OXFORD

Briefings in Bioinformatics, 2023, **24(2)**, 1–17

https://doi.org/10.1093/bib/bbad087

Advance access publication date 14 March 2023

Problem Solving Protocol

Benchmarking genome assembly methods on metagenomic sequencing data

Zhenmiao Zhang, Chao Yang, Werner Pieter Veldsman, Xiaodong Fang and Lu Zhang

Corresponding author: Lu Zhang, Department of Computer Science, Hong Kong Baptist University, Hong Kong SAR, China. Institute for Research and Continuing Education, Hong Kong Baptist University, Shenzhen, China. E-mail: ericluzhang@hkbu.edu.hk

Abstract

Metagenome assembly is an efficient approach to reconstruct microbial genomes from metagenomic sequencing data. Although short-read sequencing has been widely used for metagenome assembly, linked- and long-read sequencing have shown their advancements in assembly by providing long-range DNA connectedness. Many metagenome assembly tools were developed to simplify the assembly graphs and resolve the repeats in microbial genomes. However, there remains no comprehensive evaluation of metagenomic sequencing technologies, and there is a lack of practical guidance on selecting the appropriate metagenome assembly tools. This paper presents a comprehensive benchmark of 19 commonly used assembly tools applied to metagenomic sequencing datasets obtained from simulation, mock communities or human gut microbiomes. These datasets were generated using mainstream sequencing platforms, such as Illumina and BGISEQ short-read sequencing, 10x Genomics linked-read sequencing, and PacBio and Oxford Nanopore long-read sequencing. The assembly tools were extensively evaluated against many criteria, which revealed that long-read assemblers generated high contig contiguity but failed to reveal some medium- and high-quality metagenome-assembled genomes (MAGs). Linked-read assemblers obtained the highest number of overall near-complete MAGs from the human gut microbiomes. Hybrid assemblers using both short- and long-read sequencing were promising methods to improve both total assembly length and the number of near-complete MAGs. This paper also discussed the running time and peak memory consumption of these assembly tools and provided practical guidance on selecting them.

Keywords: genome assembly tools, metagenomic sequencing, metagenome-assembled genome, short-read sequencing, linked-read sequencing, long-read sequencing

INTRODUCTION

The aim of metagenome assembly is to reconstruct microbial genomes from metagenomic sequencing data. It is an approach that has fundamentally advanced the study of both host-associated microbial communities and free-living microbes [1–3]. Microbial genomes are traditionally reconstructed by sampling colonies cultured from isolates in a laboratory [4]. However, the majority of microbes cannot be cultured under laboratory conditions, making their genomes undetectable with traditional isolate-and-culture approaches [5, 6]. Metagenomic sequencing bypasses this shortcoming by enabling efficient, direct detection of a mixture of microbial DNA without the need for isolation, thus facilitating the reconstruction of microbial genomes with diverse characteristics. Despite the advantages that metagenomic genome reconstruction offers, the combinations of read-technologies and assembly related software in this subfield of microbial genome assembly are yet to be benchmarked to determine their suitability and comparative performance.

Short-read sequencing is the most widely adopted sequencing technology in metagenomic studies. Many assembly software tools have been developed to assemble short-reads from microbial genomes with imbalanced coverage. For example, IDBA-UD [7] resolves short repeats from low-depth regions by local assembly using the paired-end constraint of short-reads. MetaSPAdes [8] extends SPAdes [9] by incorporating graph simplification strategies to separate strains with similar sequences, and applies ExSPAnder [10] to detangle the repetitive sequences. MEGAHIT [11] constructs succinct *de Bruijn* graphs using *k*-mers (subsequences of length *k*) to fill the gaps in low-depth regions and resolve genomic repeats. Currently, the most popular commercial short-read sequencing platforms are designed by Illumina (e.g. HiSeq, NextSeq and MiSeq) and BGI (e.g. BGISEQ-500, MGISEQ-200 and MGISEQ-2000). A major drawback of short-reads is however that the following three tasks in metagenome assembly cannot be easily achieved: (1) the detection of horizontal gene transfers and transposon mobilization between microbes; (2) the deconvolution of duplicate and conserved sequences in microbial

Zhenmiao Zhang is a PhD candidate at the Department of Computer Science, Hong Kong Baptist University. His research interests include metagenome assembly, bioinformatics, and machine learning.

Chao Yang is a PhD student at the Department of Computer Science, Hong Kong Baptist University. His research interests include metagenomic sequencing, bioinformatics, and cancer genomics.

Werner Pieter Veldsman is a postdoc at the Hong Kong Baptist University. He obtained a PhD in Cell and Molecular Biology from the Chinese University of Hong Kong. His research is focused on computational analysis of biological data.

Xiaodong Fang is a vice director of BGI Research and adjunct Professor of Guangzhou University of Chinese Medicine, China. His research interests include bioinformatics, metagenomic sequencing and translational medicine.

Lu Zhang is an Assistant Professor at Department of Computer Science, Hong Kong Baptist University. His research interests include computational genomics, bioinformatics and machine learning.

Received: September 28, 2022. **Revised:** January 09, 2023. **Accepted:** February 15, 2023

genomes and (3) the generation of high-quality draft genomes for low-abundance microbes. It is for this reason that long-read and linked-read sequencing technologies (that provide long-range DNA connectedness information) are advantageous for resolving complex genomic regions and generating more complete draft genomes.

Linked-read (a.k.a read cloud) sequencing technology works by tagging short-reads with the same barcode sequence if they are derived from the same long DNA fragment. Several linked-read sequencing platforms have been developed, namely: Illumina TrueSeq Synthetic Long-Reads, LoopSeq, 10x Genomics, single-tube Long Fragment Read and Transposase Enzyme-Linked Long-read Sequencing. The 10x Genomics sequencing platform has been effectively used for metagenome assembly; Zlitni *et al.* [12] explored temporal strain-level variants in stool samples from a patient during a 2-month hematopoietic cell transplant treatment period by iteratively assembling 10x metagenomic sequencing data. Similarly, Roodgar *et al.* [13] investigated the human gut microbiome response to antibiotic treatment using longitudinal 10x linked-read sequencing and assembly. Two metagenome assemblers have been developed for 10x linked-reads. CloudSPAdes [14] solves the shortest cloud superstring problem to assemble metagenomes with 10x linked-reads as input. Athena [15] improves metagenome assembly by recruiting co-barcoded linked-reads for local assembly and demonstrated that 10x linked-reads outperformed Illumina short-reads and TrueSeq Synthetic Long-Reads in metagenome assembly.

Single-molecule long-read sequencing platforms, such as Pacific Biosciences Single-Molecule Continuous Long Read sequencing (PacBio CLR), Pacific Biosciences high-fidelity long-read sequencing (PacBio HiFi) and Oxford Nanopore Technologies sequencing (ONT), have recently been applied to many metagenomic studies. Tsai *et al.* [16] used PacBio CLR long-reads to assemble metagenomic sequencing data of human skin, and identified the previously uncharacterized Corynebacterium simulans. Another recent study investigated 2267 bacteria and archaea and found that a majority of their genomes could be assembled using PacBio CLR long-reads [17]. Many assembly tools have been developed for long-read metagenome assembly. The metaFlye assembler [18] extends Flye [19] to deal with uneven bacterial composition and intra-species heterogeneity by leveraging unique paths in repeat graphs. Canu [20] improves Celera Assembler [21, 22] to handle noisy long-reads by using multiple rounds of read error correction. Moss *et al.* [23] improved the long-read sequencing protocol of DNA extraction and developed Lathe to optimize metagenome assembly on ONT data. Several other long-read assemblers have recently been released including MECAT2 [24], NECAT [25], Shasta [26] and wtdbg2 [27], but only a few of them have been tested on metagenome assembly. There also remain some limitations to long-read sequencing that restrict its practical applications: (1) the high base error rate of long-read sequencing makes it challenging to distinguish strains and substrains with similar sequence characteristics; and (2) the high cost of long-read sequencing prevents its widespread application in large cohort studies.

Hybrid assembly is a strategy that integrates the strengths of both short-reads and long-reads. There are many genome assembly tools developed for hybrid assembly. DBG2OLC [28] aligns the contigs assembled from short-reads to long-reads, then it represents each long-read using the identifiers of aligned contigs and assembles them using the overlap-layout-consensus approach. MetaFlye supports hybrid assembly by considering the contigs assembled from both short- and long-reads as high-quality long-reads and assembling them using a special parameter (–subassemblies). Antipov *et al.* [29] developed hybridSPAdes that leverages long-reads to fill gaps and resolve repeats on the assembly graph built from short-reads. Unicycler [30] developers implemented algorithms to resolve paths on the assembly graph built from short-reads using the consensus sequence of long-reads and read depth information. MaSuRCA [31] builds longer super-reads from short-reads by extending *k*-mers, and merges the super-reads using long-reads as templates. OPERA-LG [32] links and orientates the contigs assembled from short-reads by paired-end constraint and with reference to long-reads. OPERA-MS [33] constructs a scaffold graph by linking the contigs assembled from short-reads if they are supported by long-reads. The contigs in the scaffold graph are grouped into clusters based on graph topology and read depth, and further assembled using OPERA-LG.

There are many previous studies that evaluated metagenome assemblers. Sczyrba *et al.* [34] evaluated six short-read assemblers using the Critical Assessment of Metagenome Interpretation (CAMI) datasets consisting of simulated short-reads; Latorre-Pérez *et al.* [35] benchmarked 13 short- and long-read assemblers using ONT long-reads sequenced from ZymoBIOMICS™ Microbial Communities. They evaluated assembly tools on the datasets obtained from a single sequencing technology, but did not compare different sequencing technologies. In a follow-up CAMI study, Meyer *et al.* [36] added benchmark results for more short-read, long-read, and hybrid assemblers using CAMI II datasets. The latter study only used datasets from simulation, which have the disadvantage of being idealized representations of metagenomic sequencing data. Currently, the advantages and limitations of the existing sequencing technologies and corresponding assembly tools remain unclear, and there is an urgent need for practical guidelines on how to select the best sequencing technologies and assembly tools.

In this study, we benchmarked 19 state-of-the-art tools to generate short-, linked-, long-read and hybrid assemblies using metagenomic sequencing datasets from simulation, mock communities and human gut microbiomes (Figure 1). The benchmark involved comparing the basic contig statistics, including total assembly length (*AL* for human stool datasets), genome fraction (*GF* for simulation and mock datasets), contig N50, NA50, normalized NGA50 (Methods), mismatches and misassemblies. We also evaluated the metagenome-assembled genomes (MAGs) after contig binning with respect to their contiguities (MAG N50), qualities (Methods; #*MQ*: the number of medium-quality MAGs; #*HQ*: the number of high-quality MAGs; #*NC*: the number of near-complete MAGs) and annotations of microbes (the number of microbes that can be annotated from MAGs).

Our results showed that the short-read assemblers generated the lowest contig contiguity and #*NC*. MEGAHIT outperformed IDBA-UD and metaSPAdes on the deeply sequenced datasets (>100X), and metaSPAdes obtained better results than MEGAHIT and IDBA-UD on low-complexity datasets (depth < 100X). The contig N50s of linked-read assemblies were significantly higher than those of short-read assemblies. Athena demonstrated a higher contig N50 than cloudSPAdes and generated the highest #*NC* among all of the assemblers for the datasets obtained from human gut microbiomes. Long-read assemblers demonstrated high contig N50 but generated smaller #*MQ* and #*HQ* than short- and linked-read assemblers. MetaFlye, Canu and Lathe performed much better than the other long-read assemblers. MetaFlye generated the highest *GF*s and *AL*s for both ONT and PacBio CLR datasets. Lathe produced a higher #*NC* than metaFlye

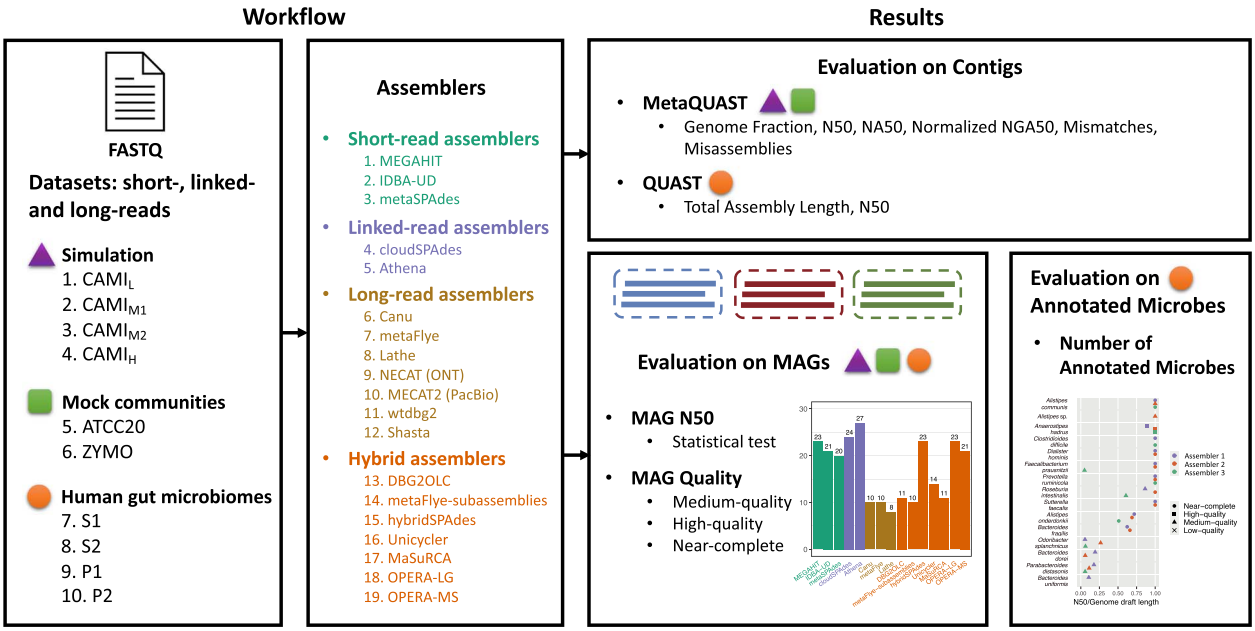

Figure 1. Data and workflow used to benchmark the 19 metagenome assembly tools. Workflow: We obtained 32 metagenomic datasets generated by short-read, linked-read and long-read sequencing from simulation, mock communities and four human gut microbiomes, which were used to evaluate the performances of 19 assembly tools. Results: The following contig statistics were used to evaluate the assemblies: total assembly length (*AL*), genome fraction (*GF*), contig N50, NA50, normalized NGA50, mismatches and misassemblies. The assemblies generated from the real datasets were also evaluated by the metagenome-assembled genome (MAG) N50; the numbers of medium-quality (#*MQ*), high-quality (#*HQ*) and near-complete (#*NC*) MAGs; and the numbers of annotated microbes.

and Canu on ONT datasets. Hybrid assemblies demonstrated higher (or at least similar) *GF*s and *AL*s than short- and long-read assemblies, and generated higher #*HQ* and #*NC* than long-read assemblies. Unicycler and MaSuRCA produced lower *GF*s and *AL*s than the other hybrid assemblers but achieved the highest contig contiguity. Unicycler or OPERA-MS generated the highest #*NC* on the real datasets sequenced by Illumina and PacBio CLR. MaSuRCA obtained more #*NC* than the other hybrid assemblers on the real datasets sequenced by Illumina and ONT.

RESULTS

Metagenomic sequencing datasets from simulation, mock communities and human gut microbiomes

In this study, 32 datasets were collected or generated using three sequencing technologies (Methods; Figure 1; Table 1): short-read sequencing (Illumina HiSeq and BGISEQ-500), linked-read sequencing (10x Chromium) and long-read sequencing (PacBio CLR and ONT). These datasets comprised the following categories: (1) Simulation datasets from four CAMI communities with low (CAMI$_L$), medium (CAMI$_{M1}$ and CAMI$_{M2}$) and high (CAMI$_H$) complexities, consisting of available short-reads, simulated 10x linked-reads and simulated ONT long-reads. We merged CAMI datasets with high complexity from five time points into CAMI$_H$ to avoid insufficient read depth. The 10x linked-reads and ONT long-reads were simulated for the four CAMI communities (Methods). (2) Mock datasets from two mock communities: 20 Strain Staggered Mix Genomic Material – MSA-1003 [37, 38] (ATCC20, consists of available Illumina HiSeq 2500, 10x Chromium and PacBio CLR reads) and ZymoBIOMICS™ Microbial Community Standard II with log distribution [39] (ZYMO, consists of available Illumina HiSeq 1500, ONT GridION and ONT PromethION reads). (3) Real datasets from stool samples of four human gut microbiomes, denoted S1 and S2 (sequenced by Illumina HiSeq 2500, BGISEQ-500, 10x Chromium and PacBio CLR; Methods; Supplementary Figure S1), and P1 and P2 [15, 23] (consisting of available Illumina HiSeq 4000, 10x Chromium and ONT reads). According to the data contributors, short-reads and linked-reads of P1 and P2 were trimmed [15] using cutadapt (v1.81) [40].

Metagenome assembly performance per short-read sequencing tool

We compared the assembly performance of MEGAHIT, IDBA-UD and metaSPAdes, the three commonly used short-read assemblers, on Illumina short-read sequencing data. The contigs generated by these three tools had comparable *GF*s (Table 2) and *AL*s (Table 2) on all datasets but nevertheless exhibited distinct characteristics on different datasets.

MEGAHIT showed the best assembly performance on the datasets with deeper sequencing depth (>100X; Table 1), including ZYMO (133X), P1 (383X) and P2 (776X). For those datasets, MEGAHIT obtained substantially higher N50s than IDBA-UD and metaSPAdes on ZYMO (136.96% higher than IDBA-UD, 208.19% higher than metaSPAdes), P1 (122.18% higher than IDBA-UD, 150.06% higher than metaSPAdes) and P2 (118.48% higher than IDBA-UD, 131.50% higher than metaSPAdes). For simulation and mock datasets, we further evaluate the contig contiguity by breaking the contigs at misassemblies and compare the NA50 and normalized NGA50 (Methods). Assemblies from MEGAHIT obtained the best contig contiguity on the datasets with deeper sequencing depth. NA50 of MEGAHIT was 1.37 and 2.10 times higher than that of IDBA-UD and metaSPAdes, respectively (NA50: MEGAHIT = 167.51 kb, IDBA-UD = 122.50 kb, metaSPAdes = 79.92 kb; Table 2; Supplementary Figure S2; Supplementary Table S1). The normalized NGA50 of MEGAHIT was also 1.15 and 1.39 times higher than IDBA-UD and metaSPAdes on ZYMO, respectively (Table 2; Supplementary Table S1). For samples from human gut

Table 1. The simulation, mock and real datasets used to evaluate the performance of metagenome assembly tools. The PacBio CLR dataset of ATCC20 was downsampled to 50% to avoid out of memory issues

Datasets		Sequencing platforms	Sources	Amount of sequencing (Gb)	Sequencing depth (X)	Average Read Length (bp)	Community composition
Simulation	$CAMI_L$	Illumina HiSeq	CAMI I	15.0	95	150	40 genomes and 20 circular elements [34]
		10x Chromium	This study	15.0	95	150	
		ONT	This study	15.0	95	4,194	
	$CAMI_{M1}$	Illumina HiSeq	CAMI I	15.0	26	150	132 genomes and 100 circular elements [34]
		10x Chromium	This study	15.0	26	150	
		ONT	This study	15.0	26	4,217	
	$CAMI_{M2}$	Illumina HiSeq	CAMI I	15.0	26	150	132 genomes and 100 circular elements [34]
		10x Chromium	This study	15.0	26	150	
		ONT	This study	15.0	26	4,186	
	$CAMI_H$	Illumina HiSeq	CAMI I	75.0	27	150	596 genomes and 478 circular elements [34]
		10x Chromium	This study	75.0	27	150	
		ONT	This study	75.0	27	4,112	
Mock	ATCC20	Illumina HiSeq	SRR8359173	1.3	19	125	20 microbes [37, 38]
		10x Chromium	SRR12283286	108.7	1,622	150	
		PacBio CLR	SRR12371719	253.5	3,784	8,394	
	ZYMO	Illumina HiSeq	ERR2935805	9.7	133	101	10 microbes [39]
		ONT GridION	ERR3152366	16.5	226	4,501	
		ONT PromethION	ERR3152367	153.7	2,105	4,446	
Real	S1	Illumina HiSeq	This study	11.6	29	150	Human gut microbiome
		BGISEQ-500	This study	16.8	42	100	
		10x Chromium	This study	58.7	147	150	
		PacBio CLR	This study	6.3	16	8,878	
	S2	Illumina HiSeq	This study	11.4	34	150	Human gut microbiome
		BGISEQ-500	This study	15.0	44	100	
		10x Chromium	This study	56.1	165	150	
		PacBio CLR	This study	8.4	25	8,973	
	P1	Illumina HiSeq	SRR6788327, SRR6807561	76.6	383	150	Human gut microbiome
		10x Chromium	SRR6760786	35.8	179	150	
		ONT MinION	SRR8427258	11.4	57	2,838	
	P2	Illumina HiSeq	SRR6788328, SRR6807555	77.6	776	150	Human gut microbiome
		10x Chromium	SRR6760782	32.6	326	150	
		ONT MinION	SRR8427257	6.1	61	1,800	

microbiomes, we grouped the contigs into MAGs and classified them based on different qualities (Methods). Here too, MEGAHIT achieved the best performance on the datasets with deeper sequencing depth (P1 and P2). MEGAHIT produced the highest #*MQ*, #*HQ* and #*NC* on P1 and P2 (#*MQ* in P1 and P2: MEGAHIT = 21, IDBA-UD = 7, metaSPAdes = 6; #*HQ* in P1 and P2: MEGAHIT = 4, IDBA-UD = 3, metaSPAdes = 1; #*NC* in P1 and P2: MEGAHIT = 1, IDBA-UD = 0, metaSPAdes = 0; Figure 2 I–K, M–O). The MAG N50s of MEGAHIT were comparable with those of IDBA-UD, and significantly higher than those of metaSPAdes, on both P1 (Wilcoxon rank-sum test *P*-value: MEGAHIT versus metaSPAdes = 1.49e-4, IDBA-UD versus metaSPAdes = 5.64e-4; Supplementary Figures S3 and S4; Supplementary Table S2; Methods) and P2 (Wilcoxon rank-sum test *P*-value: MEGAHIT versus metaSPAdes = 1.37e-3, IDBA-UD versus metaSPAdes = 2.45e-2; Supplementary Figures S3 and S4; Supplementary Table S2; Methods). Next, we annotated the MAGs with taxonomic information using Kraken 2 (Methods). The assemblies from MEGAHIT for P1 and P2 were classified as belonging to seven distinct microbes (Figure 2 C and D), which was more than that of IDBA-UD (five microbes) and metaSPAdes (two microbes).

The metaSPAdes assembler obtained much better contig contiguity than IDBA-UD and MEGAHIT on the low- and medium-complexity datasets that were not deeply sequenced (<100X), including $CAMI_L$, $CAMI_{M1}$, $CAMI_{M2}$ and ATCC20. Its contig N50s, NA50s, and normalized NGA50s were, on average, 1.58, 1.58 and 1.62 times higher than those of IDBA-UD, and 1.31, 1.35 and 1.36 times higher than those of MEGAHIT, respectively (Table 2; Figure 2 A–C and E–G; Supplementary Figures S5 and S6).

For the other datasets that were not deeply sequenced and with high complexity, including $CAMI_H$, S1 and S2, the three short-read assemblers generated comparable contig contiguity, #*MQ*, #*HQ*, #*NC* and number of annotated microbes (Table 2; Figure 3 A and B; Supplementary Table S1).

We also compared the quality of assemblies generated using different short-read sequencing platforms, i.e. the Illumina HiSeq sequencing and the BGISEQ-500 sequencing, on S1 and S2. We observed that the two short-read sequencing platforms generated

Table 2. Contig statistics of short-read, linked-read and long-read assemblies. The short-reads used for S1 and S2 in this table were generated by Illumina sequencing. The long-reads used for ZYMO in this table were generated by ONT GridION sequencing.

Assembler	MEGAHIT	IDBA-UD	metaSPAdes	cloudSPAdes	Athena	Canu	metaFlye	Lathe
$CAMI_L$								
Genome fraction (%)	**59.46**	53.35	56.54	**52.41**	46.80	39.34	**47.40**	35.04
Largest alignment (kb)	1,259.44	841.41	**1,362.26**	2,209.45	**2,214.66**	**1,426.13**	1,286.81	1,425.15
NA50 (kb)	46.72	35.96	**70.96**	92.26	**182.06**	92.13	**131.07**	111.60
Normalized NGA50	0.0065	0.0053	**0.0092**	0.0097	**0.0174**	**0.3047**	0.0215	0.0665
N50 (kb)	49.46	36.09	**71.80**	93.82	**186.99**	99.64	**159.74**	131.02
$CAMI_{M1}$								
Genome fraction (%)	**41.38**	37.55	37.94	**26.74**	21.19	24.39	**28.71**	22.18
Largest alignment (kb)	1,218.56	689.95	**1,617.41**	1,040.92	**1,519.07**	929.88	**2,233.35**	804.17
NA50 (kb)	18.62	14.89	**25.99**	42.98	**170.82**	**70.95**	60.10	69.08
Normalized NGA50	0.0033	0.0024	**0.0049**	0.0032	**0.0058**	**0.3926**	0.2606	0.0499
N50 (kb)	19.14	15.09	**26.17**	44.17	**179.52**	75.73	**89.34**	72.82
$CAMI_{M2}$								
Genome fraction (%)	**45.28**	41.37	42.55	**35.06**	24.54	24.53	**29.45**	22.98
Largest alignment (kb)	**1,427.21**	1,124.82	881.79	1,318.15	**1,411.13**	**1,967.08**	1,247.65	945.75
NA50 (kb)	29.09	29.72	**36.61**	32.68	**193.54**	58.91	**79.99**	74.50
Normalized NGA50	0.0060	0.0059	**0.0091**	0.0071	**0.0101**	**0.3827**	0.2173	0.0616
N50 (kb)	30.59	30.02	**36.89**	33.38	**196.10**	61.85	**93.65**	78.85
$CAMI_H$								
Genome fraction (%)	**69.40**	–	63.84	**23.27**	20.96	**42.48**	40.95	33.16
Largest alignment (kb)	1,623.05	–	**2,601.94**	1,798.96	**2,096.94**	**3,725.75**	2,296.49	1,473.33
NA50 (kb)	13.85	–	**16.90**	57.86	**141.58**	37.67	**54.43**	42.75
Normalized NGA50	0.0035	–	**0.0047**	0.0049	**0.0071**	**0.4597**	0.2792	0.0035
N50 (kb)	14.46	–	**17.04**	59.26	**149.57**	43.02	**82.62**	51.76
ATCC20								
Genome fraction (%)	55.89	55.77	**56.36**	–	**72.89**	60.97	**84.32**	65.63
Largest alignment (kb)	336.04	329.54	**539.00**	–	**2,279.77**	**6,387.97**	6,368.05	6,368.17
NA50 (kb)	24.69	22.32	**30.42**	–	**253.12**	1,229.85	1,369.00	**2,073.29**
Normalized NGA50	0.0088	0.0079	**0.0092**	–	**0.0859**	0.2464	**0.4236**	0.3107
N50 (kb)	25.50	22.62	**30.85**	–	**380.38**	2,519.62	3,132.67	**3,132.78**
ZYMO								
Genome fraction (%)	25.11	**26.14**	26.05	–	–	16.19	**35.63**	29.75
Largest alignment (kb)	**768.34**	465.45	641.46	–	–	**4,187.88**	4,124.75	3,445.23
NA50 (kb)	**167.51**	122.50	79.92	–	–	**2,597.88**	212.74	1,022.69
Normalized NGA50	**0.0158**	0.0137	0.0114	–	–	0.1311	**0.2036**	0.1793
N50 (kb)	**167.77**	122.50	80.58	–	–	**6,788.48**	293.03	2,537.25
S1								
Total length (Mb)	**283.38**	269.83	273.18	**518.27**	429.33	199.45	**243.88**	167.60
Largest contig (kKb)	534.60	719.21	**720.26**	844.48	**2,344.85**	3,476.84	3,388.25	**3,489.50**
N50 (Kb)	**11.58**	8.40	10.52	17.31	**128.91**	93.01	**168.44**	100.98
S2								
Total length (Mb)	**267.09**	250.15	252.27	**401.12**	356.36	190.19	**256.78**	111.55
Largest contig (kb)	384.45	467.27	**481.19**	646.68	**2,067.23**	4,055.09	**4,327.16**	3,157.79
N50 (kb)	**16.53**	10.60	13.17	28.98	**182.15**	147.96	**239.01**	118.41
P1								
Total length (Mb)	**136.90**	131.10	132.30	**237.17**	207.73	159.82	**194.64**	156.88
Largest contig (kb)	**586.48**	322.91	346.20	602.20	**2,561.26**	3,399.27	3,306.68	**4,685.91**
N50 (kb)	**15.26**	12.49	10.17	35.34	**222.85**	**192.14**	189.66	118.50
P2								
Total length (Mb)	**90.67**	84.78	87.41	**108.53**	76.80	82.73	**92.10**	73.04
Largest contig (kb)	**544.55**	489.48	157.85	443.76	**1,027.99**	3,792.50	**3,868.09**	3,779.10
N50 (kb)	**15.30**	12.91	11.63	12.32	**101.76**	**275.09**	189.91	87.32

The largest value obtained by the assemblers for each read type (short-reads, linked-reads, or long-reads) is in bold.

assemblies with comparable contig contiguity, #*MQ*, #*HQ*, #*NC* (Supplementary Table S1), and numbers of annotated microbes on S1 and S2 (Supplementary Figure S7). However, MAGs with underlying sequence differences were reconstructed from the two sequencing platforms, which led to mutually exclusive taxonomic entities being detected for the two platforms. For example, on S1 (Supplementary Figure S7), *Sutterella faecalis* was exclusively annotated to the metaSPAdes assembly of Illumina HiSeq short-read sequencing data, and *Ruminococcus bicirculans* was exclusively annotated to the metaSPAdes assembly of BGISEQ-500 short-read sequencing data. This result suggests that there are differences between the two platforms with regard to the sensitivity and specificity of taxonomic classification, and that certain microbes may be selectively targeted by the choice of sequencing platform.

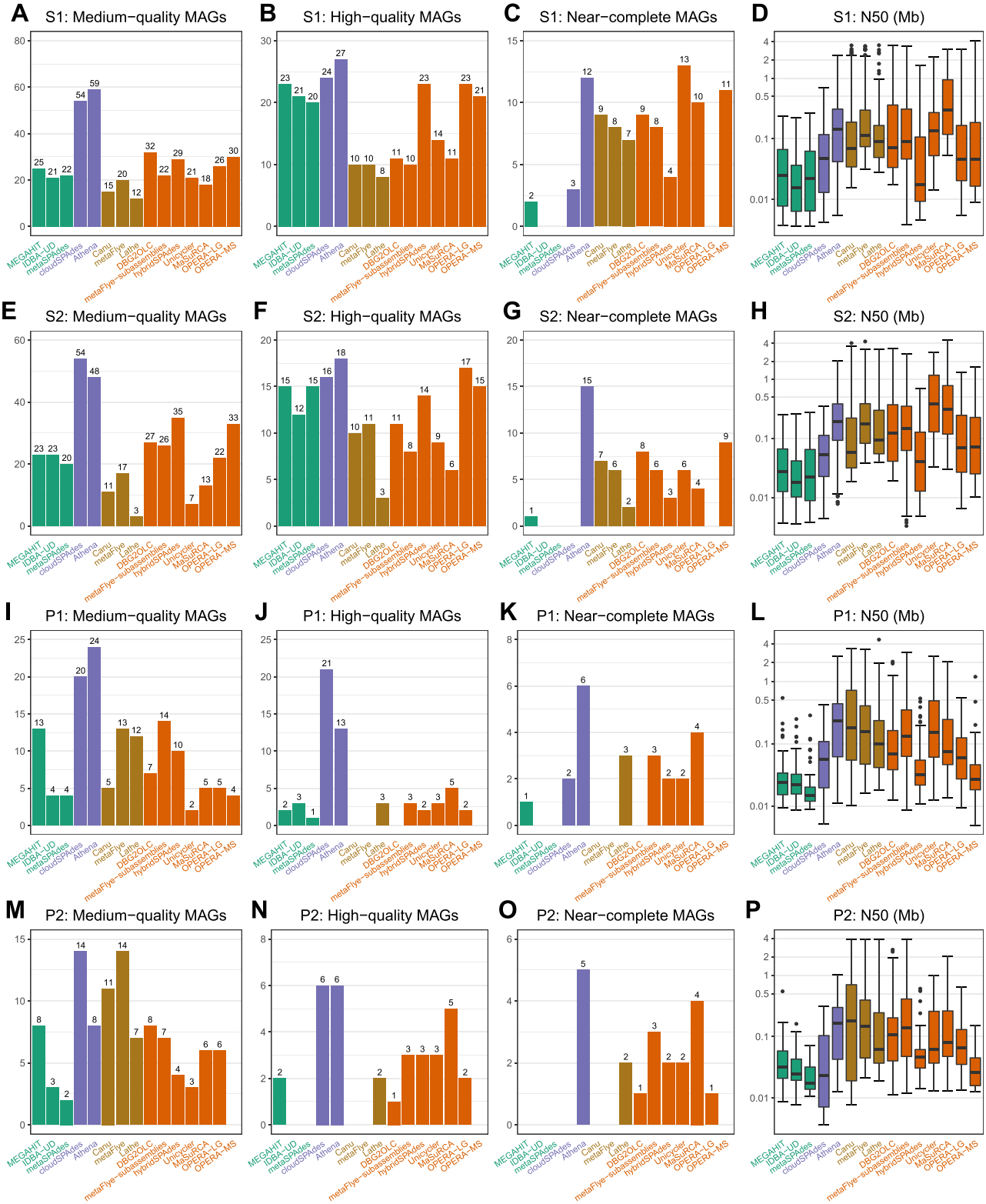

Figure 2. Numbers of medium-quality, high-quality and near-complete MAGs and the MAG N50 values for the assemblies generated from the real datasets (**A–D** for S1; **E–H** for S2; **I–L** for P1; **M–P** for P2). The short-reads used in **A–D** and **E–H** were generated by Illumina HiSeq. Color key: turquoise (short-reads), purple (linked-reads), tan (long-reads) and orange (mixed reads).

Metagenome assembly performance per linked-read sequencing tool

We compared the assemblies generated by cloudSPAdes and Athena on 10x linked-read sequencing data. CloudSPAdes failed to assemble the ATCC20 linked-read dataset due to insufficient memory (need >1TB RAM). On the other datasets, we observed that cloudSPAdes generated higher GFs than Athena on all the CAMI datasets (123.02% on average; Table 2), and produced

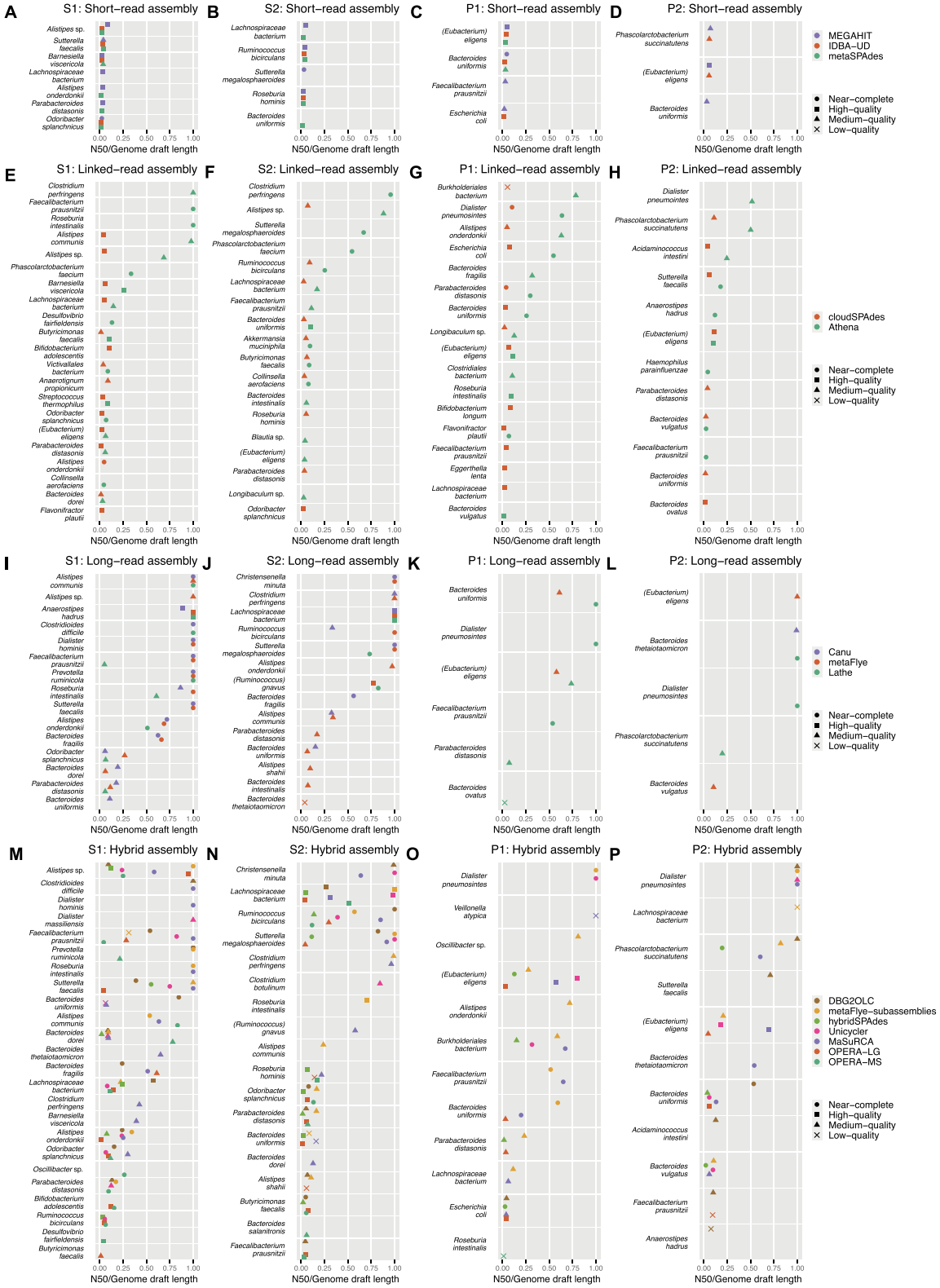

Figure 3. MAG annotations of the assemblies generated from the real datasets (**A–D** for short-read assemblies; **E–H** for linked-read assemblies; **I–L** for long-read assemblies; **M–P** for hybrid assemblies). The N50/genome draft length was used to evaluate the contiguity of MAGs. The short-reads used in **A** and **B** were sequenced by Illumina HiSeq.

higher *ALs* than Athena on all the real datasets from human gut microbiomes (122.19% on average; Table 2; Supplementary Table S1). Similar trends of *GFs* on ATCC20 were reported in a previous study [14].

Athena produced substantially higher contig contiguity than cloudSPAdes. On the CAMI simulation datasets, the contigs from Athena had higher N50s (361.37% on average), NA50s (357.92% on average), and normalized NGA50s (162.96% on average) than those from cloudSPAdes (Table 2; Supplementary Figure S5). Tolstoganov *et al*. [14] also reported higher contig N50s and NA50s from Athena on the ATCC20 mock dataset. For the real datasets, Athena obtained significantly higher contig N50s (average N50: Athena = 158.92 kb, cloudSPAdes = 23.49 kb; Table 2; Supplementary Table S1) and MAG N50s (Wilcoxon rank-sum test *P*-value: S1 = 1.99e-14, S2 = 4.46e-26, P1 = 1.33e-11, P2 = 2.26e-6; Figure 2D, H, L and P; Supplementary Figures S3, S4 and S8, S9; Supplementary Table S2) than cloudSPAdes.

The two linked-read assemblers produced comparable *#MQ* and *#HQ* (Figure 2 A, B, E, F, I, J, M and N). Athena generated substantially more *#NC* than cloudSPAdes on the real datasets (*#NC* in total: Athena = 38, cloudSPAdes = 5; Figure 2 C, G, K and O). A comparable number of microbes was annotated on the MAGs generated by the two assembly tools, and both tools identified unique microbes (Figure 3 E– He-h). For example, Athena identified three unique microbes (*Roseburia intestinalis*, *Phascolarctobacterium faecium* and *Desulfovibrio fairfieldensis*), while cloudSPAdes identified four unique microbes (*Sutterella faecalis*, *Bifidobacterium adolescentis*, *Flavonifractor plautii* and a *Longibaculum* sp.) on the S1 dataset (Figure 3E).

Metagenome assembly performance per long-read sequencing tool

We compared the performances of seven long-read assembly tools on PacBio CLR and ONT data: Shasta, wtdbg2, MECAT2 (PacBio CLR only), NECAT (ONT only), Canu, metaFlye and Lathe. Canu, metaFlye and Lathe generated assemblies with significantly higher *GFs* on the simulation and mock datasets (328.45 times on average; Supplementary Table S1) and the assemblies with much longer *ALs* on all of the real datasets (4.64 times on average; except wtdbg2; Supplementary Table S1) than the other four tools (Shasta, wtdbg2, MECAT2 and NECAT). Shasta, wtdbg2, MECAT2 and NECAT also could not generate high-quality or near-complete MAGs for the real datasets from human gut microbiomes (Supplementary Table S1. This may be because Shasta, wtdbg2, MECAT2 and NECAT were not designed for metagenome assembly and may preferentially generate long contigs derived from a single species. Therefore, we only included Canu, metaFlye and Lathe in the subsequent analysis as they generated assemblies with reasonable lengths.

The contigs generated by metaFlye had higher *GFs* than those generated by Canu and Lathe on most of the simulation and mock datasets (1.44 times on average, except Canu on $CAMI_H$; Table 2). The metaFlye also generated the highest *ALs* on the real datasets (an average of 1.40 times greater than Canu and Lathe; Table 2).

The contig contiguities of metaFlye assemblies were better than those of the corresponding Lathe assemblies on most datasets sequenced by either PacBio CLR or ONT: metaFlye generated higher contig N50s (1.60 times), NA50s (1.27 times) and normalized NGA50s (79.39 times) than Lathe on $CAMI_H$ (Table 2; Supplementary Figure S5), and higher contig N50s (average contig N50 on S1, S2 and P1: metaFlye = 199.04 Kb, Lathe = 112.63 Kb; Table 2; Supplementary Table S1) and significantly higher MAG N50s (Wilcoxon rank-sum test *P*-value: S1 = 2.27e-4, S2 = 9.85e-4, P1 = 4.83e-2; Figure 2 D, H and Ld, h and l; Supplementary Figures S3, S8 and S9; Supplementary Table S2) than Lathe on the real datasets (S1, S2 and P1).

Compared with the assemblies generated by metaFlye and Lathe, the assemblies produced by Canu had higher (or at least similar) contig contiguity on ONT data, but lower (or at best similar) contiguity on PacBio CLR data. Eight datasets were generated by ONT, including $CAMI_L$, $CAMI_{M1}$, $CAMI_{M2}$, $CAMI_H$, ZYMO (GridION and PromethION), P1 and P2. Canu generated assemblies with higher normalized NGA50s (4.77 times on average) than metaFlye on the four CAMI datasets and generated assemblies with higher contig N50s (16.40 times on average) and NA50s (8.89 times on average) than metaFlye on the GridION and PromethION datasets from ZYMO (Table 2; Supplementary Figure S5; Supplementary Table S1). Canu also generated assemblies with higher contig contiguity than those generated by metaFlye and Lathe on P1 and P2 (average contig N50s on P1 and P2: Canu = 233.62 kb, metaFlye = 189.79 kb, Lathe = 102.91 kb; Table 2; Supplementary Table S1). On the datasets sequenced by PacBio CLR (ATCC20, S1 and S2), metaFlye and Lathe produced assemblies with substantially higher contig N50s (124.33% on average), NA50s (139.95% on average) and normalized NGA50s (149.04% on average) than Canu on ATCC20 (Table 2; Supplementary Figure S10); metaFlye produced assemblies with higher contig N50s (171.32% on average) and MAG N50s (Wilcoxon rank-sum test *P*-value: S1 = 7.45e-5, S2 = 3.77e-8) than Canu on S1 and S2 (Table 2; Figure 2 D and H; Supplementary Figures S8 and S9; Supplementary Table S2).

We further evaluated the reported MAG quality metrics and annotation results of the assemblies generated by these three tools. For the PacBio CLR datasets from S1 and S2, metaFlye and Canu generated comparable *#HQ* and *#NC*, which were much larger than those generated by Lathe (*#HQ*: metaFlye = 21, Canu = 20, Lathe = 11; *#NC*: metaFlye = 14, Canu = 16, Lathe = 9; Figure 2 B, C and F, G). Metaflye and Canu also called more microbes during annotation (metaFlye = 26, Canu = 22, Lathe = 12; Figure 3 I and J). For the P1 and P2 ONT datasets, the opposite trend was observed in reported MAG quality metrics (*#HQ*: Lathe = 5, Canu = 0, metaFlye = 0; *#NC*: lathe = 5, metaFlye = 0, Canu = 0; Figure 2 J, K and N, O) and annotated microbes (Lathe = 9, metaFlye = 4, Canu = 1; Figure 3 K and L). Since the core assembly algorithms of Lathe are the same as those of Canu and metaFlye, the above observations imply that the post-processing modules of Lathe (assembly polishing and circularization) are beneficial for ONT data assembly. We also found that all three assembly tools identified uniquely annotated microbes, e.g. metaFlye, Canu and Lathe uniquely reported *Alistipes* sp. (S1), *Bacteroides uniformis* (S1) and *Dialister pneumosintes* (P1), respectively (Figure 3 I and K).

Metagenome hybrid assembly performance

We evaluated seven hybrid assembly tools, namely DBG2OLC, metaFlye-subassemblies (Methods), hybridSPAdes, Unicycler, MaSuRCA, OPERA-LG, and OPERA-MS, on short- and long-read sequencing datasets. Compared with the other tools, metaFlye-subassemblies produced assemblies with significantly higher *GF* (198.39% on average; Table 3) on ATCC20 mock datasets and significantly higher *ALs* on P1 (243.41% on average; Table 3) and P2 (181.20% on average; Table 3). The other assemblers (except Unicycler, MaSuRCA) generated comparable *GFs* and *ALs* (Table 3) on most datasets. Unicycler and MaSuRCA produced substantially less *GFs* and *ALs* (Table 3) than the other assemblers (Table 3), which might be because they were developed for the assembly of isolates and not optimized for metagenome assembly.

DBG2OLC and metaFlye-subassemblies generated contigs with substantially higher contig contiguity than hybridSPAdes, OPERA-LG and OPERA-MS on most datasets. For example, metaFlye-subassemblies produced folds better N50s (8.22 times on average), NA50s (5.99 times on average) and normalized NGA50s (5.99 times on average; Supplementary Figures S11–S13) than hybridSPAdes, OPERA-LG and OPERA-MS on ATCC20 and ZYMO mock datasets (Table 3). The metaFlye-subassemblies also achieved folds better contig N50s (3.42 times on average; Table 3) and significantly better MAG N50s than hybridSPAdes, OPERA-LG and OPERA-MS (Wilcoxon rank-sum test *P*-values were S1: hybridSPAdes = 4.34e-11, OPERA-LG = 3.27e-5 and OPERA-MS = 1.65e-4; S2: hybridSPAdes = 4.14e-13, OPERA-LG = 1.60e-4 and OPERA-MS = 1.05e-5; P1: hybridSPAdes = 6.20e-15, OPERA-LG = 8.02e-7 and OPERA-MS = 9.90e-14; P2: hybridSPAdes = 9.49e-6, OPERA-LG = 2.37e-3 and OPERA-MS = 1.79e-10; Figure 2 D, H, L and P; Supplementary Figures S3, S4 and S8, S9; Supplementary Table S2) on all of the real datasets.

When we compared metaFlye-subassemblies and DBG2OLC, we observed assemblies from metaFlye-subassemblies had higher contig N50s (27.99 times on average), NA50s (19.18 times on average) and normalized NGA50s (6.84 times on average) than those from DBG2OLC on the low-complexity mock datasets ATCC20 and ZYMO (Table 3; Supplementary Figures S11–S13; Supplementary Table S1). Results were sample dependent for assemblies generated with real data from human gut microbiomes: metaFlye-subassemblies obtained higher N50s than DBG2OLC on P1, but lower N50s than DBG2OLC on S1, S2 and P2.

Though Unicycler and MaSuRCA produced substantially lower *GF*s and *AL*s than the other assemblers on most of the datasets, the two assemblers obtained assemblies with the highest NA50s on all the simulation and mock datasets, and the highest N50s on all the real datasets (Table 3). This result may have been caused by Unicycler and MaSuRCA excluding assemblies with poor contiguity as part of their workflow, thereby producing assemblies with high contiguity and low total assembly length.

We observed inconsistent hybrid MAG quality for datasets from different sequencing platforms. On the S1 and S2 samples sequenced by Illumina HiSeq and PacBio CLR, Unicycler and OPERA-MS achieved the highest #*NC* (S1: Unicycler = 13; S2: OPERA-MS = 9; Figure 2 C and G). OPERA-LG, OPERA-MS and hybridSPAdes produced comparable #*HQ* (S1: OPERA-LG = 23, OPERA-MS = 21, hybridSPAdes = 23; S2: OPERA-LG = 17, OPERA-MS = 15, hybridSPAdes = 14; Figure 2), which were substantially higher than the other hybrid assemblers (Figure 2). On the P1 and P2 samples sequenced by Illumina HiSeq and ONT, MaSuRCA generated the highest #*NC* (S1: MaSuRCA = 4; S2: MaSuRCA = 4; Figure 2 K and O) and the highest #*HQ* (S1: MaSuRCA = 5; S2: MaSuRCA = 5; Figure 2 J and N).

The number of microbes that could be called by annotation of the hybrid assemblies also varied between datasets. On S1 (Figure 3 M), the MaSuRCA assembly was annotated with the highest number of microbes (15), while hybridSPAdes obtained the lowest microbe call rate (5). The other hybrid assemblers generated similar microbe call rates. On S2 (Figure 3 N), all the hybrid assemblers except Unicycler obtained similar microbe call rates (metaFlye-subassemblies = 10, OPERA-LG = 10, DBG2OLC = 9, MaSuRCA = 9, OPERA-MS = 8, hybridSPAdes = 8), which was significantly more than Unicycler (5). On P1 and P2 (Figure 3 P and Q), assemblies from metaFlye-subassemblies and MaSuRCA generated similar numbers (total number of annotated microbes: metaFlye-subassemblies = 14, MaSuRCA = 13). These numbers were larger than that of DBG2OLC (8), OPERA-LG (7), OPERA-MS (1), hybridSPAdes (7) and Unicycler (7).

The MAGs generated by various hybrid assembly tools were annotated to unique microbes, e.g. *Dialister hominis*, *Dialister massiliensis* and *Oscillibacter* sp. were uniquely identified from the assemblies of MaSuRCA, Unicycler and OPERA-MS on S1, respectively (Figure 3 M). *Lachnospiraceae bacterium* and *Sutterella faecalis* were uniquely annotated from the assemblies of metaFlye-subassemblies and DBG2OLC on P2, respectively (Figure 3 P).

Metagenome assembly statistics per read type

We compared assembly statistics (*AL*, *GF*, NA50, N50, #*HQ*, #*NC* and the number of annotated microbes) of the four sequencing technologies discussed above in terms of read type: (1) short-reads, (2) linked-reads, (3) long-reads and (4) mixed reads (short- and long-reads). We observed the *GF*s and *AL*s of mixed read (hybrid) assemblies were higher than or similar to those generated separately from either short-reads or long-reads for all datasets (Supplementary Table S3).

The contig contiguities of linked-read, long-read and hybrid assemblies were substantially higher than those of short-read assemblies (Supplementary Table S3). Linked-read assembly generated the highest NA50 on $CAMI_{M1}$, $CAMI_{M2}$ and $CAMI_{H}$, and the highest N50 on P1. Long-read assembly obtained the highest NA50 on ATCC20 and the highest N50 on P1 and P2. Hybrid assembly produced the highest NA50 on $CAMI_{L}$ and ZYMO, and the highest N50 on S1 and S2 (Supplementary Table S3). Linked-read assembly obtained the highest #*MQ* (151), #*HQ* (72), #*NC* (38) and identified the highest number of annotated microbes (51) on all of the real datasets (Figure 4; Supplementary Table S3). These values were higher than those of the short-read assemblies (#*MQ* = 77, #*HQ* = 44, #*NC* = 6, number of annotated microbes = 20; Figure 4; Supplementary Table S3), the long-read assemblies (#*MQ* = 64, #*HQ* = 26, #*NC* = 21, number of annotated microbes = 36; Figure 4; Supplementary Table S3), and the hybrid assemblies (#*MQ* = 89, #*HQ* = 50, #*NC* = 30, number of annotated microbes = 41; Figure 4; Supplementary Table S3).

As with the short-reads platforms discussed earlier, the assemblies generated from each read type identified unique microbes, implying that read types and combinations of read types can be employed to selectively detect certain microbes; for example, *Roseburia hominis*, *Phascolarctobacterium faecium*, *Alistipes onderdonkii* and *Roseburia intestinalis* were uniquely identified by short-read, linked-read, long-read and hybrid assembly of the S2 dataset, respectively (Figure 4 B).

MAG generation performance on simulation and mock datasets

For the simulation and mock datasets, we also evaluated the quality of the MAGs generated by different assembly tools. On the simulation datasets of CAMI, hybrid assemblies generated the highest #*NC* compared with assemblies from the other sequencing technologies (Supplementary Figure S14). The hybridSPAdes achieved the highest #*NC* (4) on $CAMI_{L}$ (Supplementary Figure S14). DBG2OLC obtained the highest #*NC*s for the remaining CAMI datasets, which were comparable with those of MaSuRCA ($CAMI_{M1}$: DBG2OLC = 7, MaSuRCA = 5; $CAMI_{M2}$: DBG2OLC = 9, MaSuRCA = 9; $CAMI_{H}$: DBG2OLC = 36, MaSuRCA = 32; Supplementary Figure S14). The #*NC*s from long-read assemblies were comparable or higher than those from short- or linked-read assemblies (Supplementary Figure S14). Canu generated the highest #*NC*s among the long-read assemblers (#*NC* in total:

Table 3. Contig statistics of hybrid assemblies. The short-reads used for S1 and S2 in this table were generated by Illumina sequencing. The long-reads used for ZYMO in this table were generated by ONT GridION sequencing.

Assembler	DBG2OLC	metaFlye – subassemblies	hybridSPAdes	Unicycler	MaSuRCA	OPERA-LG	OPERA-MS
$CAMI_L$							
Genome fraction (%)	49.98	56.79	**58.07**	31.68	44.19	57.80	50.90
Largest alignment (kb)	1,213.11	**2,219.70**	1,362.24	1,425.06	1,426.34	2,218.31	1,287.72
NA50 (kb)	76.19	126.98	91.26	**235.98**	131.16	115.93	53.42
Normalized NGA50	0.0381	0.0178	0.0134	0.0103	**0.0591**	0.0134	0.0066
N50 (kb)	78.36	141.96	93.79	**261.61**	149.27	125.23	56.21
$CAMI_{M1}$							
Genome fraction (%)	33.78	38.72	**40.14**	19.60	29.36	39.51	39.80
Largest alignment (kb)	1,725.22	**2,233.38**	1,216.50	2,231.08	2,012.29	2,231.12	1,111.21
NA50 (kb)	77.61	68.44	25.50	94.51	**129.20**	66.24	33.14
Normalized NGA50	0.1064	**0.1640**	0.0055	0.0069	0.0662	0.0109	0.0044
N50 (kb)	85.76	82.23	25.93	97.11	**151.76**	79.95	35.81
$CAMI_{M2}$							
Genome fraction (%)	30.69	43.25	**44.28**	24.32	27.70	44.03	42.89
Largest alignment (kb)	**3,463.86**	1,521.25	1,490.93	1,521.29	1,761.39	1,422.92	1,213.71
NA50 (kb)	105.11	68.34	43.46	**176.50**	175.89	76.51	33.39
Normalized NGA50	0.1031	**0.1216**	0.0125	0.0108	0.0728	0.0148	0.0062
N50 (kb)	109.01	74.64	45.05	188.74	**194.57**	81.79	34.93
$CAMI_H$							
Genome fraction (%)	63.96	65.20	66.05	–	53.71	65.48	**67.88**
Largest alignment (kb)	2,956.06	**3,442.50**	3,438.88	–	3,051.45	2,601.94	1,022.07
NA50 (kb)	92.79	46.81	27.67	–	**135.96**	47.25	26.64
Normalized NGA50	0.1071	**0.1684**	0.0077	–	0.0642	0.0116	0.0040
N50 (kb)	98.07	68.79	28.58	–	**170.05**	58.57	29.25
ATCC20							
Genome fraction (%)	40.46	**84.19**	69.46	–	22.80	56.92	57.86
Largest alignment (kb)	927.92	**4,901.00**	928.58	–	1,735.64	1,680.37	2,616.12
NA50 (kb)	96.84	**1,167.15**	92.08	–	326.88	222.90	656.28
Normalized NGA50	0.0274	**0.3722**	0.0313	–	0.0315	0.0380	0.1185
N50 (kb)	126.81	**2,526.40**	99.82	–	836.09	435.16	1,770.61
ZYMO							
Genome fraction (%)	36.49	36.41	**36.71**	13.36	18.79	30.33	33.86
Largest alignment (kb)	2,048.80	**6,274.62**	1,446.59	4,923.41	4,906.62	1,951.03	1,365.92
NA50 (kb)	30.19	211.50	34.65	**4,923.41**	1,173.30	178.90	174.75
Normalized NGA50	0.0805	**0.1856**	0.0262	0.0781	0.1423	0.0506	0.0459
N50 (kb)	32.11	285.95	35.60	**5,364.48**	2614.10	207.70	263.72
S1							
Total length (Mb)	302.03	339.23	**346.50**	217.17	156.12	334.37	341.67
Largest contig (kb)	3,456.14	3,388.44	1,635.23	2,291.92	4,144.22	3,006.11	**4,149.71**
N50 (kb)	151.78	91.19	21.51	143.17	**564.60**	51.40	52.24
S2							
Total length (Mb)	301.29	318.21	318.97	115.98	163.84	311.85	**319.30**
Largest contig (kb)	**4,663.05**	2,698.47	814.50	4,037.66	4,552.26	1,730.89	1,623.96
N50 (kb)	244.48	147.85	31.99	**605.60**	513.73	89.95	83.00
P1							
Total length (Mb)	165.72	**233.50**	139.81	43.97	108.54	144.64	95.32
Largest contig (kb)	2,069.69	2,940.83	689.74	**3,468.12**	2,120.13	831.64	1,188.06
N50 (kb)	60.46	119.49	22.80	**200.30**	130.77	40.24	19.08
P2							
Total length (Mb)	79.16	**125.86**	92.14	47.85	61.41	92.75	66.97
Largest contig (kb)	2,597.67	**3,874.78**	860.71	2,495.76	2,041.58	758.52	275.14
N50 (kb)	75.47	66.29	20.54	76.32	**105.28**	30.21	12.33

The largest value obtained by the assemblers is in bold.

Canu = 23, metaFlye = 5, Lathe = 5; Supplementary Figure S14). The numbers were higher than those of short-read assemblers (#*NC* in total: MEGAHIT = 10, IDBA-UD = 4, metaSPAdes = 5) and linked-read assemblers (#*NC* in total: cloudSPAdes = 5, Athena = 12; Supplementary Figure S14).

On the mock datasets from ATCC20 and ZYMO, long-read assemblies generated the highest #*NC*. The metaFlye obtained much more #*NC* than the other long-read assemblers on ATCC20 (metaFlye = 14, Canu = 6, Lathe = 10; Supplementary Figure S15), and Lathe was the best long-read assembler for ZYMO (Lathe = 3, Canu = 0, metaFlye = 1; Supplementary Figure S16). The hybrid assemblies generated less #*NC*s than long-read assemblies. On ATCC20, metaFlye-subassemblies and OPERA-MS produced comparable #*NC*s that were more than the other hybrid assemblers (metaFlye-subassemblies = 9, OPERA-MS = 8, OPERA-LG = 1, DBG2OLC = 0, hybridSPAdes = 0, MaSuRCA = 0;

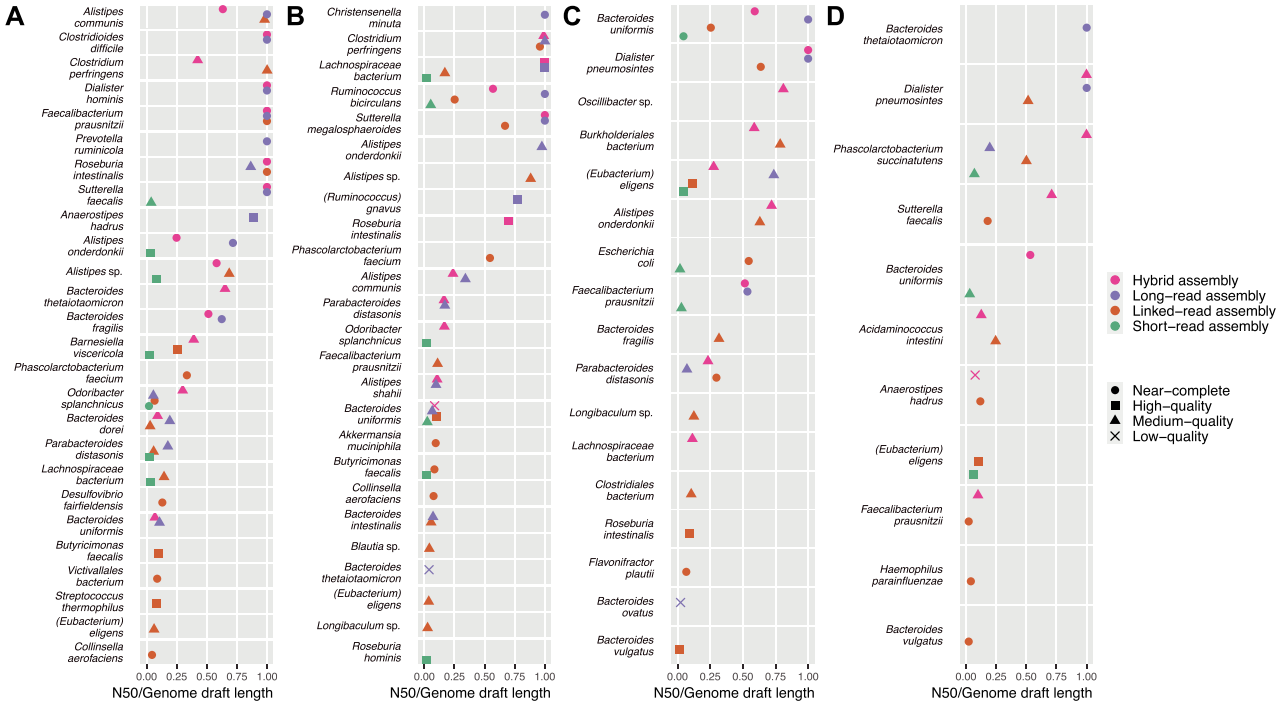

Figure 4. MAG annotations of the assemblies generated with different sequencing technologies from human gut microbiomes (**A** for S1, **B** for S2, **C** for P1 and **D** for P2).

Supplementary Figure S15). The numbers were higher than those of linked- (Athena = 3) and short-read assemblies (MEGAHIT = 1, IDBA-UD = 0, metaSPAdes = 0; Supplementary Figure S15). On ZYMO, DBG2OLC, hybridSPAdes and MaSuRCA generated the same #NCs (2), that were higher than those of metaFlye-subassemblies (1), Unicycler (1), MaSuRCA (1) and OPERA-LG (0; Supplementary Figure S16). The #NCs of hybrid assemblies were also higher than those of short-read assemblies (MEGAHIT = 0, IDBA-UD = 0, metaSPAdes = 0; Supplementary Figure S16).

Metagenome assembly performance in terms of mismatches and misassemblies

On the simulation and mock datasets, we investigated the number of mismatches and misassemblies per assembly tool and sequencing technology.

Among the short-read assemblers, metaSPAdes generated the smallest number of misassemblies on all the simulation and mock datasets (MEGAHIT: 3.28 times higher on average, IDBA-UD: 3.31 times higher on average; Supplementary Table S1). IDBA-UD generated assemblies with the lowest number of mismatches on CAMI and ATCC20 datasets (MEGAHIT: 2.43 times higher on average, metaSPAdes: 1.74 times higher on average; Supplementary Table S1). The three short-read assemblers obtained a similar number of mismatches on ZYMO (Supplementary Table S1).

The number of misassemblies made by the two linked-read assemblers, cloudSPAdes and Athena, were similar but we observed that assemblies from Athena had lower mismatches (cloudSPAdes: 1.56 times higher on average).

Among the three long-read assemblers (Canu, metaFlye and Lathe), Canu obtained the lowest number of misassemblies on $CAMI_H$ and ZYMO (Supplementary Table S1). Lathe achieved the lowest number of misassemblies on the remaining datasets ($CAMI_L$, $CAMI_{M1}$, $CAMI_{M2}$ and ATCC20; Supplementary Table S1). Canu consistently produced the lowest number of mismatches on all the long-read datasets from simulation and mock communities (Supplementary Table S1).

Among the hybrid assemblers, DBG2OLC had the lowest number of misassemblies on $CAMI_L$, $CAMI_{M2}$, $CAMI_H$ (Supplementary Table S1). Unicycler made the lowest number of misassemblies on $CAMI_{M1}$ and the ONT GridION dataset of ZYMO (Supplementary Table S1). MaSuRCA and hybridSPAdes made the lowest number of misassemblies on the ONT PromethION dataset of ZYMO and on ATCC20 (Supplementary Table S1). Although Unicycler failed to assemble $CAMI_H$, ATCC20 and the ONT PromethION dataset of ZYMO due to insufficient memory (need >1T RAM), it obtained the lowest number of mismatches on all the other simulation and mock datasets among the hybrid assemblers (Supplementary Table S1).

We also compared the number of misassemblies and mismatches between different sequencing technologies. Hybrid assembly achieved the lowest number of misassemblies on $CAMI_L$ and ZYMO. Linked-read assembly obtained the lowest number of misassemblies on $CAMI_{M1}$, $CAMI_{M2}$, $CAMI_H$. Short-read and long-read assemblies obtained the lowest number of misassemblies on ATCC20 and ZYMO, respectively. In contrast, hybrid assembly generated the lowest number of mismatches on all simulation and mock datasets except for $CAMI_H$.

Evaluation of computational time and resource requirements

We compared the running time (real time) and peak memory consumption (maximum RSS) of the assembly tools on the CAMI simulation datasets (Figure 5). MetaSPAdes and IDBA-UD had longer running times (metaSPAdes: 1.92 times on average; IDBA-UD: 1.81 times on average) and significantly higher memory usage (metaSPAdes: 8.72 times more on average; IDBA-UD: 20.82 times more on average) than MEGAHIT (Figure 5). During linked-read assembly, cloudSPAdes had a significantly longer running time (5.80 times

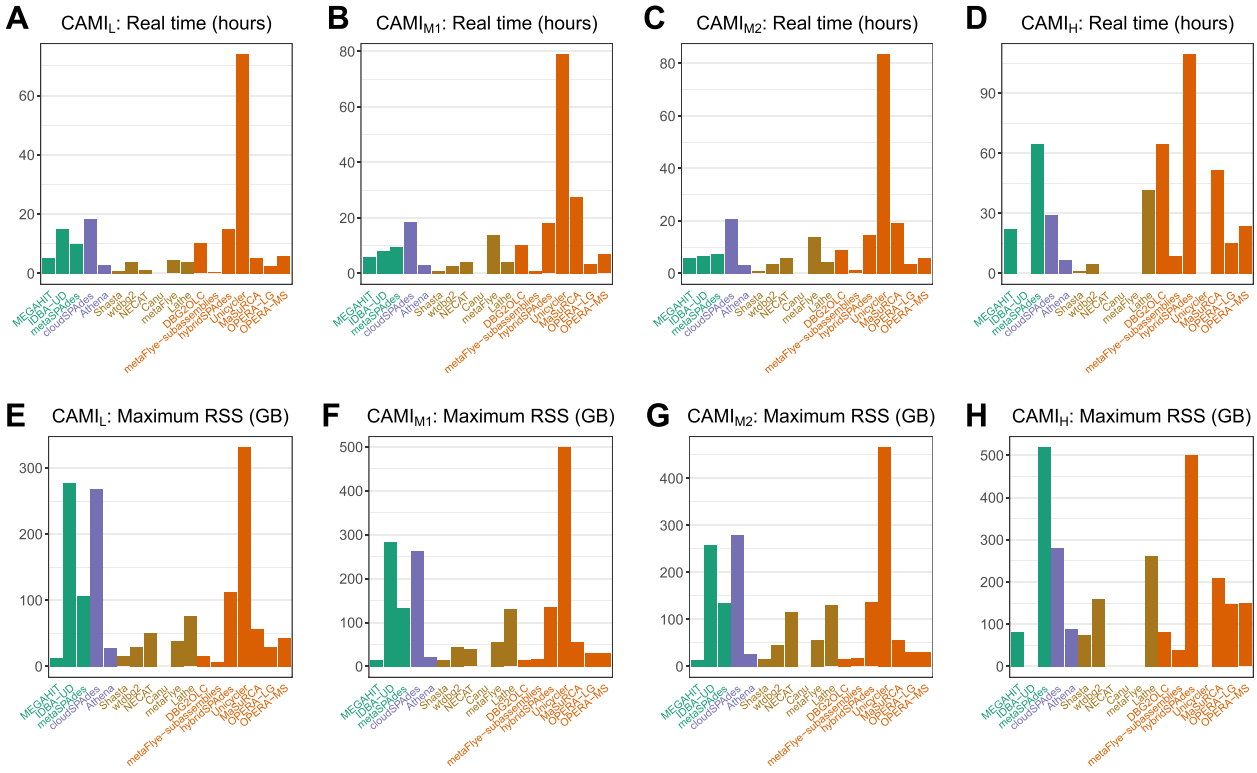

Figure 5. Computational resources (real time and resident set size [RSS]) consumed by the assembly tools in analyzing CAMI datasets. IDBA-UD, NECAT, metaFlye and Unicycler were not used for the analysis of CAMI$_H$ because it was found that they exceeded the maximum memory limitation. Canu was also not used because it exceeded our server wall-clock time (more than 7 days). Color key: turquoise (short-reads), purple (linked-reads), tan (long-reads) and orange (mixed reads).

longer on average) and consumed higher peak memory (8.73 times more on average) than Athena (Figure 5). Canu took more than 7 days to complete metagenome assembly on each of the CAMI datasets, which was more than twice as long as the other long-read assembly tools took (we excluded Canu in Figure 5 because it exceeded our server wall-clock time). Among the remaining long-read assemblers, Lathe and metaFlye had the highest peak memory consumption (3.57 times more on average) and the longest running time (5.82 times longer on average), respectively. (except for CAMI$_H$; Figure 5). MetaFlye failed to assemble CAMI$_H$ due to insufficient memory (need >1T RAM), where Lathe needed the highest peak memory consumption and running time among the others. During hybrid assembly, Unicycler was 32.37 times slower than the other tools on average and metaFlye-subassemblies was 29.74 times faster than the other tools on average (Figure 5). Unicycler required substantially more memory than all the other hybrid assemblers (16.85 times more on average; Figure 5), and, as mentioned earlier, failed to assemble the CAMI$_H$ datasets due to insufficient memory. Comparative performance in terms of recorded CPU times for all of the preceding assembly tools was similar to the trends observed during real time usage analysis (Supplementary Figure S17).

DISCUSSION

Metagenome assembly aims to reconstruct microbial genomes from metagenomic sequencing data and directly affects the quality of MAGs. Many metagenome assembly tools have been developed to assemble data derived from various sequencing technologies, but there is currently a lack of an independent, comprehensive, and up-to-date evaluation of these tools in the literature. In this study, we benchmarked the performance of 19 commonly used assemblers on simulation, mock, and real datasets with diverse complexities with the aim of providing practical guidance for MAG reconstruction.

Recent studies [41–43] have demonstrated the applicability of representing microbial genomes as MAGs generated via large short-read metagenomic sequencing cohort studies. Nevertheless, short-read assembly usually generates highly fragmented contigs, and as a result, poor quality MAGs. By comparing MEGAHIT, IDBA-UD and metaSPAdes, we observed that MEGAHIT outperformed IDBA-UD and metaSPAdes in generating assemblies from deeply sequenced datasets (>100x), probably because MEGAHIT has an optimized data structure and algorithms that were designed to analyze large datasets. However, metaSPAdes performed better than MEGAHIT and IDBA-UD in generating assemblies from low-complexity datasets, which has previously been reported [36].

We found that linked-read assemblies had consistently better contig contiguity than short-read assemblies but sometimes worse contiguity than long-read assemblies. This is probably because linked-reads help to resolve the ambiguous branches and circles from repetitive sequences in assembly graphs [14, 15]. Regardless, linked-reads fail to capture tandem repeats and highly variable regions presented within long fragments. We furthermore observed that Athena generated assemblies containing contigs with higher contiguity and higher *#NC* than those generated by cloudSPAdes, although cloudSPAdes generated higher *GF*s or *AL*s. The *#NC*s from linked-read assemblies were higher than those from long-read assemblies, which may be attributable to the higher read depth and lower base-error rate of linked-read sequencing.

Long-read assemblers can generate assemblies with high contig contiguity, but individual contigs usually do not represent circular microbial genomes and sometimes cannot be grouped into high-quality MAGs. This may be because (1) high-molecular-weight DNA cannot be easily extracted from some microbes [15]; (2) long-read sequencing is still expensive; and (3) error-prone long-reads result in assembly errors and low-quality MAGs. In our study, we compared the performances of seven state-of-the-art long-read assemblers and found that Canu, metaFlye, and Lathe performed substantially better than the other assemblers. The metaFlye generated assemblies had the highest *GF*s and *AL*s, which is supported by observations from a previous study [35]. Lathe generated assemblies with significantly more #*NC* than metaFlye and Canu from the ONT datasets, suggesting that genome polishing and circularization are essential steps to improve MAG quality generated from ONT long-reads.

Hybrid assembly has been adopted as an expedient way to correct assembly errors arising from error-prone long-reads by using short-reads with high base quality to correct errors in long reads or to polish long-read based assemblies. Our results showed that hybrid assemblies have similar or higher *GF*s and *AL*s, and higher #*NC*s than those from either short- or long-read assemblies. We also found that Unicycler and MaSuRCA generated less *GF*s and *AL*s than the other hybrid assemblers but obtained significantly higher contig contiguity. These two assemblers were not designed for metagenome assembly, but they still obtained the highest #*NC*s on the S1 (for Unicycler), P1 (for MaSuRCA) and P2 (for MaSuRCA) datasets.

Read trimming is a commonly employed pre-assembly quality control step. To evaluate the impact of read trimming, we used raw and trimmed Illumina short-reads (1.3 Gb), 10x linked-reads (downsampled to 5 Gb) and PacBio CLR long-reads (downsampled to 5 Gb) from the ATCC20 mock community. The trimming step for short- and linked-reads was conducted using fastp (v0.21.0) [44] with default parameters. NanoFilt (v2.8.0) [45] was used to filter the long-reads with parameters '–l 500 –headcrop 50' as suggested on NanoFilt's GitHub page. MetaSPAdes, Athena, metaFlye and metaFlye-subassemblies were our assemblers of choice for short-read, linked-read, long-read and hybrid assemblers, respectively. As shown in Supplementary Table S4, differences between assembly statistics before read trimming and after read trimming were negligible. Read trimming actually resulted in small NA50 decreases for metaSPAdes, Athena and metaFlye-subassemblies.

Genome polishing long-read assemblies is a beneficial post-processing step. Many tools have been developed to polish draft genomes assembled from long-reads [46–55]. These tools correct assembly errors using short- or long-reads, and have the ability to improve the accuracy and contiguity of long-read assemblies [56]. Since genome polishing is the post-processing of genome assembly, we did not include the benchmark of these tools in this study.

Reconstructing the genomes of low- (0.1%–1%) and ultra-low (<0.1%) abundance microbes is a challenging task [57, 58]. We evaluated the performances of different assemblers in assembling the genomes of microbes with low- and ultra-low abundance in ATCC20 (Supplementary Table S5). Short-read assemblers were unable to assemble any genomes of low-abundance microbes (*GF* >50%; Supplementary Figure S6), whereas all of these microbes were identified by linked-read (Athena; Supplementary Table S5), long-read (metaFlye and Shasta; Supplementary Figure S10) and hybrid assemblers (DBG2OLC, metaFlye-subassemblies and hybridSPAdes; Supplementary Figure S11). For the five low-abundance microbes in ATCC20, the long-read (metaFlye) and hybrid (metaFlye-subassemblies) assemblers generated higher contig contiguity (4.71 times on average) than the linked-read assembler (Athena) (Supplementary Table S5). Only one (ATCC_8482) of the five ultra-low abundance microbes was assembled by long-read (Lathe and metaFlye; Supplementary Figure S10) and hybrid (metaFlye-subassembly; Supplementary Figure S11) assemblers, suggesting that ultra-low abundance microbes are still difficult to be assembled even if long-reads are available.

Sequencing depth is also an important factor in the assembly of low- and ultra-low abundance microbes. We evaluated the impact of sequencing depth on assembling low and ultra-low abundance microbes on ZYMO, where we collected the Illumina short-reads (9.7Gb), and two datasets of ONT long-reads, with sequencing sizes of 16.5Gb and 153.7Gb, respectively. Among the long-read assemblers, an increase of sequencing depth cannot improve the assembly performance of low and ultra-low abundance microbes for wtdbg2, NECAT and Canu. These three assemblers failed to assemble any low- and ultra-low abundance microbes for both ONT datasets (Supplementary Table S5). Shasta, metaFlye and Lathe obtained better assembly results on the dataset with higher sequencing depth (Supplementary Table S5). The results showed the NGA50 of the low-abundance microbe *Saccharomyces cerevisiae* assembled using metaFlye was significantly improved (5.36 times) on the ONT dataset with higher sequencing depth. MetaFlye also assembled two more ultra-low abundance microbes (*Escherichia coli* and *Salmonella enterica*) on the dataset with higher sequencing depth. Among the hybrid assemblers, DBG2OLC, metaFlye-subassemblies and hybridSPAdes could assemble two more ultra-low abundance microbes on the dataset with deeper sequencing depth, and these assemblers also improved the NGA50 of the low-abundance microbe *Saccharomyces cerevisiae* (Supplementary Table S5).

PacBio HiFi technology has shown great success in reconstructing genomes from microbial communities [59]. We evaluated the long-read assemblers on PacBio HiFi dataset (merged from SRR9202034 and SRR932898) from ATCC20. Similar to our findings for PacBio CLR, we observed that metaFlye, Lathe, and Canu generated substantially higher *GF*s than Shasta, wtdbg2 and MECAT2 (8.98 times on average, Supplementary Table S1). Lathe and metaFlye obtained significantly higher N50s (2.50 times on average), NA50s (2.06 times on average), and normalized NGA50s (1.37 times on average) than Canu (Supplementary Table S1; Supplementary Figure S18). We found that the long-read assemblers on PacBio HiFi and CLR sequencing platforms produced comparable *GF*s, N50s, NA50s and normalized NGA50s (Supplementary Table S1). The misassemblies generated by Canu on PacBio CLR reads were substantially higher (2.17 times on average) than on PacBio HiFi reads (Supplementary Table S1).

CONCLUSION

In this study, we conducted a thorough benchmarking of 19 mainstream metagenomic assembly software tools using 32 metagenomic sequencing datasets generated by a range of sequencing technologies – short-read, linked-read and long-read sequencing. We present the results of our benchmark study as practical guidelines to assist end-users in selecting the best sequencing and assembly strategy for their purposes. We believe that our findings will be invaluable to the microbiome research community and may lead to improved results in future genome-based microbiome studies.

METHODS

Simulation of 10x linked-reads and ONT long-reads for CAMI datasets

We simulated 10x linked-reads and ONT long-reads using LRTK-SIM (git version d0a87c6) [60] and CAMISIM (v1.2-beta) [61] given the taxonomic composition in CAMI datasets. The simulated total number of nucleotides was the same as those of the available four short-read CAMI datasets.

Sample preparation and sequencing of S1 and S2 samples

The S1 and S2 datasets were obtained from two subjects with a typical Chinese diet and who had not taken any antibiotics, probiotics or prebiotics in the 3 months prior to the sample collection. Their stool samples were collected, aliquoted and stored at −80°C until analysis. Total microbial DNA was extracted using the QIAamp DNA stool mini kit (Qiagen, Valencia, CA, USA) according to the manufacturer's protocol. For short-read sequencing, fecal microbial DNA of the two subjects was sequenced by paired-end sequencing with a coverage of more than 30 million reads per sample using Illumina HiSeq 2500 (Illumina, CA, USA) and BGISEQ-500 (BGI, ShenZhen, China), respectively. For linked-read sequencing, we followed the strategies described by Bishara *et al.* [15] for library preparation on a 10x Chromium System (10x Genomics, CA, USA) and performed sequencing on an Illumina HiSeq 2500 (Illumina, CA, USA). The linked-reads were demultiplexed and barcodes extracted by Long Ranger (v2.2.1) [62]. For long-read sequencing, the SMRTbell libraries were prepared with the 20-kb Template Preparation using BluePippinTM Size selection system (15-kb size cutoff) protocol and were then sequenced in SMRT cells (Pacific Biosciences, CA, USA) with magnetic bead loading and P4-C2 or P6-C4 chemistry.

Metagenome assembly

We used MEGAHIT (v1.2.9)[11], IDBA-UD (v1.1.3)[7] and metaSPAdes (v3.15.0)[8] for short-read assembly; cloudSPAdes (v3.12.0-dev)[14] and Athena (v1.3)[15] for linked-read assembly; Shasta (v0.7.0)[26], wtdbg2 (v2.5)[27], MECAT2 (v20190314)[24], NECAT (v0.01)[25], metaFlye (v2.8.3)[18], Canu (v2.1.1)[20] and Lathe (git version 9cef07a)[23] for long-read assembly and DBG2OLC (git version 9514828)[28], and metaFlye (v2.8.3)[18], hybridSPAdes (v3.15.0) [29], Unicycler (v0.5.0) [30], MaSuRCA (v4.0.9) [31], OPERA-LG (v2.0.6)[32] and OPERA-MS (v0.8.3)[33] for hybrid assembly. To enable the hybrid mode of metaFlye (metaFlye-subassemblies), we combined the contigs assembled from short-reads (metaSPAdes) and long-reads (metaFlye) using the '–subassemblies' option. We used '–pacbio-hifi' for metaFlye HiFi assembly [59]. Most of the assembly tools were run using default parameters, but we adjusted the parameters in the following cases to avoid out-of-memory issues: (1) metaFlye on $CAMI_H$ was run with 'flye –asm-coverage 50'; and (2) NECAT on $CAMI_H$ was run with 'MIN_READ_LENGTH = 8000'.

The run time (user, system and real time) and maximum peak memory (RSS) usage of the software assembly tools were determined using the Linux command '/usr/bin/time -v.' All of the assembly tools were run on Linux machines with a dual 64-core AMD EPYC 7742 2.25GHz base clock speed 256MB L3 cache CPU with 1 TB memory.

Contig statistics

We generated AL, contig N50 by QUAST (v5.0.2) [63] from the assemblies after removing the contigs shorter than 1 kb. We enabled the MetaQUAST mode [64] to obtain the overall contig NA50, the NGA50 for each microbe, and the number of mismatches and misassemblies on the datasets for which reference genomes were available. We used the parameter '–fragmented –min-alignment 500 –unique-mapping' in MetaQUAST to disable ambiguous alignments. The NGA50 cannot be compared between microbes since microbes with larger genome sizes would tend to have higher NGA50s. In order to calculate the overall contig contiguity considering all the microbes in the dataset, and inspired by the definition of MAG contiguity in the previous study [23], we eliminated the impact of genome sizes on NGA50 by using NGA50/genome size, and averaged NGA50/genome size across all of the microbes in the dataset. This statistic was operationalized as normalized NGA50.

Contig binning and MAG qualities

To prepare input for MetaBat2 [65] contig binning, we respectively used BWA (v0.7.17) [66] and minimap2 (v2.17) [67] to align short-reads/linked-reads and long-reads to the contigs. For minimap2 we respectively used the parameters '-ax map-pb' and '-ax map-ont' to align PacBio CLR and ONT reads. For hybrid assembly, the short-read alignment was adopted as the input of MetaBat2. The alignment file was sorted by coordinates using SAMtools (v1.9) [68], and the contig coverage was extracted by the 'jgi_summarize_bam_contig_depths' program in MetaBat2. MetaBat2 (v2.12.1)[65] was used to group the contigs into MAGs using both contig coverage and sequence characteristics. We calculated the N50 for each MAG using QUAST (v5.0.2) [63]. The N50s of MAGs were compared between different assemblers, using the non-parametric Wilcoxon Rank Sum test, implemented in the compare_means function of the R package ggpubr (v0.5.0), with parameters 'paired = F' and 'alternative = less' (so that the alternative hypothesis was group 1 was larger than group 2). The single-copy gene completeness and contamination of each MAG were identified using CheckM (v1.1.2) [69]. The transfer RNAs (tRNAs) and ribosomal RNAs (5S, 16S and 23S rRNAs) were detected by ARAGORN (v1.2.38) [70] and barrnap (v0.9) [71], respectively. MAGs were defined as high-quality (completeness > 90%, contamination < 5%), medium-quality (completeness ≥ 50%, contamination < 10%) or low-quality (otherwise). Near-complete MAGs were those high-quality MAGs with 5S, 16S and 23S rRNAs, and at least 18 tRNAs [72].

MAG taxonomic classification

We removed poorly assembled MAGs with contig N50s < 50 kb, completeness < 75% or contamination > 25%, and annotated the contigs in MAGs with Kraken2 (v2.1.2) [73] using its standard database. To determine the dominant microbes identified for each MAG, we used 'assign_species.py' from 'metagenomics_workflows' [74], which has been adopted in previous studies [15, 23]. We used dRep [75] to remove redundant MAGs from the same microbial cluster.

LIST OF TERMS

MAG: the metagenome-assembled genomes; the bins generated by contig binning tools after metagenome assembly
AL: the total assembly length
GF: the genome fraction
#LQ: the number of low-quality MAGs
#MQ: the number of medium-quality MAGs
#HQ: the number of high-quality MAGs
#NC: the number of near-complete MAGs

N50: the contig length that contigs with higher or equal lengths produce more than 50% of the total assembly length
NA50: the N50 of contigs after breaking the contigs at misassemblies
NGA50: the contig length that contigs (after breaking the contigs at misassemblies) with higher or equal lengths produce more than 50% of the genome size

Key Points

- We comprehensively evaluated 19 genome assemblers on 32 metagenomic sequencing datasets generated from simulation, mock and real microbial communities, and provided detailed pros and cons of these assembly tools for each mainstream sequencing technology.
- Linked-read assemblers obtained the highest number of overall near-complete MAGs from the human gut microbiomes.
- We observed long-read assemblers achieved high contig contiguity but failed to reveal a considerable fraction of total assembly length and many medium- and high-quality MAGs.
- Hybrid assemblers using short- and long-reads were promising tools to both improve total assembly length and the number of near-complete MAGs.

SUPPLEMENTARY DATA

Supplementary data are available online at http://bib.oxfordjournals.org/.

ACKNOWLEDGMENTS

The authors thank the Research Grants Council of Hong Kong, Hong Kong Baptist University, HKBU Research Committee and Shenzhen Science and Technology Innovation Commission for their kind support of this project.

FUNDING

This research was partially supported by the Guangdong Basic and Applied Basic Research Foundation (No. 2019A1515011046 and No. 2021A1515012226); the Hong Kong Research Grant Council Early Career Scheme (HKBU 22201419), HKBU IRCMS (No. IRCMS/19-20/D02) and a grant from Shenzhen Science and Technology Innovation Commission (SZSTI) – Shenzhen Virtual University Park (SZVUP) Special Fund Project (No. 2021Szvup135). This project is also supported by open project of BGI-Shenzhen, Shenzhen 518000, China. The design of the study and collection, analysis and interpretation of data were partially supported by the Science Technology and Innovation Committee of Shenzhen Municipality, China (SGDX20190919142801722).

DATA AVAILABILITY

The CAMI short-reads were downloaded from '1st CAMI Challenge Dataset 1 CAMI_low', '1st CAMI Challenge Dataset 2 CAMI_medium' and '1st CAMI Challenge Dataset 3 CAMI_high' at https://data.cami-challenge.org/participate. Illumina short-reads, 10x linked-reads and PacBio CLR long-reads of ATCC20 are available under the Sequence Read Archive (SRA) accession codes SRR8359173, SRR12283286 and SRR12371719, respectively. We also used the PacBio HiFi long-reads of ATCC20 from SRR9202034 and SRR9328980. Illumina HiSeq short-reads, ONT GridION and the ONT PromethION long-reads of ZYMO were collected from ERR2935805, ERR3152366 and ERR3152367, respectively. Illumina short-reads (P1: SRR6788327, SRR6807561; P2: SRR6788328, SRR6807555), 10x linked-reads (P1: SRR6760786; P2: SRR6760782) and the ONT long-reads (P1: SRR8427258; P2: SRR8427257) of P1 and P2 were also downloaded from the SRA. Illumina short-reads, 10x linked-reads and long-reads of S1 and S2 are available in the SRA under project PRJNA841170. The source code of this study is available at https://github.com/ericcombiolab/Benchmark-metagenome-assemblers.

AUTHORS' CONTRIBUTIONS STATEMENT

L.Z. conceived the study; Z.M.Z. conducted the experiments and analyzed the results; Z.M.Z. and L.Z. wrote the manuscript; C.Y. and W.P.V. reviewed and revised the manuscript; X.D.F. helped with the library preparation and sequencing of S1 and S2. All of the authors have read and approved the final manuscript.

References

1. Yang C, Chowdhury D, Zhang Z, *et al.* A review of computational tools for generating metagenome-assembled genomes from metagenomic sequencing data. *Comput Struct Biotechnol J* 2021; **19**:6301–14.
2. Ghurye JS, Cepeda-Espinoza V, Pop M. Focus: microbiome: metagenomic assembly: overview, challenges and applications. *Yale J Biol Med* 2016; **89**(3): 353.
3. Ayling M, Clark MD, Leggett RM. New approaches for metagenome assembly with short reads. *Brief Bioinform* 2020; **21**(2): 584–94.
4. Simon C, Daniel R. Metagenomic analyses: past and future trends. *Appl Environ Microbiol* 2011; **77**(4): 1153–61.
5. Bharti R, Grimm DG. Current challenges and best-practice protocols for microbiome analysis. *Brief Bioinform* 2021; **22**(1): 178–93.
6. Berg G, Rybakova D, Fischer D, *et al.* Microbiome definition revisited: old concepts and new challenges. *Microbiome* 2020; **8**(1): 1–22.
7. Peng Y, Leung HCM, Yiu S-M, *et al.* IDBA-UD: a de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth. *Bioinformatics* 2012; **28**(11): 1420–8.
8. Nurk S, Meleshko D, Korobeynikov A, *et al.* metaSPAdes: a new versatile metagenomic assembler. *Genome Res* 2017; **27**(5): 824–34.
9. Bankevich A, Nurk S, Antipov D, *et al.* SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. *J Comput Biol* 2012; **19**(5): 455–77.
10. Prjibelski AD, Vasilinetc I, Bankevich A, *et al.* ExSPAnder: a universal repeat resolver for DNA fragment assembly. *Bioinformatics* 2014; **30**(12): i293–301.
11. Li D, Liu C-M, Luo R, *et al.* MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct *de Bruijn* graph. *Bioinformatics* 2015; **31**(10): 1674–6.
12. Zlitni S, Bishara A, Moss EL, *et al.* Strain-resolved microbiome sequencing reveals mobile elements that drive bacterial competition on a clinical timescale. *Genome Med* 2020; **12**(1): 1–17.
13. Roodgar M, Good BH, Garud NR, *et al.* Longitudinal linked-read sequencing reveals ecological and evolutionary responses of a

human gut microbiome during antibiotic treatment. *Genome Res* 2021; **31**(8): 1433–46.
14. Tolstoganov I, Bankevich A, Chen Z, *et al.* cloudSPAdes: assembly of synthetic long reads using de Bruijn graphs. *Bioinformatics* 2019; **35**(14): i61–70.
15. Bishara A, Moss EL, Kolmogorov M, *et al.* High-quality genome sequences of uncultured microbes by assembly of read clouds. *Nat Biotechnol* 2018; **36**(11): 1067–75.
16. Tsai Y-C, Conlan S, Deming C, *et al.* Resolving the complexity of human skin metagenomes using single-molecule sequencing. *MBio* 2016; **7**(1): e01948–15.
17. Koren S, Harhay GP, Smith TPL, *et al.* Reducing assembly complexity of microbial genomes with single-molecule sequencing. *Genome Biol* 2013; **14**(9): 1–16.
18. Kolmogorov M, Bickhart DM, Behsaz B, *et al.* metaFlye: scalable long-read metagenome assembly using repeat graphs. *Nat Methods* 2020; **17**(11): 1103–10.
19. Kolmogorov M, Yuan J, Lin Y, *et al.* Assembly of long, error-prone reads using repeat graphs. *Nat Biotechnol* 2019; **37**(5): 540–6.
20. Koren S, Walenz BP, Berlin K, *et al.* Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation. *Genome Res* 2017; **27**(5): 722–36.
21. Myers EW, Sutton GG, Delcher AL, *et al.* A whole-genome assembly of drosophila. *Science* 2000; **287**(5461): 2196–204.
22. Miller JR, Delcher AL, Koren S, *et al.* Aggressive assembly of pyrosequencing reads with mates. *Bioinformatics* 2008; **24**(24): 2818–24.
23. Moss EL, Maghini DG, Bhatt AS. Complete, closed bacterial genomes from microbiomes using nanopore sequencing. *Nat Biotechnol* 2020; **38**(6): 701–7.
24. Xiao C-L, Chen Y, Xie S-Q, *et al.* MECAT: fast mapping, error correction, and *de novo* assembly for single-molecule sequencing reads. *Nat Methods* 2017; **14**(11): 1072–4.
25. Chen Y, Nie F, Xie S-Q, *et al.* Efficient assembly of nanopore reads via highly accurate and intact error correction. *Nat Commun* 2021; **12**(1): 1–10.
26. Shafin K, Pesout T, Lorig-Roach R, *et al.* Nanopore sequencing and the shasta toolkit enable efficient de novo assembly of eleven human genomes. *Nat Biotechnol* 2020; **38**(9): 1044–53.
27. Ruan J, Li H. Fast and accurate long-read assembly with wtdbg2. *Nat Methods* 2020; **17**(2): 155–8.
28. Ye C, Hill CM, Shigang W, *et al.* DBG2OLC: efficient assembly of large genomes using long erroneous reads of the third generation sequencing technologies. *Sci Rep* 2016; **6**(1): 1–9.
29. Antipov D, Korobeynikov A, McLean JS, *et al.* hybridSPAdes: an algorithm for hybrid assembly of short and long reads. *Bioinformatics* 2016; **32**(7): 1009–15.
30. Wick RR, Judd LM, Gorrie CL, *et al.* Unicycler: resolving bacterial genome assemblies from short and long sequencing reads. *PLoS Comput Biol* 2017; **13**(6): e1005595.
31. Zimin AV, Puiu D, Luo M-C, *et al.* Hybrid assembly of the large and highly repetitive genome of *Aegilops tauschii*, a progenitor of bread wheat, with the MaSuRCA mega-reads algorithm. *Genome Res* 2017; **27**(5): 787–92.
32. Gao S, Bertrand D, Chia BKH, *et al.* OPERA-LG: efficient and exact scaffolding of large, repeat-rich eukaryotic genomes with performance guarantees. *Genome Biol* 2016; **17**(1): 1–16.
33. Bertrand D, Shaw J, Kalathiyappan M, *et al.* Hybrid metagenomic assembly enables high-resolution analysis of resistance determinants and mobile elements in human microbiomes. *Nat Biotechnol* 2019; **37**(8): 937–44.
34. Sczyrba A, Hofmann P, Belmann P, *et al.* Critical assessment of metagenome interpretation—a benchmark of metagenomics software. *Nat Methods* 2017; **14**(11): 1063–71.
35. Latorre-Pérez A, Villalba-Bermell P, Pascual J, *et al.* Assembly methods for nanopore-based metagenomic sequencing: a comparative study. *Sci Rep* 2020; **10**(1): 1–14.
36. Meyer F, Lesker T-R, Koslicki D, *et al.* Tutorial: assessing metagenomics software with the CAMI benchmarking toolkit. *Nat Protoc* 2021; **16**(4): 1785–801.
37. Zhang L, Fang X, Liao H, *et al.* A comprehensive investigation of metagenome assembly by linked-read sequencing. *Microbiome* 2020; **8**(1): 1–11.
38. Hon T, Mars K, Young G, *et al.* Highly accurate long-read hifi sequencing data for five complex genomes. *Scientific Data* 2020; **7**(1): 1–11.
39. Nicholls SM, Quick JC, Tang S, *et al.* Ultra-deep, long-read nanopore sequencing of mock microbial community standards. *Gigascience* 2019; **8**(5): giz043.
40. Martin M. Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet J* 2011; **17**(1): 10–2.
41. Parks DH, Rinke C, Chuvochina M, *et al.* Recovery of nearly 8,000 metagenome-assembled genomes substantially expands the tree of life. *Nat Microbiol* 2017; **2**(11): 1533–42.
42. Almeida A, Mitchell AL, Boland M, *et al.* A new genomic blueprint of the human gut microbiota. *Nature* 2019; **568**(7753): 499–504.
43. Almeida A, Nayfach S, Boland M, *et al.* A unified catalog of 204,938 reference genomes from the human gut microbiome. *Nat Biotechnol* 2021; **39**(1): 105–14.
44. Chen S, Zhou Y, Chen Y, *et al.* fastp: an ultra-fast all-in-one FASTQ preprocessor. *Bioinformatics* 2018; **34**(17): i884–90.
45. De Coster W, D'hert S, Schultz DT, *et al.* NanoPack: visualizing and processing long-read sequencing data. *Bioinformatics* 2018; **34**(15): 2666–9.
46. Huang N, Nie F, Ni P, *et al.* BlockPolish: accurate polishing of long-read assembly via block divide-and-conquer. *Brief Bioinform* 2022; **23**(1): bbab405.
47. Jiang H, Fan J, Sun Z, *et al.* NextPolish: a fast and efficient genome polishing tool for long-read assembly. *Bioinformatics* 2020; **36**(7): 2253–5.
48. Walker BJ, Abeel T, Shea T, *et al.* Pilon: an integrated tool for comprehensive microbial variant detection and genome assembly improvement. *PloS One* 2014; **9**(11): e112963.
49. Vaser R, Sović I, Nagarajan N, *et al.* Fast and accurate de novo genome assembly from long uncorrected reads. *Genome Res* 2017; **27**(5): 737–46.
50. Wick RR, Holt KE. Polypolish: short-read polishing of long-read bacterial genome assemblies. *PLoS Comput Biol* 2022; **18**(1): e1009802.
51. Zimin AV, Salzberg SL. The genome polishing tool polca makes fast and accurate corrections in genome assemblies. *PLoS Comput Biol* 2020; **16**(6): e1007981.
52. Huang N, Nie F, Ni P, *et al.* NeuralPolish: a novel nanopore polishing method based on alignment matrix construction and orthogonal Bi-GRU networks. *Bioinformatics* 2021; **37**(19): 3120–7.
53. Huang Y-T, Liu P-Y, Shih P-W. Homopolish: a method for the removal of systematic errors in nanopore sequencing by homologous polishing. *Genome Biol* 2021; **22**(1): 1–17.
54. Shafin K, Pesout T, Chang P-C, *et al.* Haplotype-aware variant calling with PEPPER-Margin-DeepVariant enables high accuracy in nanopore long-reads. *Nat Methods* 2021; **18**(11): 1322–32.
55. Warren RL, Coombe L, Mohamadi H, *et al.* ntEdit: scalable genome sequence polishing. *Bioinformatics* 2019; **35**(21): 4430–2.

56. Zhang X, Liu C-G, Yang S-H, *et al.* Benchmarking of long-read sequencing, assemblers and polishers for yeast genome. *Brief Bioinform* 2022; **23**(3): bbac146.
57. Cleary B, Brito IL, Huang K, *et al.* Detection of low-abundance bacterial strains in metagenomic datasets by eigengenome partitioning. *Nat Biotechnol* 2015; **33**(10): 1053–60.
58. Luo Y, Yu YW, Zeng J, *et al.* Metagenomic binning through low-density hashing. *Bioinformatics* 2019; **35**(2): 219–26.
59. Bickhart DM, Kolmogorov M, Tseng E, *et al.* Generating lineage-resolved, complete metagenome-assembled genomes from complex microbial communities. *Nat Biotechnol* 2022; **40**(5): 711–9.
60. Zhang L, Zhou X, Weng Z, *et al.* Assessment of human diploid genome assembly with 10x linked-reads data. *Gigascience* 2019; **8**(11): giz141.
61. Fritz A, Hofmann P, Majda S, *et al.* CAMISIM: simulating metagenomes and microbial communities. *Microbiome* 2019; **7**(1): 1–12.
62. Zheng GXY, Lau BT, Schnall-Levin M, *et al.* Haplotyping germline and cancer genomes with high-throughput linked-read sequencing. *Nat Biotechnol* 2016; **34**(3): 303–11.
63. Gurevich A, Saveliev V, Vyahhi N, *et al.* QUAST: quality assessment tool for genome assemblies. *Bioinformatics* 2013; **29**(8): 1072–5.
64. Mikheenko A, Saveliev V, Gurevich A. MetaQUAST: evaluation of metagenome assemblies. *Bioinformatics* 2016; **32**(7): 1088–90.
65. Kang DD, Li F, Kirton E, *et al.* MetaBAT 2: an adaptive binning algorithm for robust and efficient genome reconstruction from metagenome assemblies. *PeerJ* 2019; **7**:e7359.
66. Li H. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. *arXiv* 2013; 1303.3997.
67. Li H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* 2018; **34**(18): 3094–100.
68. Li H, Handsaker B, Wysoker A, *et al.* The sequence alignment/map format and SAMtools. *Bioinformatics* 2009; **25**(16): 2078–9.
69. Parks DH, Imelfort M, Skennerton CT, *et al.* CheckM: assessing the quality of microbial genomes recovered from isolates, single cells, and metagenomes. *Genome Res* 2015; **25**(7): 1043–55.
70. Laslett D, Canback B. ARAGORN, a program to detect tRNA genes and tmRNA genes in nucleotide sequences. *Nucleic Acids Res* 2004; **32**(1): 11–6.
71. Seemann T. Barrnap. https://github.com/tseemann/barrnap, 2018.
72. Bowers RM, Kyrpides NC, Stepanauskas R, *et al.* Minimum information about a single amplified genome (MISAG) and a metagenome-assembled genome (MIMAG) of bacteria and archaea. *Nat Biotechnol* 2017; **35**(8): 725–31.
73. Wood DE, Jennifer L, Langmead B. Improved metagenomic analysis with Kraken 2. *Genome Biol* 2019; **20**(1): 1–13.
74. Moss EL. https://github.com/elimoss/metagenomics_workflows, 2019.
75. Olm MR, Brown CT, Brooks B, *et al.* dRep: a tool for fast and accurate genomic comparisons that enables improved genome recovery from metagenomes through de-replication. *ISME J* 2017; **11**(12): 2864–8.