioRxiv*, 2017. doi: 10.1101/099127. URL
519 <http://biorxiv.org/content/early/2017/01/09/099127>.

520 [4] I. Sharon, M. J. Morowitz, B. C. Thomas, E. K. Costello, D. A. Relman,
521 and J. F. Banfield. Time series community genomics analysis reveals rapid
522 shifts in bacterial species, strains, and phage during infant gut colonization.
523 *Genome Research*, 23(1):111–120, aug 2012. doi: 10.1101/gr.142315.112. URL
524 <https://doi.org/10.1101/gr.142315.112>.

525 [5] Jorge F Vázquez-Castellanos, Rodrigo García-López, Vicente Pérez-Brocal,
526 Miguel Pignatelli, and Andrés Moya. Comparison of different assembly and
527 annotation tools on analysis of simulated viral metagenomic communities in
528 the gut. *BMC genomics*, 15(1):1, 2014.

529 [6] Konstantinos Mavromatis, Natalia Ivanova, Kerrie Barry, Harris Shapiro, Eu-
530 gene Goltsman, Alice C McHardy, Isidore Rigoutsos, Asaf Salamov, Frank
531 Korzeniewski, Miriam Land, et al. Use of simulated data sets to evaluate the
532 fidelity of metagenomic processing methods. *Nature methods*, 4(6):495–500,
533 2007.

- 534 [7] David B Jaffe, Jonathan Butler, Sante Gnerre, Evan Mauceli, Kerstin
535 Lindblad-Toh, Jill P Mesirov, Michael C Zody, and Eric S Lander. Whole-
536 genome sequence assembly for mammalian genomes: Arachne 2. *Genome*
537 *research*, 13(1):91–96, 2003.
- 538 [8] Samuel Aparicio, Jarrod Chapman, Elia Stupka, Nik Putnam, Jer-ming Chia,
539 Paramvir Dehal, Alan Christoffels, Sam Rash, Shawn Hoon, Arian Smit, et al.
540 Whole-genome shotgun assembly and analysis of the genome of *fugu rubripes*.
541 *Science*, 297(5585):1301–1310, 2002.
- 542 [9] Anveshi Charuvaka and Huzefa Rangwala. Evaluation of short read metage-
543 nomic assembly. *BMC genomics*, 12(2):1, 2011.
- 544 [10] Jared T Simpson, Kim Wong, Shaun D Jackman, Jacqueline E Schein,
545 Steven JM Jones, and Inanç Birol. Abyss: a parallel assembler for short read
546 sequence data. *Genome research*, 19(6):1117–1123, 2009.
- 547 [11] Shakya Migun, Christopher Quince, James Campbell, Zamin Yang, Christo-
548 pher Schadt, and Mircea Podar. Comparative metagenomic and rRNA microbial
549 diversity characterization using archaeal and bacterial synthetic communities.
550 *Environmental Microbiology*, 15(6):1882–1899, 2013.
- 551 [12] Brandon K. B. Seah and Harald R. Gruber-Vodicka. gbtools: In-
552 teractive visualization of metagenome bins in R. *Frontiers in Mi-*
553 *crobiology*, 6, dec 2015. doi: 10.3389/fmicb.2015.01451. URL
554 <https://doi.org/10.3389/fmicb.2015.01451>.
- 555 [13] Dinghua Li, Ruibang Luo, Chi-Man Liu, Chi-Ming Leung, Hing-
556 Fung Ting, Kunihiko Sadakane, Hiroshi Yamashita, and Tak-Wah
557 Lam. MEGAHIT v1.0: A fast and scalable metagenome assembler
558 driven by advanced methodologies and community practices. *Meth-*
559 *ods*, 102:3–11, jun 2016. doi: 10.1016/j.ymeth.2016.02.020. URL
560 <https://doi.org/10.1016/j.ymeth.2016.02.020>.
- 561 [14] Andries Johannes van der Walt, Marc Warwick Van Goethem,
562 Jean-Baptiste Ramond, Thulani Peter Makhalanyane, Oleg Reva,
563 and Don Arthur Cowan. Assembling metagenomes, one com-
564 munity at a time. *bioRxiv*, 2017. doi: 10.1101/120154. URL
565 <http://biorxiv.org/content/early/2017/06/06/120154>.
- 566 [15] William W. Greenwald, Niels Klitgord, Victor Seguritan, Shibu Yooseph,
567 J. Craig Venter, Chad Garner, Karen E. Nelson, and Weizhong Li. Utilization
568 of defined microbial communities enables effective evaluation of meta-genomic
569 assemblies. *BMC Genomics*, 18(1), apr 2017. doi: 10.1186/s12864-017-3679-5.
570 URL <https://doi.org/10.1186/s12864-017-3679-5>.

- [16] 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.
- [17] 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>.
- [18] 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.
- [19] 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.
- [20] Anthony M. Bolger, Marc Lohse, and Bjoern Usadel. Trimmomatic: A flexible trimmer for illumina sequence data. *Bioinformatics*, 30(15):2114–2120, 2014.
- [21] Matthew D MacManes. On the optimal trimming of high-throughput mrna sequence data. *Frontiers in genetics*, 5:13, 2014.
- [22] Heng Li and Richard Durbin. Fast and accurate short read alignment with burrows–wheeler transform. *Bioinformatics*, 25(14):1754–1760, 2009.
- [23] C. Titus Brown and Luiz Irber. sourmash: a library for MinHash sketching of DNA. *The Journal of Open Source Software*, 1(5), sep 2016. doi: 10.21105/joss.00027. URL <https://doi.org/10.21105/joss.00027>.
- [24] Brian D. Ondov, Todd J. Treangen, Páll Melsted, Adam B. Mallonee, Nicholas H. Bergman, Sergey Koren, and Adam M. Phillippy. Mash: fast genome and metagenome distance estimation using MinHash. *Genome Biology*, 17(1), jun 2016. doi: 10.1186/s13059-016-0997-x. URL <https://doi.org/10.1186/s13059-016-0997-x>.
- [25] David Koslicki and Daniel Falush. Metapalette: a k-mer painting approach for metagenomic taxonomic profiling and quantification of novel strain variation. *mSystems*, 1(3), 2016. doi: 10.1128/mSystems.00020-16. URL <http://msystems.asm.org/content/1/3/e00020-16>.
- [26] Zhang Qingpeng, Awad Sherine, and Brown Titus. Crossing the streams: a framework for streaming analysis of short dna sequencing reads. *PeerJ PrePrints 3:e1100* <https://dx.doi.org/10.7287/peerj.preprints.890v1>, 2015.

- [27] MR Crusoe, HF Alameldin, S Awad, E Boucher, A Caldwell, R Cartwright, A Charbonneau, B Constantinides, G Edverson, S Fay, J Fenton, T Fenzl, J Fish, L Garcia-Gutierrez, P Garland, J Gluck, I Gonzlez, S Guermond, J Guo, A Gupta, JR Herr, A Howe, A Hyer, A Hrpfer, L Irber, R Kidd, D Lin, J Lippi, T Mansour, P McA’Nulty, E McDonald, J Mizzi, KD Murray, JR Nahum, K Nanlohy, AJ Nederbragt, H Ortiz-Zuazaga, J Ory, J Pell, C Pepe-Ranne, ZN Russ, E Schwarz, C Scott, J Seaman, S Sievert, J Simpson, CT Skenner, J Spencer, R Srinivasan, D Standage, JA Stapleton, SR Steinman, J Stein, B Taylor, W Trimble, HL Wiencko, M Wright, B Wyss, Q Zhang, e zyme, and CT Brown. The khmer software package: enabling efficient nucleotide sequence analysis [version 1; referees: 2 approved, 1 approved with reservations]. *F1000Research*, 4(900), 2015. doi: 10.12688/f1000research.6924.1.
- [28] Heng Li, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer, Gabor Marth, Goncalo Abecasis, Richard Durbin, et al. The sequence alignment/map format and samtools. *Bioinformatics*, 25(16):2078–2079, 2009.
- [29] Stefan Kurtz, Adam Phillippy, Arthur L Delcher, Michael Smoot, Martin Shumway, Corina Antonescu, and Steven L Salzberg. Versatile and open software for comparing large genomes. *Genome biology*, 5(2):1, 2004.
- [30] J. H. Kim, M. S. Waterman, and L. M. Li. Diploid genome reconstruction of *Ciona intestinalis* and comparative analysis with *Ciona savignyi*. *Genome Research*, 17(7):1101–1110, jun 2007. doi: 10.1101/gr.5894107. URL <https://doi.org/10.1101/gr.5894107>.
- [31] 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>.
- [32] 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.
- [33] 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>.
- [34] Richard Allen White, Eric M. Bottos, Taniya Roy Chowdhury, Jeremy D. Zucker, Colin J. Brislawn, Carrie D. Nicora, Sarah J. Fansler, Kurt R. Glaes-

semann, Kevin Glass, and Janet K. Jansson. Moleclo 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>.

[35] 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>.

[36] 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>.

[37] Caiti Smukowski Heil, Joshua N. Burton, Ivan Liachko, Anne Friedrich, Noah A. Hanson, Cody L. Morris, Joseph Schacherer, Jay Shendure, James H. Thomas, and Maitreya J. Dunham. Identification of a novel interspecific hybrid yeast from a metagenomic open fermentation sample using hi-c. *bioRxiv*, 2017. doi: 10.1101/150722. URL <http://biorxiv.org/content/early/2017/06/15/150722>.

[38] Joshua N. Burton, Ivan Liachko, Maitreya J. Dunham, and Jay Shendure. Species-level deconvolution of metagenome assemblies with hi-c–based contact probability maps. *G3*, 4(7):1339–1346, may 2014. doi: 10.1534/g3.114.011825. URL <https://doi.org/10.1534/g3.114.011825>.

[39] Martial Marbouty, Axel Cournac, Jean-François Flot, Hervé Marie-Nelly, Julien Mozziconacci, and Romain Koszul. Metagenomic chromosome conformation capture (meta3c) unveils the diversity of chromosome organization in microorganisms. *eLife*, 3, dec 2014. doi: 10.7554/elife.03318. URL <https://doi.org/10.7554/elife.03318>.

[40] Christopher W. Beitel, Lutz Froenicke, Jenna M. Lang, Ian F. Korf, Richard W. Micheltore, Jonathan A. Eisen, and Aaron E. Darling. Strain- and plasmid-level deconvolution of a synthetic metagenome by sequencing proximity ligation products. *PeerJ*, 2:e415, may 2014. doi: 10.7717/peerj.415. URL <https://doi.org/10.7717/peerj.415>.

- 687 [41] Adina Chuang Howe, Janet K Jansson, Stephanie A Malfatti, Susannah G
688 Tringe, James M Tiedje, and C Titus Brown. Tackling soil diversity with the
689 assembly of large, complex metagenomes. *Proceedings of the National Academy
690 of Sciences*, 111(13):4904–4909, 2014.
- 691 [42] Bonnie L. Brown, Mick Watson, Samuel S. Minot, Maria C.
692 Rivera, and Rima B. Franklin. MinIONTM nanopore sequenc-
693 ing of environmental metagenomes: a synthetic approach. *Giga-
694 Science*, 6(3):1–10, feb 2017. doi: 10.1093/gigascience/gix007. URL
695 <https://doi.org/10.1093/gigascience/gix007>.
- 696 [43] Susannah J Salter, Michael J Cox, Elena M Turek, Szymon T Calus,
697 William O Cookson, Miriam F Moffatt, Paul Turner, Julian Parkhill,
698 Nicholas J Loman, and Alan W Walker. Reagent and laboratory
699 contamination can critically impact sequence-based microbiome analyses.
700 *BMC Biology*, 12(1), nov 2014. doi: 10.1186/s12915-014-0087-z. URL
701 <https://doi.org/10.1186/s12915-014-0087-z>.
- 702 [44] Cedric C Laczny, Christina Kiefer, Valentina Galata, Tobias Fehlmann,
703 Christina Backes, and Andreas Keller. Busybee web: metagenomic data anal-
704 ysis by bootstrapped supervised binning and annotation. *Nucleic Acids Re-
705 search*, page gkx348, 2017.
- 706 [45] Brian Cleary, Ilana Lauren Brito, Katherine Huang, Dirk Gevers, Terrance
707 Shea, Sarah Young, and Eric J Alm. Detection of low-abundance bacte-
708 rial strains in metagenomic datasets by eigengenome partitioning. *Nature
709 Biotechnology*, 33(10):1053–1060, sep 2015. doi: 10.1038/nbt.3329. URL
710 <https://doi.org/10.1038/nbt.3329>.