Efficient Algorithms and Pipelines for Microbiome Analysis

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von

M.Sc. Caner Bağcı

aus Konak / Türkei

Tübingen

2022

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	01.02.2023
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr. Daniel H. Huson
2. Berichterstatter:	Prof. Dr. Detlef Weigel

Abstract

Microbes are essential in our lives in many aspects. They inhabit our bodies often in a commensal relationship. We utilize them in many fields of biotechnology and pharmacy. They are also causative agents of many infections as pathogens. Until the development and widespread availability of next-generation sequencing technologies, it was possible to study them only under traditional culture conditions. However, even today, we can grow only a minority (reported as low as 1%) of them in cultures. In nature, they live in interaction with other microbial species, called the microbiome, as well as their host and other factors in the environment. Metagenomics is the study of microbial genomes as the total collection of their genetic material sequenced directly from a microbiome.

This dissertation presents several novel methodological advancements to the study of microbial genomes, with a focus on microbiomes. The emergence of long-read sequencing technologies has enabled researchers to study microbial genomes in more detail, faster, and more conveniently. The computational methods to analyze metagenomes were initially developed for short-read sequencing technologies. They needed adaptations before the long-read sequencing methods could be used for microbiome research.

The first challenge addressed here is the taxonomic and functional binning of long-reads from a long-read metagenomic sequencing dataset. A new lowest-common-ancestor (LCA) algorithm was developed in the Metagenome Analyzer (MEGAN) tool, such that it is capable of accurately binning long-read metagenomic datasets. In the second study presented, this algorithm was applied to the output of a pipeline that was developed to assemble metagenome-assembled-genomes (MAG) from environmental samples using only long-read sequencing. The systematic frameshift errors in long-read assemblies were also addressed to enable further downstream analysis on them, such as quality con-

trol of the assembled contigs and annotation. It was demonstrated that obtaining even closed, circular chromosomes as contigs from long-read only metagenomic sequencing of ordinarily complex environmental samples is possible.

In a separate study, a novel software and a set of algorithms called MAIRA were developed to help researchers in biological and clinical fields to analyze microbiomes in a time and cost-efficient manner. MAIRA is aimed to run a modern laptop without requiring access to compute servers and perform taxonomic and functional analysis of metagenomes in real-time, and in the field of sequencing. It takes the real-time basecalling and portability advantages of Nanopore sequencing, and couples them with efficient algorithms to report which species are present in an environmental sample and which antimicrobial resistance or virulence factor genes they carry.

Lastly, it was demonstrated how phylogenetic outlines can be applied to the phylogenetic context of microbes to evaluate their evolutionary relationships. Novel microbial genomes are efficiently compared against a database of publicly available microbial genomes to estimate their evolutionary distances to each other. These distances are then used to construct a phylogenetic outline, which is shown to be an alternative and appropriate approach to determining the phylogenetic context of novel microbial genomes.

Altogether, the methods presented here progress our capabilities in the computational analysis of microbial genomes. As the rapid developments in sequencing technologies continue, the demand for novel computational methods to analyze the ever-growing size of microbial datasets will continue to exist.

Kurzfassung

Mikroben sind in vielerlei Hinsicht von wesentlicher Bedeutung für unser Leben. Sie siedeln in unserem Körper, oft in einer Art des Kommensalismus. Wir nutzen sie in vielen Bereichen der Biotechnologie und Pharmazie. Aber als Krankheitserreger sind sie auch die Verursacher vieler Infektionen. Bis zur Entwicklung und breiten Verfügbarkeit von Next-Generation-Sequencing-Technologien war es nur möglich, sie unter traditionellen Kulturbedingungen zu untersuchen. Doch selbst heute kann nur ein geringer Anteil von ihnen in Kultur gezüchtet werden (Berichten zufolge bis zu 1%). In der Natur leben sie in Interaktion mit anderen mikrobiellen Spezies, dem so genannten Mikrobiom, sowie mit ihrem Wirt und weiteren Faktoren ihrer Umwelt. Unter Metagenomik versteht man die Untersuchung mikrobieller Genome als Gesamtheit ihres genetischen Materials, das direkt aus einem Mikrobiom sequenziert wird.

In dieser Dissertation werden mehrere neue methodische Fortschritte bei der Erforschung von mikrobiellen Genomen vorgestellt, wobei der Schwerpunkt auf Metagenomen liegt. Das Aufkommen von Long-Read-Sequenzierungstechnologien hat es Forschern ermöglicht, mikrobielle Genome detaillierter, schneller und bequemer zu untersuchen. wurden Ursprünglich wurden Berechnungsmethoden für die Metagenomanalyse für Short-Read-Sequenzierungstechnologien entwickelt. Sie mussten angepasst werden, bevor die Long-Read-Sequenzierungsmethoden für die Mikrobiomforschung eingesetzt werden konnten.

Die erste Herausforderung, mit der wir uns hier befassen, ist das taxonomische und funktionelle Binning von Longreads aus einem Long-Read Metagenom Sequenzierungsdatensatz. Ein neuer LCA-Algorithmus (Lowest Common-Ancestor) wurde im Metagenome Analyzer (MEGAN) entwickelt, der in der Lage ist, metagenomische Long-Reads

akkurater Bins zuzuordnen. In der zweiten vorgestellten Studie wurde dieser Algorithmus auf die Ergebnisse einer Pipeline angewandt, die entwickelt wurde, um Metagenom-assemblierte Genome (MAG) aus Umweltproben zu assemblieren, die ausschließlich unter Verwendung von Long-Read-Sequenzierung generiert wurden. Die systematischen Frameshift Fehler in Long-Read-Assemblies wurden ebenfalls berücksichtigt, um weitere nachgelagerte Analysen zu ermöglichen, wie z. B. die Qualitätskontrolle der assemblierten Contigs und die Annotation. Es wurde gezeigt, dass es möglich ist, selbst geschlossene, zirkuläre Chromosomen als Contigs aus der reinen Long-Read-Metagenomsequenzierung von normalerweise komplexen Umweltproben zu erhalten.

In einer separaten Studie wurden eine neuartige Software und eine Reihe von Algorithmen mit dem Namen MAIRA entwickelt, um Forschern in biologischen und klinischen Bereichen bei der zeit- und kosteneffizienten Analyse von Mikrobiomen zu helfen. MAIRA ist darauf ausgerichtet, auf einem modernen Laptop zu laufen, ohne dass ein Zugang zu Compute-Servern erforderlich ist, und taxonomische und funktionelle Analysen von Metagenomen in Echtzeit und im Bereich der Sequenzierung durchzuführen. MAIRA nutzt die Vorteile des Echtzeit-Basecalls und der Portabilität der Nanopore-Sequenzierung und verbindet sie mit effizienten Algorithmen, um zu ermitteln, welche bakteriellen Spezies in einer Umweltprobe vorhanden sind und welche Gene für antimikrobielle Resistenz oder Virulenzfaktoren diesen zugeordnet werden können.

Schließlich wurde gezeigt, wie phylogenetische Outlines auf den phylogenetischen Kontext von Mikroben angewendet werden können, um ihre evolutionären Beziehungen zu bewerten. Neuartige mikrobielle Genome werden hierbei effizient mit einer Datenbank öffentlich zugänglicher mikrobieller Genome verglichen, um ihre evolutionären Distanzen zueinander abzuschätzen. Diese Distanzen werden dann verwendet, um ein phylogenetisches Outline zu erstellen, dass sich als alternativer und gut geeigneter Ansatz zur Bestimmung des phylogenetischen Kontextes neuartiger mikrobieller Genome erweist.

Insgesamt verbessern die hier vorgestellten Methoden unsere Fähigkeiten bei der computergestützten Analyse mikrobieller Genome. Da die rasante Entwicklung der Sequenzierungstechnologien anhält, wird der Bedarf an neuartigen Berechnungsmethoden zur Analyse der immer größer werdenden mikrobiellen Datensätze weiter bestehen.

Acknowledgments

Firstly, I would like to deeply thank my supervisor Prof. Dr. Daniel H. Huson for allowing me to do this research in his lab, for his patience, constant support, kindness, providing a nice working environment, and guiding me in my pursuit of becoming a scientist. I am very grateful to have done my doctoral studies under his supervision. I would also like to thank my TAC members, Prof. Dr. Detlef Weigel and Prof. Dr. Ruth Ley, for their guidance, the discussions we have had, and their time.

I am thankful to all of the past and present members of Algorithms in Bioinformatics that I have worked together with. I would like to especially thank Dr. Benjamin Albrecht, who I worked worked together with on many of the projects presented in this dissertation, for being the best office-mate, for the hours-long fruitful discussions that we enjoyed, and for his help when I was lost in my work. I am also thankful to all of the students that I have worked together with, or those who attended my tutorials. I always enjoyed supervising or teaching part of my studies, thanks to you guys.

I would like to thank all of my collaborators, who I enjoyed working together with and exchanging ideas during my studies: Prof. Dr. Peter Lockhart, Dr. Rohan Williams, Prof. Dr. Lars Angenent, Prof. David Bryant, Prof. Dr. Laura Weyrich, Dr. Sofia Esquivel Elizondo, and Dr. Raphael Eisenhofer. I am especially grateful to Pete (and Trish, of course) for hosting me in Palmerston North during the crazy times of the COVID outbreak and for all the scientific and or non-scientific discussions that I enjoyed having with him.

This thesis would probably not be possible without the support of my friends here. I thank you all: Achim, Ali, Ania, Bilge, Dennis, Direnç, Efe B., Efe K., Elif, Emre, Ezgi,

Gözde, Jonas, Mehmet, Mete, Michael, Murat, Nils, Sascha, Xixi, Yasemin.

Last but not least, I would like to express my gratitude to my parents, Cengiz and Hediye, and my sister, Damla, for their genuine love and support throughtout my life.

Contents

Acknowledgments **vii**

Background **1**

1 Taxonomic and Functional Characterization of Microbes 1

2 Computational analysis of metagenomes 5

3 Long-read sequencing 9

Publications **11**

Objectives and Contributions **15**

Conclusions **17**

1 Taxonomic Binning of Long-Reads 17

2 Assembly and Downstream Analysis of Microbiomes 26

3 Accurate and Real-Time Analysis of Microbes 30

4 Phylogenetic context 36

5 Outlook 39

Bibliography **41**

Appendices **51**

Appendix I 52

Appendix II 70

Appendix III 84

Appendix IV 97

Appendix V 108

Background

1 Taxonomic and Functional Characterization of Microbes

Microbes play an essential role in many aspects of life, from medicine to biotechnology. They are the causative agents of a range of infections in humans. They inhabit many sites of our bodies, and every food we consume, often living unaware of their host and not intending to cause harm. Many studies have shown correlations between certain compositions of microbes (microbiota) and the state of their host in the past few decades (Schwabe and Jobin, 2013; Vandenkoornhuyse *et al.*, 2015; Blaser *et al.*, 2014). In humans, they are known to affect weight, the immune system, or can cause pathogenicity (Blaser *et al.*, 2014). In agriculture, they can affect the growth of the plants (Vandenkoornhuyse *et al.*, 2015), as well as the metabolism, and thus levels of carbon-emission of cattle (Mizrahi and Jami, 2018). They are utilized in many areas of biotechnology for the diverse metabolic capabilities of them. They are used in wastewater treatment plants (Wagner *et al.*, 2002), in fertilization of food products, in producing pharmaceutical products (Shen *et al.*, 2001). They are also the source of most antibiotics in use today in order to, paradoxically, treat diseases caused by other harmful microbes (Hutchings *et al.*, 2019).

Until the emergence and the widespread use of DNA-sequencing methods became common, microbes were studied in pure cultures. A vast majority of the microbes can still not be cultured, making them near-impossible to study with the traditional methods (Riesenfeld *et al.*, 2004). In addition to them not being possible to apply to many mi-

crobes, the traditional culture-based methods are also time-consuming. In the case of an infection caused by a microbial pathogen, the time needed to isolate and grow the microbes in culture is measured in terms of days, even up to 45 days in some cases (Lagier *et al.*, 2015). Further diagnostics, such as the characterization of the antimicrobial resistance profile of the isolated microbe, require even more work and time, as well as *a priori* knowledge of the microbe in question. The isolation and characterization of microbes from more complex or less widely studied sources for applications in non-medical fields is yet more challenging or, for some, not possible (Rinke *et al.*, 2013). This has led to most microbial diversity not being studied and understood in environments such as soil (Schmeisser *et al.*, 2007).

Next-generation sequencing technologies started to appear and become widely used at the beginning of the century. They allowed the researchers to characterize the genetic materials of organisms in an accelerated and high-throughput manner, where many fragments of nucleic acids are sequenced simultaneously and rapidly (Mardis, 2008). This has also led to a shift in the study of microbes, as in other biological and medical disciplines. It has become possible to sequence the genetic material of complex communities of microbes (e.g. the human gut) as a whole (Qin *et al.*, 2010), as well as to sequence genomes of organisms from diverse environments consisting of microbes that have not been discovered before (e.g. from groundwater) (Tyson *et al.*, 2004).

The complex communities of microbes (microbiome) are studied to understand both the taxonomic composition of organisms present in environmental samples and also their functional capabilities (Simon and Daniel, 2011) The taxonomic profiling of microbiomes is performed using both targeted and untargeted sequencing strategies. In targeted sequencing methods, conserved regions of the genomes of microbes are selectively amplified before the sequencing. The analysis is focused on characterizing the taxonomic profile of the sample either by using previous knowledge of the targeted sequence or by *de novo* clustering of the sequence data. 16S ribosomal RNA (rRNA) gene is the most commonly chosen marker in bacteria and archaea for targeted analysis of microbiomes, as it contains highly conserved regions, as well as regions that are relatively more variable (Ranjan *et al.*, 2016). On the other hand, the untargeted sequencing methods rely on information gained from the complete genetic material present in a sample (metagenome) (Thomas *et al.*, 2012).

Metagenomics, the study of the metagenome, can be used to resolve both the taxonomic profile of environmental samples, and the functions the organisms present in the samples can carry out (Simon and Daniel, 2011). Metagenomics provides a deeper understanding of the microbiomes than the targeted sequencing methods. It allows a more resolved taxonomic profiling, overcoming the limitation of 16S rRNA gene-based taxonomic analysis being bounded by genus-level assignments, for instance (Poretsky *et al.*, 2014). Additionally, it provides information on the gene content of the genomes sequences, such as antimicrobial resistance genes, or virulence factors (Dos Santos *et al.*, 2017; Padmanabhan *et al.*, 2013).

Numerous computational methods have been developed to uncover both the taxonomic composition and the functional capabilities of microbes from metagenomic datasets (Breitwieser *et al.*, 2019). These can, first, be divided into two categories based on the type of the input sequence data that they can work with; namely those which work directly with the raw sequencing reads (read-based methods) (Simon *et al.*, 2019), and those which work with contigs assembled from the raw sequencing reads (assembly-based methods) (Sczyrba *et al.*, 2017). Further, they can be classified into different categories, based on their computational approach. There exist alignment-based methods, which attempt at assigning sequencing reads or assembled contigs to a taxonomic (and functional) class based on their alignments to databases of known sequences (Breitwieser *et al.*, 2019). The employed databases can differ in the type of sequences they contain (DNA or protein); and the information that they contain (whole genomes or proteomes, or selected marker sequences from those) (Breitwieser *et al.*, 2019). Alignment-free approaches have also been developed, providing rapid insights into the sample, such as using k-mers, or maximum exact matches (Ren *et al.*, 2018).

Although there have been an extensive collection of computational methods developed to analyse the metagenomes sequenced by the second-generation NGS technologies, the question of pairing the taxonomic and functional information gained from the sequence data has been challenging to overcome (Beier *et al.*, 2017). This has remained a limitation primarily due to the short lengths of the raw sequencing reads obtained from the second-generation sequencing technologies, such as Illumina, which ranges from 100-300 base pairs. Read-based methods, as mentioned above, can often only give insights into either the taxonomic or the functional composition of metagenomes, lacking

the connection between the two. Specifically, one can answer the question of what microbes are present in the sample, and what functions they are capable of; however it remains questionable which microbes can carry out which functions. To some degree, assembly-based methods can attempt to overcome this problem, by first assembling short sequence reads into longer contigs, which are more informative at providing answers to both the taxonomy and the gene content of the organisms present in the sample (Prakash and Taylor, 2012). However, assembling short-reads and binning the resulting contigs from metagenomes remains a challenging problem, as metagenome assemblies generated from short-read sequencing are often very fragmented (Sczyrba *et al.*, 2017).

Establishing the connection between the taxonomic and functional classification of microbiome samples is especially important in specific scenarios, such as in the surveillance of pathogens (Afshinnekoo *et al.*, 2017). In order to specifically characterize the pathogenic microbes, it is often required to know the composition and the organization of the genes in their genome (Boolchandani *et al.*, 2019). This includes the potential repertoire of all antimicrobial resistance (AMR) genes or traits they may carry, their virulence factors, and their organisation within the genome (as in pathogenicity islands) (Bashir *et al.*, 2016). The isolation and (patho)genomic characterization of microbial isolates in the case of infection is often time-consuming and challenging, requiring the samples to be analyzed instead by directly sequencing the environmental samples (Pendleton *et al.*, 2017). The challenges mentioned above, regarding the coupling taxonomic and functional analysis for metagenome datasets generated by second-generation sequencing technologies, made this approach practically not feasible. Furthermore, the high capital costs, the requirement of specifically trained personnel, and the time it takes from preparing the libraries to the data analysis have also contributed to the impracticality of using second-generation sequencing technologies in a medical setting to survey pathogens (Petersen *et al.*, 2019).

2 Computational analysis of metagenomes

Alignment-based approaches

Metagenomic classification methods that utilize alignments of reads or assembled contigs to a database of known sequences are generally considered more accurate, although they often require more resources and time (Sczyrba *et al.*, 2017). The alignment of the reads can be against either databases consisting of nucleotide sequences, such as genomes, or against databases that contain protein sequences (Breitwieser *et al.*, 2019). The alignment-based approaches are also more favourable when the functional classification of genes on the reads or contigs is desired. In the case of protein databases, sequencing reads or assembled contigs are translated into all six reading frames, and these are aligned against amino acid sequences (as in BLASTX) (Buchfink *et al.*, 2015). Protein sequences are evolutionary more conserved than nucleotides, thus making the classification more sensitive when closely related references of the sequenced organisms are missing in the database (Menzel *et al.*, 2016). Bacterial genomes can also undergo significant structural changes in a relatively short amount of evolutionary time, complicating the alignment of longer sequences to them (such as aligning assembled contigs or long-reads to a database of genomes) (Bernard *et al.*, 2016).

After the alignment of reads against a database of known sequences with taxonomic annotations, they can be assigned to a taxonomic unit in multiple ways. The possible approaches can be fundamentally divided into two categories, namely read-binning and taxonomic profiling (Breitwieser *et al.*, 2019). While the read-binning methods attempt to assign a taxonomic classification to each read individually, regardless of the taxonomic rank (some may be classified more specifically than others); taxonomic profiling methods aim to produce an overview of the relative abundances of all taxonomic units within the sample, often not specifically assign each read to a certain taxonomic unit (Segata *et al.*, 2012).

Lowest common ancestor (LCA), introduced by MEGAN (Huson *et al.*, 2007) to be used in binning of metagenomic reads, is one of the most widely used approaches to

assign a read (or contig) to a taxonomic unit after it was aligned to a database of known sequences. It aims at overcoming the limitation of the naïve best-hit approach by binning reads (or assembled contigs) to potentially less specific taxonomic units, which agree on all (or the majority) of the significant hits from the database. Each read is individually placed on the taxonomic unit, which is the lowest common ancestor of all taxa with significant hits to it. Many other read-binning methods developed later, also use the naïve or modified versions of the LCA algorithm, such as kraken (Wood and Salzberg, 2014), and kaiju (Menzel *et al.*, 2016).

Other methods, such as MetaPhlAn (Segata *et al.*, 2012), attempt to produce a taxonomic profile of the sample by aligning reads against databases of clade-specific marker genes. The taxonomic profiling methods differ from read-binning methods, as not every read is employed in the analysis, but rather a subset that aligns to the clade-specific marker genes (Sunagawa *et al.*, 2013).

Alignment-free approaches

Although alignment-based approaches provide more useful information in metagenomic analysis, they come at an increased cost of computational resources and time (Breitwieser *et al.*, 2019). The faster, yet still at a comparable accuracy, alignment-free metagenomic classification methods are a more popular choice when a rapid analysis is desired.

Kraken (Wood and Salzberg, 2014), being the first such tool, is built around the idea of extracting subsequences at a fixed length from both reads and reference sequences, called k-mers (of length k), and binning each read based on exact matches of k-mers from the reads to the database. The database contains k-mers extracted from the reference sequences, and they are mapped to the LCA of all references they are seen in. The classification is then based on a pruned subtree of taxa with matching k-mers to the read, and the read gets assigned to the leaf node in the pruned subtree with the highest summed weight from root to leaf (highest count of matching k-mers). The first implementation of Kraken used only genomic references; however, it has later been adopted to use protein references, as well, in Kraken2 (Wood *et al.*, 2019). Kraken2 has also made improve-

ments to the speed and accuracy of the classification by introducing minimers and spaced seeds.

On the other hand, Kaiju (Menzel *et al.*, 2016), aims to combine the strengths of both alignment-based and alignment-free methods. It aims at identifying maximal-exact-matches (MEM) between query sequences and a database of either protein or nucleotide sequences. Identifying MEMs using an FM-index is often the first step in read alignment (Liu and Schmidt, 2012). Kaiju, however, gains a major speed-up over alignment-based methods as it stops at this step and does not further extend the MEMs to gapped alignments.

Assembly-based approaches

Sequence assembly is the computational process of merging short sequencing reads based on the overlapping sequences among themselves in order to reconstruct longer, contiguous fragments (contig) of the original molecule that had given rise to said reads (Nagarajan and Pop, 2013). It can also be applied to metagenomic sequencing datasets to reconstruct genomes of organisms that are present in the sample. Sequence assembly itself is an algorithmically challenging problem due to the nature of genomes and the methods available to us to sequence them (Ghurye *et al.*, 2016). Repeated regions in the genomes, and the short length of sequencing reads that often cannot span them make it difficult to assemble even single genomes into full chromosomes. Metagenomes, consisting of a mixture of several genomes, suffer from this problem even further. On top of the repeated regions within a genome, there can also be regions that are completely or nearly identical in the genomes of different organisms. The unevenness of the coverage along a genome (both due to similar regions from different organisms but also to the uneven abundance of organisms in the actual sample) adds additional challenges to the metagenome assembly (Breitwieser *et al.*, 2019).

The product of a metagenome assembly, especially from short sequencing reads, is often a collection of contigs at varying lengths and numbers depending on the complexity of the environmental sample and the sequencing technology applied (Meyer *et al.*,

2022). The following challenge in the taxonomic and functional analysis of the sample is typically grouping the assembled contigs into different sets, each of which typically represents a genome (metagenome assembled genome - MAG) (Breitwieser *et al.*, 2019). Computational methods to bin assembled contigs into MAGs (genome binning), employ compositional features of the contigs (such as tetra-nucleotide frequencies, or GC content), differences in the coverages of the contigs (Kang *et al.*, 2019), and recently deep learning methods (Nissen *et al.*, 2021) or evaluating the raw assembly graphs that gave rise to these contigs (Mallawaarachchi *et al.*, 2020).

The assembly and the consecutive contig binning of metagenomic sequencing datasets are often desired, as it has the potential to convey more information than the raw sequence reads. The contigs can subsequently be decorated with functional features, such as coding sequences on them being called *de novo*, and annotated for the functions they carry (Thomas *et al.*, 2012). The enzymes that are coded by the assembled genomes, and the pathways that they take a role in can be studied in more detail. Phylogenetic relationships of the organisms both within the sample, and also with the publicly available genomes can also be established in a more robust way from the assembled contigs (Sunagawa *et al.*, 2013). Although the assembly and binning of metagenome derived from short-read sequencing technologies still remain a challenge, the introduction of long-read sequencing technologies has resolved some of the issues. Several studies have recently reported chromosome-level MAGs derived from long-read metagenomes (Arumugam *et al.*, 2021; Singleton *et al.*, 2021).

Functional analysis of metagenomes

Even though the taxonomic composition and the shifts in that are essential questions in microbiome studies, the functional capabilities of the organisms are even more important to understand what they are capable of doing and how they interact with the environment they live in and with each other. This becomes even more pronounced considering the fact that even two different strains of the same species can differ substantially in certain functions, such as some *Escherichia* strains being pathogenic to humans, whereas others having a commensal relationship with humans as their host (Bashir *et al.*, 2016). There-

fore, it is not always possible to answer what a microbe is capable of doing when only a taxonomic name can be assigned to it.

The simplest approach to studying the function of a microbiome is to predict it using the taxonomic information, as introduced in PICRUSt (Douglas *et al.*, 2018). PICRUSt was developed mainly for 16S rRNA amplicon sequencing of microbiomes, although they can also be used with shotgun metagenomics datasets. It predicts the functional potential of a microbiome based on the taxonomic profile assigned to it using any method. The prediction of the functional potential is then based on the known functional capabilities of the assigned taxa. Therefore it works reliably only when the taxonomic assignments are very specifically to the said strain from the database (Agrawal *et al.*, 2019), which is seldomly possible in microbiome studies, and even less so with amplicon sequencing as it was originally designed for.

Deeper analysis of function in metagenomes typically employs alignment and/or assignment of reads or assembled contigs against specialized databases, such as KEGG (Kanehisa and Goto, 2000) for pathway mapping of proteins, CARD (Alcock *et al.*, 2020) for antibiotic resistance genotypes, and VFDB (Chen *et al.*, 2005) for virulence factors. Both MEGAN (Beier *et al.*, 2017) and MG-RAST (Glass *et al.*, 2010) work in similar ways to assign function to metagenomic reads, by first aligning them to a comprehensive database and then assigning function to the reads that hit a protein with a functional annotation from one of the databases they employ.

3 Long-read sequencing

The introduction of single-molecule sequencing technologies (third-generation sequencing, long-read sequencing) by Pacific Biosciences (PacBio), and Oxford Nanopore Technologies (ONT) in 2015, has opened up new prospects in the field of microbiome studies since the second-generation sequencing technologies (Petersen *et al.*, 2019). The leading characteristic single-molecule sequencing technologies have presented is their capability to sequence fragments of DNA that are much longer than second-generation sequenc-

ing methods (Nicholls *et al.*, 2019). The length of the raw reads produced by them ranges from kilobases to millions of bases in just a single read (on average 15 kilobases with ONT today) (Nicholls *et al.*, 2019). They owe this capability to sequencing single molecules of DNA without undergoing an enzymatic synthesis, thus overcoming the enzymatic limitation of synthesis and being limited only to the length of the fragments in the prepared library (Wang *et al.*, 2021).

MinION^{TM}, developed by Oxford Nanopore Technologies, is one of the first sequencing devices to operate using single-molecule sequencing technology. It was announced in 2012, has been commercially available since 2014, and is still under development regarding both its chemical and computational components (Wang *et al.*, 2021). On top of being a single-molecule sequencing device, MinION offers additional features making it a more practical tool for medical, and in-field sequencing applications. It is a palm-sized, portable device weighing only 90 grams. It can be operated almost anywhere connected to a computer as a USB device. This portability overcomes some of the limitations of the second-generation sequencing technologies to be applied in a medical or in-field setting, such as requiring extensive laboratory infrastructure and capital costs (Pendleton *et al.*, 2017). All devices, including MinION, developed by Oxford Nanopore Technologies, have the added advantage that the sequencing data becomes available in ”real-time”, while the sequencing is going on. This enables researchers also analyze the data in real-time, hence removing the requirement to wait for the sequencing run and additional base-calling to finish to start the computational analysis of the data. It overcomes the next limitation of the second-generation sequencing technologies by removing the time needed between library preparation and data analysis (Parker *et al.*, 2017).

Despite all the advantages that have become available with the development of third-generation sequencers, at the time of their launch, they also posed many novel computational challenges as all of the existing algorithms and pipelines to analyze sequencing data have been initially developed for the second-generation sequencers (Magi *et al.*, 2018).

Publications

Peer reviewed publications included in this dissertation

1. Daniel H Huson, Benjamin Albrecht, Caner Bağcı, Irina Bessarab, Anna Gorska, Dino Jolic, and Rohan BH Williams. MEGAN-LR: new algorithms allow accurate binning and easy interactive exploration of metagenomic long reads and contigs. *Biology Direct*, 13(1):6, 2018.

2. Krithika Arumugam, Caner Bağcı, Irina Bessarab, Sina Beier, Benjamin Buchfink, Anna Gorska, Guanglei Qiu, Daniel H Huson, and Rohan BH Williams. Annotated bacterial chromosomes from frame-shift-corrected long-read metagenomic data. *Microbiome*, 7(1):61, 2019

3. Benjamin Albrecht*, Caner Bağcı* and Daniel H. Huson (2020). MAIRA – real-time taxonomic and functional analysis of long reads on a laptop. *BMC Bioinformatics*, 21.13 1-12, 2020

4. Caner Bağcı, David Bryant, Banu Cetinkaya, and Daniel H. Huson. Phylogenetic context using phylogenetic outlines. *Genome Biology and Evolution*, 13.9, 2021

5. Anupam Gautam, Hendrik Felderhoff, Caner Bağcı, and Daniel H. Huson. Using AnnoTree to get more assignments, faster, in DIAMOND+MEGAN microbiome analysis. *mSystems*, 7.1, 2022

* denotes equal contribution.

Peer reviewed papers and book chapters not included in this dissertation

- Caner Bağcı, Benjamin Albrecht, and Daniel H. Huson. MAIRA: protein-based analysis of MinION reads on a laptop. *Accepted for publication in "Metagenomic Data Analysis: Methods and Protocols" Book chapter by Humana Press*.

- Caner Bağcı*, Sascha Patz*, and Daniel H. Huson. DIAMOND+ MEGAN: Fast and Easy Taxonomic and Functional Analysis of Short and Long Microbiome Sequences. *Current Protocols* 1.3:e59, 2021.

- Sofia Esquivel-Elizondo, Caner Bağcı, Monika Temovska, Byoung Seung Jeon, Daniel H. Huson and Largus T. Angenent. Caproate production and genomic characterization of *Caproiciproducens sp.* 7D4C2, isolated from a chain elongating bioreactor. *Frontiers in Microbiology*, 11, 2021

- Caner Bağcı, Sina Beier, Anna Górska, and Daniel H Huson. Introduction to the analysis of environmental sequences: metagenomics with MEGAN. In *Evolutionary Genomics*, pages 591-604. Humana, New York, NY, 2019.

* denotes equal contribution.

Personal Contributions

Paper 1: All authors contributed to the study design. AG, DH and RW designed the visualization techniques. DH implemented the algorithms and visualization techniques. IB and DJ performed Nanopore sequencing. BA implemented the MAF2DAA program. BA and CB performed the simulation study. DH and RW lead the project and wrote the manuscript. All authors read and approved the final manuscript.
Personal contributions in the study design and the simulation study: CB contributed to the study design by specifying the need for frameshift-aware alignment and specialized LCA algorithms for long-reads. CB and BA contributed equally to the simulation study.

Paper 2: GQ developed and performed the enrichment reactor experiment and obtained samples IB designed and performed the sequencing experiment. RBHW and DHH designed the analysis strategies. KA, CB, SB, AG, DHH, and RBHW performed the data analysis. BB implemented the frame-shift alignment and range-culling in DIAMOND. DHH implemented frame-shift correction in MEGAN. DHH and RBHW wrote the manuscript, and all other authors contributed. All authors read and approved the final manuscript.
Personal contributions in the analysis: CB performed the assemblies together with KA, performed the taxonomic assignment, visualization of the assembly, contributed ideas to the frameshift correction, performed the comparison of LR- and SR- and RefSeq assemblies, performed the break-point and repeat analysis.

Paper 3: DHH proposed and guided the project and wrote the manuscript. BA and CB contributed to the writing. BA and CB wrote the software framework. BA designed and implemented the built-in alignment tool ELLA. CB designed and implemented the protein-graph approach. All authors read and approved the final manuscript. BA and CB contributed equally to this work.

Paper 4: D.B. and D.H.H. conceptualized the project. D.B. and D.H.H. developed the outline algorithm. D.H.H. designed and implemented the software. C.B. and B.C. designed and populated the database. C.B. performed all data analysis and comparisons. D.H.H. and C.B. wrote the original draft of the manuscript and all authors edited the manuscript.

Paper 5: D.H.H. and C.B. conceptualized the project. H.F. and C.B. performed the computations. A.G. and H.F. analyzed the results. A.G. and D.H.H. wrote the manuscript. All authors edited the manuscript.
Personal contributions in the analysis: C.B. conceptualized and supervised the master thesis of H.F., which resulted in the publication.

Objectives and Contributions

This dissertation aims at introducing novel methods that are more efficient in terms of accuracy and computational needs in the analysis of metagenomics datasets. The methods introduced range from new pipelines and algorithms to analyze long-read metagenomics datasets, to utilization of alternative databases and phylogenetic outlines for the taxonomic and functional analysis of both reads and metagenome-assembled-genomes.

It presents extensions to the Metagenome Analyzer (MEGAN) to be used in the context of long-read metagenomic datasets. The algorithms and the analysis pipeline developed in MEGAN-LR (Huson *et al.*, 2018) allow long-read metagenomic datasets to be accurately analyzed for their taxonomic and functional content. This approach is further extended to taxonomic binning of contigs assembled from such datasets, resulting in metagenome-assembled-genomes (Arumugam *et al.*, 2019), as well as correcting systematic frameshift errors in contigs resulting from long-read metagenomic sequencing datasets. It also introduces an alternative database, AnnoTree (Mendler *et al.*, 2019), to be used as the reference instead of NCBI-nr in DIAMOND+MEGAN based pipelines, in order to both decrease the computational resources and time required for the analysis, while at the same time increasing the taxonomic and functional assignment rates (Gautam *et al.*, 2022).

It introduces MAIRA (Albrecht *et al.*, 2020), a software consisting of novel algorithms for the real-time and in-field (on a laptop) analysis of microbiomes using Nanopore sequencing. MAIRA is a software with a graphical user interface, designed for easy use by clinicians and biologists, that can perform taxonomic and functional (antibiotic resistance and virulence factors) analysis of metagenomic datasets sequenced with Nanopore devices. Its main target is diagnostic use in a clinical setting, where both

accuracy and time are valuable, and access to large infrastructures is limited.

Finally, it demonstrates how phylogenetic outlines can be used in the phylogenetic context of microbes to evaluate their evolutionary relationships (Bagci *et al.*, 2021). In this method, the evolutionary distances of query genomes against a database of reference genomes are calculated using the Mash algorithm (Ondov *et al.*, 2016). The phylogenetic outline constructed from these distances are proposed as an alternative and faster method to phylogenetic placement to evaluate the phylogenetic context of the query genomes.

Conclusions

This chapter explains the main challenges tackled and the major contributions achieved in the publications that are included in this dissertation. Section 1 introduces the Paper 1 and the Paper 5 (Appendix I and Appendix V), on the extensions of MEGAN to long-reads and using AnnoTree as a reference database. Section 2 introduces Paper 2 (Appendix II) on metagenomic assembly, binning, and the frameshift correction of contigs assembled from Nanopore metagenomic datasets. Section 3 introduces MAIRA, a software and group of algorithms to analyze Nanopore metagenomic datasets in real-time and in-field. Section 4 introduces the use of phylogenetic outlines to determine the phylogenetic context of microbial genomes using Mash distances. Finally, an outlook is given.

1 Taxonomic Binning of Long-Reads

Since the introduction of next-generation sequencing technologies and their application to microbiomes, a great amount of work has been done on the taxonomic and functional classification of metagenomic reads and contigs. One of the first and simplest approaches to assign taxonomy to each metagenomic read was using the Basic Local Alignment Search Tool (BLAST) (Altschul *et al.*, 1997) to align all reads against all existing reference sequences in a database (such as BLASTx against NCBI-nr). Each read would then be assigned to a taxonomic unit by analysing the significant hits that BLAST has generated, as in using the lowest common ancestor (LCA) algorithm (Huson *et al.*, 2007).

However, NGS technologies have, at the same time, caused an exponential growth of reference databases, as sequencing has become cheaper and available to everybody. As of August 2021, NCBI's GenBank database contains a total of 941 billion nucleotides in sequences, compared to 56 billion at the end of 2005, when the first next-generation sequencing platforms became available. The exponential growth of the databases has become even more dramatic recently, with GenBank growing from 654 billion nucleotides to 941 billion nucleotides in the span of a year, from August 2020 to August 2021; and even almost doubling in size from 367 billion to 654 billion from August 2019 to August 2020 [1]

The growth of the reference databases, and the massive amount of data generated for each sample in microbiome studies, required the researchers to adopt the computational methods to classify metagenomic reads and contigs. Some methods have shifted away from computing computationally expensive alignments of reads against all known reference sequences. Kraken (Wood and Salzberg, 2014), for example, assigns reads to taxonomic units based on the exact matches of k-mers, with a simple hashing strategy, decreasing the computational resources needed by far. It has also recently been reported to suffer from the growth of databases (Nasko *et al.*, 2018). Kaiju (Menzel *et al.*, 2016) and Centrifuge (Kim *et al.*, 2016), on the other hand, use more complicated data structures to allow either exact or inexact but gapless matches of any length to provide a more sensitive assignment of reads than simply relying on fixed-length k-mers. Different to read binning, MetaPhlAn (Segata *et al.*, 2012) aims at generating the taxonomic profile of the sample by computing alignments of the reads against a much smaller database of clade-specific marker genes. As not all reads align to this small set of genes, instead of binning each read to a taxonomic unit MetaPhlAn rather generates an estimation of the relative abundances of organisms present in the sample.

Another pursuit of dealing with the growing size of reference databases and sequencing runs has been to significantly accelerate the alignment process without losing significant sensitivity of it. DIAMOND (Buchfink *et al.*, 2015) is one of the first such tools to make a substantial improvement over BLAST, aligning reads 20,000 times faster than it, with a similar degree of sensitivity. MEGAN was developed to analyze large metage-

[1] https://www.ncbi.nlm.nih.gov/genbank/statistics/ , accessed August 2021

nomic datasets using the LCA algorithm to classify reads based on their significant alignments to all reference sequences in a database. DIAMOND+MEGAN pipeline has thus taken advantage of DIAMOND being 20,000 times faster than BLASTx, and still allows taxonomic binning of reads based on full alignments of reads against comprehensive databases, such as NCBI's non-redundant protein database (NCBI-nr), at acceptable speeds and computational resources needed. Since it uses complete databases instead of a reduced set of genes and still relies on complete alignments instead of alignment-free k-mer or pseudo-alignment approaches, it arguably achieves better sensitivity when assigning reads, especially from complex environments, which are not well represented in the reference databases. Additionally, using alignments of reads again a database (preferably against a protein database) is the only way to obtain a functional profile of a microbiome sample, as has been implemented in DIAMOND+MEGAN pipeline, as well as in approaches such as HUMAnN2 (Franzosa *et al.*, 2018).

Although a substantial amount of research has been carried out, and many different methods have been established successfully in order to bin short metagenomic reads into taxonomic classes, as well as to functionally profile them; when long-reads had started being used in metagenomics, most these methods were not suitable for the same tasks. The two features of long-reads, namely them being longer and non-uniform in length, and being more error-prone than short-reads, have caused problems for existing taxonomic binners and functional profilers for metagenomic data (Huson *et al.*, 2018).

Centrifuge (Kim *et al.*, 2016) and Kaiju (Menzel *et al.*, 2016) were two of the first tools developed to overcome the problem of taxonomic profiling and binning of long-reads. Although both were initially developed for short-reads, both have also been shown to be capable of working with long-reads, too. Both of them use a similar idea, with an FM-index, that allows maximal exact matches. Kaiju can work with protein references, as well, whereas Centrifuge employs only nucleotide sequences from genome databases. The maximal exact matches overcome the problem of the high rate of errors in long-reads, which would result in many missing matches if fixed-length k-mers were used (Liu and Schmidt, 2012).

The protein-alignment and LCA-based approaches, such as in DIAMOND+MEGAN, were affected to a greater degree than alignment-free approaches. The length of the long-

reads breaks the assumption that one read usually aligns to one protein (even partially to a protein), as short-reads are generally shorter than the average length of a protein. The long-reads, however, can span multiple genes on a single read, considering their average length is measured in tens of kilobases. This results not only in the LCA algorithm considering all alignments from different genes equally the same, but also the heuristics aligners employ not to report most of the significant alignments. DIAMOND, for example, was initially developed with short-reads in mind and thus also makes the assumption that one read can span only one protein. Therefore it evaluates all alignments equally and reports only the best-scoring few of them.

The high error-rate of the long-reads brings a more obvious yet more challenging problem for the aligners. Most of the errors on raw long-reads are insertions and deletions rather than substitutions (Mikheyev and Tin, 2014). These insertions and deletions induce frameshifts when the reads are translated into amino acid space, as in translated BLASTx alignments. The frameshifts, therefore, lead to premature termination in the translated alignment of error-prone long-reads against protein databases, resulting in many short alignments that are broken into pieces at every site of an insertion or deletion error.

We addressed both of these problems in MEGAN-LR in order to allow accurate taxonomic and functional binning of error-prone metagenomic long-reads using protein references and compared them to the existing methods in a comprehensive simulation study. The first pipeline that we suggested, LAST+MEGAN-LR (Huson *et al.*, 2018), replaced the alignment component of DIAMOND with the LAST aligner (Kielbasa *et al.*, 2011), and introduced a novel LCA algorithm for binning long-reads that span multiple genes.

LAST is a versatile aligner that can carry out many tasks, from genome-to-genome alignment to DNA-to-protein alignments. One important feature of it is the frameshift-aware alignment (Sheetlin *et al.*, 2014) in DNA-to-protein mode, which allowed us to overcome the problem of insertion and deletion errors breaking the translated alignments for error-prone long-reads. It is also an all-against-all aligner, thus, does not employ a heuristic to filter alignments before they are reported, making it suitable for long-reads which span multiple genes. In the frameshift-aware mode, LAST translates DNA sequences into all six reading-frames, and can switch the frame of the alignment back and

forth when an insertion or deletion is observed in the read. At the time of the publication of MEGAN-LR, the frameshift-aware alignment was under development for DIAMOND. Later, we adopted the pipeline to use DIAMOND instead of LAST (Arumugam *et al.*, 2019; Bağcı *et al.*, 2021), when it adopted a similar frameshift-aware alignment algorithm. The main advantage of using DIAMOND over LAST for the alignment step is the output DAA format, which is much more compressed and more suitable for the downstream analysis by MEGAN-LR.

The naïve LCA algorithm that MEGAN had successfully been using for the taxonomic binning of short-reads was not suitable for long-reads, as it often resulted in unspecific and rather spurious binning of the reads. The naïve LCA algorithm assumes that a read is spanned by a single gene, as is often the case with short Illumina reads. Thus, it considers all alignments to a read equal on many levels, such as their length, bitscore, and position on the read. It proceeds by taking those alignments whose bitscore lie within a specified percentage of the best scoring alignment for the read, and assigns the read to the lowest common ancestor of the taxonomic mappings of the proteins from which alignments pass the filter. When applied to long-reads, or long-read contigs, such as those of sizes millions of base pairs, this simple filter results in taking only the longest protein(s) on the read/contig into account as those are often the ones that produce the highest bitscores.

Therefore, we developed the *interval-union LCA* algorithm, which aims at considering all genes (regions) on a long-read from which significant alignments arise and assign the read to the lowest common ancestor that agrees with the majority of the taxonomic mappings of the proteins from said regions. Instead of taking all alignments from a read into account at once, the *interval-union LCA* algorithm first divides a read into ”intervals”. The reads are scanned from beginning to the end, and each time a new alignment starts or ends, a new interval is defined whose start point is the previous start/end event, and the end point is the current start/end event. As in naïve LCA algorithm, only those alignments whose bitscores are within a percentage of the bitscore of the best scoring alignment within an interval are considered for the further steps.

In the second step, in order to determine the taxonomic assignment of the long-read, we compute the union of the intervals in which an alignment from a protein of

a taxonomic unit or any of its descendants is present. Then, the taxonomic tree with all taxa having interval-union scores is traversed in a post-order traversal until a taxon whose interval-union score is above a specified threshold (80% in the original paper, later adopted to 51%). In more detail, the traversal starts at the root of the tree and proceeds as follows, with three conditions:

- if there exists more than one child with an interval-union score above or equal to the threshold, terminate and report the current taxon as the LCA
- if there does not exist any child with an interval-union score above or equal to the threshold, terminate and report the current taxon as the LCA
- if there exists one and only one child with an interval-union score above or equal to the threshold, recurse the traversal from that child

As there exist two termination criteria, an LCA of a long-read can either mean that there does not exist a convincing amount of references to assign the read to a lower-rank taxon (first condition); or the read can be assigned to multiple taxa with the same confidence and thus is placed on the lowest common ancestor of those (second condition). With environmental samples, both of these cases are possible in different scenarios. It is often difficult to confidently assign reads to lower-rank taxa in metagenomes from poorly studied environments, while it is often the case that there are multiple strains or species with similar scores for organisms from extensively studied environments with vast amounts of references.

The interval-union LCA algorithm differs from the naïve LCA algorithm also in the way that it reports the abundance of organisms in the sample. The naïve LCA algorithm reports the number of reads assigned to each taxon. However, this is not suitable for long reads/contigs as their lengths can be largely varying, whereas the short-reads are often uniform in length. The interval-union LCA algorithm instead reports the sum of the number of assigned bases on reads (i.e. the sum of the interval-scores mentioned above) for each taxon.

The naïve LCA algorithm can assign each read to a single functional class, as it applies the same assumption in taxonomic binning that each read spans one gene. In the

case of long-reads, however, the interval-union LCA algorithm assigns a function to each interval, thus allowing a long-read to be assigned to multiple functional classes, each of which may be carried out by one of the several genes it may span.

To show the accuracy of the method developed here, we designed a simulation study where the whole LAST+MEGAN-LR pipeline is run against reads simulated from more than a thousand different genomes, and its performance is compared to that of Kaiju using the same set of simulated reads. Kaiju, at the time, was the only tool that performed translated nucleotide searches against protein databases, and we did not compare the performance of LAST+MEGAN-LR to methods that employ nucleotide databases.

In a real environmental sample, the organisms that are being sequenced are very unlikely to be exactly the same as an organism that had been sequenced and deposited in public databases before. Therefore, to simulate the real metagenomic question of assigning taxonomy to novel organisms, we designed our simulation in such a way that when we simulated reads from a genome and ran it against the pipeline, we took out all the references from the database that came from the said genome. We also limited our simulation to those organisms whose genus contains between 2 and 10 complete reference genomes. The reason for this is that it would have been impossible to assign a genus-level classification if the genus contained only one genome and it had been taken out in our simulation, or it would become relatively too easy to assign the correct genus with too many references as also indicated in our results.

For each of the 1151 genomes that we selected based on the above criteria, we simulated 2,000 long-reads coming either from R7.3 or R9 chemistries of Oxford Nanopore Technologies, with R7.3 being more error-prone. Then we constructed both LAST and Kaiju databases with the references from the genome in question taken out. Finally, we ran both LAST+MEGAN-LR pipeline and Kaiju with the simulated set of reads and database for each of the 1151 genomes and compared their outputs at the read level.

We calculated the sensitivity and the precision of assignments from both tools at the genus level, as the simulated genomes were chosen based on the number of existing genomes within their respective genus. MEGAN-LR outperformed Kaiju both on sensitivity, the rate of reads that are assigned to the correct genus or below; and also precision,

the rate of reads assigned correctly, ignoring those unassigned or assigned to an ancestor of the true genus. The results showed similar trends both for R7 and R9 chemistries, although both tools performed better on the less error-prone R9 chemistry.

As expected, the assignment of reads coming from genera that are represented better in the database generally resulted in better sensitivity and precision. Per sample comparison of MEGAN-LR to Kaiju has also shown that MEGAN-LR outperformed Kaiju in nearly every dataset, with a small number of exceptions. On the other hand, Kaiju was many times faster than LAST+MEGAN-LR pipeline.

In addition, we also performed a parameter scanning study to evaluate the effect of the threshold for the interval-union score (i.e. percentToCover), as well as the threshold to select alignments based on their bitscore in relation to the top-scoring alignment. We carried out the parameter scanning on a mock community for which the organisms present and their relative abundances were known. Lower values of the topPercent parameter increase the specificity of the assignment, although at the cost of increasing false-positives. While increasing values for the percentToCover parameter increases the specificity of the assignments, as well as increasing the rate of false positives. The organisms present in the mock community were all well-studied and well-represented in the databases. Thus, we recommended a slightly more relaxed set of parameters than those which performed the best.

LAST+MEGAN-LR pipeline has achieved both taxonomic and functional binning of long-reads at significantly higher sensitivity and precision than what had been available at the time. It has been developed further to allow working with metagenomic assemblies of long-reads and correct for frameshift errors that appear in them, as described in Section 2.

AnnoTree as an Alternative Database

NCBI-nr (Benson *et al.*, 2005) has been the standard database of an analysis carried out by MEGAN. It is currently the most comprehensive protein database, consisting of a

non-redundant collection of all protein sequences available in the databases of NCBI. However, the exponentially growing size of it has recently made computational analysis using it computationally very expensive.

AnnoTree (Mendler *et al.*, 2019) is a webserver and a database of annotated protein sequences from the Genome Taxonomy DataBase (GTDB) (Parks *et al.*, 2018). It automatically re-annotates the proteins from the genomes present in GTDB using KEGG (Kanehisa and Goto, 2000), PFAM (Finn *et al.*, 2008) and TIGRFAMs (Haft *et al.*, 2003) functional classifications.

Since GTDB provides a systematic taxonomy and also includes MAGs recovered from environmental samples, AnnoTree prodives a more complete database of functional annotations of proteins, and is smaller in size compared to NCBI-nr, as it contains only bacterial and archaeal proteins. Thus, we designed a study to evaluate the performance of AnnoTree as an alternative to NCBI-nr in both the taxonomic and functional analysis of metagenomes (Gautam *et al.*, 2022).

We used ten different publicly available samples, from different environmental sources, such as human gut or soil. Using AnnoTree as the reference database instead of NCBI-nr has generally resulted in more reads being assigned to a taxonomic class. This is most likely due to GTDB including more environmental samples than NCBI-nr. The assignments from AnnoTree and NCBI-nr also agreed with each other, showing that for the taxonomic assignments AnnoTree was a suitable, if not better, replacement for NCBI-nr.

In case of functional analysis of microbiomes, using AnnoTree database assigned approximately 70% more reads than NCBI-nr to the KEGG classification. PFAM and TIGRFAMs classifications are not available in MEGAN, thus were not tested. Besides assigning more reads to taxonomic and functional classes, replacing NCBI-nr with AnnoTree as the database resulted in a two-fold decrease in the computational time of DIAMOND+MEGAN pipeline.

Here we showed the potential of using AnnoTree for faster and better taxonomic and functional analysis of metagenomic datasets using DIAMOND+MEGAN-LR pipeline. One major disadvantage of this approach was the lack of reference sequences for eu-

karyotic and viral organisms in the AnnoTree databases. This would potentially lead to missing or false assignment of reads originating from them in samples containing higher amounts of eukaryotic and viral material than the samples that were tested in our study.

2 Assembly and Downstream Analysis of Microbiomes

Although short and near perfect, Illumina sequencing has long been the gold standard in metagenomics; long-read technologies have also been gaining popularity in recent years. Much work has been done in assigning individual reads to a taxonomic unit (i.e. read-binning) or in predicting their relative abundances within one sample (i.e. profiling). However, in order to study the genomic capabilities of an organism as a whole, its genome is still desired at satisfying completeness and contiguity. The assembly of metagenomes into genomes at useful completeness and contiguity has been near impossible with short-reads, as the assemblies often get very fragmented. The long-reads, however, can reach the sizes of the contigs from short-read assemblies, even in the form of raw data. Therefore, they have become an attractive tool to study microbiomes as metagenomic assembled genomes (MAG) (Frank *et al.*, 2016). We, here, developed a pipeline to assemble MAGs from long-read sequencing data, assign taxonomy to them, report a quality measure on them, and annotate them at an acceptable quality to enable the study of their functional capabilities. The whole pipeline takes about 6 hours once the sequencing finishes.

We applied Nanopore sequencing on an enriched microbiome from a bioreactor run on sludge from a waste-water treatment plant to target phosphate-accumulating organisms. The sequencing yielded near 700,000 reads, with an average length of 9 kb. The first step in our pipeline was to assemble these reads into contigs using Unicycler (Wick *et al.*, 2017). Although Unicycler has been designed to assemble isolated microbial genomes rather than metagenomes, its computational requirements over the accuracy trade-off were better than the state-of-the-art Canu (Koren *et al.*, 2017) at the time of this work. Later, we adopted the pipeline to use metaFlye (Kolmogorov *et al.*, 2020; Bağcı *et al.*, 2021), which has specifically been designed to assemble metagenomic long-

read samples and still runs at an acceptable timeframe with acceptable computational resources. From our assembly, we have obtained 1702 contigs, with an average length of 61 kb. However, Unicycler was also able to assemble five circular, complete genomes with sizes ranging from 2.7 to 4.2 Mb. There were another five linear contigs over the length of 1Mb.

Fundamentally, the long-read metagenome assemblies can be binned into MAGs without using sophisticated contig binning algorithms from environments of such diversity, as some of the assembled contigs are already complete or near-complete genomes. We, thus, performed the taxonomic binning of these contigs using MEGAN-LR with DIAMOND (Buchfink *et al.*, 2015) as the frameshift-aware aligner. We then simply extracted all of the contigs that are assigned to each taxonomic unit by MEGAN-LR as a "contig bin", potentially representing a MAG.

The quality of MAGs produced from the assembly, and binning of microbiomes are often deemed using CheckM (Parks *et al.*, 2015) or similar approaches. CheckM uses lineage-specific marker genes that are ubiquitous and appear in a single copy for near all organisms of that lineage in order to determine the completeness and contamination of a bin. The MAGs are also often annotated with their coding sequences to study their functional capabilities using tools such as Prokka (Seemann, 2014). Both of these methods require the *de novo* identification of coding sequences as they rely on alignments and profile searches on protein sequences. The contigs assembled from erroneous long-reads contain a significant amount of errors, although after being corrected multiple times. The majority of these errors are insertions and deletions of one or multiple bases instead of substitutions of one base with another. The insertions and deletions that end up in the final contigs cause frameshifts when translated into coding sequences and thus disrupt the *de novo* prediction of coding sequences. This limitation makes tools such as CheckM and Prokka unusable for contigs assembled solely from long-reads.

In order to address this issue, we implemented a simple yet effective method in MEGAN-LR to correct the frameshift errors in contigs assembled from long-read sequencing methods. After performing the frame-shift aware alignment using DIAMOND, we consider the top-scoring alignment on each interval of a long-read, as described in Section 1, to correct for frameshifts using the protein references. Frameshifts causing the

frame of the alignment to decrease are depicted with forward-slashes, and those causing the frame to increase are depicted with reverse-slashes in frameshift alignments. A decreasing frame can be caused by either deletion of one (or 1 + 3N) or the addition of two (or 2 + 3N) nucleotides in the contig, and an increasing frame can be caused by either addition of one (or 1 + 3N) or the deletion of two (or 2 + 3N) nucleotides. To bring the frame back to its starting point, we add one unspecific base (N) or two unspecific bases (NN) when a forward or reverse slash is observed in the alignment, respectively.

After binning the contigs with MEGAN-LR, and correcting for frameshift errors in the binned MAGs, we evaluated their quality according to the MIMAG standards using CheckM and annotated the genomes using Prokka. We obtained seven high-quality draft genomes according to MIMAG (Bowers *et al.*, 2017) standards, that are more than 90% complete and less than 5% contaminated. In addition, we obtained four medium and three low-quality draft genomes. All of the seven high-quality draft genomes are derived from single complete, five of them being closed circular, contigs, thus representing the full chromosome of the assembled genome in its full contiguity. The counts and lengths of coding sequences, tRNAs, and rRNAs predicted by Prokka also lay within the ranges usually seen in bacteria.

One of the seven assembled chromosomes belonged to the genus *Candidatus Accumulibacter*. This known phosphate-accumulating organism that is commonly found in in waste-water treatment plants and was the target of the enrichment in this study. Two other circular chromosomes belonged to the organisms *Bacteroidetes bacterium* OLB8 and OLB12, also known to occur in anammox bioreactor communities. There existed reference metagenome-assembled genomes in public repositories for all of these three organisms. We performed a genome-wide alignment of our assembled chromosomes against the reference assemblies. The assembled chromosomes aligned end-to-end to the references, all of which were quite fragmented.

To further check if the assembled chromosomes contained any major misassembly or chimerism, we also obtained a set of paired-end short-reads from the same community and aligned them to our assembled contigs. From the mapping of short-reads to the contigs, we calculated short-read mapping coverage along the long-read contigs and detected the ”break-points” where the coverage of short-reads dropped significantly. We identi-

fied eleven of these in the seven long-read chromosomes. In addition, we performed an assembly of the short-reads, which resulted in many fragmented contigs. We aligned these short-read contigs to the long-read chromosomes, akin to the genome-wide alignment to the references as above. We again observed a linear alignment of the short-read contigs to our long-read chromosomes.

We also performed an additional verification of repeats assembled in our long-read chromosomes by aligning them against themselves. We calculated the percentage of repeated locations on the chromosomes from the alignments and compared them to all complete bacterial reference genomes from RefSeq (O'Leary *et al.*, 2016). We observed that the rate of repeats in our long-read chromosomes is similar to those of reference genomes for all bacteria, although two of them showed rather a high percentage of repeated locations (7 to 8% of all positions on the chromosome), which was the driving reason for this verification step.

With the short-read data mentioned above, we performed a "hybrid" correction of the long-read assemblies using pilon (Walker *et al.*, 2014) and mappings of short-reads to our long-read contigs. We then calculated the rate of frameshifts for the original, unpolished assembly and for the short-read polished assembly. We found out that polishing long-read assemblies with short-reads significantly reduces the rate of frameshifts from 6 per kilobyte to 1.2 per kilobyte. However, running pilon required extensive computational resources and time, and still did not correct for all of the insertions and deletions, thus still requiring a frameshift-correction step.

Here, we showed that long-read-only assemblies of microbiomes have the potential to recover complete, or near-complete, fully assembled chromosomes as MAGs. There are limitations in the downstream analysis of long-read assembled contigs; however, these can be overcome using our frameshift correction method.

3 Accurate and Real-Time Analysis of Microbes

The MinION sequencing platform released by Oxford Nanopore Technologies in 2014 opened up the opportunity to sequence and analyze biological samples in the field. In addition to its long-read sequencing capabilities, MinION is also a portable, palm-sized device that operates with power from a USB connection. Such portability allows the sequencing to be taken to the field and does not require any complex infrastructure. Another major advantage introduced by ONT sequencing platforms is the capability of streaming data. The raw electrical signals measured by the sequencing device, such as the MinION, becomes available in real-time. These raw signals can then be basecalled almost in real-time, either on a laptop or by special hardware ONT offers (MinIT, or recently MinION Mk1C).

In order to take advantage of both portability and real-time sequencing MinION offers, we have developed the software MAIRA (Albrecht *et al.*, 2020), which can be used to analyze microbial samples sequenced by the MinION, in real-time, and on a laptop. MAIRA is the first of its kind to be able to analyze samples from Nanopore sequencing devices in real-time, and on portably a laptop, without requiring *a priori* knowledge of the sample. MAIRA takes input the basecalled reads as they are produced and analyzes both the taxonomic content of the sample, and also functional capabilities of the microbes that are identified for their antibiotic resistance potential and virulence factors that they may carry. It is designed with clinical microbial data in mind, but the functional analysis capabilities can be extended to other fields as well. It provides a graphical user interface (GUI) to set up the analysis and to monitor the results of the taxonomic and functional content of the sample. Taking advantage of long-reads, it can also connect the taxonomic and functional information gained, which misses in many short-read microbiome analyzing software. MAIRA is implemented in Java, and thus can be run on all three major operating systems.

Similar to MEGAN-LR (Huson *et al.*, 2018), MAIRA bases its taxonomic and functional classification of microbes on protein alignments. As it can be envisioned, the alignment of reads against a very large database would not be feasible on a laptop in real-time. To overcome this, MAIRA operates in two steps. In the first step, it aligns all

reads against a genus-specific marker database, with the attempt to predict microbial genera that are potentially present in the sample. In a second step, it aligns all reads that are basecalled so far against all proteins of the genera that are identified in the first step, in order to classify the species that are present. In this step, the functional classification is also carried out, which is achieved by aligning the reads against specific functional databases, namely CARD (Alcock *et al.*, 2020) for antibiotic resistance and VFDB (Chen *et al.*, 2005) for virulence factors. The alignments are carried out in frameshift-aware sense, as the long-reads generated by ONT platforms are prone to insertion and deletion errors, as discussed above.

The first step of genus-level analysis is a rapid step to provide an overview of the taxonomic content. The ”genus marker” database for this step is computed and made available from proteins that are specific to a single genus, which does not appear in any other genera with an identity over 80%. To increase the speed of the alignment, they are also clustered within the genus, that is only one representative protein is kept for those sharing over 90% identity.

The genus-level analysis provides a very fast overview of the genera that are potentially present in the sample, yet it is not very specific. Although it can predict the genera that are present correctly, it often makes false-positive calls as well. The next step of the species and functional identification of the sample is a more in-depth analysis, aiming at accurately classifying the species that are present in the sample. The decision of which genera to be analyzed further is made either automatically for those having a predicted score above 0.8 by default or by the user who can activate or deactivate the genera to be analyzed. All reads are then aligned against all proteins of the activated genera in a frameshift-aware manner for each batch of the reads. MAIRA supports the use of frameshift-aware aligners, LAST (Kielbasa *et al.*, 2011) and in-house developed ELLA.

The species-level analysis is activated at the end of each batch when all reads are aligned against the genus databases of all active genera. In this step, MAIRA uses a so-called ”synteny-graph”, analogous to overlap graphs used in the assembly process of (meta)genomes. The synteny-graph takes input as the all alignments of all reads against the activated genus databases and outputs metrics for the relative abundance and the confidence of the species that it predicts to be present.

The synteny-graph is built around the idea that long-reads span multiple genes, and a more precise call on the presence of an organism can be made if the conserved synteny of these genes in all of the reads seen so far is taken into account. The graph consists of nodes which represent gene families. These gene families are determined from the alignments of reads to the proteins. They are then connected by an undirected edge when they are seen to occur subsequently on a long-read, i.e. in immediate synteny to each other.

MAIRA starts with an empty synteny-graph, which gets populated with the alignments of the reads at the end of each batch of the analysis. Each long-read is processed individually in this process. The process starts with filtering the alignments. The alignments that do not cover a certain fraction of the reference protein (80%, by default) are not taken into account. The reasoning here is to exclude those alignments which result from highly conserved domains on the proteins while a reasonable amount of the protein remains unaligned. Next, alignments that still cover a good majority of the protein, however, are very divergent from the reference are excluded. The percentage of positives is used as a threshold here and is set to 60% by default. The percentage of positives is a measure that is calculated from the ratio of the number of positive scoring matches (coming from the substitution matrix used) to the length of the alignment. We use the percentage of positives, instead of percentage of identity, since a positive match in protein alignments indicate that the substitution in that position may not lead to significant functional changes in the protein, although the amino acids are not identical.

All of the alignments that pass the said filters from all genus databases are then gathered together for a long-read. In order to determine the gene families that would represent the nodes in the synteny graph, a ”binning” of the alignments based on their locations on the reads is performed. The alignments are stacked together when they overlap at the same locus on a long-read. An overlap is considered valid when at least 2/3 of the smaller alignment overlaps with the longer alignment. It is also taken into account that when a batch of long-reads is aligned against a genus database at a later point because it becomes activated at a later stage of the analysis, all previous alignments from that read are re-analyzed for binning. Each ”alignment bin” represents a gene family and is modelled as nodes in the synteny graph. For each read, individually, the alignments within these bins are filtered further by their raw score, only taking those whose raw scores lie

within the 90% of the score of the best scoring alignment. This is done to reduce the load of the alignments to consider at the next step, as most of those low-scoring alignments are likely from organisms closely related to the actual organism that the read comes from. Further, it helps with the alignments that pass the initial filters due to database inconsistencies, such as partial proteins or only conserved domains being present in the database, which would still make it through the coverage filter.

The same procedure is applied to alignments from functional databases, and they are added to the nodes in the graph. Each node is then populated with the taxonomic and functional annotations of the proteins it contains. Both of these are retrieved from an SQLite database, which contains the information on all organisms a protein has been seen in so far instead of the lowest common ancestor of them. Although this results in more computations being carried out, it allows MAIRA to be more specific about the call it makes.

Finally, the nodes that appear next to each other on a long-read are connected by an undirected edge, if it does not already exist, representing their synteny. The read is added as an attribute to the edge. The direction of the edges in regard to the overall orientation of the genes on the genome cannot be determined reliably, as it can be influenced by two independent factors. The read may come from either strand of the DNA; thus, the orientation of the direction would have to be reversed half of the time, which is computationally easy to detect. However, the orientation of family genes that are often in synteny may also be completely or partially changed in different organisms, as well. This, in turn, makes it impossible to detect whether the orientation of the synteny needs to be reserved, as it can be either of the two cases.

Once all alignments from all genus-specific and functional databases have been added to the graph, MAIRA proceeds to call the taxonomic and functional content of the sample. The synteny-graph helps in eliminating false-positive calls by filtering out those species from which alignments are obtained due to genomic similarity. The graph is initially induced for each species, and as a first step, the species whose induced graphs are not well-connected are filtered out. This filter helps to filter out species that are similar in genomic context (share similar genes), but their overall genomic structure is different from the true species present in the sample; thus, their synteny-graphs are not

well-connected.

After filtering out non-connected induced graphs, MAIRA proceeds to check the containment of induced graphs among all. Although uncommon, it is possible that two species share the same set of gene families and in synteny. These are often two very similar species from the same genus, and big portions of their genomes are similar. In this case, MAIRA eliminates the false-positive calls by checking the containment of one genome within another. The idea here is to filter out the species which has produced alignments due to their similarity to the true-positive call by checking the uniqueness of protein nodes in the synteny graph. The species whose nodes are contained in another by more than 85% are filtered out by default.

In order to determine the functional capabilities of the species present in a sample, MAIRA adds all functional hits that it detects to the synteny graph. It can then proceed in two modes to assign the functional hits to the called organisms. In a conservative approach, it can assign functional hits to a species that are contained within the induced synteny graph of that organism. In addition, in a more relaxed approach, it can assign the hits that are contained in a node that is a certain number of edges away from any node of that species. The latter considers that these genes may have been horizontally transferred and have not been seen before in any reference to that organism.

In order to evaluate the performance of MAIRA, we carried out four difference studies. First, we obtained a mock community dataset sequenced by Nicholls *et al.* (2019) using a GridION sequencing platform. The mock community consisted of eight microbial and two fungal organisms. Carrying out an analysis on a high-end laptop, by taking 10,000 reads at each batch, MAIRA required only five batches of reads (50,000) and slightly above two hours to detect the presence of all eight bacterial species. It also detected one false-positive species with a relatively high completeness-score of 0.75, however, this stayed below the default threshold of 0.80. The false-positive species *Bacillus amyloliquefaciens* is highly similar to one of the true-positive species in the samples, *Bacillus subtilis*.

In order to have a better understanding of the performance of MAIRA in more realistic settings, we also carried out simulation studies. First simulation study focused on

functional identification of antibiotic resistance and virulence factor genes on the species that MAIRA calls. We simulated Nanopore reads from ten different species, found in the commercially available mock community designed for pathogen detection: ATCC MSA-4000TM. We have detected the antibiotic resistance and virulence factor genes of these organisms by aligning their genomes against CARD and VFDB databases using DIAMOND in an independent manner. MAIRA have identified all ten species present in the sample, as well as reporting three false-positive species. The false-positive species reported were highly similar to four other species present in the sample, coming from two different genera: *Staphylococcus* and *Streptococcus*. On identification of antibiotic resistance and virulence factors, MAIRA achieved a high rate of true-positives, while maintaining a very low rate of false-positives.

In a separate simulation study, we compared the performance of MAIRA against Centrifuge (Kim *et al.*, 2016). Here, we randomly selected 100 genomes that were made available after 2016. We built protein databases for MAIRA and genome indices for Centrifuge consisting of genomes that were published before 2016, in order to simulate the situation that the analyzed genomes do not exist in the public repositories. When the simulated genome came from a novel species, that did not have any genomes available in the databases before 2016, MAIRA reported a false-positive in 5% of the cases. In contrast, Centrifuge reported a false-positive in all cases. As the number of assemblies from different strains of the species analyzed increased in the database, Centrifuge eventually outperformed MAIRA. When there were more 15 assemblies for a species already available in the database, Centrifuge reported the true-positive in all simulated cases (25 simulations). On the other hand, MAIRA, failed to identify any species four times and reported a false-positive one time.

By employing the ideas of checking for completeness, uniqueness, and synteny of genes, MAIRA achieves a high sensitivity of calling the true positive species. Its main weakness lies in calling organisms that are not well represented in its default RefSeq database, as seen in the second simulation study against Centrifuge. RefSeq contains only high-quality genomes from isolated organisms, thus MAIRA fails to make reliable, if any, calls from environments which do not contain many cultured microbes, such as soil. It also maintains a strength in such cases, by trying to avoid reporting false-positive species, unlike many other taxonomic identification tools, as seen in the second simula-

tion study against Centrifuge.

4 Phylogenetic context

As described in Section 2, the assembly of microbiomes results in MAGs, which are often desired to be analyzed both taxonomically and functionally. The taxonomic analysis of resulting MAGs is often done by calculating their ”phylogenetic context” of them. The phylogenetic context is the indication of phylogenetic relationships with the given query genomes and other organisms that are found to be similar to them from a database of genomes based on their taxonomic similarities.

This can be done using fast, alignment-free methods such as Mash (Ondov *et al.*, 2016), marker-gene-based phylogenetic placement methods such as GTDB-Tk (Chaumeil *et al.*, 2020), or using methods that align whole genomes against protein or nucleotide references such as DIAMOND+MEGAN-LR (Huson *et al.*, 2018), as introduced above. All of these methods require extensive scripting and many further steps to obtain a final visual representation of the phylogenetic context of the query genomes.

Here, we introduced a fast and interactive method, implemented in SplitsTree5, to explore the phylogenetic context of a given query genome or a set of genomes (Bagci *et al.*, 2021). We use our implementation of Mash to query the given genomes against the reference set of GTDB genomes to figure out their similarities to known organisms quickly and visualize their phylogenetic context from the distances calculated from our Mash implementation in a phylogenetic outline.

The relationship of organisms is often visualized using phylogenetic or taxonomic trees. The taxonomic trees are fixed, may contain human errors, and provide very little indication of uncertainties. On the other hand, phylogenetic trees cluster organisms definitely, not allowing alternative groupings. Microbes, however, often do not evolve as a definite tree due to events such as horizontal gene transfer. Phylogenetic networks, rather than phylogenetic trees, can show these uncertain relationships among organisms.

Phylogenetic outlines display all of the splits calculated in a phylogenetic network. They are, however, much less cluttered and use much fewer nodes and edges, as they display only the outline of the network. Their main advantage in this work is that they are much faster to compute than phylogenetic networks, as they require only $O(n)^2$ time, compared to $O(n)^4$ of phylogenetic networks. (n being the number of taxa).

We used the Genome Taxonomy Database (GTDB) (Parks *et al.*, 2018) as the source of our reference genomes as it provides a genome-wise similarity-based taxonomy for bacteria and archaea and contains nearly 32,000 species, all of which are represented by a so-called ”representative genome”. We downloaded these representative genomes and computed a Mash sketch for each one of them using our implementation of the Mash algorithm in SplitsTree5. We assigned these Mash sketches to the leaves of the GTDB taxonomy and computed a Bloom filter (Bloom, 1970) for the higher-level nodes, representing the set of all k-mers in all of the sketches of leaves below that node. The taxonomy, Mash sketches and Bloom filters were stored in an SQLite database for fast access.

The Bloom filter allowed us to filter out the taxa that were out of interest early in the search. The search begins at the root of the tree and traverses the tree in a top-down way as long as the bloom filter of the queries node contains a number of k-mers that are above the threshold given by the user. Then, finally, when a leaf node (Mash sketch for the representative genome of a species) is reached, we calculate the Mash distance between the query genome and this genome. The results are shown to the user in a histogram, and the user decides how many genomes to include in the phylogenetic context based on a distance threshold. Then a more specific Mash distance, using a smaller k-mer size is calculated between the chosen genomes and the query genomes, and a phylogenetic outline of them is visualized.

We applied the newly developed method to our long-read metagenome-assembled genomes (MAG) from Section 2 and compared the calculated phylogenetic contexts to the phylogenetic placement calculated by GTDB-Tk and the taxonomic labels reported by MEGAN-LR in the original study. In all cases, the phylogenetic context was compatible with the taxonomic assignments we calculated using MEGAN-LR in Section 2. For all of the MAGs from our previous study, the closest organism reported by SplitsTree

was the one it was assigned to by MEGAN-LR. In one case, the closest organism was a rather new genome that was not available at the time of the study presented in Section 2. The second closest genome was still the one reported by MEGAN-LR.

In the phylogenetic context of *Candidatus Accumulibacter* MAG, a species of *Xanthomonadales* reported by SplitsTree. It was unexpected as this organism comes from a different taxonomic class and is not related to *Candidatus Accumulibacter*. Upon further inspection, we found out the reference genome for this *Xanthomonadales* species also comes from a granular sludge microbiome, from which two other species of *Candidatus Accumulibacter* were assembled as well. Thus, we suspect this draft reference genome to be contaminated with *Candidatus Accumulibacter* contigs. This shows that our method can also assist in detecting such anomalies.

The GTDB-Tk placement of *Candidatus Accumulibacter* genome agreed very well with the phylogenetic context calculated by SplitsTree5. All reference genomes reported by SplitsTree5 were in the immediate neighbourhood of this draft genome in the GTDB-Tk placement, as well, and they had similar distances.

The phylogenetic context for the draft genome from the genus *Thauera* also showed a similar pattern as above. It contained all references from the genus *Thauera* in GTDB taxonomy and had a similar topology and distances to the GTDB-Tk placement. Both the phylogenetic context and GTDB-Tk place this draft genome closer to the species *Thauera aminoaromatica* S2, with a Mash distance of 0.2, suggesting a more refined classification than that was reported by MEGAN-LR.

However, in the case of low-quality draft genome *Betaproteobacteria*, we observe a different behaviour in the phylogenetic context. Our method places this draft genome near organisms from *Candidatus Accumulibacter* genus, although at large distances. GTDB-Tk, on the other hand, places it nearer to the genus *Azonexus*, but outside of it too. *Candidatus Accumulibacter* is a sister genus of *Azonexus* in GTDB taxonomy. This confirms the known fact that ANI values, as approximated by Mash distances, are not suitable for comparison of genomes that do not belong to the genus and can only provide a poor estimation of the evolutionary distance between them.

Similar trends were also observed for the rest of the eleven MAGs reported in the original study, where SplitsTree5 calculated a phylogenetic outline similar to GTDB-Tk when the query genome lied within the genus boundaries and disagreed more with it when the query genomes were outside genus boundaries or did not find any similar genomes at all.

Here we developed a new method using references and taxonomy from GTDB, Mash sketches and their Bloom filters, and phylogenetic outlines in order to calculate phylogenetic context of a given draft genome. We used the MAGs assembled in our previous study, presented in Section 2, to assess the performance of the method in comparison to the original taxonomic labels reported by MEGAN-LR and phylogenetic placement calculated by GTDB-Tk. We showed that our method can be a useful tool to calculate fast outlines of newly assembled genomes and provides a visual representation of their evolutionary relationship to the reference genomes within an acceptable distance.

5 Outlook

This dissertation explored novel methods for the analysis of whole-genome shotgun metagenomic sequencing datasets, especially in the context of long-reads and their taxonomic and functional assignments. The emerging long-read sequencing technologies have developed rapidly over the last years, and have been utilized in many aspects of biological research, including microbial genetics. They allowed more detailed analysis of taxonomic and functional contents of microbiomes. It has become possible to assemble near-complete, closed chromosomes from raw sequences of complex microbiome samples, which was beyond imagination using short-read sequencing. The on-the-fly basecalling and portability of Nanopore sequencing has presented opportunities to take the analysis to the field and carry it out in real-time, directly starting after the sample has been loaded to the flowcell.

All considered, together with the growth of efficient algorithms and higher quality databases, microbiome research advances at a fast pace. All methods presented in this

dissertation have, and will continue to have, many aspects in which they can be improved with the changing state of both sequencing technologies and computational resources available to us. Simply, the Q20+ chemistry of Nanopore, which became available to the public in the last quarter of 2022, has greatly improved the quality of the raw sequencing reads, potentially meaning that the methods introduced here may also benefit from adjustments to the new technology. GTDB, as discussed in several sections of this dissertation, is rapidly becoming the *de-facto* accepted bacterial taxonomy by the commmunity, considering that now IJSEM also proposes names for the *Candidatus* taxa presented in there (Pallen *et al.*, 2022). Being a purely methodological, standardized taxonomy, it offers many conveniences and improvements that can be taken advantage of by any method developed to analyze microbial genomics data. All these rapid developments in the chemical, biological, clinical, and computational aspects of microbiome research indicates that we are far away from its maturation and a future with vast amounts of biological data and findings, requiring a constant need for development of computational methods, awaits.

Bibliography

Afshinnekoo, E., Chou, C., Alexander, N., Ahsanuddin, S., Schuetz, A. N., and Mason, C. E. (2017). Precision metagenomics: rapid metagenomic analyses for infectious disease diagnostics and public health surveillance. *Journal of Biomolecular Techniques: JBT*, **28**(1), 40.

Agrawal, S., Kinh, C. T., Schwartz, T., Hosomi, M., Terada, A., and Lackner, S. (2019). Determining uncertainties in picrust analysis–an easy approach for autotrophic nitrogen removal. *Biochemical Engineering Journal*, **152**, 107328.

Albrecht, B., Bağcı, C., and Huson, D. H. (2020). MAIRA-real-time taxonomic and functional analysis of long reads on a laptop. *BMC bioinformatics*, **21**(13), 1–12.

Alcock, B. P., Raphenya, A. R., Lau, T. T., Tsang, K. K., Bouchard, M., Edalatmand, A., Huynh, W., Nguyen, A.-L. V., Cheng, A. A., Liu, S., *et al.* (2020). CARD 2020: antibiotic resistome surveillance with the comprehensive antibiotic resistance database. *Nucleic acids research*, **48**(D1), D517–D525.

Altschul, S., Madden, T., Schaffer, A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Arumugam, K., Bağcı, C., Bessarab, I., Beier, S., Buchfink, B., Gorska, A., Qiu, G., Huson, D. H., and Williams, R. B. (2019). Annotated bacterial chromosomes from frame-shift-corrected long-read metagenomic data. *Microbiome*, **7**(1), 1–13.

Arumugam, K., Bessarab, I., Haryono, M. A., Liu, X., Zuniga-Montanez, R. E., Roy, S., Qiu, G., Drautz-Moses, D. I., Law, Y. Y., Wuertz, S., *et al.* (2021). Recovery of

complete genomes and non-chromosomal replicons from activated sludge enrichment microbial communities with long read metagenome sequencing. *npj Biofilms and Microbiomes*, **7**(1), 1–13.

Bağcı, C., Patz, S., and Huson, D. H. (2021). DIAMOND + MEGAN: fast and easy taxonomic and functional analysis of short and long microbiome sequences. *Current protocols*, **1**(3), e59.

Bagci, C., Bryant, D., Cetinkaya, B., and Huson, D. H. (2021). Microbial phylogenetic context using phylogenetic outlines. *Genome biology and evolution*, **13**(9), evab213.

Bashir, M., Ahmed, M., Weinmaier, T., Ciobanu, D., Ivanova, N., Pieber, T. R., and Vaishampayan, P. A. (2016). Functional metagenomics of spacecraft assembly cleanrooms: presence of virulence factors associated with human pathogens. *Frontiers in microbiology*, **7**, 1321.

Beier, S., Tappu, R., and Huson, D. H. (2017). Functional analysis in metagenomics using MEGAN 6. In *Functional metagenomics: Tools and applications*, pages 65–74. Springer.

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., and Wheeler, D. L. (2005). Genbank. *Nucleic acids research*, **33**(suppl_1), D34–D38.

Bernard, G., Chan, C. X., and Ragan, M. A. (2016). Alignment-free microbial phylogenomics under scenarios of sequence divergence, genome rearrangement and lateral genetic transfer. *Scientific reports*, **6**(1), 1–12.

Blaser, M. J. *et al.* (2014). The microbiome revolution. *The Journal of clinical investigation*, **124**(10), 4162–4165.

Bloom, B. H. (1970). Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, **13**(7), 422–426.

Boolchandani, M., D'Souza, A. W., and Dantas, G. (2019). Sequencing-based methods and resources to study antimicrobial resistance. *Nature Reviews Genetics*, **20**(6), 356–370.

Bowers, R. M., Kyrpides, N. C., Stepanauskas, R., Harmon-Smith, M., Doud, D., Reddy, T., Schulz, F., Jarett, J., Rivers, A. R., Eloe-Fadrosh, E. A., *et al.* (2017). Minimum information about a single amplified genome (MISAG) and a metagenome-assembled genome (MIMAG) of bacteria and archaea. *Nature biotechnology*, **35**(8), 725–731.

Breitwieser, F. P., Lu, J., and Salzberg, S. L. (2019). A review of methods and databases for metagenomic classification and assembly. *Briefings in bioinformatics*, **20**(4), 1125–1136.

Buchfink, B., Xie, C., and Huson, D. (2015). Fast and sensitive protein alignment using DIAMOND. *Nature Methods*, **12**, 59–60.

Chaumeil, P.-A., Mussig, A. J., Hugenholtz, P., and Parks, D. H. (2020). GTDB-Tk: a toolkit to classify genomes with the genome taxonomy database.

Chen, L., Yang, J., Yu, J., Yao, Z., Sun, L., Shen, Y., and Jin, Q. (2005). VFDB: a reference database for bacterial virulence factors. *Nucleic acids research*, **33**(suppl_1), D325–D328.

Dos Santos, D. F. K., Istvan, P., Quirino, B. F., and Kruger, R. H. (2017). Functional metagenomics as a tool for identification of new antibiotic resistance genes from natural environments. *Microbial ecology*, **73**(2), 479–491.

Douglas, G. M., Beiko, R. G., and Langille, M. G. (2018). Predicting the functional potential of the microbiome from marker genes using picrust. In *Microbiome analysis*, pages 169–177. Springer.

Finn, R. D., Tate, J., Mistry, J., Coggill, P. C., Sammut, S. J., Hotz, H.-R., Ceric, G., Forslund, K., Eddy, S. R., Sonnhammer, E. L. L., and Bateman, A. (2008). The Pfam protein families database. *Nucleic Acids Res*, **36**, D281–D288.

Frank, J. A., Pan, Y., Tooming-Klunderud, A., Eijsink, V. G., McHardy, A. C., Nederbragt, A. J., and Pope, P. B. (2016). Improved metagenome assemblies and taxonomic binning using long-read circular consensus sequence data. *Scientific reports*, **6**(1), 1–10.

Franzosa, E. A., McIver, L. J., Rahnavard, G., Thompson, L. R., Schirmer, M., Weingart, G., Lipson, K. S., Knight, R., Caporaso, J. G., Segata, N., *et al.* (2018). Species-level

functional profiling of metagenomes and metatranscriptomes. *Nature methods*, **15**(11), 962–968.

Gautam, A., Felderhoff, H., Bağci, C., and Huson, D. H. (2022). Using AnnoTree to get more assignments, faster, in diamond+ megan microbiome analysis. *Msystems*, **7**(1), e01408–21.

Ghurye, J. S., Cepeda-Espinoza, V., and Pop, M. (2016). Focus: microbiome: metagenomic assembly: overview, challenges and applications. *The Yale journal of biology and medicine*, **89**(3), 353.

Glass, E. M., Wilkening, J., Wilke, A., Antonopoulos, D., and Meyer, F. (2010). Using the metagenomics RAST server (MG–RAST) for analyzing shotgun metagenomes. *Cold Spring Harb Protoc.*, **2010**(1), pdb.prot5368.

Haft, D. H., Selengut, J. D., and White, O. (2003). The TIGRFAMs database of protein families. *Nucleic Acids Res*, **31**(1), 371–373.

Huson, D. H., Auch, A. F., Qi, J., and Schuster, S. C. (2007). MEGAN analysis of metagenomic data. *Genome Res*, **17**(3), 377–386.

Huson, D. H., Albrecht, B., Bağcı, C., Bessarab, I., Gorska, A., Jolic, D., and Williams, R. B. (2018). MEGAN-LR: new algorithms allow accurate binning and easy interactive exploration of metagenomic long reads and contigs. *Biology direct*, **13**(1), 1–17.

Hutchings, M. I., Truman, A. W., and Wilkinson, B. (2019). Antibiotics: past, present and future. *Current opinion in microbiology*, **51**, 72–80.

Kanehisa, M. and Goto, S. (2000). KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res*, **28**(1), 27–30.

Kang, D. D., Li, F., Kirton, E., Thomas, A., Egan, R., An, H., and Wang, Z. (2019). MetaBAT 2: an adaptive binning algorithm for robust and efficient genome reconstruction from metagenome assemblies. *PeerJ*, **7**, e7359.

Kielbasa, S. M., Wan, R., Sato, K., Horton, P., and Frith, M. C. (2011). Adaptive seeds tame genomic sequence comparison. *Genome Research*, **21**(3), 487–493.

Kim, D., Song, L., Breitwieser, F. P., and Salzberg, S. L. (2016). Centrifuge: rapid and sensitive classification of metagenomic sequences. *Genome Research*, **26**, 1721–1729.

Kolmogorov, M., Bickhart, D. M., Behsaz, B., Gurevich, A., Rayko, M., Shin, S. B., Kuhn, K., Yuan, J., Polevikov, E., Smith, T. P., *et al.* (2020). metaflye: scalable long-read metagenome assembly using repeat graphs. *Nature Methods*, **17**(11), 1103–1110.

Koren, S., Walenz, B. P., Berlin, K., Miller, J. R., Bergman, N. H., and Phillippy, A. M. (2017). Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation. *Genome research*, **27**(5), 722–736.

Lagier, J.-C., Edouard, S., Pagnier, I., Mediannikov, O., Drancourt, M., and Raoult, D. (2015). Current and past strategies for bacterial culture in clinical microbiology. *Clinical microbiology reviews*, **28**(1), 208–236.

Liu, Y. and Schmidt, B. (2012). Long read alignment based on maximal exact match seeds. *Bioinformatics*, **28**(18), i318–i324.

Magi, A., Semeraro, R., Mingrino, A., Giusti, B., and D'aurizio, R. (2018). Nanopore sequencing data analysis: state of the art, applications and challenges. *Briefings in bioinformatics*, **19**(6), 1256–1272.

Mallawaarachchi, V., Wickramarachchi, A., and Lin, Y. (2020). Graphbin: refined binning of metagenomic contigs using assembly graphs. *Bioinformatics*, **36**(11), 3307–3313.

Mardis, E. R. (2008). The impact of next-generation sequencing technology on genetics. *Trends in genetics*, **24**(3), 133–141.

Mendler, K., Chen, H., Parks, D. H., Lobb, B., Hug, L. A., and Doxey, A. C. (2019). AnnoTree: visualization and exploration of a functionally annotated microbial tree of life. *Nucleic acids research*, **47**(9), 4442–4448.

Menzel, P., Ng, K. L., and Krogh, A. (2016). Fast and sensitive taxonomic classification for metagenomics with Kaiju. *Nature Communications*, **7**, 11257.

Meyer, F., Fritz, A., Deng, Z.-L., Koslicki, D., Lesker, T. R., Gurevich, A., Robertson, G., Alser, M., Antipov, D., Beghini, F., *et al.* (2022). Critical assessment of metagenome interpretation: the second round of challenges. *Nature methods*, **19**(4), 429–440.

Mikheyev, A. S. and Tin, M. M. (2014). A first look at the oxford nanopore minion sequencer. *Molecular ecology resources*, **14**(6), 1097–1102.

Mizrahi, I. and Jami, E. (2018). The compositional variation of the rumen microbiome and its effect on host performance and methane emission. *Animal*, **12**(s2), s220–s232.

Nagarajan, N. and Pop, M. (2013). Sequence assembly demystified. *Nature Reviews Genetics*, **14**(3), 157–167.

Nasko, D. J., Koren, S., Phillippy, A. M., and Treangen, T. J. (2018). RefSeq database growth influences the accuracy of k-mer-based lowest common ancestor species identification. *Genome biology*, **19**(1), 1–10.

Nicholls, S. M., Quick, J. C., Tang, S., and Loman, N. J. (2019). Ultra-deep, long-read nanopore sequencing of mock microbial community standards. *Gigascience*, **8**(5), giz043.

Nissen, J. N., Johansen, J., Allesøe, R. L., Sønderby, C. K., Armenteros, J. J. A., Grønbech, C. H., Jensen, L. J., Nielsen, H. B., Petersen, T. N., Winther, O., *et al.* (2021). Improved metagenome binning and assembly using deep variational autoencoders. *Nature biotechnology*, **39**(5), 555–560.

O'Leary, N. A., Wright, M. W., Brister, J. R., Ciufo, S., Haddad, D., McVeigh, R., Rajput, B., Robbertse, B., Smith-White, B., Ako-Adjei, D., *et al.* (2016). Reference sequence (RefSeq) database at ncbi: current status, taxonomic expansion, and functional annotation. *Nucleic acids research*, **44**(D1), D733–D745.

Ondov, B. D., Treangen, T. J., Melsted, P., Mallonee, A. B., Bergman, N. H., Koren, S., and Phillippy, A. M. (2016). Mash: fast genome and metagenome distance estimation using minhash. *Genome biology*, **17**(1), 1–14.

Padmanabhan, R., Mishra, A. K., Raoult, D., and Fournier, P.-E. (2013). Genomics and metagenomics in medical microbiology. *Journal of microbiological methods*, **95**(3), 415–424.

Pallen, M. J., Rodriguez-R, L. M., and Alikhan, N.-F. (2022). Naming the unnamed: over 65,000 Candidatus names for unnamed Archaea and Bacteria in the Genome Taxonomy Database. *International Journal of Systematic and Evolutionary Microbiology*, **72**(9), 005482.

Parker, J., Helmstetter, A. J., Devey, D., Wilkinson, T., and Papadopulos, A. S. (2017). Field-based species identification of closely-related plants using real-time nanopore sequencing. *Scientific reports*, **7**(1), 1–8.

Parks, D. H., Imelfort, M., Skennerton, C. T., Hugenholtz, P., and Tyson, G. W. (2015). CheckM: assessing the quality of microbial genomes recovered from isolates, single cells, and metagenomes. *Genome research*, **25**(7), 1043–1055.

Parks, D. H., Chuvochina, M., Waite, D. W., Rinke, C., Skarshewski, A., Chaumeil, P.-A., and Hugenholtz, P. (2018). A standardized bacterial taxonomy based on genome phylogeny substantially revises the tree of life. *Nature biotechnology*, **36**(10), 996–1004.

Pendleton, K. M., Erb-Downward, J. R., Bao, Y., Branton, W. R., Falkowski, N. R., Newton, D. W., Huffnagle, G. B., and Dickson, R. P. (2017). Rapid pathogen identification in bacterial pneumonia using real-time metagenomics. *American journal of respiratory and critical care medicine*, **196**(12), 1610–1612.

Petersen, L. M., Martin, I. W., Moschetti, W. E., Kershaw, C. M., and Tsongalis, G. J. (2019). Third-generation sequencing in the clinical laboratory: exploring the advantages and challenges of nanopore sequencing. *Journal of clinical microbiology*, **58**(1), e01315–19.

Poretsky, R., Rodriguez-R, L. M., Luo, C., Tsementzi, D., and Konstantinidis, K. T. (2014). Strengths and limitations of 16s rrna gene amplicon sequencing in revealing temporal microbial community dynamics. *PloS one*, **9**(4), e93827.

Prakash, T. and Taylor, T. D. (2012). Functional assignment of metagenomic data: challenges and applications. *Briefings in bioinformatics*, **13**(6), 711–727.

Qin, J., Li, R., Raes, J., Arumugam, M., Burgdorf, K. S., Manichanh, C., Nielsen, T., Pons, N., Levenez, F., Yamada, T., *et al.* (2010). A human gut microbial gene catalogue established by metagenomic sequencing. *nature*, **464**(7285), 59–65.

Ranjan, R., Rani, A., Metwally, A., McGee, H. S., and Perkins, D. L. (2016). Analysis of the microbiome: Advantages of whole genome shotgun versus 16s amplicon sequencing. *Biochemical and biophysical research communications*, **469**(4), 967–977.

Ren, J., Bai, X., Lu, Y. Y., Tang, K., Wang, Y., Reinert, G., and Sun, F. (2018). Alignment-free sequence analysis and applications. *Annual Review of Biomedical Data Science*, **1**, 93.

Riesenfeld, C. S., Schloss, P. D., and Handelsman, J. (2004). Metagenomics: genomic analysis of microbial communities. *Annu. Rev. Genet.*, **38**, 525–552.

Rinke, C., Schwientek, P., Sczyrba, A., Ivanova, N. N., Anderson, I. J., Cheng, J.-F., Darling, A., Malfatti, S., Swan, B. K., Gies, E. A., *et al.* (2013). Insights into the phylogeny and coding potential of microbial dark matter. *Nature*, **499**(7459), 431–437.

Schmeisser, C., Steele, H., and Streit, W. R. (2007). Metagenomics, biotechnology with non-culturable microbes. *Applied microbiology and biotechnology*, **75**(5), 955–962.

Schwabe, R. F. and Jobin, C. (2013). The microbiome and cancer. *Nature Reviews Cancer*, **13**(11), 800–812.

Sczyrba, A., Hofmann, P., Belmann, P., Koslicki, D., Janssen, S., Dröge, J., Gregor, I., Majda, S., Fiedler, J., Dahms, E., *et al.* (2017). Critical assessment of metagenome interpretation—a benchmark of metagenomics software. *Nature methods*, **14**(11), 1063–1071.

Seemann, T. (2014). Prokka: rapid prokaryotic genome annotation. *Bioinformatics*, **30**(14), 2068–2069.

Segata, N., Waldron, L., Ballarini, A., Narasimhan, V., Jousson, O., and Huttenhower, C. (2012). Metagenomic microbial community profiling using unique clade-specific marker genes. *Nat Meth*, **9**(8), 811–814.

Sheetlin, S. L., Park, Y., Frith, M. C., and Spouge, J. L. (2014). Frameshift alignment: statistics and post-genomic applications. *Bioinformatics*, **30**(24), 3575–3582.

Shen, B., Du, L., Sanchez, C., Edwards, D., Chen, M., and Murrell, J. (2001). The biosynthetic gene cluster for the anticancer drug bleomycin from streptomyces verticillus atcc15003 as a model for hybrid peptide–polyketide natural product biosynthesis. *Journal of Industrial Microbiology and Biotechnology*, **27**(6), 378–385.

Simon, C. and Daniel, R. (2011). Metagenomic analyses: past and future trends. *Applied and environmental microbiology*, **77**(4), 1153–1161.

Simon, H. Y., Siddle, K. J., Park, D. J., and Sabeti, P. C. (2019). Benchmarking metagenomics tools for taxonomic classification. *Cell*, **178**(4), 779–794.

Singleton, C. M., Petriglieri, F., Kristensen, J. M., Kirkegaard, R. H., Michaelsen, T. Y., Andersen, M. H., Kondrotaite, Z., Karst, S. M., Dueholm, M. S., Nielsen, P. H., *et al.* (2021). Connecting structure to function with the recovery of over 1000 high-quality metagenome-assembled genomes from activated sludge using long-read sequencing. *Nature communications*, **12**(1), 1–13.

Sunagawa, S., Mende, D. R., Zeller, G., Izquierdo-Carrasco, F., Berger, S. A., Kultima, J. R., Coelho, L. P., Arumugam, M., Tap, J., Nielsen, H. B., *et al.* (2013). Metagenomic species profiling using universal phylogenetic marker genes. *Nature methods*, **10**(12), 1196–1199.

Thomas, T., Gilbert, J., and Meyer, F. (2012). Metagenomics-a guide from sampling to data analysis. *Microbial informatics and experimentation*, **2**(1), 1–12.

Tyson, G. W., Chapman, J., Hugenholtz, P., Allen, E. E., Ram, R. J., Richardson, P. M., Solovyev, V. V., Rubin, E. M., Rokhsar, D. S., and Banfield, J. F. (2004). Community structure and metabolism through reconstruction of microbial genomes from the environment. *Nature*, **428**(6978), 37–43.

Vandenkoornhuyse, P., Quaiser, A., Duhamel, M., Le Van, A., and Dufresne, A. (2015). The importance of the microbiome of the plant holobiont. *New Phytologist*, **206**(4), 1196–1206.

Wagner, M., Loy, A., Nogueira, R., Purkhold, U., Lee, N., and Daims, H. (2002). Microbial community composition and function in wastewater treatment plants. *Antonie Van Leeuwenhoek*, **81**(1), 665–680.

Walker, B. J., Abeel, T., Shea, T., Priest, M., Abouelliel, A., Sakthikumar, S., Cuomo, C. A., Zeng, Q., Wortman, J., Young, S. K., *et al.* (2014). Pilon: an integrated tool for comprehensive microbial variant detection and genome assembly improvement. *PloS one*, **9**(11), e112963.

Wang, Y., Zhao, Y., Bollas, A., Wang, Y., and Au, K. F. (2021). Nanopore sequencing technology, bioinformatics and applications. *Nature biotechnology*, **39**(11), 1348–1365.

Wick, R. R., Judd, L. M., Gorrie, C. L., and Holt, K. E. (2017). Unicycler: resolving bacterial genome assemblies from short and long sequencing reads. *PLoS computational biology*, **13**(6), e1005595.

Wood, D. E. and Salzberg, S. L. (2014). Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome Biology*, **15**, R46.

Wood, D. E., Lu, J., and Langmead, B. (2019). Improved metagenomic analysis with Kraken 2. *Genome biology*, **20**(1), 1–13.

Appendices

In the following Appendices, all of the publications listed in the Chapter 3 ”Publications”, under ”Peer reviewed publications included in this dissertation”, are given as an appendix in the same order as they appear in Chapter 3.

Appendix I

Huson *et al. Biology Direct* (2018) 13:6
https://doi.org/10.1186/s13062-018-0208-7

Biology Direct

RESEARCH Open Access

MEGAN-LR: new algorithms allow accurate binning and easy interactive exploration of metagenomic long reads and contigs

Daniel H. Huson[1,2*], Benjamin Albrecht[1], Caner Bağcı[1,5], Irina Bessarab[4], Anna Górska[1,5], Dino Jolic[3,5] and Rohan B. H. Williams[4]

Abstract

Background: There are numerous computational tools for taxonomic or functional analysis of microbiome samples, optimized to run on hundreds of millions of short, high quality sequencing reads. Programs such as MEGAN allow the user to interactively navigate these large datasets. Long read sequencing technologies continue to improve and produce increasing numbers of longer reads (of varying lengths in the range of 10k-1M bps, say), but of low quality. There is an increasing interest in using long reads in microbiome sequencing, and there is a need to adapt short read tools to long read datasets.

Methods: We describe a new LCA-based algorithm for taxonomic binning, and an interval-tree based algorithm for functional binning, that are explicitly designed for long reads and assembled contigs. We provide a new interactive tool for investigating the alignment of long reads against reference sequences. For taxonomic and functional binning, we propose to use LAST to compare long reads against the NCBI-nr protein reference database so as to obtain frame-shift aware alignments, and then to process the results using our new methods.

Results: All presented methods are implemented in the open source edition of MEGAN, and we refer to this new extension as MEGAN-LR (MEGAN long read). We evaluate the LAST+MEGAN-LR approach in a simulation study, and on a number of mock community datasets consisting of Nanopore reads, PacBio reads and assembled PacBio reads. We also illustrate the practical application on a Nanopore dataset that we sequenced from an anammox bio-rector community.

Reviewers: This article was reviewed by Nicola Segata together with Moreno Zolfo, Pete James Lockhart and Serghei Mangul.

Conclusion: This work extends the applicability of the widely-used metagenomic analysis software MEGAN to long reads. Our study suggests that the presented LAST+MEGAN-LR pipeline is sufficiently fast and accurate.

Keywords: Microbiome, Long reads, Sequence analysis, Taxonomic binning, Functional binning, Algorithms, Software, Nanopore, PacBio

*Correspondence: daniel.huson@uni-tuebingen.de
[1]Center for Bioinformatics, University of Tübingen, Sand 14, 72076 Tübingen, Germany
[2]Life Sciences Institute, National University of Singapore, 28 Medical Drive, Singapore 117456, Singapore
Full list of author information is available at the end of the article

Background

There are numerous computational tools for taxonomic or functional binning or profiling of microbiome samples, optimized to run on hundreds of millions of short, high quality sequencing reads [1–4]. Alignment-based taxonomic binning of reads is often performed using the naïve LCA algorithm [5], because it is fast and its results are easy to interpret. Functional binning of reads usually involves a best-hit strategy to assign reads to functional classes.

Software or websites for analyzing microbiome shotgun sequencing samples usually provide some level of interactivity, such as MG-RAST [2]. The interactive microbiome analysis tool MEGAN, which was first used in 2006 [6], is explicitly designed to enable users to interactively explore large numbers of microbiome samples containing hundreds of millions of short reads [1].

Illumina HiSeq and MiSeq sequencers allow researchers to generate sequencing data on a huge scale, so as to analyze many samples at a great sequencing depth [7–9]. A wide range of questions, in particular involving the presence or absence of particular organisms or genes in a sample, can be answered using such data. However, there are interesting problems that are not easily resolved using short reads. For example, it is often very difficult to determine whether two genes that are detected in the same microbiome sample also belong to the same *genome*, even if they are located close to each other in the genome, despite the use of metagenomic assembly in combination with contig binning techniques and paired-end reads [10].

Current long read sequencing technologies, such as provided by Oxford Nanopore Technologies (ONT) or Pacific Biosciences (PacBio), produce smaller numbers (in the range of hundreds of thousands) of longer reads (of varying lengths in the range of 10 kb – 300 kb, say) of lower quality (error rates around 10%) [11, 12]. There is increasing interest in using long reads in microbiome sequencing and there is a need to adapt short read tools to long read datasets. There are a number of tools that are applicable to long reads, such as WIMP [13], Centrifuge [14] or Kaiju [15]. While the two former are based on comparing against DNA references, the latter can also use a protein reference database.

In this paper, we focus on protein-alignment-based approaches. One reason for this is that existing DNA reference databases cover only a small fraction of the genome sequences believed to be present in the environment [16], although much work has been done on sequencing human-associated microbes [17]. This problem can be ameliorated, to a degree, by using protein alignments, because amino acid sequences are more conserved than DNA sequences. Moreover, work on bacterial pangenomes suggest that the association between species level taxonomic assignment and coding gene content can be weak [18]. Finally, questions going beyond taxonomic profiling and correlation studies will usually require knowledge of the functional content.

Here we present a new classification pipeline for taxonomic and functional analysis of long reads and contigs, based on protein alignments. The pipeline, LAST+MEGAN-LR, consists of first running the alignment tool LAST and then processing the resulting DNA-to-protein alignments using new algorithms provided in MEGAN-LR. We perform a simulation study to evaluate the performance of the method in the context of the taxonomic assignment and compare it with Kaiju, one of the few other tools that use protein references. We also investigate the performance of the pipeline using mock-community datasets and illustrate its application on Nanopore reads sequenced from an anammox enrichment bio-rector.

Methods

Long read taxonomic binning

The naïve LCA (lowest common ancestor) algorithm is widely used for binning short reads onto the nodes of a given taxonomy (such as the NCBI taxonomy), based on alignments [5]. Consider a read r that has significant alignments $a_1, \ldots, a_k$ to reference sequences associated with taxa $t_1, \ldots, t_k$. The naïve LCA assigns r to the lowest taxonomic node that lies above the set of all nodes representing $t_1, \ldots, t_k$. The set of *significant* alignments is defined to consist of those alignments whose score lies close to the best score achieved for the given read, defined, say, as those that have a bit score that lies within 10% of the best bit score.

The naïve LCA algorithm is fast, easy to implement and the results are easy to interpret. When applied to protein alignments, an implicit assumption of the algorithm is that any read aligns to only one gene and so all associated taxa are "competing" for the same gene; this justifies the above definition of significant alignments. While reads that are only a few hundred base pairs long usually fulfill this assumption, longer reads or assembled contigs often overlap with more than one gene and so the naïve algorithm is not suitable for them.

To make the naïve algorithm applicable to protein alignments on a long read or contig r, a simple idea is to first determine "conserved genes" as regions along the read where alignments accumulate. The second step is to apply the naïve LCA to each of these regions individually. The placement of the read is finally determined using the LCA of all these gene-based LCAs. There are two problems here. First, because protein alignments around the same location can have quite different lengths, delineating different "conserved genes" can be difficult in practice. Second, because a large proportion of genes on a long read or contig may be conserved to different extents across different taxonomic groups, the placement

of the read will often be to a high-level (or "unspecific") taxon.

To address these issues, we present a new taxonomic binning for long reads that we call the *interval-union LCA* algorithm. This algorithm processes each read r in turn, in two steps. First, the read is partitioned into a set of intervals $v_1, \ldots, v_m$ that have the property that every alignment associated with r starts and ends at the beginning or end of some interval, respectively. In other words, a new interval starts wherever some alignment begins or ends. We say that an alignment a_i is *significant* on an interval v_j, if its bit score lies within 10% (by default) of the best bit score seen for any alignment that covers v_j. In MEGAN-LR this threshold is referred to as the `topPercent` parameter.

In the second step, for each taxon t that is associated with any of the alignments, let $I(t)$ denote the union of all intervals for which there exists some significant alignment a_i associated with taxon t. In a post-order traversal, for each higher-rank taxonomic node s we compute $I(s)$ as the union of the intervals covered by the children of s. In result, every node of the taxonomy is labeled by a set of intervals. Note that, during the computation of the union of interval sets, we merge any overlapping intervals into a single interval.

The read r is then placed on the taxon s that has the property that its set of intervals $I(s)$ covers 80% (by default) of the total aligned or covered portion of the read, while none of its children does (see Fig. 1). In MEGAN-LR this threshold is referred to as the `percentToCover` parameter. Note that it is possible that there are multiple nodes that have this property, in which case the read is assigned to the LCA of all such nodes.

Long read functional binning and annotation

Functional binning of short reads is usually performed by assigning each read to a class in a functional classification system such as InterPro [19], eggNOG [20] or KEGG [21], based on its alignments.

This is often done using a simple *best-hit* strategy, as follows. For a short read r, let a denote the highest-scoring alignment of r to a reference protein for which the functional class c is known. Assign r to the functional class c. For example, c might be an InterPro family or an eggNOG cluster. In short read analysis, each read is assigned to at most one class in any given functional classification. Many reads remain unclassified because all the reference proteins that they align to are unclassified.

A long read may contain multiple genes, and for each gene, there may be many alignments involving different taxa. To avoid redundancy in functional assignments when processing alignments between the long read and different taxa, we consider the "dominance" of individual alignments (as defined below).

Let r be a long read and let $a_1, \ldots, a_k$ be a set of DNA-to-protein alignments from r to a suitable protein reference sequences. Note that this set will often include alignments between the read and the same homologue in different taxa.

To reduce the number of redundant functional classes associated with r, we introduce the following concept. We say that an alignment a_i *dominates* an alignment a_j, if (1)

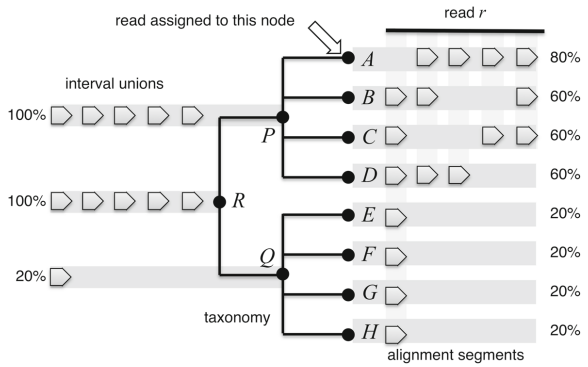


Fig. 1 To illustrate the interval-union LCA algorithm, here we show eight hypothetical species $A, B, \ldots, H$ separated into two genera, P and Q, belonging to the same family R. Alignments from the read r to proteins associated with the species are indicated by arrows on the right and cover between 80% (for A) and 20% (for H) of the aligned read. Using arrows, on the left we depict the sets of intervals computed for nodes P, Q, R as the union of the sets of intervals of the children of each node. Nodes R and P each cover 100% of the aligned read. The read r is placed on A as it is the lowest taxonomic node with $\geq$80% coverage. Note that, if A only covered 60% of the aligned read, then the read would be assigned to the higher taxon P (and this would remain the case even if one of the taxa below Q had 60% coverage)

a_i covers more than 50% of the read that is covered by a_j, (2) if the bit score of a_i is greater than that of a_j, and (3) both alignments lie on the same strand of r. Optionally, one might also require that the taxonomic identity of each protein reference sequence under consideration is compatible with the taxonomic bin assigned to the read r.

The set of functional classes associated with a long read r is then given by the functional classes associated with those alignments of r that are not dominated by some other alignment of r. Each read can be binned to all functional classes associated with it. Moreover, the set of associated classes can be used to provide simple, functional annotation of the read or contig.

To exploit that latter, we provide a dialog for exporting taxonomic and functional annotations in GFF3 format. It can be applied to any selection of taxonomic or functional classification nodes, or to a set of selected reads in the new *long read inspector*, which is described in more detail below. The user chooses a classification, and then each alignment to a reference sequence associated with that classification is exported as a CDS item. By default, only those alignments that are not dominated by another alignment are exported. In addition, the user can decide to export only those items for which the taxon associated with the corresponding reference sequence is compatible with the taxon assigned to the read.

Reporting counts

In taxonomic or functional binning of short reads, it usually suffices to report the number of reads assigned to a specific classification node, because all reads are of a very similar length and all alignments have much the same length as the reads. For long reads or contigs, the lengths and alignment coverage can vary widely. Moreover, the number of reads contained in a contig, or contig coverage, is an additional factor to be considered. To address this, in MEGAN-LR each node can be labeled by one of the following:

1. the number of reads assigned,
2. the total length of all reads assigned,
3. the total number of aligned bases of all reads assigned, or
4. in the case of contigs, the total number of reads contained in all assigned contigs.

For long reads, by default, MEGAN–LR reports (3), the number of aligned bases, rather than (2), as this down-weights any long stretches of unaligned sequence. In addition, we use this value to determine the minimum support required for a taxon to be reported. By default, a taxon is only reported if it obtains at least 0.05% of all aligned bases. In MEGAN-LR, this is called the `minSupport` parameter. If the number of aligned bases assigned to a taxon t does not meet this threshold, then the assigned bases are pushed up the taxonomy until a taxon is reached that has enough aligned bases to be reported.

Long read alignment

In this paper, we focus on taxonomic and functional binning of long reads using DNA-to-protein alignments. Currently long read sequencing technologies (Oxford Nanopore and PacBio) exhibit high rates of erroneous insertions and deletions [11, 12]. Consequently, programs such as BLASTX [22] are not suitable for such reads as they cannot handle frame-shifts.

The LAST program [23, 24] uses a frame-shift aware algorithm to align DNA to proteins and produces long protein alignments on long reads, even in the presence of many frame-shifts. Initial indexing of the NCBI–nr database (containing over 100 million sequences) by LAST takes over one day on a server. However, once completed, alignment of reads against the NCBI-nr database using the index is fast; the alignment of Nanopore reads takes roughly one hour per gigabase on a server.

The DIAMOND program [25] is widely used in microbiome analysis to compute alignments of short metagenomic reads against a protein reference database such as NCBI–nr. A new frame-shift aware alignment mode is currently under development and DIAMOND will provide an alternative to LAST in the future.

Long read analysis

LAST produces output in a simple text–based multiple alignment format (MAF). For performance reasons, LAST processes all queries and all reference sequences in batches and alignments associated with a given query are not reported consecutively, but rather in batches.

In addition, the size of a MAF file is often very large and subsequent sorting and parsing of alignments can be time consuming. To address these issues, we have implemented a new program called "MAF2DAA" that takes MAF format as input, either as a file or piped directly from LAST, and produces a DAA ("Diamond alignment archive") file as output [25]. The program processes the input in chunks, first filtering and compressing each chunk of data on-the-fly, and then interleaving and filtering the results into a single DAA file that contains all reads with their associated alignments. During filtering, MAF2DAA removes all alignments that are *strongly dominated* by some other alignment, to reduce a large number of redundant alignments.

In more detail, for a given read r, we say that an alignment a of r *strongly dominates* an alignment b for r, if it covers most of b (by default, we require 90% coverage) and if its bit score is significantly larger (by default, we require that $0.9 \times \text{bitscore}(a) > \text{bitscore}(b)$).

A DAA file obtained in this way can then be processed by MEGAN's Meganizer program that performs taxonomic and functional binning, and indexing, of all reads in the DAA file. This program does not produce a new file but appends the results to the end of the DAA file, and any such "meganized" DAA file can be directly opened in MEGAN for interactive analysis. We have modified MEGAN so that it supports frame-shift containing alignments. The final DAA file is usually around ten times smaller than the MAF file produced by LAST.

Long read visualization

Interactive analysis tools for short read microbiome sequencing data usually focus on representing the taxonomic and functional classifications systems used for binning or profiling the reads, for example reporting the number of reads assigned to each class. In addition, some tools provide a reference-centric visualization that displays how the reads align against a given reference sequence. However, visualizations of the short reads themselves are usually not provided.

For long read or contigs, there is a need for visualization techniques that make it easy to explore the taxonomic and functional identity of reference sequences to which the reads align. To address this, we have designed and implemented a *long read inspector* (using JavaFX) that allows one to investigate all long reads assigned to a given taxonomic or functional class (see Fig. 2).

In this tool, each long read or contig r is represented by a horizontal line and all corresponding aligned reference sequences are shown as arrows above (forward strand alignments) or below (reverse strand alignments) the line. The user can select which annotations to display in the view. For example, if the user requests Taxonomy and InterPro annotations, then all reference sequences will be labeled by the associated taxonomic and InterPro classes. The user can search for functional attributes in all loaded reads.

Let a be an arrow representing an alignment of r to a reference sequence associated with taxon s. We use a hierarchical coloring scheme to color such arrows. Initially, we implicitly assign a color index to each taxon, e.g., using the hash code of the taxon name. For each arrow a with associated reference taxon s we distinguish between three different cases. First, if $s = t$, then we use the color assigned to t to color a. Second, if s is a descendant of t, then t has a unique child u that lies on the path from t down to s and we use the color of u to color a. Otherwise, we color a gray to indicate that the taxon associated with a is either less specific or incompatible with t.

For example, if a read r is assigned to the genus *Candidatus Brocadia* and has an alignment to the strain *Candidatus Brocadia sinica JPN1*, then we color the corresponding arrow a using the color that represents the species *Candidatus Brocadia sinica*.

This is a useful strategy when used in combination with the taxonomic binning procedure described above: a read

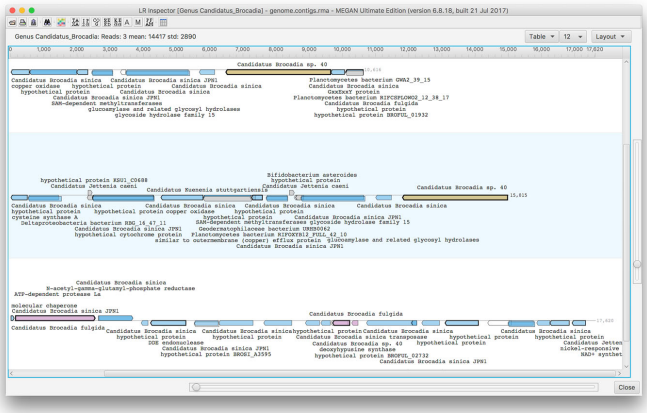


Fig. 2 This screen shot of the MEGAN-LR long read inspector shows three contigs assigned to the genus *Candidatus Brocadia*, with alignments to more specific taxa. Alignments to reference protein sequences are shown as arrows, colored by species of the references; blue for *Candidatus Brocadia sinica*, brown for *Candidatus Brocadia sp. 40* and pink for *Candidatus Brocadia fulgida*. Alignments are labeled by taxonomic and functional classes associated with the corresponding reference proteins

r is binned to the lowest taxon *t* that covers 80% (by default) of the aligned read and the taxonomy-based coloring makes it easy to see how the different taxonomic classes below *t* contribute. For example, if all arrows on one half of the read have one color and all arrows on the other half have some other color, then this may indicate a chimeric read or misassembled contig.

As discussed above, an alternative approach is to export reads and their alignments in GFF3 format and then to use a genome browser such as IGB [26] to explore them (see Fig. 3).

LAST+MEGAN-LR

In summary, we propose to use the following pipeline to analyze metagenomic long reads and contigs (see Fig. 4):

- Align all reads against a protein reference database (such as NCBI-nr) using LAST, producing MAF output.
- Either pipe the output of LAST directly to MAF2DAA, or apply MAF2DAA to the MAF file generated by LAST, to obtain a much smaller output file in DAA format.
- Meganize the DAA file either using the Meganizer command-line tool or interactively in MEGAN.
- Open the meganized DAA file in MEGAN for interactive exploration using the long-read inspector. Export annotated reads in GFF3 format for further investigation, e.g. using a genome browser such as IGB [26] or Artemis [27].

Nanopore sequencing

To obtain a Nanopore dataset, we sequenced the genomic DNA of the Microbial Mock Community B (even, high concentration, catalog nr. HM-276D, BEI Resources). Library preparation was performed using a Low Input by PCR Genomic Sequencing Kit SQK-MAP006 (Oxford Nanopore Technologies, Oxford, UK) for 2D sequencing. Briefly, 100 ng of genomic DNA was sheared in a Covaris g-TUBE (Covaris, Inc., Woburn, MA, USA) at 6000 rpm, treated with PreCR (New England Biolabs, Ipswich, MA, USA) and used as input for adapter ligation according to the ONT protocol. Adapter-ligated DNA was further amplified with the LongAmp Taq 2X Master Mix (NEB) using the following program: 95°C 3 min; 18 cycles of 95°C 15 sec, 62°C 15 sec, 65°C 10 min; 65°C 20 min. Sequencing was performed using an early access MinION device (ONT) on a FLO-MAP003 flowcell (ONT). Raw fast5 files were obtained with MinKNOW (v0.50.2.15, ONT) using a 48 h genomic sequencing protocol, basecalled with ONT's proprietary Metrichor cloud-based basecalling service and the 2D Basecalling for SQK-MAP006 v1.34 workflow.

Genomic DNA from the lab scale Anammox enrichment reactor described in Liu et al. [28] was extracted using the FastDNA SPIN Kit for Soil with 4x homogenization on the FastPrep instrument (MP Bio). The DNA was further purified using Genomic DNA Clean and Concentrator -10 Kit (Zymo Research). Approximately 1700 ng of extracted DNA was used for library preparation using a Ligation Sequencing Kit SQK-LSK108 (Oxford Nanopore Technologies, Oxford, UK) for 1D sequencing according to the manufacturer protocol. Sequencing was performed using an early access MinION device (ONT) on a SpotON FLO-MIN106 flowcell (R9.4). The run was stopped after 22 h due to low number of active pores. Fast5 files were obtained with MinKNOW (v1.3.30, ONT) using a 48 h genomic sequencing protocol. Basecalling was performed using Metrichor (Instance ID:135935, 1D Basecalling for FLO-MIN106 450 bps_RNN (rev.1.121)).

Parameters

The MEGAN-LR approach employs a number of different user-specified parameters. The main effect of changing

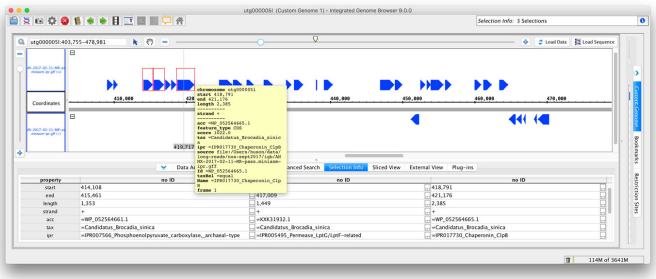


Fig. 3 Example of long read data exported from MEGAN-LR and imported into the IGB genome browser [26]

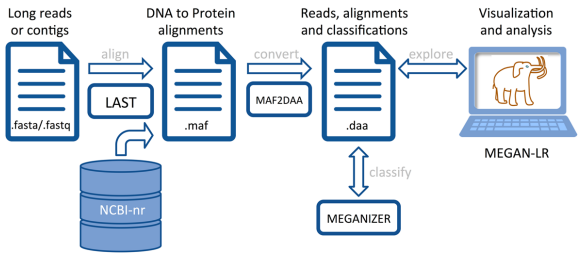


Fig. 4 The LAST+MEGAN-LR pipeline. Long reads or contigs are aligned against the NCBI-nr database using LAST and the resulting MAF file (multiple alignment format) is converted to DAA format (Diamond alignment format), including filtering of dominated alignments. Taxonomic and functional binning of the reads or contigs is then performed using the Meganizer program and the results are appended to the DAA file. The meganized DAA file can then opened and interactively analyzed in MEGAN-LR

any of these is usually a shift in the trade-off between false positive and false negative taxonomic assignments. What balance of false positives and false negatives is ideal depends on the biological question at hand, and so the parameters may have to be adjusted by the user.

The `minSupport` parameter (default setting 0.05%) sets the "level of detection", that is, it is used to decide whether a taxonomic node has been assigned enough weight (such as number of reads or number of aligned bases, say) so as to appear in the displayed tree. If the threshold is not met, then the weights are pushed up the tree until enough weight has been accumulated. Lowering this threshold will improve sensitivity for low-abundance species while increasing the risk of false positives induced by the erroneous assignment of individual reads, i.e., due to random hits or database errors. Increasing this threshold will decrease false positives while causing more low-abundance taxa to be missed.

The `topPercent` parameter (default value 10%) is used to determine which alignments on the same interval of a read are considered significant. An alignment is only considered significant if its bitscore lies within the given percentage of the bitscore for the best alignment. Setting this threshold too small will result in false positive assignments based on chance differences in alignment score, whereas setting this threshold too large will result in false negatives on lower taxonomic ranks due to assignment to higher taxonomic classes.

The `percentToCover` parameter (default value 80%) influences at what rank of the taxonomy a long read will be placed. Setting this parameter too high or too low will usually result in less specific assignments.

LAST alignment of long reads against the NCBI-nr database can produce very large files due to large numbers of alignments covering the same segment of reads. The concept of strong-domination was developed to address this issue. By default, MEGAN-LR uses a setting of `MinPercentCoverToStronglyDominate` = 90% and `TopPercentScoreToStronglyDominate`=90% to filter reads.

When reporting functional classes of intervals of a long read, a key problem is which alignments to report on. In practice, using all alignments found for a read produces too many redundant gene calls. Here MEGAN-LR uses a parameter `MinPercentCoverToDominate` = 50% to filter the alignments that are reported.

In the "Results" section, we illustrate the effect of varying most of these parameters on the performance of MEGAN-LR on mock community data.

Simulation study

To evaluate the performance of the proposed LAST+MEGAN-LR approach and, in particular, of the interval-union LCA algorithm, we undertook a simulation study to estimate the sensitivity and precision of the algorithm, following the protocol reported in [15], as defined below. We attempted to model two major obstacles in metagenomic studies, namely sequencing errors and the incompleteness of reference databases.

Our simulation study is based on a set P of 4282 prokaryotic genomes from NCBI for which both annotated genomes and annotated sets of proteins are available, downloaded in March 2017. In addition, we identified a subset Q of 1151 genomes that consists of all those organisms in P whose genus contains at least 2 and at most 10 organisms in P, and for which a full taxonomic classification is given. Note that Q can be partitioned into nine different categories, based on the number $2-10$ of organisms in Q that the corresponding genus contains.

For each target species t in Q, we performed the following "leave-one-out" evaluation:

- First, we collected a set of R of 2000 simulated reads from the genome sequence of t using NanoSim [29], a read simulator that produces synthetic reads that

reflect the characteristic base-calling errors of ONT reads, running in linear mode.
- Second, we constructed a protein reference database $D_{\hat{t}}$ that contained all proteins associated with all organisms in P except for t ("leave one out").
- Third, we performed taxonomic binning of all reads in R using LAST+MEGAN-LR as follows. We first build a LAST reference index on $D_{\hat{t}}$, then aligned all reads in R against $D_{\hat{t}}$ using LAST, with a frameshift cost of 15, and then performed taxonomic binning of all reads in MEGAN using the interval-union LCA algorithm (default parameters).
- Fourth, for comparison, we also ran the taxonomic binning program Kaiju[15] on R and $D_{\hat{t}}$, building a custom Kaiju index on $D_{\hat{t}}$. We performed taxonomic binning of simulated reads using Kaiju's greedy mode, with the maximum number of allowed substitutions set to 5.

To be precise, we ran each of the four steps twice to produce two simulation datasets, each containing 2,000 reads per target species. The first dataset was produced using the ecoli_R73_2D (R7.3) simulator profile, whereas the second was produced using the ecoli_R9_2D (R9) profile. Both profiles were downloaded from the NanoSim FTP address (http://ftp.bcgsc.ca/supplementary/NanoSim/) in April 2017. The R7.3 profile introduces more errors in reads and should make it harder for analysis methods to identify appropriate reference sequences.

To compare the performance of MEGAN-LR and Kaiju, we calculated the sensitivity and precision of taxonomic assignments at the genus, family and order levels. In more detail, following the approach used in [15], we define *sensitivity* as the percentage of reads in R that are assigned either to the correct taxon or to one of its descendants. We define *precision* as the percentage of reads that are assigned correctly, out of all reads that were binned to any node that is not an ancestor of the correct taxon.

Results

We have implemented the interval-union LCA algorithm and the modified functional binning algorithm. In addition, we have implemented a new long read interactive viewer. We provide methods for exporting long read annotations in GFF3 format. Our code has been integrated into the open source edition of MEGAN. In addition, we have modified MEGAN (and all tools bundled with MEGAN) so as to support DNA-to-protein alignments that contain frame-shifts. We use the term MEGAN-LR (MEGAN long read) to refer to this major extension of MEGAN.

Simulation study

The results of our simulation study are shown in Fig. 5, where we summarize the sensitivity and precision scores achieved at genus level by LAST+MEGAN-LR and Kaiju, for both the R7.3 and R9 datasets. In all cases, LAST+MEGAN-LR shows better sensitivity and precision than Kaiju. As expected, both methods are less sensitive on the R7.3 data, as many reads remain unclassified. However, the difference in performance between the two methods is larger on the R7.3 data, and we suspect that this is due to the ability of LAST to perform frame-shift aware alignments and thus to accommodate erroneous insertions and deletions.

Per-dataset performance analysis of LAST+MEGAN-LR and Kaiju is presented in Fig. 6. This shows that LAST+MEGAN-LR outperforms Kajiu on a vast majority of the simulated datasets, with Kajiu sometimes showing better performance when the sensitivity or precision is very low.

Kaiju is many times faster than LAST+MEGAN-LR. However, the latter approach computes and uses all relevant protein alignments, and these are also used to perform functional analysis of the reads or contigs. Hence, we suggest to use Kaiju to obtain a fast, first taxonomic profile for a set of long reads or contigs, and then to use LAST+MEGAN-LR to perform a more accurate and detailed subsequent analysis.

PacBio reads on HMP mock community

To test LAST+MEGAN-LR on a publicly available PacBio mock community dataset, we downloaded "HMP dataset 7" from the PacBio website https://github.com/PacificBiosciences/DevNet/wiki/Human_Microbiome_Project_MockB_Shotgun in April 2017. This dataset contains 319,703 reads of average length 4,681 bp. It was sequenced using the P5 polymerase and C3 chemistry.

LAST alignment against the NCBI-nr database (downloaded January 2017) resulted in protein alignments for 284,728 reads (89% of all reads). MEGAN-LR analysis using the interval-union LCA algorithm assigned 1054 megabases (Mb) aligned bases to taxonomic nodes. Of these, 945.3 Mb were assigned to bacterial genera, with no false positives. A total of 758.4 Mb of aligned sequences were assigned to bacterial species, of which 755 Mb were assigned to true positive species (that is, species known to be contained in the mock-community), whereas approximately 3.4 Mb (0.4%) were assigned to false positive species. The 20 bacterial species in the mock community received between 2.8 Mb (0.37%) and 145 Mb (19%) aligned bases assigned at the species level, whereas the highest false positive species obtained 1.1 Mb (0.14%).

Kaiju classified 280,465 of these reads, assigning 128,774 to a species or lower rank node with a true positive rate of 76.9%. 209,435 reads were assigned to a genus or lower rank node with a true positive rate of 84.5%.

To investigate the use of LAST+MEGAN-LR on assembled reads, we assembled this set of reads using minimap

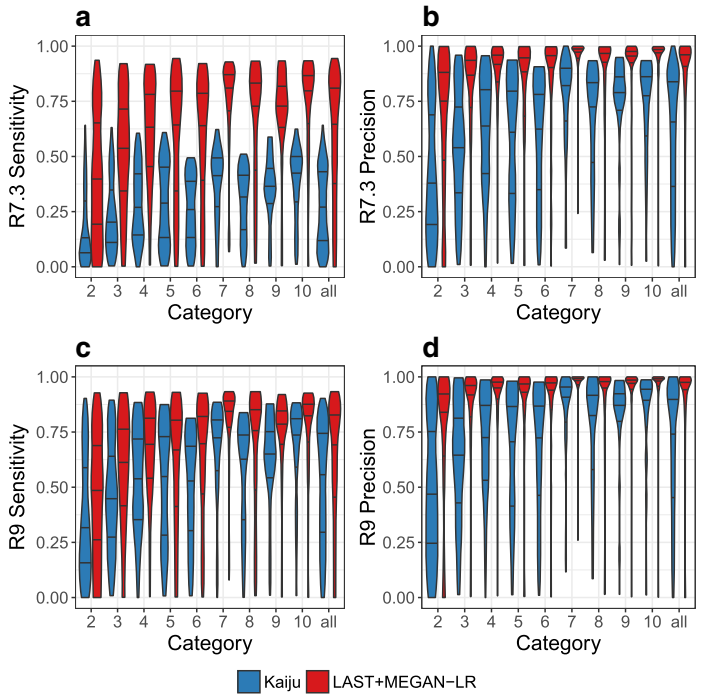


Fig. 5 Violin plots comparing the performance of LAST+MEGAN-LR and Kaiju for two simulation studies, one based on a R7.3 Nanopore chemistry profile and the other based on a R9 Nanopore chemistry profile. In both cases, we report the sensitivity (percentage of reads assigned to the correct taxon) and precision (percentage of reads assigned correctly out of all reads not binned to an ancestor of the correct taxon) of taxonomic assignments. This is done at the genus level for nine different categories of genera (reflecting the number of species in the genus from which the target species was removed), and for all. Results for the R7.3 profile are shown in **a** and **b**, and results for the R9 profile are shown in **c** and **d**

(options -Sw5 -L100 -m0 -t8) and miniasm (version 0.2, default options) [30] and obtained 1130 contigs, with a mean length of 43,976 and maximum length of 1,272,994. LAST alignment against the NCBI-nr database resulted in 41.8 Mb of aligned sequences. Of this, 41.1 Mb and 38.6 Mb, were assigned to bacterial genus and species nodes, respectively, with no false positives and only one false negative species.

PacBio reads on Singer et al. mock community

Our analysis of PacBio reads recently published on a mock-community containing 26 bacterial and archaeal species [31] gave rise to results of similar quality. Of 53,654 reads of average length 1,041 and maximum length 16,403, exactly 51,577 received LAST alignments against NCBI-nr. Of 49.5 Mb of aligned sequences, 45.8 Mb were assigned to prokaryotic genera, with no assignments to false positive species. The amount of sequence assigned at the species level was 36.8 Mb, all of which was assigned to true positive species.

Of the 26 species in the mock community, two are not reported in the analysis and therefore constitute false negative species. These make up approximately 0.01% (*Nocardiopsis dassonvillei*) and 0.1% (*Salmonella bongori*) of the community and are thus on the borderline of detection using the default settings of MEGAN-LR. By default, MEGAN-LR requires that a taxon receives at least 0.05% of all aligned bases before it is reported.

On this data, Kaiju assigned 47,056 reads at the species level, with a true positive rate of 98.7%.

Nanopore reads on HMP mock community

To perform the first test of our new methods on Nanopore data, we sequenced the content of the Genomic DNA from Microbial Mock Community B, as described in the "Methods" section. We obtained 124,911 pass reads of

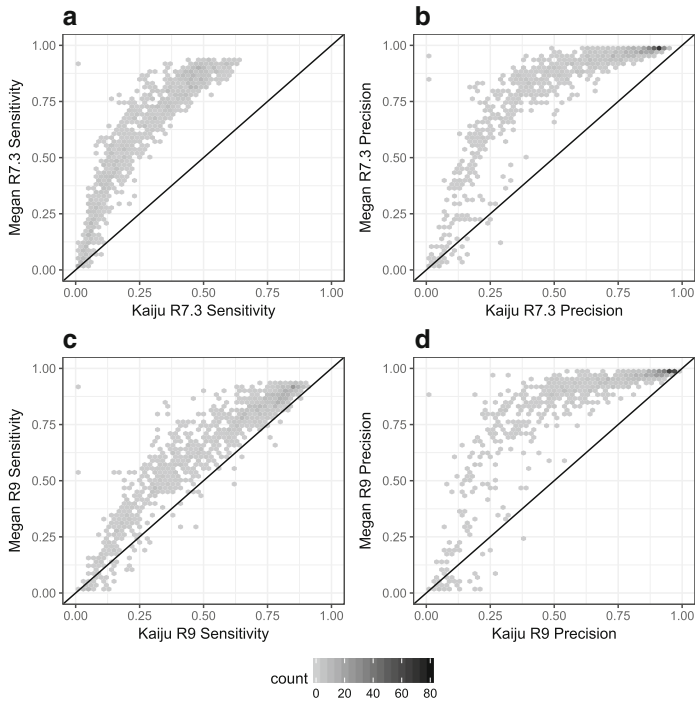


Fig. 6 Here we plot the sensitivity and precision at genus level for Kaiju versus LAST+MEGAN-LR on the R7.3 samples in **a** and **b**, and on the R9 samples in **c** and **d**, respectively

average length 2870, including all template-, complement- and 2D reads.

The LAST alignment against the NCBI-nr database resulted in protein alignments for 57,026 reads (45.6% of all reads). MEGAN-LR analysis assigned a total of 110 Mb aligned bases. Of these, 100 Mb were assigned to bacterial genera, with a false positive assignment rate of 0.1%. Approximately 71.9 Mb of aligned sequences were assigned at the species level, with a false positive rate of 0.9%. The 20 bacterial species in the mock community received between 0.36 Mb (0.5%) and 12.2 Mb (17%) aligned bases assigned at the species level, whereas the highest false positive species obtained 0.21 Mb (0.3%). Around 66 kb of all aligned sequences (0.05%) were falsely assigned to Eukaryota.

Kaiju exhibited a higher false positive rate than LAST+MEGAN-LR on these Nanopore reads, namely 19.8% and 12.6% at the species and genus level, respectively. The program assigned 22,433 reads at the species level and 39,173 reads at the genus level.

Application to anammox data

To illustrate the utility of our new methods in a research context, we applied Nanopore sequencing to a sample obtained from a laboratory bio-reactor enriched for anaerobic ammonium oxidizing bacteria (AnAOB) [32], as described in the "Methods" section. We obtained 71,411 reads of average length 4658 and maximum length 30,846.

LAST alignment against the NCBI-nr database resulted in protein alignments for 64,097 reads (90% of all reads). MEGAN-LR analysis assigned a total of 212 Mb aligned bases. Of these, 94 Mb were assigned to bacterial genera and 112 Mb to bacterial species. The reason why there are more assignments to species than there are to genera is that some of the species present do not have a genus designation in the NCBI taxonomy. The top ten bacterial species assignments are shown in Table 1. This indicates that the most abundant organism in the sample is *Candidatus Brocadia sinica*, a known AnAOB species.

Functional binning in MEGAN-LR allows one to summarize counts at different levels of detail. For example,

Table 1 The ten top bacterial species identified in a Nanopore dataset taken from an anammox enrichment bioreactor, by the number of bases aligned to corresponding reference proteins

Species	Aligned (Mb)
Candidatus Brocadia sinica	84.9
Armatimonadetes bacterium OLB18	8.8
Bacteroidetes bacterium OLB12	4.8
Rhodocyclaceae bacterium UTPRO2	2.9
Chloroflexi bacterium OLB13	2.7
Nitrospira sp. OLB3	1.5
Streptomyces sp. SolWspMP-5a-2	1.1
Anaerolineae bacterium UTCFX5	0.6
Pseudorhodoplanes sinuspersici	0.4

For *Candidatus Brocadia sinica*, this suggests at least ten-fold coverage of the genome

in Table 2 we list the number of alignments to genes for the main KEGG categories of metabolism. MEGAN-LR also makes it possible to investigate function in detail. For example, the anammox process relies on the extremely reactive intermediate hydrazine, produced by the enzyme hydrazine synthase, comprised of the three protein subunits HSZ-α, HZS-β and HZS-γ [33]. Using MEGAN-LR, we identified eight reads that together contain all three subunits, see Fig. 7.

To illustrate the use of LAST+MEGAN-LR on assembled reads, we assembled this set of reads using minimap (options -Sw5 -L100 -m0 -t8) and miniasm (default options) [30] and obtained 31 contigs, with a mean length of 129,601 and maximum length of 750,799. LAST alignment against the NCBI-nr database resulted in 2.98 Mb of aligned sequences. The interval-union LCA algorithm assigned 13 contigs and 96% of all aligned bases to *Candidatus Brocadia sinica*.

Table 2 For each of the main KEGG categories of metabolism, we report the number of alignments against KEGG Orthology reference sequences for the given category, and the number of different KEGG Orthology groups (KOs) involved in such alignments

KEGG metabolism categories	# Alignments	# KOs
Carbohydrate metabolism	9691	347
Amino acid metabolism	8519	371
Energy metabolism	4909	225
Metabolism of cofactors and vitamins	2826	197
Nucleotide metabolism	2675	124
Lipid metabolism	2564	95
Xenobiotics biodegradation and metabolism	1738	116
Glycan biosynthesis and metabolism	1684	114
Metabolism of other amino acids	1344	73
Metabolism of terpenoids and polyketides	1156	63
Biosynthesis of other secondary metabolites	1076	54

These results are based on a LAST+MEGAN-LR analysis of Nanopore reads from an anammox enrichment bioreactor

Performance

To illustrate the computational resources required by the LAST+MEGAN-LR approach, we measured the wall-clock time and memory consumption on the four datasets discussed above. In addition, we considered a further unpublished Nanopore dataset obtained from cheese, consisting of 34 million reads of average length 1460 and maximum length 229,439 (unpublished data provided by the Dutton Lab, UCSD, during the Santa Barbara Advanced School of Quantitative Biology 2017). The programs were run on a Linux server with 32 cores and 512 GB of main memory.

We ran LAST using a volume size setting (parameter -s) of 20 GB (the maximum value), and recorded the peak memory used by the program. We set the maximum memory limit of MEGAN to between 5 GB and 10 GB, depending on the input size. We summarize our measurements in Table 3. The LAST alignment of reads was performed against the entire NCBI-nr protein database and the total size of the LAST index was 215 GB. This step took between a few minutes and a few hours, depending on the size of the input file. The subsequent two steps of conversion and meganization took less than half as long as alignment. By using a smaller LAST volume size, the whole pipeline can also be run on a computer with 16 GB main memory, such as a laptop.

Parameters

To investigate the effect of setting particular parameter values, we analyzed the three mock communities employing a range of different values for `minSupport`, `topPercent` and `percentToCover`. We used the values 0, 0.025, 0.05, 0.075 and 0.1 for `minSupport`; 0, 5, 10 and 20 for `topPercent`; and 50, 60, 70, 80, 90 and 100 for `percentToCover`, respectively. Starting with the DAA file containing the LAST alignments of the reads against NBCI-nr, we ran the classification step of the MEGAN-LR pipeline on all possible combinations of values for the three parameters, with all other parameters set to their default values. We turned off the strong-domination filter for the cases in which `topPercent` equals 20, because that filter removes any alignment whose score lies 10% below that of the best overlapping hit.

For all combinations of parameters, we calculated the rate of true positives and false positives for the number of assigned bases at the species and genus ranks, as well as for the number of assigned bases at any rank above genus. Figure 8 shows these values for Nanopore reads on HMP

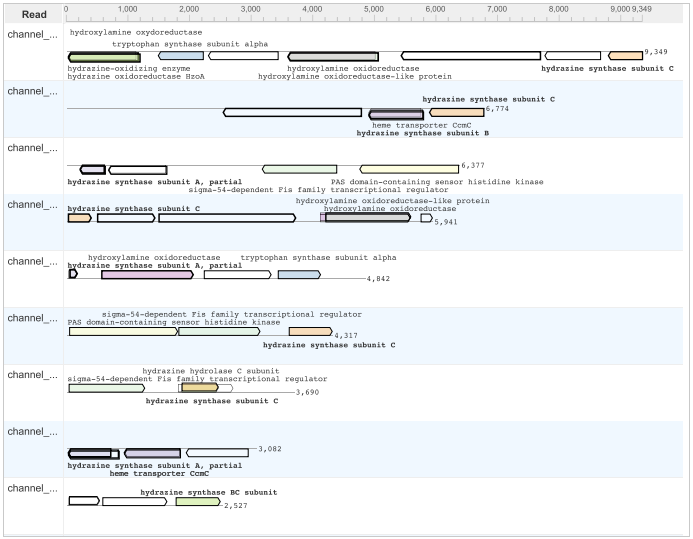


Fig. 7 Long read inspector showing nine reads in the anammox sample that together contain all three subunits of the hydrazine synthase gene, labeled `hydrazine synthase subunit A, partial`, `hydrazine synthase subunit B` and `hydrazine synthase subunit C`

mock community. The figures for PacBio reads on the HMP and the Singer et al. mock community are available in the supplementary material. We also decided to omit the `minSupport` parameter in the figures as it showed little to no variability for any value above 0. Turning off `minSupport` causes spurious assignments of some reads (up to 4% at species level).

As depicted in Fig. 8, increasing the `percentToCover` parameter improves the specificity of the true positive assignments (i.e. more reads are binned at lower ranks), but also increases the rate of false positives.

Using a higher value of the `topPercent` parameter results in more alignments being considered by the LCA algorithm and thus results in a more conservative or less specific binning of reads.

We would like to emphasize that the datasets tested for the effects of parameters in this study are mock communities of species whose proteins are well represented in the reference database. While Fig. 8 suggests setting `TopPercent` to 5% and `percentToCover` to 90%, we suggest that in practice both values should be relaxed slightly, to 10 and 80%, respectively, so as to account for the fact that environmental microbes are usually not so well represented by reference sequences.

Discussion

The application of long read sequencing technologies to microbiome samples promises to provide a much more informative description of the genetic content of environmental samples. The alignment of long reads against a protein reference database is a key step in the functional analysis of such data. Here we show that such protein alignments can also be used to perform accurate taxonomic binning using the interval-union LCA algorithm.

Our simulation study suggests that LAST+MEGAN-LR performs taxonomic binning more accurately than Kaiju. The reported results on mock community datasets indicate a high level of accuracy down to the species level when the corresponding species are represented in the protein reference database. In addition, the computed protein alignments can be used to identify genes and MEGAN-LR provides a useful visualization of the annotated sequences.

The main motivation for developing these new methods is to assist our work on the study of microbial communities in enrichment bio-rectors, where long read sequencing promises to provide access to near-complete genome sequences of the dominating species.

The simple assembly of the anammox data presented in this paper places the dominant species into 11 contigs of

Table 3 Performance of the LAST+MEGAN-LR pipeline

Step	Input	Output	Runtime	Memory
PacBio reads on HMP mock community				
Align	Reads file (1.5 GB)	MAF file	119 min	23 GB
Convert	MAF file (49 GB)	DAA file	29 min	5 GB
Classify	DAA file (4.2 GB)	Meganized DAA file (4.5 GB)	6 min	5 GB
PacBio reads on Singer et al. mock community				
Align	Reads file (56 MB)	MAF file	10 min	22 GB
Convert	MAF file (8.9 GB)	DAA file	5 min	5 GB
Classify	DAA file (197 MB)	Meganized DAA file (415 MB)	1 min	5 GB
Nanopore reads on HMP mock community				
Align	Reads file (191 MB)	MAF file	10 min	22 GB
Convert	MAF file (6.1 GB)	DAA file	3 min	5 GB
Classify	DAA file (553 MB)	Meganized DAA file (644 MB)	1 min	5 GB
Anammox data				
Align	Reads file (336 MB)	MAF file	31 min	24 GB
Convert	MAF file (8.5 GB)	DAA file	4 min	5 GB
Classify	DAA file (371 MB)	Meganized DAA file (500 MB)	2 min	5 GB
Cheese data				
Align	Reads file (5.1 GB)	MAF file	251 min	24 GB
Convert	MAF file (93 GB)	DAA file	90 min	10 GB
Classify	DAA file (3.1 GB)	Meganized DAA file (3.5 GB)	5 min	10 GB

For each of five long read datasets, we report the wall-clock time and main memory required by LAST to align against the NCBI-nr database, for MEGAN to convert the LAST MAF output files into DAA format, and then for MEGAN to classify the reads so as to meganize the DAA file, respectively. The computations were performed on a Linux server with 32 cores and 512GB memory

length greater than 100 kb, containing about 2.8 Mb of aligned sequence and 3.7 Mb of total sequence. This suggests that a more careful assembly, assisted by a set of high quality MiSeq reads, should result in a nearly complete genome.

Our simulation study did not incorporate chimerism or similar artifacts. Because Kaiju uses a heuristic based on the longest match found, we suspect that Kaiju will perform poorly on chimeric reads or misassembled contigs, assigning such a read to one of the source taxa. In contrast, the interval-union LCA algorithm requires by default that 80% of the aligned read is assigned to a taxon and so in practice, such reads will often be placed on a higher taxonomic node.

All datasets discussed in this paper are available here: http://ab.inf.uni-tuebingen.de/software/downloads/megan-lr.

Conclusions

There is increasing interest in using long reads in microbiome sequencing and there is a need to adapt short read tools to long read datasets. In this paper we present an extension of the widely-used metagenomic analysis software MEGAN to long reads. With MEGAN-LR, we provide new algorithms for taxonomic binning, functional annotation and easy interactive exploration of metagenomic long reads and contigs, based on DNA-to-protein alignments. Our work suggests that the presented LAST+MEGAN-LR pipeline is sufficiently fast and accurate.

Reviewers' comments

Reviewer's report 1: Nicola Segata and Moreno Zolfo

Reviewer's comments: The authors present here a novel computational pipeline to address the issue of taxonomical and functional classification of long reads. The authors correctly underline that long reads from emerging sequencing technologies are currently a computational challenge in the field of metagenomics. Indeed, not much attention has been dedicated to the taxonomic identification of long reads, and the author developed an extension of the previously published MEGAN software, which they call MEGAN-LR. The pipeline works with long nucleotide reads which are mapped against a protein database using

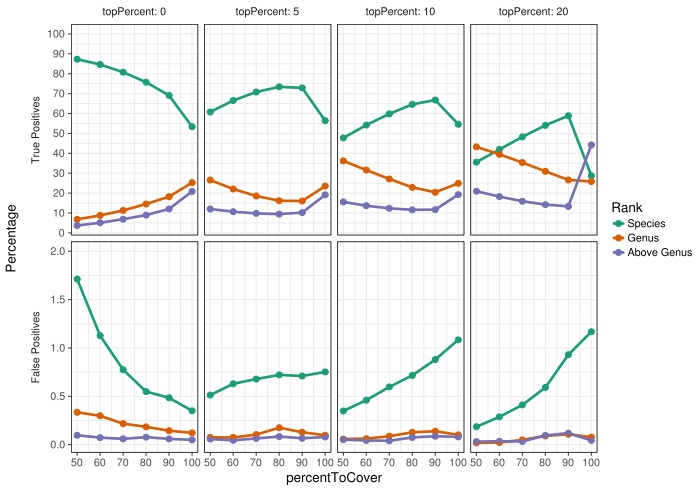


Fig. 8 The effect of changing the `topPercent` and `percentToCover` parameters for the analysis of the Nanopore HMP mock community. True positive and false positive rates are reported for each combination of parameters at the levels of species and genus, and for the sum of ranks above genus. The rate is calculated as the number of correctly assigned bases divided by the total number of bases assigned at the respective taxonomic level

LAST, it accounts for read that align against more than one protein, and is frameshift aware. The authors provide convincing evidences on the accuracy and precision of MEGAN-LR on synthetic data and mock communities sequenced ad-hoc. This review was performed by Nicola Segata and Moreno Zolfo

As summarized in my comments above, I think this is a well written and clear paper. I do not think there are many major issues, but there are several points that the authors should at least consider addressing to improve the paper:

1. It would be useful for the general comprehension of the frameset in which MEGAN-LR is set, to understand why the authors decided to focus on protein-based taxonomic assignment. Most of the other existing algorithms use nucleotide-based approaches. I would suggest to add a paragraph exploring the advantages and disadvantages of the two approaches.
   Author's response: *We have added a paragraph discussing this to the Background section.*
2. The default threshold to report the presence for a taxon is set to 0.05% of the total aligning bases. Since the overall performance of the algorithm could be dramatically affected by this parameter, it would be nice to see how the precision and specificity of MEGAN- LR vary when changing the threshold. Also, I think that the authors should clarify on how this threshold was chosen as default: was it the result of a parameter- optimization of some sort?
   Author's response: *We have added a section on "Parameters" to Methods.*
3. Similarly, one could test the impact of the threshold that is used to determine whether a LAST alignment is strongly dominated by another alignment. Since this value is set by default to 90%, it would be interesting to see the behaviour of the mapper at different thresholds.
   Author's response: *We have added a section on "Parameters" to Methods.*
4. The fact that some alignments in the MAF file are eliminated if they are strongly dominated by another alignment can affect the correct placement of a read. How did the authors decide the default thresholds by which this mechanism is implemented in MEGAN-LR?
   Author's response: *We have added a section on "Parameters" to Methods.*
5. Overall, a precise estimate on the memory and CPU requirements of MEGAN-LR is not provided. I think this point should be reported more clearly, by providing the computational resources used by MEGAN-LR in the analysis. Specifically, I think it would be useful to report how much CPU time and memory were required in each of the validations step. Moreover, it would be also useful to have an estimate

on the order of magnitude of time required to analyse a whole average PacBio/Nanopore metagenome.
Author's response: *We have added a section on "Performance" to Results.*

6. Figure 5, the performances of Kaiju and LAST+MEGAN-LR are binned by the number of species in the genus. It would be interesting to see in the same box plot also the summed (i.e. overall) distributions for each subplot.
Author's response: *To each subplot, we have added a category that summarizes all datasets.*
7. The comparison between Kaiju and MEGAN-LR is performed only on the simulated dataset. I would suggest to run Kaiju also on the PacBio and Nanopore reads from the mock communities, if the genomes of the species present in the communities are available and well annotated. This should provide further support to the higher specificity and precision of MEGAN-LR.
Author's response: *We have added true positive and false positive rates of Kaiju's assignments for mock communities against NCBI-nr to their respective sections.*
8. Another computational tool that is addressing the problem of long-reads mapping is MinHash (Jain et al., https://doi.org/10.1101/103812). It is understandable that the validation was conducted only on Kaiju (as it is the only tool using protein-alignments). Nevertheless, it would be interesting to see the other approaches compared.
Author's response: *A comparison against DNA-based analysis approaches is beyond the scope of this paper.*
9. There is no much on the task of "functional classification" in the "Results" section. Estimating the functional potential of a microbiome is an important task, and it would be very nice if the authors provide some details, validation, and application on real data for this. ror example could the authors provide some comments on the functional landscape detectable with MEGAN-LR of the anammox dataset?
Author's response: *We have added a high-level summary genes assigned to KEGG metabolic categories and also a detailed inspection of the key hydrazine syntase subunits for the anammox sample.*

Reviewer's report 2: Pete James Lockhart

Reviewer's comments: The manuscript by Huson et al. describes and evaluates a novel approach for analyzing long sequence reads and these to taxa and functional categories. The approach will be welcomed by biologists as it provides objective criteria and an interactive means to evaluate the taxonomic identity of species in metagenomics samples.

Identify genome functional characteristics. The latter will include e.g. virulence and pathogenicity, and provides a means e.g. for assessing health risk posed by micro- organisms in metagenomics samples. I have indicated some minor points of communication that should be considered.

1. Also a number of default thresholds are indicated for different stages of analysis, e.g. 80% threshold for the LCA assignment, 50% for the alignment dominance criterion, 0.05% for MEGAN-LR reporting. It would help potential users to have more insight into the thinking behind these values , and whether or not additional threshold values should be considered.
Author's response: *We have added a section on "Parameters" to Methods.*

Reviewer's report 3: Serghei Mangul

Reviewer's comments:

1. The authors propose protein based alignment. Is there an advantage to use protein-based alignment versus nucleotide-based alignment?
Author's response: *We have added a paragraph discussing this to the Background section.*
2. The nucleotide- based methods (for example Centrifuge) have been excluded from the comparison. Including those methods (by using the comparable database with nucleotide sequences) can be valuable. Also, this will provide a general comparison of nucleotide-based versus protein based performance of metagenomic tools.
Author's response: *While we agree that such a comparison would be useful, such a comparison against DNA-based analysis approaches is beyond the scope of this paper.*
3. p.9, line 46. More information about the leave-one-out experiment is required. What is the motivation for the experiment? Does it refer to removing one reference genome, from which reads were simulated? Such experiment can quantify, the possibility of misassignment of reads to the close-related genome, due to the incompleteness of the reference.
Author's response: *Yes, all genes associate with the source genome are removed from the reference database.*
4. p.10, line 18. What is the maximum number of mismatches allowed by MEGAN-LR? The effect of this parameter on the performance of both Megan-LR and Kaiju needs to be explored.
Author's response: *While the number of mismatches is an important parameter for DNA-DNA alignments, it does not usually play a role in amino-acid alignments.*

5. p.10. How was the performance on the species level?
   Author's response: *Our study follows the one published in the Kaiju paper and does not allow an assessment of species-level performance due to its 'leave one species out" approach.*
6. p.10. The paper report sensitivity and precision on the read level. It would be interesting to know such performance on different taxa levels. In such, case sensitivity, for example, would be the percentage of taxa correctly identified.
   Author's response: *We have added supplementary plots for higher taxonomic levels to the companion website.*
7. p.11. The contribution of LAST algorithms to the superiority of MEGAN-LR in comparison to other methods needs to be quantified. One way to do so is to compare the performance of Kaiju with LAST instead of current alignment algorithm.
   Author's response: *As an aligner, LAST does not perform taxonomic binning and so a comparison of Kaiju with LAST without MEGAN-LR is not possible.*
8. p.12, line 24. A more extensive analysis is required. Besides, FN species, it will be interesting to know the number of TP, FP and general sensitivity and precision of each taxonomic level.
   Author's response: *FN levels are very low for the mock data. We now report TP and FP in Fig. 8.*

Abbreviations
MEGAN-LR: long read extension of the metagenome analysis tool MEGAN

Acknowledgements
We thank Dr Gayathri Natarajan and Dr Ying Yu Law for their assistance with obtaining samples from the anammox bioreactor.

Funding
The authors acknowledge support by the High Performance and Cloud Computing Group at the Zentrum für Datenverarbeitung of the University of Tübingen, the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant no INST 37/935-1 FUGG and grant no HU 566/12-1.
This work supported in part by the Singapore National Research Foundation and Ministry of Education under the Research Centre of Excellence Programme, and by a program grant from the Environment and Water Industry Programme Office (EWI), project number 1301–RIS–59.
This research was supported in part by the National Science Foundation under grant no NSF PHY-1748958.
We acknowledge support by the Open Access Publishing Fund of University of Tübingen.

Availability of data and materials
http://ab.inf.uni-tuebingen.de/software/downloads/megan-lr/.

Authors' contributions
All authors contributed to the study design. AG, DH and RW designed the visualization techniques. DH implemented the algorithms and visualization techniques. IB and DJ performed Nanopore sequencing. BA implemented the MAF2DAA program. BA and CB performed the simulation study. DH and RW lead the project and wrote the manuscript. All authors read and approved the final manuscript.

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details
[1]Center for Bioinformatics, University of Tübingen, Sand 14, 72076 Tübingen, Germany. [2]Life Sciences Institute, National University of Singapore, 28 Medical Drive, Singapore 117456, Singapore. [3]Max-Planck Institute for Developmental Biology, 72076 Tübingen, Germany. [4]Singapore Centre for Environmental Life Sciences Engineering, National University of Singapore, 28 Medical Drive, Singapore 117456, Singapore. [5]IMPRS 'From Molecules to Organisms', Tübingen, Germany.

Received: 19 October 2017 Accepted: 29 March 2018
Published online: 20 April 2018

References
1. Huson DH, Beier S, Flade I, Górska A, El-Hadidi M, Mitra S, Ruscheweyh H-J, Tappu R. MEGAN Community Edition - interactive exploration and analysis of large-scale microbiome sequencing data. PLoS Comput Biol. 2016;12(6):1004957. https://doi.org/10.1371/journal.pcbi.1004957.
2. Glass EM, Wilkening J, Wilke A, Antonopoulos D, Meyer F. Using the metagenomics RAST server (MG–RAST) for analyzing shotgun metagenomes. Cold Spring Harb Protoc. 2010;2010(1):5368.
3. Wood DE, Salzberg SL. Kraken: ultrafast metagenomic sequence classification using exact alignments. Genome Biol. 2014;15:46. https://doi.org/10.1186/gb-2014-15-3-r46.
4. Segata N, Waldron L, Ballarini A, Narasimhan V, Jousson O, Huttenhower C. Metagenomic microbial community profiling using unique clade-specific marker genes. Nat Meth. 2012;9(8):811–4. https://doi.org/10.1038/nmeth.2066.
5. Huson DH, Auch AF, Qi J, Schuster SC. MEGAN analysis of metagenomic data. Genome Res. 2007;17(3):377–86. https://doi.org/10.1101/gr.5969107.
6. Poinar HN, Schwarz C, Qi J, Shapiro B, Macphee RDE, Buigues B, Tikhonov A, Huson DH, Tomsho LP, Auch A, Rampp M, Miller W, Schuster SC. Metagenomics to paleogenomics: large–scale sequencing of mammoth DNA. Science. 2006;311(5759):392–4. https://doi.org/10.1126/science.1123360.
7. Mackelprang R, Waldrop M, DeAngelis K, David M, Chavarria K, Blazewicz S, Rubin E, Jansson J. Metagenomic analysis of a permafrost microbial community reveals a rapid response to thaw. Nature. 2011;480(7377):368–71. https://doiorg/101038/nature10576.
8. Human Microbiome Project Consortium. Structure, function and diversity of the healthy human microbiome. Nature. 2012;486(7402):207–14.
9. Willmann M, El–Hadidi M, Huson DH, Schütz M, Weidenmaier C, Autenrieth IB, Peter S. Antibiotic selection pressure determination through sequence–based metagenomics. Antimicrob Agents Chemother. 2015;59(12):7335–45.
10. Kang DD, Froula J, Egan R, Wang Z. MetaBAT, an efficient tool for accurately reconstructing single genomes from complex microbial communities. PeerJ. 2015;3:1165. https://doi.org/10.7717/peerj.1165.
11. Rhoads A, Au KF. Pacbio sequencing and its applications. Genomics, Proteomics Bioinforma. 2015;13(5):278–89. SI: Metagenomics of Marine Environments.
12. Jain M, Olsen HE, Paten B, Akeson M. The Oxford Nanopore MinION: delivery of nanopore sequencing to the genomics community. Genome Biol. 2016;17:239.
13. Juul S, Izquierdo F, Hurst A, Dai X, Wright A, Kulesha E, Pettett R, Turner DJ. What's in my pot? Real–time species identification on the MinION. bioRxiv 2015;030742. https://doi.org/10.1101/030742. https://www.biorxiv.org/content/early/2015/11/06/030742.full.pdf.

14. Kim D, Song L, Breitwieser FP, Salzberg SL. Centrifuge: rapid and sensitive classification of metagenomic sequences. Genome Res. 2016;26:1721–9. http://genome.cshlp.org/content/early/2016/10/17/gr.210641.116.full.pdf+html.
15. Menzel P, Ng KL, Krogh A. Fast and sensitive taxonomic classification for metagenomics with Kaiju. Nat Commun. 2016;7:11257.
16. Wu D, Hugenholtz P, Mavromatis K, Pukall R, Dalin E, Ivanova NN, Kunin V, Goodwin L, Wu M, Tindall BJ, Hooper SD, Pati A, Lykidis A, Spring S, Anderson IJ, D'Haeseleer P, Zemla A, Singer M, Lapidus A, Nolan M, Copeland A, Han C, Chen F, Cheng J-F, Lucas S, Kerfeld C, Lang E, Gronow S, Chain P, Bruce D, Rubin EM, Kyrpides NC, Klenk H-P, Eisen JA. A phylogeny-driven genomic encyclopaedia of bacteria and archaea. Nature. 2009;462(7276):1056–60.
17. Weinstock GM. Genomic approaches to studying the human microbiota. Nature. 2012;489(7415):250–6. https://doi.org/10.1038/nature11553.
18. Gordienko EN, Kazanov MD, Gelfand MS. Evolution of pan-genomes of Escherichia coli, Shigella spp, and Salmonella enterica. J Bacteriol. 2013;195(12):2786–92. https://doi.org/10.1128/JB.02285-12.
19. Mitchell A, Chang H-Y, Daugherty L, Fraser M, Hunter S, Lopez R, McAnulla C, McMenamin C, Nuka G, Pesseat S, Sangrador-Vegas A, Scheremetjew M, Rato C, Yong S-Y, Bateman A, Punta M, Attwood TK, Sigrist CJA, Redaschi N, Rivoire C, Xenarios I, Kahn D, Guyot D, Bork P, Letunic I, Gough J, Oates M, Haft D, Huang H, Natale DA, Wu CH, Orengo C, Sillitoe I, Mi H, Thomas PD, Finn RD. The InterPro protein families database: the classification resource after 15 years. Nucleic Acids Res. 2015;43 (Database Issue):213–21. http://doi.org/10.1093/nar/gku1243. http://nar.oxfordjournals.org/content/43/D1/D213.full.pdf+html.
20. Powell S, Szklarczyk D, Trachana K, Roth A, Kuhn M, Muller J, Arnold R, Rattei T, Letunic I, Doerks T, Jensen LJ, von Mering C, Bork P. eggNOG v3.0: orthologous groups covering 1133 organisms at 41 different taxonomic ranges. Nucleic Acids Res. 2012;40(Database Issue):284–9.
21. Kanehisa M, Goto S. KEGG: Kyoto Encyclopedia of Genes and Genomes. Nucleic Acids Res. 2000;28(1):27–30.
22. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res. 1997;25:3389–402.
23. Kiełbasa SM, Wan R, Sato K, Horton P, Frith MC. Adaptive seeds tame genomic sequence comparison. Genome Res. 2011;21(3):487–93.
24. Sheetlin SL, Park Y, Frith MC, Spouge JL. Frameshift alignment: statistics and post-genomic applications. Bioinformatics. 2014;30(24):3575–82.
25. Buchfink B, Xie C, Huson DH. Fast and sensitive protein alignment using DIAMOND. Nat Methods. 2015;12:59–60.
26. Nicol JW, Helt GA, Blanchard Jr SG, Raja A, Loraine AE. The Integrated Genome Browser: free software for distribution and exploration of genome–scale datasets. Bioinformatics. 2009;25(20):2730–1. https://doi.org/10.1093/bioinformatics/btp472./oup/backfile/content_public/journal/bioinformatics/25/20/10.1093/bioinformatics/btp472/2/btp472.pdf.
27. Rutherford K, Parkhill J, Crook J, Horsnell T, Rice P, Rajandream MA, Barrell B. Artemis: sequence visualization and annotation. Bioinformatics. 2000;16(10):944–5.
28. Liu X, Arumugam K, Natarajan G, W ST, Drautz–Moses DI, Wuertz S, Yingyu L, Williams RBH. Draft genome sequence of a *Candidatus* brocadia bacterium enriched from tropical–climate activated sludge. BioRvix 2017. https://doi.org/10.1101/123943.
29. Yang C, Chu J, Warren RL, Birol I. NanoSim: Nanopore sequence read simulator based on statistical characterization. GigaScience. 2017;6(4):1–6. https://doi.org/10.1093/gigascience/gix010.
30. Li H. Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences. Bioinformatics. 2016;32(14):2103–10. https://doi.org/10.1093/bioinformatics/btw152./oup/backfile/content_public/journal/bioinformatics/32/14/10.1093_bioinformatics_btw152/3/btw152.pdf.
31. Singer E, Andreopoulos B, Bowers RM, Lee J, Deshpande S, Chiniquy J, Ciobanu D, Klenk H-P, Zane M, Daum C, Clum A, Cheng J-F, Copeland A, Woyke T. Next generation sequencing data of a defined microbial mock community. Sci Data. 2016;3:160081.
32. Kartal B, de Almeida NM, Maalcke WJ, Op den Camp HJM, Jetten MSM, Keltjens JT. How to make a living from anaerobic ammonium oxidation. FEMS Microbiol Rev. 2013;37:428–61.
33. Dietl A, Ferousi C, Maalcke WJ, Menzel A, de Vries S, Keltjens JT, Jetten MSM, Kartal B, Barends TRM. The inner workings of the hydrazine synthase multiprotein complex. Nature. 2015;527:394.

Appendix II

Arumugam *et al. Microbiome* (2019) 7:61
https://doi.org/10.1186/s40168-019-0665-y

Microbiome

SHORT REPORT Open Access

Annotated bacterial chromosomes from frame-shift-corrected long-read metagenomic data

Krithika Arumugam[1], Caner Bağcı[2,3], Irina Bessarab[4], Sina Beier[2], Benjamin Buchfink[5], Anna Górska[2,3], Guanglei Qiu[1], Daniel H. Huson[2,6*†] and Rohan B. H. Williams[4*†]

Abstract

Background: Short-read sequencing technologies have long been the work-horse of microbiome analysis. Continuing technological advances are making the application of long-read sequencing to metagenomic samples increasingly feasible.

Results: We demonstrate that whole bacterial chromosomes can be obtained from an enriched community, by application of MinION sequencing to a sample from an EBPR bioreactor, producing 6 Gb of sequence that assembles into multiple closed bacterial chromosomes. We provide a simple pipeline for processing such data, which includes a new approach to correcting erroneous frame-shifts.

Conclusions: Advances in long-read sequencing technology and corresponding algorithms will allow the routine extraction of whole chromosomes from environmental samples, providing a more detailed picture of individual members of a microbiome.

Keywords: Microbiome, Long-read sequencing, Microbial genomics, Sequence assembly, Frame-shifts, Algorithms, Software

Background

Second-generation sequencing has been the work-horse of metagenomic analysis of microbiomes, with typical studies based on hundreds of millions of short reads [1, 2]. While the taxonomic and functional binning of short metagenomics read data are reasonably straightforward computational problems [3], much recent work has focused on the challenge of assembling and binning metagenomic contigs, a procedure which provides invaluable working models of the genomes of member species [4]. However, the assembly of whole bacterial chromosomes from short metagenomic reads has proven to be an all but impossible task.

Third generation sequencing promises to allow the extraction of whole genomes from environmental samples with ease [5]. This promise is now beginning to be fulfilled. Here, we report on the results of a single ONT MinION run on a microbial community from an enrichment bioreactor targeting polyphosphate accumulating organisms (PAO), that had been inoculated with activated sludge from a full-scale water reclamation plant in Singapore.

Results

Running a MinION sequencer for 1 day, we obtained ≈ 695,000 long reads with an average length of 9 kb, totaling approximately 6 Gb of sequence (Additional file 1: Table S1). Using Unicycler [6–8], we assembled these into 1702 contigs (LR contigs) of average length 61 kb (Additional file 2: Table S2). We observed 10 contigs over 1 Mb in length, including five circular contigs between 2.7 and 4.2 Mb long (see Fig. 1a). In principle, long-read assembly procedures could generate complete genomes *de novo*, without the need for complex contig binning procedures, and accordingly we designed tools and analyses to determine the extent to which such long

*Correspondence: daniel.huson@uni-tuebingen.de; lsirbhw@nus.edu.sg
†Daniel H. Huson and Rohan B. H. Williams contributed equally to this work.
[4]Singapore Centre for Environmental Life Sciences Engineering, National University of Singapore, 28 Medical Drive, Singapore 117456, Singapore
Full list of author information is available at the end of the article

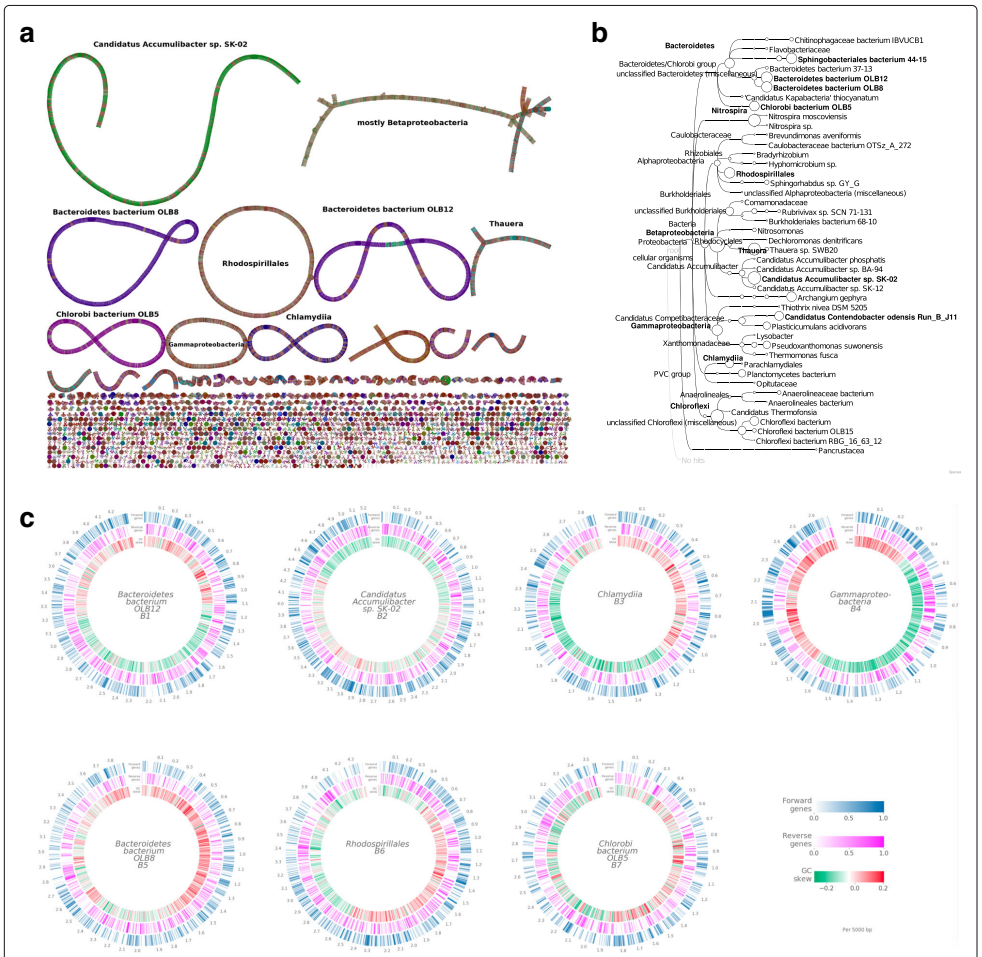


Fig. 1 Summary of results. **a** Bandage [30] visualization of the Unicycler assembly graph before final segmentation into contigs. The largest connected components are labeled by the corresponding taxonomic bins, and the nodes are colored by the MEGAN taxonomic classification of the corresponding long reads. The seven longest linear and circular components correspond to the seven LR-chromosomes. **b** MEGAN-LR taxonomic binning: nodes are scaled to indicate the number of aligned bases in each bin. Bins that are more than 50% complete are shown in bold. **c** Annotation of the seven LR-chromosomes, labeled by the corresponding taxonomic bins. The three circular tracks indicate the genes annotated by Prokka on the forward strand (blue) and reverse strand (pink), and GC-skew (green and red indicate lower or higher than average GC content, respectively)

contigs represented genomes of member species of the community. Our analyses are based on (1) the analysis of genome completeness and quality, (2) whole genome comparisons to reference genomes, and (3) comparison with metagenome-assembled genomes recovered from short reads sequenced from the same DNA sample.

Long reads, and, to a lesser degree, LR contigs, suffer from a high rate of erroneous insertions and deletions, which lead to frame-shifts in translated alignments. For the data presented here, the average number of frame-shifts per kilobyte of aligned sequence is 14.8, for unassembled long reads, and 6, for LR contigs, with a standard deviation of 2.9 and 2, respectively. For this reason,

genome evaluation tools (such as CheckM [9]) and annotation workflows (such as Prokka [10]), which typically employ translated alignments, perform poorly on current long-read data.

To address this deficiency, we have developed a two-step frame-shift correction technique. First, we have modified DIAMOND [11] (v 0.9.23) so as to perform a *frame-shift aware* DNA-to-protein alignment [12] of the sequences against the NCBI-nr protein reference database [13]. Second, based on the location of frame-shifts reported in the alignments, we insert Ns into the sequences so as to maintain the frame (see Fig. 2b). Sequences corrected in this way can be evaluated and annotated using conventional genome quality and annotation tools.

We performed initial taxonomic analysis of all LR contigs using MEGAN-LR [14] (v 6.13.3), obtaining 106 taxonomic bins at different taxonomic ranks (see Fig. 1b and Additional file 3: Table S3). To determine whether these taxonomic bins might harbor complete genomes, we applied CheckM to the set of frame-shift-corrected LR contigs contained in each taxonomic bin. This analysis indicates that 14 of the bins are more than 50% complete. Of these, six fulfill the definition of a "high quality draft" metagenome-assembled genome (namely, completeness $>$ 90% and contamination $<$ 5%). For purposes of this paper, we also consider the seventh bin listed in Table 1 as high quality, as it consists of only one circular LR contig and is of chromosomal length. There are four additional bins that reach the level of "medium quality draft" (completeness $>$ 50% and contamination $<$ 10%) [15].

In all seven high-quality bins, the CheckM results derive from a single long contig, of length 2.7 − 5.2 Mb, with the numbers of cognate rRNA and tRNA genes, and protein coding genes, as reported by Prokka, all lying within the range usually seen for bacterial genomes (see Table 1 and Additional file 3: Table S3). Throughout this paper, we will refer to these long contigs as the seven *LR chromosomes*.

From the seven high quality taxonomic bins, we obtained a near-complete LR chromosome (number B2 in Table 1) that is binned to *Candidatus Accumulibacter*, a polyphosphate accumulating organism (PAO) that is commonly observed in waste-water treatment plants and is the target of our enrichment protocol [16]. Two circular LR chromosomes (B1 and B5) are binned to the species *Bacteroidetes bacterium* OLB8 and OLB12,

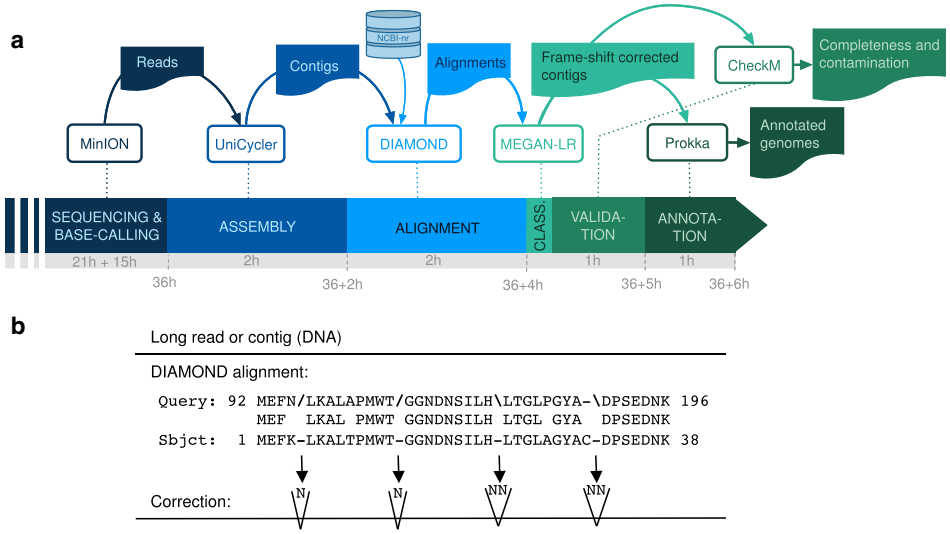


Fig. 2 Analysis. **a** Long-read analysis pipeline shown from left to right. MinION sequencing produces a set of reads. These are assembled into contigs using Unicycler and aligned against the NCBI-nr database using DIAMOND. The contigs and alignments are processed by MEGAN so as to perform taxonomic binning and also to produce frame-shift-corrected contigs. These are analyzed using CheckM and annotated using Prokka. The duration of each step is shown in wall-clock hours. MEGAN analysis took less than 10 min. **b** Frame-shift correction: in frame-shift alignments, forward slashes, and backward slashes indicate a frame decrease, or increase, by one, respectively. Correction is performed by inserting one or two unspecified nucleotides into the sequence, respectively

Table 1 Summary of results

	(a) DIAMOND+MEGAN taxonomic bin	(b) Unicycler contigs	(c) Total (Mb)	(d) Aligned (Mb)	(e) Average coverage	(f) CheckM Complete.	(g) Prokka contam.	(h) rRNA	(i) tRNA	(j) CDS
High-quality draft genomes:										
B1	*Bacteroidetes bacterium* OLB12	1	4.2	3.5	57.3	95%	0.1%	6	39	4,163
B2	*Candidatus Accumulibacter* SK-02	1	5.2	4.1	384.2	94%	0.6%	4	53	4,915
B3	*Chlamydiia* (class)	1	2.8	1.8	48.8	94%	2%	6	39	3,387
B4	*Gammaproteobacteria* (class)	43	4.7	3.0		93%	2%	6	52	4,833
	-Longest contig		2.7	1.6	25.1	93%	0.2%	3	40	3,359
B5	*Bacteroidetes bacterium* OLB8	1	3.8	3.0	52.1	93%	1%	6	37	3,394
B6	*Rhodospirillales* (order)	1	4.4	3.0	29.5	92%	0.5%	3	47	4,015
B7	*Chlorobi bacterium* OLB5	1	3.5	2.5	38.7	88%	1%	3	41	4,131
Medium quality draft genomes:										
B8	*Thauera* (genus)	25	4.6	4.0		89%	4%	12	64	4,040
	-Longest contig		0.8	0.7	32.7	14%	0%	0	5	672
B9	*Sphingobacteriales bacterium* 44-15	59	3.2	2.8		76%	1%	2	17	2,953
	-Longest contig		0.2	0.1	10.2	0%	0%	0	0	172
B10	*Bacteroidetes* (phylum)	43	3.9	2.6		72%	7%	1	12	1,997
	-Longest contig		1.2	0.8	14.1	32%	0%	0	3	807
B11	*Candidatus Contendobacter* B J11	39	2.5	2.0		59%	9%	2	37	2,668
	-Longest contig		0.3	0.3	15.4	19%	0%	0	7	295
Low quality draft genomes:										
B12	*Betaproteobacteria* (class)	111	6.6	5.5		89%	79%	6	71	4,655
	-Longest contig		0.4	0.3	37.1	10%	0%	0	1	372
B13	*Nitrospira* (genus)	34	4.2	3.7		83%	13%	0	6	563
	-Longest contig		1.1	0.9	17.6	27%	0%	0	2	99
B14	*Chloroflexi* (phylum)	151	5.4	4.3		71%	29%	0	11	3,565
	-Longest contig		0.2	0.2	13.3	8%	0%	0	1	86

For all 14 taxonomic bins B1–B14 that CheckM deems ≥ 50% complete (a), and -in cases where the bin contains more than one contig- also for the longest contig, in descending order of assembly quality, we report (b) the number of contigs produced by Unicycler, (c) the total number of bases, (d) the number of bases aligned by DIAMOND to some protein reference, (e) the average coverage by long reads (based on the longest contig), (f) the %-completeness and (g) %-contamination reported by CheckM, and (h)–(j), the number of rRNA, tRNA and coding sequences reported by Prokka, respectively

both of which were originally recovered as metagenome-assembled genomes from a partial-nitritation anammox (PNA) bioreactor community, where they are thought to function as aerobic heterotrophs [17]. All three of these LR-chromosomes align end-to-end to their corresponding (fragmented) reference genomes (see Fig. 3).

The remaining four are closed circular chromosomes that do not align to any current reference genome and thus most likely represent novel organisms. One of these (B3) is binned to the class of *Chlamydiia*. Although normally considered an obligate intracellular pathogen in humans, members of the phylum *Chlamydiae* are known to occur in microeukaryotes that occur as predators in such reactor communities [18]. Another (B6) is binned to *Rhodospirillales* and contains a 16S sequence that maps to the genus *Defluviicoccus*. Some members of this genus compete with PAO for carbon sources and are commonly observed in PAO enrichment reactors [19]. Another LR chromosome

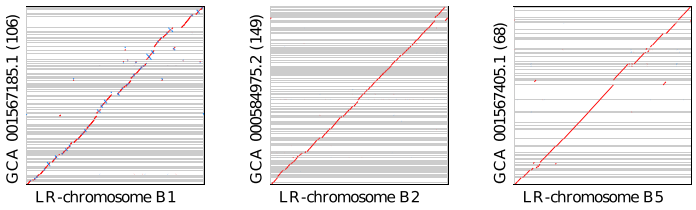


Fig. 3 Dot plots for the three LR-chromosomes that have high similarity to reference genome assemblies, namely, B1 against GCA_001567185.1 (*Bacteroidetes bacterium* OLB12), B2 against GCA_000584975.2 (*Candidatus accumulibacter* sp. SK-02), and B5 against GCA_001567405.1 (*Bacteroidetes bacterium* OLB8). Forward alignments are shown in red, whereas reverse complemented alignments are shown in blue, and gray lines indicate contig boundaries in the reference assemblies. The number of contigs in each reference sequence is given in brackets

(B4) is binned to the class *Gammaproteobacteria*. Finally, we obtained an LR chromosome (B7) that is binned to *Chlorobi bacterium OLB5*, an organism previously observed in waste-water [17].

For all seven LR-chromosomes, Silva analysis [20] of the contained 16S sequences confirm the taxon bin assignment obtained by MEGAN analysis (see Table 2).

Solely for the purpose of verification, we also produced a second independent set of paired reads from the same DNA aliquot using Illumina short-read sequencing. First, we used the short-read clone coverage to detect potential break-points in the assemblies of 7 LR chromosomes that might indicate long-read assembly errors, and found 11. All but one of these positions have very good long-read coverage, making an assembly error unlikely at these positions. Second, we assembled the short reads and aligned the short-read contigs against the long-read contigs, and this comparison shows a very high degree of co-linearity within the SR contigs (see Fig. 5). Third, we performed metagenomic binning of the short-read contigs and compared the short-read bins with the long-read chromosomes, confirming a very high level of concordance between the two assemblies (see Fig. 6 and Additional file 11: Figure S3).

Table 2 For all seven LR chromosomes, we list the MEGAN and Silva taxonomic assignments

Bin	MEGAN assignment	Silva assignment
B1	*Bacteroidetes bacterium* OLB12	*Bacteroidetes; Bacteroidia; Cytophagales; Microscillaceae;* OLB12
B2	*Candidatus Accumulibacter* sp. SK-02	*Proteobacteria; Gammaproteobacteria; Betaproteobacteriales; Rhodocyclaceae; Candidatus Accumulibacter*
B3	*Chlamydiia* (class)	*Chlamydiae; Chlamydiae; Chlamydiales; Parachlamydiaceae*
B4	*Gammaproteobacteria* (class)	*Proteobacteria; Gammaproteobacteria; Coxiellales; Coxiellaceae; Coxiella*
B5	*Bacteroidetes bacterium* OLB8	*Bacteroidetes; Bacteroidia; Chitinophagales; Saprospiraceae;* OLB8
B6	*Rhodospirillales* (order)	*Proteobacteria; Alphaproteobacteria; Rhodospirillales; Rhodopirillaceae; Defluviicoccus*
B7	*Chlorobi bacterium* OLB5	*Ignavibacteriae; Ignavibacteria; Ignavibacteriales; Ignavibacteriaceae*

Discussion

In this study, a single run of a nanopore MinION device on an enriched bioreactor community gave rise to a high coverage (384×) of the target polyphosphate accumulating organism, *Candidatus Accumulibacter*, but also 10–60× coverage for 13 other taxa. From this data, in total, seven high quality draft genomes were obtained, six of which as closed circular chromosomes. Only three of these draft genomes have closely related reference genomes at NCBI. In all three cases, the LR chromosomes display a major improvement in continuity over the fragmented reference genomes, which were obtained by metagenomic assembly of short reads.

A potential concern might be that the reported megabase-sized contigs might be chimeric or otherwise incorrect. The results reported by CheckM and Prokka suggest that these sequences are entirely consistent with being complete bacterial chromosomes. Moreover, our comparison with a set of short reads sequenced from the same DNA provides further evidence that the reported LR chromosomes are correct, and that an extremely high degree of recapitulation is obtained when compared to draft genomes obtained from the same DNA extraction. However, it is possible that some parts of the reported LR chromosomes might locally represent a mixture of closely related strains.

One current issue with long-read sequencing technologies is that they produce a significant rate of erroneous insertions and deletions, which cause problems when performing translated alignments. Our work suggests that

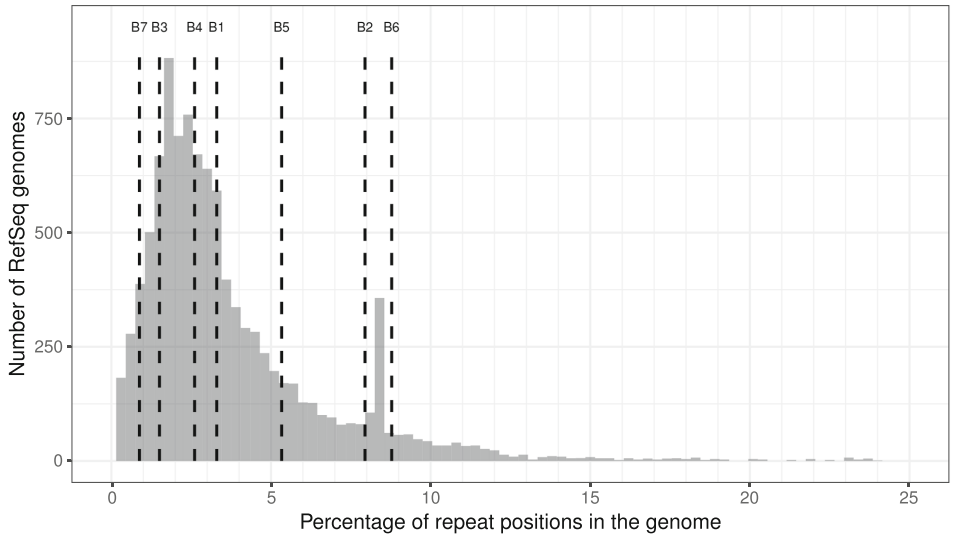


Fig. 4 Distribution of repeat rates in all complete bacterial genomes in RefSeq. Vertical lines show the repeat rate of the seven LR chromosomes. Additional 17 data points that have a repeat percentage above 25% are not shown

frame-shift aware alignment techniques can be used to reduce such problems. If short reads are available for the same DNA, then these can be used to polish the LR contigs so as to further reduce the frame-shift problem. On the data presented in this paper, short-read polishing reduced the average number of frame-shifts per kilobyte of aligned sequence to 1.2.

A major challenge for the use of long-read sequencing technologies in metagenomics is that the use of more aggressive DNA extraction techniques to access the DNA molecules of more robust cells may lead to more fracturing of the DNA molecules, which will limit the length of the sequencing reads. In this paper, our focus was on obtaining long enough reads to allow the assembly of complete chromosomes, so organisms present in low abundance or with more robust cells are underrepresented in the long-read data, as indicated in Additional file 9: Figure S1.

Conclusions

This work suggests that it is now possible to obtain complete bacterial chromosomes from an enriched microbial community using Nanopore sequencing. We provide a straight-forward pipeline for processing such data. It performs assembly, alignment against NCBI-nr, taxonomic binning, frame-shift correction, bin quality analysis and annotation, in less than 6 h (see Fig. 2a).

The application of long-read sequencing techniques promises to allow the routine extraction of whole chromosomes from environmental samples, providing a much more detailed picture of individual members of a microbiome.

Methods

EPBR bioreactor

A sequencing batch reactor (SBR) with 5.4 L working volume was inoculated with activated sludge from an EBPR mother reactor. A slow feeding strategy was applied for the reactor operation, which has been shown to benefit the proliferation of *Ca.* Accumulibacter [21]. The SBR was operated in 6 h cycles, including 60 min feeding, 20 min anaerobic, 180 min aerobic, and a 100 min settling/decant stage. In each cycle, 2.35 L of synthetic waste-water composed of 0.53 L of solution A (containing 1.02 g/L NH_4Cl, 1.2 g/L $MgSO_4$ $7H_2O$, 0.01 g/L peptone, 0.01 g/L yeast extract, and 6.8 g/L sodium acetate) and 1.82 L of solution B (0. 312 g/L $K_2HPO_43H_2O$, 0.185 g/L KH_2PO_4, 0.75 mg/L $FeCl_3$ $6H_2O$ 0.015 mg/L $CuSO_4$ $5H_2O$, 0.03 mg/L $MnCl_2$, 0.06 mg/L $ZnSO_4$, 0.075 mg/L $CoCl_2$, 0.075 mg/L H_3BO_3, 0.09mg/L KI, and 0.06 mg/L Na_2MoO_4 $2H_2O$) (modified from [22]) was introduced into the reactor. The

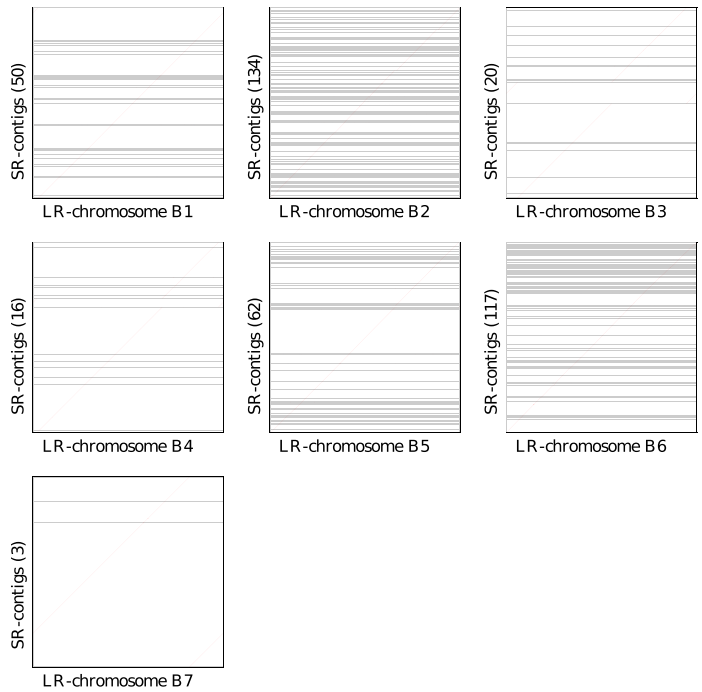


Fig. 5 For each of the seven LR chromosomes (B1–B7), we show a dot plot comparison against the set of SR contigs that align, reporting their number in brackets

reactor was operated at 30 °C with an hydraulic retention time (HRT) and a solid retention time (SRT) of 12 h and 11 days, respectively. The pH was controlled at 7.00–7.60 with DO levels maintained at 0.8–1.2 mg/L during the aerobic phase. The SBR achieved P-release of 180–200 mg/L with complete P removal observed after a 6-month operation. The reactor was sampled on day 267 of the operation.

DNA extraction

Genomic DNA was extracted from the sampled biomass with the FastDNA™SPIN kit (MP Biomedicals) for soil, using 2× bead beating with a FastPrep homogenizer (MP Biomedicals). The DNA was then size-selected on a Blue Pippin DNA size selection device (SageScience) using a BLF-7510 cassette with high pass filtering with a 8 kb cut-off.

Nanopore sequencing

The sequencing library was constructed from approximately 4μg of genomic DNA using the SQK-LSK 108 Ligation Sequencing Kit (Oxford Nanopore Technologies). Sequencing was performed on a MinION Mk1B instrument (Oxford Nanopore Technologies) using a SpotON FLO MIN106 flowcell (FAH85393) and R 9.4 chemistry, running for approximately 24 h. Data acquisition was performed using MinKNOW version 1.14.1 running on a HP ProDesk 600G2 computer (64-bit, 16 GB RAM, 2 Tb SSD HD) running Windows10. Base-calling was performed using Albacore version 2.3.1. Adaptor trimming was performed using Porechop [23] with default settings. This produced 694,955 reads of average length 9 kb (range 2 bp–66 kb). A summary of the long-read statistics is given in Additional file 1: Table S1.

Long-read assembly

Long-read assembly was performed using Unicycler (v 0.4.6) running with default settings. Assembly of the 694,955 long reads produced 1702 LR contigs of average length 61 kb (1.3 kb-5.2 Mb). This took 104 wall-clock minutes (10.2 CPU hours) on a server. (All timings

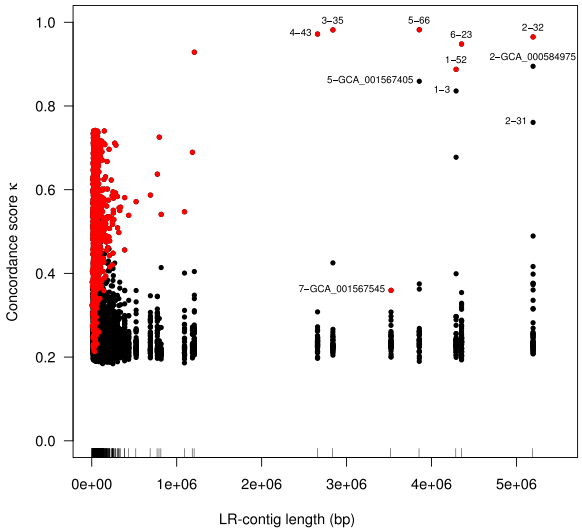


Fig. 6 Overview of the concordance score for LR contigs, on the one hand, and SR-bins and reference genomes, on the other. The *x* axis shows the length of each LR contig, with the position of each tagged with a tick on the axis; the *y* axis shows the value of concordance score κ, and data points represent pairs of LR chromosomes and SR bins, or references genomes. Selected pairs with high concordance score are labeled with "⟨LR-chromosome.id⟩—⟨SR-bin.id⟩" for comparisons to SR-bins or "⟨LR-chromosome.id⟩—⟨GCA_id⟩" for comparisons to references. Within each set of the seven LR chromosome alignments, the pair with the maximum concordance score is shown in red. All LR chromosomes have highly concordant counterpart SR-bins, with the exception of LR chromosome 5. Further details on individual LR chromosomes are reported in Additional file 10: Figure S2

in this paper were measured on a server with AMD Opteron(TM) Processor 6274, 64 × 2.2 GHz, 512 GB memory). A summary of the long-read contig statistics is given in Additional file 2: Table S2.

DIAMOND options for long reads

This paper introduces two new features in DIAMOND for use with error-prone long reads or contigs. First, the program now provides a *frame-shift mode* that performs frame-shift alignment of DNA sequences against a protein reference database [12]. This feature is activated using the command line option `-F 15`, which also sets the frame-shift dynamic programming penalty to a specific value, in this case 15.

Second, the program now provides the option to perform *range-culling*. This feature determines which alignments are reported to output. Without range-culling, the program reports the most significant alignments for the query, up to a given count or score, independent of their position along the query. With range-culling, the decision whether to report an alignment is made locally. By default, any alignment *A* found is reported, unless there exists another alignment *B* that covers at least 50% of *A* on the query and whose bit-score is significantly larger, by defaulting requiring that the score of *A* is less than 90% of the score of *B*. This feature is activated using the command line options `--range-culling` and `--top 10`.

DIAMOND alignment

In preparation of running DIAMOND on the Unicycler LR contigs, we downloaded the NCBI-nr database in November 2018, obtaining 177.6 million protein reference sequences. DIAMOND required about 1 h to initially process the database.

DIAMOND was run on the set of LR contigs with the following options: `--range-culling --top 10 -F 15 --outfmt 100 -c1 -b12 -t /dev/shm`. The program required 140 walk-clock minutes (81 CPU hours) to align all 1702 LR contigs against the NBCI-nr database and obtain 1.8 million alignments for 1695 contigs.

In comparison, running DIAMOND on the LR contigs without using the long-read specific options take only 40 wall-clock minutes (15 CPU hours), but only finds 42,230 alignments, and is thus not useful in practice.

Frame-shift correction

In Fig. 2b, we illustrate how to correct frame-shift errors in a given query DNA sequence, based on an alignment

computed by DIAMOND in frame-shift mode. In a frame-shift alignment, a '/' in the alignment transcript indicates that the aligner inreased the current frame of the query sequence by 1 at the given position, whereas a '\' indicates the current frame was increased by 1, as in http://last.cbrc.jp/doc/lastal.html. To perform frame-shift correction, in the former case, we insert a single unspecified nucleotide 'N' into the query sequence, whereas in the latter case, we insert two unspecified nucleotides 'NN.'

To perform this correction on a long read or LR contig, we greedily select a maximal set of non-overlapping alignments for the whole query and use this set for correction. This is implemented in MEGAN.

MEGAN analysis and frame-shift correction

The output file of DIAMOND was prepared for analysis with MEGAN using the program daa-meganizer, which is part of the MEGAN Community Edition suite, version 6.13.1. The following command line options were used:

```
 --longReads --lcaAlgorithm
longReads --lcaCoveragePercent 51
--readAssignmentMode alignedBases
--acc2taxa prot_acc2tax-Nov2018X1.abin
```

The first three options select MEGAN's long-read analysis mode and sets the amount of aligned sequence to be covered by a taxon during the LCA analysis to 51% [14]. The fourth option requests that the primary count associated with each taxon is the number of aligned reads contained in the contigs binned to that taxon. The final option instructs the program to use the November 2018 mapping of NCBI accessions to NCBI taxa. This "meganization" step took less than five wall-clock minutes (0.2 CPU hours).

A summary of the taxon bins obtained by MEGAN analysis is given in Additional file 3: Table S3.

Frame-shift correction was performed on all LR contigs using MEGAN's `Export Frame-Shift Corrected Reads...` menu item, and the resulting sequences were saved into taxon-specific files, in just over 2 min.

CheckM

The frame-shift-corrected bins were analyzed for their completeness and contamination using CheckM (v1.0.12) in `lineage_wf` mode. Data files for CheckM were downloaded on 26.11.2018 from https://data.ace.uq.edu.au/public/CheckM_databases. The full output of CheckM is provided in Additional file 4: Table S4.

Prokka

We annotated the frame-shift-corrected bins using Prokka (v1.12) in metagenome annotation mode without specifying taxa. The taxonomic database for this version of Prokka is based on Rfam 1.12.

16S analysis

For all seven LR chromosomes, we extracted all 16S sequences annotated by Prokka and performed taxonomic classification of them using Silva [20], obtaining the correspondence between the MEGAN assignments and the Silva assignments (note that *Ignavibacteriaceae* appears within the *Chlorobi* group in the NCBI taxonomy) reported in Table 2.

All assignments were obtained using a threshold of 95% identity, except for the case of bin B4, where a lower threshold of 90% identity was needed to obtain an assignment.

Comparison with genomic references

For each of the seven LR chromosomes, we determined the reference taxon that occurs the most times in DIAMOND alignments of the contig against NCBI-nr. We then aligned the LR chromosomes to the corresponding reference assemblies using Minimap2 [7] (v2.14-r883) with parameters `-cx asm20 -t32 --secondary=yes -P`. We found a significant level of DNA similarity in three cases, which we summarize here as dot plots (see Fig. 3). The other four LR chromosomes did not align to their corresponding reference sequences (less than 1% of the total chromosome covered by an alignment), or, indeed, to any genome in the whole of NCBI.

Repeat analysis

We used Minimap2 to align all seven LR-chromosomes against themselves with parameters `-cx asm10 -t32 --secondary=yes -P` to find repeated regions in them. The option `-c` generates CIGAR strings in the output, `-x asm10` is a preset of parameters for comparing assemblies with up to 10% divergence, `-t32` sets the number of threads, `--secondary=yes` reports secondary alignments (by default Minimap2 reports only the best alignment), and `-P` retains all chains and attempts to elongate them. We then marked the positions that are within alignments of length equal to or greater than 500 in a contig to itself as repeat regions.

In order to check whether the repeat rates obtained for our contigs are typical for bacterial genomes, we performed the same analysis on all complete bacterial genomes in RefSeq (downloaded on 01.06.2018). Figure 4 suggests that the seven LR chromosomes have repeat-rates that are similar to those observed for complete bacterial genomes in RefSeq.

Additional short-read sequencing

To support the evaluation of the long-read contigs, we performed additional short-read sequencing from the

same sample. Genomic DNA library preparation was performed using a modified version of the Illumina TruSeq DNA Sample Preparation protocol. Sequencing was performed on an Illumina HiSeq 2500 using a read length of 301 bp (paired-end). The raw gDNA FASTQ files were processed using cutadapt (v 1.14) in paired-end mode (with default arguments except `-overlap 10 -m 30 -q 20,20`). We obtained 43,856,872 short reads in total. Summary statistics for the short reads are provided in Additional file 5: Table S5.

Break-point and coverage analysis using short reads

We aligned all short reads against the LR contigs using Minimap2, with options `-2 -f 0 -t 32 -F 10000 -ax sr --secondary=yes -N 10000`. Then, considering each pair of reads, a valid clone, if the two aligned reads have the correct orientation with respect to each other and a distance below 800, we determined the clone coverage of each LR contig. Any stretch of LR contig, for which the clone-coverage is zero, is considered a potential break-point. We identified 11. All but one of these are covered by multiple long reads, and so we assume that they are not indicative of a long-read assembly error. The coordinates of the potential break-points are reported in Additional file 6: Table S6.

A comparison of the SR-coverage and LR-coverage of the 14 longest LR contigs reported in Table 1 yields a strong positive correlation (Pearson's $R = 0.9988$), see Table 3.

LR contig polishing using short reads

In the case that short reads are available, polishing of LR contigs using short reads will lead to a reduction of frame-shift error. To investigate this, we used pilon [24] (with minimap2 mapping of short reads to LR contigs as described above) to polish the LR contigs using our short reads. We then analyzed the polished LR contigs using DIAMOND + MEGAN and frame-shift correction, as described above. The resulting number of frame-shifts per kilobyte of aligned sequence was 1.2 (standard deviation 1.6), compared to 6 for unpolished LR contigs.

Assembly of short reads

The 43.86 million short reads were assembled using SPAdes-3.12.0 [25] (default parameters except `-meta -k 21,33,55,77,99,127 -t 30`). We obtained a total of 539,404 short-read contigs (SR contigs) of at least 500 bp in length. See Additional file 5: Table S5.

Comparison of α diversity between short and long reads

To compare the α-diversity represented in the short reads and SR contigs, on the one hand, and in the long reads and LR contigs, on the other, we used the program metaxa2 [26] to extract and taxonomically bin 16S sequences. We then computed the Shannon index based on the genus-level bins. The values for the short reads, SR contigs, LR reads, and LR contigs are 2.9, 3.9, 3.4, and 2.3, respectively. This indicates that the short-read dataset captures more diversity than the long-read dataset.

Comparison of SR contigs and LR chromosomes

To verify the correctness of the seven LR chromosomes, we aligned them against the set of SR contigs using Minimap2, as described above in the section on repeat analysis, and present the results using dot-plots in Fig. 5. These plots indicate a perfect concordance between the LR chromosomes and corresponding SR contigs. (What appear to be breaks in four of the diagonals are artifacts due to "wraparound" in the circular chromosomes.)

For each of the seven LR chromosomes, we aligned all corresponding SR contigs against the corresponding reference genomes using Minimap2 (as described above) and find significant alignments only for SR contigs corresponding to the LR chromosomes 1, 2, and 5. This supports the conclusion that only three of the LR chromosomes are present in the current reference databases.

Metagenomic binning of short-read assembly

Genome binning was performed on all SR contigs that were at least 2 kb in length using MetaBAT [27] (v2.12.1, using default parameters). This was followed by bin evaluation using CheckM (v1.0.11) (default parameters except `lineage_wf -t 29`). This gave rise to 80 bins, of which 21 (26%) fulfill the definition of "high quality" and 14 (18%) are considered "medium quality" [15]. We performed a CheckM analysis of these bins, and the result is reported in Additional file 7: Table S7.

We screened for 16S genes within the SR contigs using the USEARCH [28] module `--search16s` (v 10.0.240, 64 bit), and annotated these sequences using Silva.

In addition, for ease of comparison with the long read results summarized in Table 1, we also performed DIAMOND + MEGAN taxonomic binning of the SR contigs (using the same parameters as for the LR contigs, but without frame-shift correction), followed by CheckM analysis, and present the results in Additional file 8: Table S8.

Measuring the concordance between SR bins and LR chromosomes

Here, we introduce the *concordance score*, which provides a measure of concordance between SR bins and LR contigs. In more detail, we used BLASTN [29] (version 2.4.0+, default parameters) to align all SR contigs (as queries) against all LR contigs (as subjects), retaining only the best hit for each pair of sequences. Based on this, for each SR bin and LR contig, we computed four scores:

Table 3 Comparison of LR and SR coverage

Bin	MEGAN assignment	Completeness longest contig (%)	GC content (%)	LR coverage	SR coverage
B1	*Bacteroidetes bacterium* OLB12	95	43.6	57.3	117.5
B2	*Candidatus Accumulibacter* SK-02	94	61.3	384.2	707.8
B3	*Chlamydiia* (class)	94	38.2	48.8	107.6
B4	*Gammaproteobacteria* (class)	93	40.5	25.1	56.2
B5	*Bacteroidetes bacterium* OLB8	93	41.2	52.1	109.9
B6	*Rhodospirillales* (order)	92	63.6	29.5	56.1
B7	*Chlorobi bacterium* OLB5	88	38.1	38.7	90.2
B8	*Thauera* (genus)	14	68.9	32.7	60.5
B9	*Sphingobacteriales bacterium* 44-15	0	40.6	10.2	23.2
B10	*Bacteroidetes* (phylum)	32	43.5	14.1	27.7
B11	*Candidatus Contendobacter* B J11	19	62.6	15.4	22.3
B12	*Betaproteobacteria* (class)	10	62.9	37.1	66.0
B13	*Nitrospira* (genus)	27	60.4	17.6	28.7
B14	*Chloroflexi* (phylum)	8	51.8	13.3	18.9

For the longest contigs in each of the 14 bins reported in Table 1, we report the completeness as determined by CheckM, the average CG content, the average long-read coverage, and the average short-read coverage

- The average ratio of the alignment length to the length of the SR contig,
- The average sequence identity reported by BLASTN,
- The proportion of the LR contig that is covered by aligned SR contigs, and
- The proportion of the SR contigs in the bin that are aligned on the LR contig.

The concordance score κ is then defined as the mean of these four values. So, for a given LR contig, if we select an SR bin whose concordance score κ is close to 1, then that bin will consist mostly of contigs that tile the LR contig at a high level of sequence identity. We also use κ to measure the concordance between the contigs of a reference genome assembly and a LR chromosome.

Comparison of SR bins and LR chromosomes

LR chromosome 1 is contained in the MEGAN taxonomic bin labeled *Bacteroidetes bacterium* OLB12. This LR chromosome is tiled by contigs from SR bin 52 (a medium quality "metagenome-assembled genome" (MAG), with a concordance score of $\kappa = 0.88$), and from SR bin 3 ($\kappa = 0.84$), which cover the first third and second two-thirds of the LR chromosome, respectively. SR bin 52 is annotated by CheckM to UID2570, which is selective for members of phyla *Chlorobi*, *Bacteroidetes*, and *Ignavibacteriae*, and thus taxonomically ambiguous. See Additional file 10: Figure S2a.

LR chromosome 2 is contained in the MEGAN taxonomic bin labeled *Candidatus Accumulibacter* sp. SK-02, and is tiled by contigs in SR-bin 32 (high quality MAG, $\kappa = 0.97$). CheckM annotates this to lineage marker set UID3971, which is selective for *Accumulibacter*, *Dechloromonas* and *Azospira*, all contained in the Order of *Rhodocyclaceae*. See Additional file 10: Figure S2b. Examination of the alignments between LR chromosome 1 and the closely related SR-bin 31 ($\kappa = 0.76$) shows that the contigs from SR-bin 31 fill a major gap in the coverage of LR-chromosome 1 by the members of SR-bin 32. See Additional file 10: Figure S2c. This suggests that SR-bin 32 and SR-bin 31 should be a single bin. The closest reference genome identified by MEGAN-LR is GCA_000584975.1 (*Candidatus Accumulibacter* sp. SK-02), with $\kappa = 0.90$.

LR chromosome 3 is contained in the MEGAN-LR taxonomic bin labeled *Chlamydiia* and is covered by contigs from SR-bin 35 (high quality MAG, $\kappa = 0.98$). SR bin 35 is annotated by CheckM to UID2982, which selects for members of phylum *Chlamydiae* and phylum *Verrucomicrobia*. We confirmed that LR chromosome 6 (and SR bin 35) are members of phylum *Chlamydiae* using a Minimap2 alignment against all extant reference or draft genomes in the PVC superphylum (data not shown). See Additional file 10: Figure S2d.

LR chromosome 4 is contained in the MEGAN-LR taxonomic bin labeled *Gammaproteobacteria* and is covered by contigs from SR-bin 43 (high quality MAG, $\kappa = 0.97$), which is annotated to *Gammaproteobacteria* by CheckM via UID4266. See Additional file 10: Figure S2e.

LR chromosome 5 is contained in the MEGAN-LR taxonomic bin labeled *Bacteroidetes bacterium* OLB8, is

aligned to by SR-bin 66 (high quality MAG, $\kappa = 0.98$), which is annotated to *Bacteroidetes* by CheckM via UID2591. See Additional file 10: Figure S2f.

LR chromosome 6 is contained in the MEGAN-LR taxonomic bin labeled *Rhodospirillales*. This LR chromosome is tiled by contigs from SR-bin 23 (high-quality MAG, $\kappa = 0.95$), which is annotated to the order of *Rhodospirillales* by CheckM. SR-bin 23 contains a full length 16S sequence, which Silva assigns to the genus *Defluviicoccus* (a member of order *Rhodospirillales*). See Additional file 10: Figure S2g.

LR chromosome 7 is contained in the MEGAN-LR taxonomic bin labeled *Chlorobi bacterium* OLB5. While there is a good coverage of this LR chromosome by SR contigs, these are not contained in any SR-bin identified by MetaBAT. The closest reference genome, GCA_001567546, has a κ value of only 0.36. See Additional file 10: Figure S2h.

Additional files

Additional file 1: Table S1. Summary of LR read data. (TXT 1 kb)

Additional file 2 : Table S2. Summary of LR contig data. (TXT 1 kb)

Additional file 3: Table S3. Summary of LR contig taxonomic bins computed using DIAMOND + MEGAN-LR. (TXT 5 kb)

Additional file 4: Table S4. CheckM results for all 106 LR contig taxonomic bins. (TXT 24 kb)

Additional file 5: Table S5. Summary of short-read data. (TXT 1 kb)

Additional file 6: Table S6. Potential break-points in seven LR chromosomes, inferred as locations that have no short read clone coverage. (TXT 2 kb)

Additional file 7: Table S7. Summary of short-read assembly binning using MetaBAT. (TXT 7.94 kb)

Additional file 8: Table S8. SR assembly statistics and CheckM results for 14 taxonomic bins. (PDF 35.7 kb)

Additional file 9: Figure S1. Using Minimap2, we aligned all SR contigs against all LR reads and LR contigs. Here, we show, for a given level of average coverage of a SR contig by short reads, how many bases of the SR-contigs align to long reads only ("in LR"), or to LR contigs ("in LR contigs"), or not ("not in LR"). There are 221 SR contigs that have a coverage greater than 150 but are not shown in the plot. They cover 5.3 Mb in total, of which 49.6% is aligned to long reads and 50.4% to LR contigs. (PDF 20 kb)

Additional file 10: Figure S2. Concordance statistics for SR contigs against LR chromosomes. In each plot, the LR chromosome is represented by the *x* axis, and the five panels, from top to bottom, represent: (A) the locations of alignments to the LR chromosome, (B) the corresponding percent identity, (C) the alignment-length to query-length ratio, (D) the alignment length and (E) the query length. The colors red and black are used to distinguish between alignments to different SR-bins or reference genomes, as described in the text. (PDF 72 kb)

Additional file 11: Figure S3. Plot of LR contig length vs concordance score κ; highlighting pairs of LR chromosomes/contigs and SR bins or references that show high levels of concordance. (PDF 378 kb)

Acknowledgements
We thank Daniela Drautz-Moses and colleagues for Illumina library preparation and short read sequencing.

Funding
This work was supported in part by the Singapore National Research Foundation and Ministry of Education under the Research Centre of Excellence Programme, and by a program grant from the Environment and Water Industry Programme Office (EWI), project number 1301-RIS-59. The computational work was partially performed on resources of the National Supercomputing Centre, Singapore (https://www.nscc.sg). The authors acknowledge support by the High Performance and Cloud Computing Group at the Zentrum für Datenverarbeitung of the University of Tübingen, the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant no INST 37/935-1 FUGG and grant no HU 566/12-1.

Availability of data and materials
Data from this paper study are available from the NCBI via BioProject accession `PRJNA509764` and Short Read Archive accessions `SRX5120474` and `SRX5126404` for long and short read data, respectively.
DIAMOND, including all modifications introduced in this paper, is open source and available here: `https://github.com/bbuchfink/diamond`.
MEGAN Community Edition, including the algorithm for frame-shift correction introduced in this paper, is open source and available here: `http://ab.inf.uni-tuebingen.de/data/software/megan6/download`.
Files containing the LR contigs and SR contigs, the two corresponding meganized DIAMOND files, and the output of MetaBAT on the SR contigs, can be downloaded here: https://ab.inf.uni-tuebingen.de/data/external/Arumugam-et-al-2019/.

Authors' contributions
GQ developed and performed the enrichment reactor experiment and obtained samples IB designed and performed the sequencing experiment. RBHW and DHH designed the analysis strategies. KA, CB, SB, AG, DHH, and RBHW performed the data analysis. BB implemented the frame-shift alignment and range-culling in DIAMOND. DHH implemented frame-shift correction in MEGAN. DHH and RBHW wrote the manuscript, and all other authors contributed. All authors read and approved the final manuscript.

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note

Author details
[1]Singapore Centre for Environmental Life Sciences Engineering, Nanyang Technological University, 60 Nanyang Drive, SBS-01N-27, Singapore 637551, Singapore. [2]Department of Computer Science, University of Tübingen, Sand 14, 72076 Tübingen, Germany. [3]International Max Planck Research School From Molecules to Organisms, Max Planck Institute for Developmental Biology and Eberhard Karls University Tübingen, Max-Planck-Ring 5, 72076 Tübingen, Germany. [4]Singapore Centre for Environmental Life Sciences Engineering, National University of Singapore, 28 Medical Drive, Singapore 117456, Singapore. [5]Max-Planck-Institute for Developmental Biology, Max-Planck-Ring 5, 72076 Tübingen, Germany. [6]Life Sciences Institute, National University of Singapore, 28 Medical Drive, Singapore 117456, Singapore.

Received: 4 January 2019 Accepted: 11 March 2019
Published online: 16 April 2019

References

1. The Human Microbiome Project Consortium. Structure, function and diversity of the healthy human microbiome. Nature. 2012;486(7402):207–14.
2. Willmann M, El-Hadidi M, Huson DH, Schütz M, Weidenmaier C, Autenrieth IB, Peter S. Antibiotic selection pressure determination through sequence-based metagenomics. Antimicrob Agents Chemother. 2015;59(12):7335–45.
3. Huson DH, Beier S, Flade I, Górska A, El-Hadidi M, Mitra S, Ruscheweyh HJ, Tappu R. MEGAN Community Edition - interactive exploration and analysis of large-scale microbiome sequencing data. PLoS Comput Biol. 2016;12(6):1004957.

4. Quince C, Walker AW, Simpson JT, Loman NJ, Segata N. Shotgun metagenomics, from sampling to analysis. Nat Biotechnol. 2017;35:833.
5. Jain M, Olsen HE, Paten B, Akeson M. The Oxford Nanopore MinION: delivery of nanopore sequencing to the genomics community. Genome Biol. 2016;17:239.
6. Wick RR, Judd LM, Gorrie CL, Holt KE. Unicycler: resolving bacterial genome assemblies from short and long sequencing reads. PLoS Comput Biol. 2017;13(6):1–22.
7. Li H. Minimap2: pairwise alignment for nucleotide sequences. Bioinformatics. 2018;1:7.
8. Vaser R, Sović I, Nagarajan N, Šikić M. Fast and accurate de novo genome assembly from long uncorrected reads. Genome Res. 2017;27(5):737–46.
9. Parks DH, Imelfort M, Skennerton CT, Hugenholtz P, Tyson GW. Assessing the quality of microbial genomes recovered from isolates, single cells, and metagenomes. Genome Res. 2014;25:1043–55.
10. Seemann T. Prokka: rapid prokaryotic genome annotation. Bioinformatics. 2014;30(14):2068–9.
11. Buchfink B, Xie C, Huson DH. Fast and sensitive protein alignment using DIAMOND. Nat Methods. 2015;12:59–60.
12. Kielbasa SM, Wan R, Sato K, Horton P, Frith MC. Adaptive seeds tame genomic sequence comparison. Genome Res. 2011;21(3):487–93.
13. Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Wheeler DL. Genbank. Nucleic Acids Res. 2005;1(33):34–8.
14. Huson DH, Albrecht B, Bağcı C, Bessarab I, Górska A, Jolic D, Williams RBH. MEGAN-LR: new algorithms allow accurate binning and easy interactive exploration of metagenomic long reads and contigs. Biol Direct. 2018;13(1):6.
15. Bowers RM, Kyrpides NC, Stepanauskas R, Harmon-Smith M, Doud D, Reddy TBK, Schulz F, Jarett J, Rivers AR, Eloe-Fadrosh EA, Tringe SG, Ivanova NN, Copeland A, Clum A, Becraft ED, Malmstrom RR, Birren B, Podar M, Bork P, Weinstock GM, Garrity GM, Dodsworth JA, Yooseph S, Sutton G, Glöckner FO, Gilbert JA, Nelson WC, Hallam SJ, Jungbluth SP, Ettema TJG, Tighe S, Konstantinidis KT, Liu WT, Baker BJ, Rattei T, Eisen JA, Hedlund B, McMahon KD, Fierer N, Knight R, Finn R, Cochrane G, Karsch-Mizrachi I, Tyson GW, Rinke C, Consortium TGS, Schriml L, Hugenholtz P, Yilmaz P, Meyer F, Lapidus A, Parks DH, Murat Eren A, Banfield JF, Woyke T. Minimum information about a single amplified genome (misag) and a metagenome-assembled genome (mimag) of bacteria and archaea. Nat Biotechnol. 2017;35:725–31.
16. Skennerton CT, Barr JJ, Slater FR, L BP, Tyson GW. Expanding our view of genomic diversity in *Candidatus* accumulibacter clades. Environ Microbiol. 2015;17:1574–85.
17. Speth DR, in 't Zandt MH, Guerrero-Cruz S, Dutilh BE, Jetten MSM. Genome–based microbial ecology of anammox granules in a full-scale wastewater treatment system. Nat Commun. 2016;7:11172.
18. Collingro A, Poppert S, Heinz E, Schmitz-Esser S, Essig A, Schweikert M, Wagner M, Horn M. Recovery of an environmental chlamydia strain from activated sludge by co-cultivation with *acanthamoeba sp.* Microbiology. 2005;151:301–30.
19. Burow LC, Kong Y, Nielsen JL, Blackall LL, Nielsen PH. Abundance and ecophysiology of Defluviicoccus spp, glycogen-accumulating organisms in full-scale wastewater treatment processes. Microbiology. 2007;153(1):178–85.
20. Quast C, Pruesse E, Yilmaz P, Gerken J, Schweer T, Yarza P, Peplies J, Glöckner FO. The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. NAR. 2013;41:590–6.
21. Tu Y, Schuler AJ. Low acetate concentrations favor polyphosphate-accumulating organisms over glycogen-accumulating organisms in enhanced biological phosphorus removal from wastewater. Environ Sci Technol. 2013;47(8):3816–24.
22. Lu H, Oehmen A, Virdis B, Keller J, Yuan Z. Obtaining highly enriched cultures of candidatus accumulibacter phosphates through alternating carbon sources. Water Res. 2006;40(20):3838–48.
23. Wick RR, Judd LM, Gorrie CL, Holt KE. Completing bacterial genome assemblies with multiplex minion sequencing. Microb Genomics. 2017;10:000132–000132000132.
24. Walker BJ, Abeel T, Shea T, Priest M, Abouelliel A, Sakthikumar S, Cuomo CA, Zeng Q, Wortman J, Young SK, et al. Pilon: an integrated tool for comprehensive microbial variant detection and genome assembly improvement. PloS ONE. 2014;9(11):112963.
25. Bankevich A, Nurk S, Antipov D, Gurevich AA, Dvorkin M, Kulikov AS, Lesin VM, Nikolenko SI, Pham S, Prjibelski AD, Pyshkin AV, Sirotkin AV, Vyahhi N, Tesler G, Alekseyev MA, Pevzner PA. SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. J Comput Biol J Comput Mol Cell Biol. 2012;19(5):455–77.
26. Bengtsson-Palme J, Hartmann M, Eriksson KM, Pal C, Thorell K, Larsson DGJ, Nilsson RH. metaxa2: improved identification and taxonomic classification of small and large subunit rRNA in metagenomic data. Mol Ecol Resour. 2015;15(6):1403–14.
27. Kang DD, Froula J, Egan R, Wang Z. MetaBAT, an efficient tool for accurately reconstructing single genomes from complex microbial communities. PeerJ. 2015;3:1165.
28. Edgar R. Search and clustering orders of magnitude faster than BLAST. Bioinformatics (Oxford, England). 2010;26(19):2460–1.
29. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res. 1997;25:3389–402.
30. Wick RR, Schultz MB, Zobel J, Holt KE. Bandage: interactive visualization of de novo genome assemblies. Bioinformatics. 2015;31(20):3350–2.

Appendix III

Albrecht *et al. BMC Bioinformatics* 2020, **21**(Suppl 13):390
https://doi.org/10.1186/s12859-020-03684-2

BMC Bioinformatics

RESEARCH Open Access

MAIRA- real-time taxonomic and functional analysis of long reads on a laptop

Benjamin Albrecht[1†], Caner Bağcı[1,2†] and Daniel H. Huson[1*]

From The 18th Asia Pacific Bioinformatics Conference
Seoul, Korea. 18–20 August 2020

*Correspondence:
daniel.huson@uni-tuebingen.de
†Benjamin Albrecht and Caner Bağcı contributed equally to this work.
[1]Department of Computer Science, University of Tübingen, Sand 14, Tübingen, Germany
Full list of author information is available at the end of the article

Abstract

Background: Advances in mobile sequencing devices and laptop performance make metagenomic sequencing and analysis in the field a technologically feasible prospect. However, metagenomic analysis pipelines are usually designed to run on servers and in the cloud.

Results: MAIRA is a new standalone program for interactive taxonomic and functional analysis of long read metagenomic sequencing data on a laptop, without requiring external resources. The program performs fast, online, genus-level analysis, and on-demand, detailed taxonomic and functional analysis. It uses two levels of frame-shift-aware alignment of DNA reads against protein reference sequences, and then performs detailed analysis using a protein synteny graph.

Conclusions: We envision this software being used by researchers in the field, when access to servers or cloud facilities is difficult, or by individuals that do not routinely access such facilities, such as medical researchers, crop scientists, or teachers.

Keywords: Metagenomics, Microbiome, Long read sequencing, Taxonomic analysis, Functional analysis, Mobile computing, Open source software, Antibiotic resistance, Virulence factors

Background

The Oxford Nanopore MinION USB sequencing device, the MinIT USB base-calling device and advances in lab-on-a-chip technologies allow sequencing to be taken into field [1]. With ever rising sequencing yield and continuously growing reference databases, the computational analysis of such data is challenging, and much work is being done to address this efficiently, usually on a server or using cloud computing [2, 3]. Modern laptops provide a lot of computational power, main memory and fast access to SSD storage, and thus it should be possible to perform detailed analysis of microbiome sequencing data in the field, on a laptop.

In this paper, we present a new program called MAIRA (mobile analysis of long reads) for analyzing long metagenomic reads on a laptop, without requiring external resources. The program first performs fast genus-level analysis in real time, and then also performs detailed species-level taxonomic and functional analysis. The latter refers to basic analysis of antibiotic resistance potential [4] and virulence factors [5].

We envision this software being used by researchers in the field, when access to servers or cloud facilities is difficult [6], or by individuals that do not routinely use such facilities, such as medical researchers, crop scientists, or teachers, say. The program is able to analyze sequencing data in real-time and thus may be applicable in time-critical applications.

Following the approach established in [7, 8], we base the taxonomic and functional alignment of metagenomics sequencing reads on the alignment of the reads against a protein reference database; here we use all bacterial proteins in RefSeq [9].

There are three main challenges.

1. Current long reads contain many erroneous insertions and deletions, causing frame shifts that disrupt translated alignments and result in very poor performance of methods such as BLASTX [10].
2. The alignment of gigabases of sequencing reads against a reference database containing on the order of one hundred million reference proteins requires substantial computational resources, despite major advances [11, 12].
3. For many applications, such as the assessment of antibiotic resistance or virulence, say, it is crucial to link the associated genes to specific organisms.

Here we describe a new open source program called MAIRA that addresses these three main challenges.

Implementation

Challenge 1 can be solved using a frame-shift aware alignment tool [13], and MAIRA supports the use of LAST [11], but also contains a new, built-in frame-shift aligner called ELLA (manuscript in preparation). To address Challenge 2, the program performs a fast online genus-level analysis, whereas a more detailed analysis can be computed on-demand, for selected genera of interest. To address Challenge 3, we introduce the concept of a protein synteny graph (manuscript in preparation) whose nodes represent gene families and whose edges reflect synteny.

MAIRA is a standalone computer program for analyzing sequencing reads from a mobile sequencing device or long reads from other sources. MAIRA is written in Java and runs on all three major operating systems. The program is designed to be used interactively, however a command-line mode is also provided.

Online and on-demand

Initial analysis is performed "online" in the sense that sequencing reads are processed incrementally in batches, in real-time, as they become available. The displayed results are updated after completion of genus-level analysis of each batch. Detailed analysis is performed "on-demand" in the sense that the user may select or prioritize the genera to be analyzed in detail, so as to focus the laptop's computational resources on the analysis

of specific organisms of interest. The displayed results are updated after completion of the analysis of each batch.

We will now discuss the main workflow of MAIRA, summarized in Fig. 1.

Input

The input consists of files of sequences in FastA or FastQ format. Input can be specified as a single file or a directory of input files. All reads are then loaded and processed by the program in batches. Alternatively, one can specify a "monitor directory" that the program will periodically inspect to determine whether any new files of reads are available, which are then processed in batches. In particular, when sequencing with a MinION and running a MinIT to perform base-calling, the MinIT can write FastQ files onto its local hard drive, which can be accessed remotely from a laptop and MAIRA will load its input from there.

Genus-level alignment and analysis

As indicated in Fig. 1a, MAIRA processes incoming reads in batches (by default, 10,000 reads per batch). These reads are aligned against a precomputed database of genus-marker genes, currently representing 2,418 genera, based on the bacterial RefSeq protein database.

Frame-shift aware DNA-to-protein alignment is performed either using LAST [11] or the built-in alignment tool ELLA (manuscript in preparation). For aligning reads against a database both aligners use a frame-shift penalty of 15 and a multiplicity value of 10 (for the marker database) and 50 (for genus-specific databases, see below), respectively. ELLA is written in Java and uses the same algorithmic approach as LAST. It has similar specificity and sensitivity, but is about 2.5 times slower. We have implemented and incorporated ELLA so as to make MAIRA completely self-contained and platform-independent.

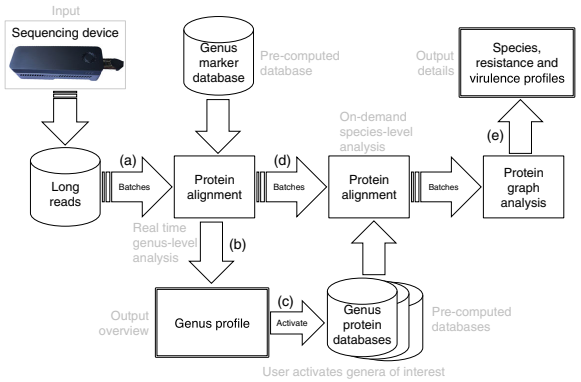


Fig. 1 MAIRA workflow. In an online manner, **a** batches of long reads obtained from a sequencing device are subjected to protein alignment against a precomputed database of genus-specific marker genes. Based on this, **b** the list of detected genera is updated in real-time. **c** Substantial evidence for the presence of a specific genus triggers **d** batch-wise protein alignment against a full database of proteins associated with the genus. Based on this, a protein graph is updated and used to perform species and-strain level analysis of the long reads, including the identification of antibiotic resistance and virulence factors

The marker database that we provide for download contains 9,434,634 marker genes. All results presented in this paper were obtaining using this database.

The database was precomputed from data downloaded from RefSeq in February 2019, comprising 142,092 bacterial genomes in 2,420 different genera. For each genus, we first constructed a graph whose nodes are proteins and any two nodes v and w are connected by an edge from v to w, if the two protein sequences have a similarity of $\geq$ 90% and if v covers at least 90% of w, using DIAMOND [12] in BLASTP mode.

A set of candidate marker genes was then computed by greedily determining a minimum *dominating set* of nodes, that is, a subset of nodes D such that every node of the graph is either contained in D, or is adjacent to some node in D. We then turned each candidate set of proteins into a set of genus representative proteins. This was done by aligning all pairs of candidate sets from different genera against each other using DIAMOND, applying an identity threshold of 80% and a coverage threshold of 80%, so as to detect and remove proteins that are not unique to a single genus.

Based on these sets of representative proteins, we then computed the marker database. Using the graph described above, we first removed all nodes that are not associated with a representative protein. Next, we greedily selected a minimum set of nodes so that all genomes are covered by a fixed minimum number of proteins, namely 1000, 500 or 100, for each of the large, medium or small database, respectively. Here, a genome is considered covered by one of its proteins v, if v is selected, or if v is adjacent to a selected node.

This resulted in 9,434,634 genus-specific *marker proteins* that represent 2,418 genera, with a median of 1,121 (mean 3,903) marker proteins (s.d. 10,588). There are 4 genera that contain less than 100 proteins and there are 6 genera that contain more than 100,000 proteins, with Bacillus (205,864), Streptomyces (194,850), and Lactobacillus (158,640) having the largest counts.

While aligning reads against the genus-marker database, for each genus G, we maintain and report a *presence score* p that heuristically seeks to approximate the probability of genus G being present in the sample. This is calculated as follows: First, each marker protein v associated with G is given a *weight* $\omega(v) = \frac{1}{\bar{n}}$, where $\bar{n}$ is the average number of marker proteins over all genomes that either contain the marker protein v itself or a similar one (both, similarity and coverage $\geq$ 90%). The presence score for G is then given by the sum of weights of all marker proteins for G to which a read has been aligned to.

The genus-level analysis of the content of the sample being sequenced is summarized on a tree in the main window of the program. Within minutes, the program will provide a first crude estimation of the genus-level content of the sample being sequenced (or read from an input file). With the completion of genus-level analysis of each batch, this summary is updated.

Species-level alignment

The above described genus-level analysis is designed to provide a very quick taxonomic analysis of long read sequencing data, akin to traditional 16S rRNA taxonomic profiling.

In addition, the program is able to perform an in-depth, protein-alignment-based taxonomic and functional analysis of the incoming reads, in batches. To save computational resources, the in-depth analysis is performed on demand, for a specified subset of taxa.

In more detail, using the bacterial RefSeq protein database, for each genus we pooled all proteins that either correspond to the genus itself, or to an ancestor or descendant taxon. We obtained a set of 2,420 genus-specific databases, each containing an average of 43,849 proteins (median 6,310, s.d. 251,520).

Batches of incoming reads (Fig. 1d) are aligned against the reference databases for all genera that have been "activated" (Fig. 1c), either explicitly by the user, or automatically, because their presence score has exceeded the activation threshold (80%, by default). The program tracks all batches of reads and ensures that each batch of reads will eventually be compared against every activated genus-specific database, independent of when the batch of reads was loaded into the program or when the genus was activated.

Protein synteny graph

The alignments computed during species-level analysis are used to build a *protein synteny graph PSG*. In this graph, each node represents a protein gene family. Gene families are defined as follows: The alignments produced from the genus specific databases for each long-read are pooled together, and filtered by the coverage of the reference protein (default: 80%), the percentage of positives in the alignments (default: 60%), and the raw-score of the alignment (default: 100). The alignments that stack (at least 2/3 of the smaller alignment overlapping with the longer alignment) at different loci are then binned together so as to obtain gene families. These bins are represented as *protein nodes* in the graph. Each such *protein node* v is annotated by the set $\tau(v)$ of all taxa and functional classes associated with members of the gene family.

A protein node is placed into the graph, if there exists a sequencing read that aligns to a gene sequence in the corresponding family. Two such nodes v and w in the graph are joined by an undirected edge $\{v, w\}$ if the two corresponding gene families appear next to each other in some sequencing read. Each edge e is annotated by the set $\rho(e)$ of all such reads.

Let t be a taxon. The taxon-specific *induced* protein synteny graph $PSG|_t$ is given by the set of all protein nodes v whose annotation contains the taxon t, that is, for which $t \in \tau(v)$ holds, together with all other nodes that lie between them in the graph along some read. Any two nodes in the resulting graph are connected by an edge, if and only if they were in the original graph.

Taxonomic analysis

Using these concepts, we declare a taxon t to be *present* in the sample, if the corresponding induced graph $PSG|_t$ contains enough nodes. In more detail, we define the naïve *completeness score* $k(t)$ as the number of nodes in the induced graph divided by the median number of proteins present for the same or (a similar) taxon in the RefSeq database. A taxon is considered present, if its completeness score exceeds a specified threshold (80%, by default).

We further compare all induced graphs to all others that are above the detection criteria, and eliminate those that are largely contained within the induced graph of another taxon (default: 85% of the nodes), without containing the same percentage of the nodes from the other itself. We do this in order to eliminate false-positive taxa that can still produce highly connected and complete subgraphs due to partial but not complete similarity to the true positive taxon.

Antibiotic resistance and virulence analysis

To identify potential antibiotic resistance genes, we use information from the *comprehensive antibiotic resistance database* (CARD) [4] to annotate nodes that represent resistance-associated genes. Similarly, to identify potential virulence factors, we use information from the *virulence factor database* (VFDB) [5] to annotate nodes that are considered virulence factors. For a given taxon t, we report all CARD and VFDB annotations present in the induced graph $PSG|_t$.

Results

We demonstrate the use of the program using three datasets.

Nanopore mock community

The Nanopore mock community published in [14] consists of 3,491,078 long reads with an average length of 4,012 bases, sequenced on a GridION. The source consists of eight bacterial strains and two yeast strains.

We ran our pipeline on 25 batches of 10,000 reads each, using both LAST and ELLA. In both cases, MAIRA correctly determined all 8 present genera of bacteria after the third batch (requiring 5 minutes using LAST and 11 minutes using ELLA). Both aligners also resulted in false-positive identification of 2 genera after the third batch: *Enterobacter* and *Shigella*. Another false-positive *Citrobacter* also passed the default genus detection threshold of 0.8 for the 15-th batch for ELLA and 11-th batch for LAST (see Fig. 2a).

Running the species-level analysis to completion on 25 batches required $9\frac{1}{2}$ hours using LAST and nearly 26 hours using ELLA. All eight true-positive species were identified with strong signal (completeness >90%) using both aligners after the end of the 4-th batch, after one hour for LAST and three hours for ELLA; and their scores thereafter remained constant.

MAIRA identified one false-positive species, *Bacillus amyloliquefaciens*, using both aligners. Its completeness score, however, stayed below 0.75 using ELLA and 0.85 using LAST (see Fig. 2b). We report the coverage of genomes in Fig. 2c.

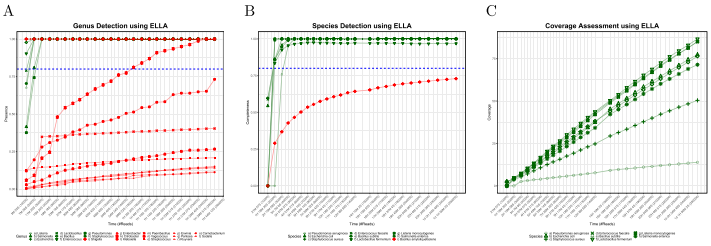


Fig. 2 Performance on mock community. **a** The presence score for true positive (green) and false positive (red) genera as a function of computation time and number of reads processed. The blue line indicates the threshold used to deem a genus "present". **b** The completeness of true positive (green) and false positive (red) species as a function of computation time and number of reads processed. The blue line indicates the threshold used to deem a species "present". **c** The coverage of species, as a function of time and number of reads processed

Running the software on further batches eventually produces one more false positive genus *Klebsiella*.

Simulated pathogen mock community

We now consider the simulation of a pathogen mock community. We simulated an even community consisting of ten different bacterial species, each represented by 10,000 simulated Nanopore reads generated using NanoSim [15]. The ten source genomes correspond to the components of a commercially available metagenomic mock community for pathogen detection called ATCC® MSA-4000™. This consists of 11 strains of 10 species in eight different genera. The genome sequence of one of the strains (ATCC BAA-1718) is very fragmented (174 contigs) and so we excluded it from our study.

Complete analysis of all 100,000 simulated long reads took slightly over six hours using MAIRA and LAST. The approach correctly determined all eight genera at the end of the 7-th batch, while indicating two incorrect genera, *Shigella* and *Enterobacter*. The species-level analysis resulted in identification of all 10 species present, while indicating three incorrect species, *Staphylococcus mitis*, *Streptococcus pseudopneumoniae* and *Streptococcus oralis*. Note that the dataset contains multiple strains of both *Staphylococcus* and *Streptococcus*.

To assess the ability of MAIRA to correctly determine the presence of CARD or VFDB genes, for both functional classifications, we aligned the set of their reference proteins against all coding sequences (CDSs) in the source genomes, using DIAMOND [12]. Then, for both classifications, any protein that aligned to a CDS with a percent positives of $\geq$ 80% and a reference coverage above $\geq$ 90% was considered an actual positive.

For each species, MAIRA recovered almost all actual positives for both CARD and VFDB, with mostly low numbers of false positives, see Table 1.

Carbapenemase-producing gram-negative bacteria isolates

The long read dataset published in [16] consists of 110 MinION long read sequencing datasets, of which we were able to download 109 (sample ERR2797062 could not be found). These datasets were sequenced from 58 *Klebsiella pneumoniae*, 28 *Escherichia coli*, 13 *Pseudomonas aeruginosa*, and 10 *Acinetobacter baumannii* isolates. The authors report that 63 isolates are suspected to have carbapenemase resistance, 34 have other types of antibiotic resistance, and indicate that they failed to produce evidence for antibiotic resistance in 13 isolates.

We ran MAIRA on the first 10,000 reads of each sample (or all reads, where there were less than 10,000). It took about one minute per sample to complete genus-level analysis using MAIRA and LAST.

For all 109 samples, MAIRA reports the correct species and no false positives. However, the initial, fast genus-level analysis did report some false positive genera. For example, in the case of *Klebsiella pneumoniae*, three false positives were listed. None of the false positive genera were confirmed by the species-level analysis. The full running time varied between five and 20 minutes per sample, depending on the organism.

In order to determine carpabanem-resistance of isolates, we checked for CARD ontology terms with annotation *confers_resistance_to: ARO:0000020* and found homology-based evidence of resistance in 102 isolates with varying ranges of support. The number of reads covering resistance genes ranged from 1 to 35.

Table 1 Assessment of CARD and VFDB hits reported by MAIRA

Organism	Classif.	TP (CDS)	FN (CDS)	TP (Term)	FP (Term)	FN (Term)
Acinetobacter baumannii	CARD	25	3	145	6	65
	VFDB	31	9	31	0	10
Enterococcus faecalis	CARD	22	5	24	2	26
	VFDB	23	3	23	0	5
Escherichia coli	CARD	69	6	81	22	22
	VFDB	102	27	123	0	36
Klebsiella pneumoniae	CARD	63	7	85	20	282*
	VFDB	33	8	34	0	8
Neisseria meningitidis	CARD	15	1	16	1	5
	VFDB	31	8	30	0	12
Pseudomonas aeruginosa	CARD	55	15	87	7	38
	VFDB	119	16	119	0	22
Staphylococcus aureus	CARD	27	2	32	2	7
	VFDB	57	5	55	0	5
Streptococcus agalactiae	CARD	4	7	4	4	20
	VFDB	31	0	32	0	1
Streptococcus pneumoniae	CARD	17	1	19	8	14
	VFDB	11	2	11	0	4
Streptococcus pyogenes	CARD	4	8	4	4	16
	VFDB	29	1	29	0	3

For all ten bacterial species present in a simulated pathogen mock community, we report the number of true postives (TP), false positives (FP) and false negatives (FN), based on the coding sequences (CDS) in the source genomes, and based on the terms reported (Term). (*) The large false negative value for CARD terms reported for *Klebsiella pneumoniae* is based on a single CDS that is annotated with over 250 different CARD terms (WP_004176269.1) and thus gives an exaggerated impression

Discussion

In Fig. 3 we show a screen shot of the user interface. The user sets up an analysis using the dialog shown at the bottom left. The program then processes the given data in batches, frequently updating the tables and trees that represent the taxonomic and functional analysis of the data. The user can interactively explore the taxonomic and functional analyses (CARD and VFDB), can export all tables and trees in different formats, save and reopen the analysis.

With MAIRA, we intend to provide a powerful and accurate, standalone long read analysis software that runs on a laptop and can be used in the field. The use of a comprehensive protein reference database ensures wide applicability of the software. The two step approach, light-weight genus-level analysis, followed by on-demand species-level analysis of selected genera, allows the user to interactively control how the computational resources of the laptop are used.

The presented results suggest that the software is able to perform analysis of real data in acceptable running time and with very good accuracy. Alternative approaches currently rely on the use of servers or cloud resources to perform detailed analysis.

MAIRA is a general purpose tool. Moving forward, we intend to add special purpose modules that are dedicated to specific applications, such as the detection of specific pathogens. Moreover, antibiotic resistance is a complex question that is not adequately addressed by the tool at present. In future work, we intend to also consider DNA alignments of resistance-related genes to perform a more detailed analysis. Future versions

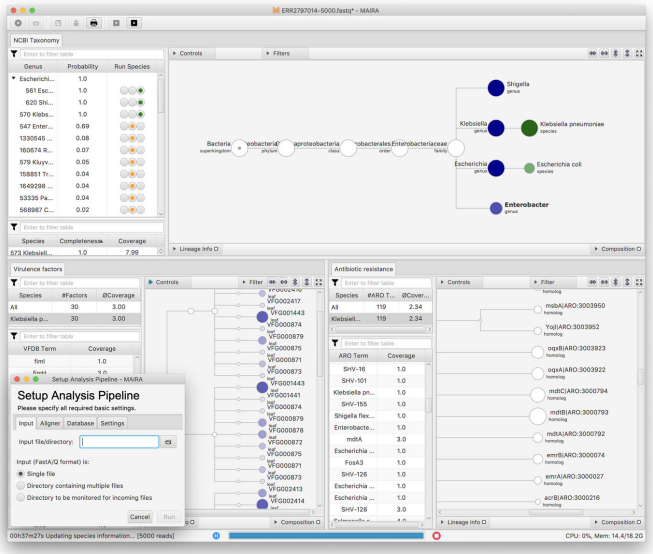


Fig. 3 MAIRA user interface. During processing of an input file or directory, MAIRA provides an overview of the genera, species and functional classes discovered so far. The user can explore and download results while they are being produced, so as to allow insights into the processed sequences

of the program will use locality of resistance genes and virulence factors in the protein synteny graph to provide more meaningful results.

Initial genus-level analysis performed by MAIRA is very fast and can be applied online during sequencing. Detailed, species-level analysis on individual genera is also fast enough to keep pace with a current MinION+MinIT sequencing setup. Complete analysis of one Gb of long read data requires about 10 hours using LAST and can be considered comfortably doable on a laptop.

One current limitation is that MAIRA focuses on identifying genera and species. Moving forward, we plan to lift this restriction so that assignments can be made to higher level taxonomic ranks, as well. Another limitation is given by the choice of reference databases. In this paper we describe the use of bacterial RefSeq, and the use of this database will avoid false positive identifications. If false negative results are to be avoided, then the use of the more comprehensive, but less curated, NR database may be more appropriate.

The data analysis presented here suggests the current algorithm for performing a fast genus-level analysis tends to yield false positive genera that are then not subsequently confirmed by the detailed species-level analysis. This indicates that there is room for improving the accuracy of the fast genus-level analysis and we will address this in future work.

Comparison with centrifuge

In order to further evaluate the performance of MAIRA, we compared it to Centrifuge [17], which is the backend for the real-time analysis software WIMP [18], developed by ONT. Centrifuge is designed to work with nucleotide databases on a server. We designed our evaluation so as to simulate a common obstacle in metagenomics, namely that the organism to be identified is not present in the reference databases. For this, we extracted genome assemblies from RefSeq that were published before 2016, and built MAIRA databases for them, as explained above. For Centrifuge, we were only able to build an index for the subset of complete assemblies, as using all assemblies produced an out-of-memory error on a server with 600GB of RAM. Also, the pre-built NCBI-nt index obtained from the Centrifuge website produced the same error.

We used NanoSim to collect 20x long reads on each of 100 different genomes available after 2016. These genomes were selected so as to equally cover four different categories, reflecting what information is available for the corresponding MAIRA and/or Centrifuge indices.

The 4 categories are: (a) no information at species level for MAIRA or Centrifuge; (b) no information at species level for Centrifuge, some for information for MAIRA from incomplete assemblies; (c) only one genome from the species for both MAIRA and Centrifuge; and (d) more than 15 genomes from the species for both MAIRA and Centrifuge. Species assignment accuracy was based on the top 10% of the scores of each tool (top 10% completeness for MAIRA, top 10% number of reads assigned for Centrifuge).

For category (a), MAIRA reported a false species in 5 cases, and did not report any assignment in the other 20; whereas Centrifuge reported a false assignment for all 25 genomes. For category (b), MAIRA reported the true assignment in 22 cases, failed to assign a species in 3 cases, and reported an additional species in 5 cases; whereas Centrifuge again reported a false assignment in all 25 cases. For category (c), MAIRA failed to identify the species in 3 cases, while Centrifuge failed in one case. MAIRA reported 4 false positives, while Centrifuge reported one false positive. For category (d), Centrifuge reported only the true positive species in all cases, whereas MAIRA failed to identify 4, and reported a false positive for one case.

In summary, Centrifuge shows better performance in the ideal case that the sequenced genomes are present in the reference DNA database. However, in the more realistic scenario in which the sequenced genomes are not present in verbatim in the reference databases, MAIRA shows better performance. Moreover, an added benefit of MAIRA is that it also provides a functional analysis, which Centrifuge does not attempt.

Conclusions

This paper demonstrates that laptop-based analysis of sequencing reads from mobile sequencing devices is possible and this should extend the range of mobile sequencing. As an easy-to-install, standalone application that runs on all three major operating systems, MAIRA provides a complete analysis solution that does not require access to additional computational resources.

Beyond the practical uses of the software, this work illustrates that protein-alignment-based analysis can be performed in real-time on a laptop, that an on-demand design allows a user to direct their computational resources to species of interest, and it shows that the protein synteny graph is a useful concept for the analysis of long reads. Moreover,

while designed for running on a laptop, the command-line version can also be run on a server, where the user can also benefit from its efficient design.

We believe that this type of software may play an important role in practical pathogen detection in the future [19].

Abbreviations
PSG: protein synteny graph; CDS: coding sequence; Gb: gigabase

Acknowledgements
Not applicable.

About this supplement
This article has been published as part of *BMC Bioinformatics Volume 21 Supplement 13, 2020: Selected articles from the 18th Asia Pacific Bioinformatics Conference (APBC 2020): bioinformatics*. The full contents of the supplement are available online at https://bmcbioinformatics.biomedcentral.com/articles/supplements/volume-21-supplement-13.

Authors' contributions
DHH proposed and guided the project and wrote the manuscript. BA and CB contributed to the writing. BA and CB wrote the software framework. BA designed and implemented the built-in alignment tool ELLA. CB designed and implemented the protein-graph approach. All authors read and approved the final manuscript.

Funding
The authors acknowledge hardware support by the High Performance and Cloud Computing Group at the Zentrum für Datenverarbeitung of the University of Tübingen, the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant no INST 37/935-1 FUGG. BA and CB salaries were supported the German Research Foundation (DFG) through grant no HU 566/12-1. Publication costs were supported by the Open Access Publishing Fund of University of Tübingen.

Availability of data and materials
The publicly available datasets used in the study are available from following BioProject accessions: PRJEB29504 and PRJEB28660.
Project name: MAIRA - mobile analysis of long reads.
Project home page: https://ab.inf.uni-tuebingen.de/software/maira
Operating system(s): Platform independent.
Programming language: Java (OpenJDK 12 and OpenJFX).
Other requirements: High-end laptop, ≥32 Gb of memory, 500G of SSD
License: GNU GPL
Any restrictions to use by non-academics: None

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Department of Computer Science, University of Tübingen, Sand 14, Tübingen, Germany. [2]International Max Planck Research School From Molecules to Organisms, Max Planck Institute for Developmental Biology and Eberhard Karls University Tübingen, Max-Planck-Ring 5, 72076 Tübingen, Germany.

Published: 17 September 2020

References
1. Tyler AD, Mataseje L, Urfano CJ, Schmidt L, Antonation KS, Mulvey MR, Corbett CR. Evaluation of Oxford Nanopore's MinION sequencing device for microbial whole genome sequencing applications. Sci Rep. 2018;8(1):1–12.
2. Huson DH, Beier S, Flade I, Górska A, El-Hadidi M, Mitra S, Ruscheweyh H-J, Tappu R. MEGAN community edition - interactive exploration and analysis of large-scale microbiome sequencing data. PLoS Comput Biol. 2016;12(6): 1004957.

3. Weber N, Liou D, Dommer J, MacMenamin P, Quinones M, Misner I, Oler AJ, Wan J, Kim L, Coakley McCarthy M, Ezeji S, Noble K, Hurt DE. Nephele: a cloud platform for simplified, standardized and reproducible microbiome data analysis. Bioinformatics. 2018;34(8):1411–3.
4. Jia B, Raphenya AR, Alcock B, Waglechner N, Guo P, Tsang KK, Lago BA, Dave BM, Pereira S, Sharma AN, Doshi S, Courtot M, Lo R, Williams LE, Frye JG, Elsayegh T, Sardar D, Westman EL, Pawlowski AC, Johnson TA, Brinkman FSL, Wright GD, McArthur AG. CARD 2017: expansion and model-centric curation of the comprehensive antibiotic resistance database. Nucleic Acids Res. 2017;45(D1):566–73.
5. Chen L, Yang J, Yu J, Yao Z, Sun L, Shen Y, Jin Q. VFDB: a reference database for bacterial virulence factors. Nucleic Acids Res. 2005;33(suppl_1):325–8.
6. Quick J, Loman NJ, Duraffour S, Simpson JT, Severi E, Cowley L, Bore JA, Koundouno R, Dudas G, Mikhail A, Ouédraogo N, Afrough B, Bah A, Baum JHJ, Becker-Ziaja B, Boettcher JP, Cabeza-Cabrerizo M, Camino-Sánchez Á, Carter LL, Doerrbecker J, Enkirch T, Dorival IG, Hetzelt N, Hinzmann J, Holm T, Kafetzopoulou LE, Koropogui M, Kosgey A, Kuisma E, Logue CH, Mazzarelli A, Meisel S, Mertens M, Michel J, Ngabo D, Nitzsche K, Pallasch E, Patrono LV, Portmann J, Repits JG, Rickett NY, Sachse A, Singethan K, Vitoriano I, Yemanaberhan RL, Zekeng EG, Racine T, Bello A, Sall AA, Faye O, Faye O, Magassouba N, Williams CV, Amburgey V, Winona L, Davis E, Gerlach J, Washington F, Monteil V, Jourdain M, Bererd M, Camara A, Somlare H, Camara A, Gerard M, Bado G, Baillet B, Delaune D, Nebie KY, Diarra A, Savane Y, Pallawo RB, Gutierrez GJ, Milhano N, Roger I, Williams CJ, Yattara F, Lewandowski K, Taylor J, Rachwal P, Turner DJ, Pollakis G, Hiscox JA, Matthews DA, Shea MKO, Johnston AM, Wilson D, Hutley E, Smit E, Di Caro A, Wölfel R, Stoecker K, Fleischmann E, Gabriel M, Weller SA, Koivogui L, Diallo B, Keïta S, Rambaut A, Formenty P, Günther S, Carroll MW. Real-time, portable genome sequencing for Ebola surveillance. Nature. 2016;530:228–32.
7. Huson DH, Auch AF, Qi J, Schuster SC. MEGAN analysis of metagenomic data. Genome Res. 2007;17(3):377–86. https://doi.org/10.1101/gr.5969107.
8. Huson DH, Albrecht B, Bağcı C, Bessarab I, Górska A, Jolic D, Williams RBH. MEGAN-LR: new algorithms allow accurate binning and easy interactive exploration of metagenomic long reads and contigs. Biol Direct. 2018;13(1):6.
9. Pruitt KD, Tatusova T, Klimke W, Maglott DR. NCBI Reference Sequences: current status, policy and new initiatives. Nucleic Acids Res. 2009;37:32–6.
10. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res. 1997;25:3389–402.
11. Sheetlin SL, Park Y, Frith MC, Spouge JL. Frameshift alignment: statistics and post-genomic applications. Bioinformatics. 2014;30(24):3575–82.
12. Buchfink B, Xie C, Huson DH. Fast and sensitive protein alignment using DIAMOND. Nat Methods. 2015;12:59–60.
13. Arumugam K, Bağcı C, Bessarab I, Beier S, Buchfink B, Gorska A, Qiu G, Huson DH, Williams RBH. Annotated bacterial chromosomes from frame-shift-corrected long read metagenomic data. Microbiome. 2019;7(1):1–13.
14. Nicholls SM, Quick JC, Tang S, Loman NJ. Ultra-deep, long-read nanopore sequencing of mock microbial community standards. GigaScience. 2019;8(5):043.
15. Yang C, Chu J, Warren R, Birol I. NanoSim: nanopore sequence read simulator based on statistical characterization. GigaScience. 2017;6(4):1–6.
16. Noll N, Urich E, Wüthrich D, Hinic V, Egli A, Neher R. Resolving structural diversity of Carbapenemase-producing gram-negative bacteria using single molecule sequencing. bioRxiv. 2018456897.
17. Kim D, Song L, Breitwieser FP, Salzberg SL. Centrifuge: rapid and sensitive classification of metagenomic sequences. Genome Res. 2016;26(12):1721–9.
18. Juul S, Izquierdo F, Hurst A, Dai X, Wright A, Kulesha E, Pettett R, Turner DJ. What's in my pot? Real-time species identification on the MinION. bioRxiv. 2015030742.
19. Smolinski MS, Hamburg MA, Lederberg J. Microbial Threats to Health: Emergence, Detection, and Response. Washington, DC: The National Academies Press; 2003. https://doi.org/10.17226/10636.

Publisher's Note

Appendix IV

GBE

Microbial Phylogenetic Context Using Phylogenetic Outlines

Caner Bagci[1], David Bryant[2], Banu Cetinkaya[3], and Daniel H. Huson [1,4,*]

[1]Algorithms in Bioinformatics, University of Tübingen, Germany
[2]Department of Mathematics, University of Otago, Dunedin, New Zealand
[3]Computer Science Program, Sabanci University, Tuzla/İstanbul, Turkey
[4]Cluster of Excellence: Controlling Microbes to Fight Infection, University of Tübingen, Tübingen, Germany

*Corresponding author: E-mail: daniel.huson@uni-tuebingen.de.

Accepted: 6 September 2021

Abstract

Microbial studies typically involve the sequencing and assembly of draft genomes for individual microbes or whole microbiomes. Given a draft genome, one first task is to determine its phylogenetic context, that is, to place it relative to the set of related reference genomes. We provide a new interactive graphical tool that addresses this task using Mash sketches to compare against all bacterial and archaeal representative genomes in the Genome Taxonomy Database taxonomy, all within the framework of SplitsTree5. The phylogenetic context of the query sequences is then displayed as a phylogenetic outline, a new type of phylogenetic network that is more general than a phylogenetic tree, but significantly less complex than other types of phylogenetic networks. We propose to use such networks, rather than trees, to represent phylogenetic context, because they can express uncertainty in the placement of taxa, whereas a tree must always commit to a specific branching pattern. We illustrate the new method using a number of draft genomes of different assembly quality.

Key words: phylogeny, genomes, *k*-mers, phylogenetic networks, algorithms, software.

Significance

Metagenomic sequencing allows the construction of draft genomes of unknown microbial species. There is a need for tools that make it easy to determine the possible taxonomic identity of such a genome. Here, we provide a fast and interactive software for computing the "taxonomic context" of a draft genome that is represented as a novel "phylogenetic outline."

Introduction

In the study of microbes using sequencing, assembly, and contig binning, one important task is to calculate the "phylogenetic context" of a given draft genome, contig, or bin of contigs. This requires that we first determine which known microbes have similar sequences to the query, and then produce a suitable indication of the phylogenetic relationships.

Pairwise distances between genome-scale sequences can be quickly calculated using *k*-mer methods such as Mash (Ondov et al. 2016). In this type of approach, the *k*-mer content (words of a fixed length *k*) of a sequence is represented by a reduced "sketch" and such sketches are compared using the Jaccard index and derived distance measures that approximate average nucleotide identity (ANI).

The Genome Taxonomy Database (GTDB) (Parks et al. 2020) provides a similarity-based taxonomy for ≈195,000 bacterial and archaeal genomes obtained from the NCBI assembly database (Kitts et al. 2016). A representative subset of ≈32,000 reference genomes is provided for taxonomic analysis and the GTDB-tk tool kit provides associated analysis tools (Chaumeil et al. 2019).

Here, we propose to compute a Mash sketch for each representative reference genome in the GTDB, and to assign

a Bloom filter (Bloom 1970) to each internal node of the taxonomy so as to represent the set of all *k*-mers present in reference genomes below the node (Solomon and Kingsford 2016; Pierce et al. 2019). For a given set of query sequences, this will allow one to determine all similar reference genomes quickly enough for use in an interactive program. Mash can then be used to compute a distance matrix on the query and (a subset of) all sufficiently similar references.

Given such a matrix of pairwise distances, one option is to compute a phylogenetic tree to represent the data, using an algorithm such as neighbor-joining (Saitou and Nei 1987). Phylogenetic trees are often used to represent such data, because evolution is assumed to be predominantly driven by speciation events. In addition, phylogenetic trees have low complexity, employing only a linear number $O(n)$ of nodes and edges to represent n taxa.

However, in the evolution of microbes, reticulate events, such as horizontal gene transfer and recombination, may play a significant role (Huson et al. 2010). In addition, when using *k*-mer features and distance-based phylogenetic methods, the accuracy of the resulting phylogenetic trees may be poor. Hence, the use of phylogenetic networks, rather than phylogenetic trees, can be more appropriate.

One popular approach to obtaining a phylogenetic network (Huson and Bryant 2006) is to apply the neighbor-net algorithm (Bryant and Moulton 2004) on the distances and to represent the output as a splits network (Dress and Huson 2004), requiring $O(n^4)$ nodes and edges, in the worst case.

Here, we present a new type of phylogenetic network that we call a "phylogenetic outline" (fig. 1). A phylogenetic outline is also computed from the output of the neighbor-net algorithm and has the mathematical properties of a splits network. It displays all the calculated splits, but uses substantially fewer nodes and edges to do so. Indeed, phylogenetic outlines are only quadratic in size, containing at most $O(n^2)$ nodes and edges. By default, phylogenetic outlines are unrooted, however, we also provide algorithms for both midpoint and outgroup rooting.

Although our focus here is on using phylogenetic outlines to represent phylogenetic context, please note that phylogenetic outlines can be used to represent the output of the neighbor-net algorithm in all other settings, as well.

The entire procedure described here has been implemented as part of SplitsTree5. The implementation carries out a Mash comparison of a set of query sequences against a database representing the GTDB, so as to determine the phylogenetic context of the queries, then computes and visualizes a phylogenetic outline of the sequences.

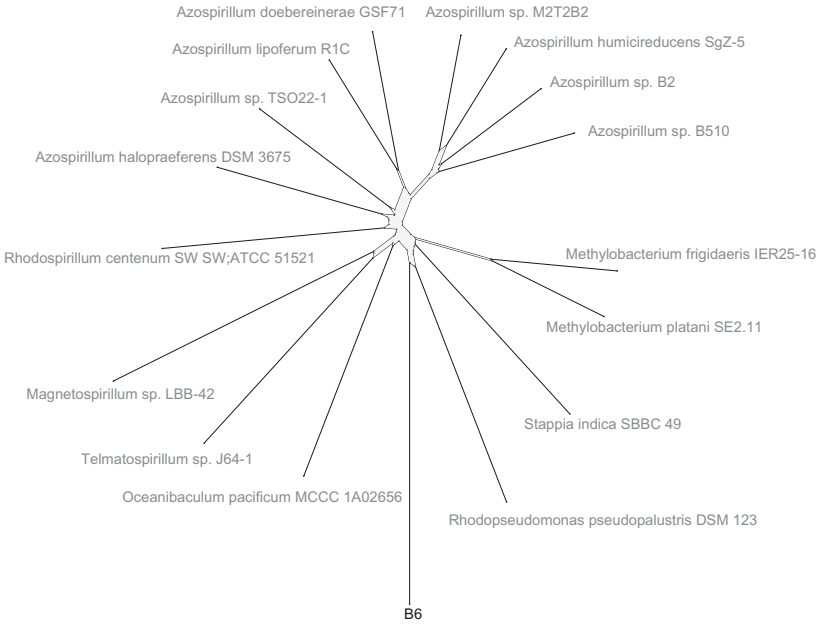


Fig. 1.—A phylogenetic outline, displaying the phylogenetic context of the metagenomic draft genome B6 from Arumugam et al. (2019).

Using a single dialog, the user selects the files containing the query sequences, loads a database containing all reference data and then obtains a phylogenetic outline of the queries, interactively in minutes. Unlike other approaches (Ondov et al. 2016; Chaumeil et al. 2019; Pierce et al. 2019), no scripting or running of multiple programs is required.

Conceptually, the calculation of "phylogenetic context" lies between "phylogenetic placement" (Matsen et al. 2010), in which one or more query sequences are placed into a precomputed phylogenetic tree, and ab initio phylogenetic tree inference, in which a phylogenetic tree is calculated for all query sequences and a subset of the reference sequences. The GTDB-tk toolkit provides tools for performing phylogenetic placement and ab initio tree inference. In both cases, the result is a phylogenetic tree that can be viewed in a program such as Dendroscope (Huson and Scornavacca 2012).

To illustrate our method, we apply it to a number of metagenomic draft genomes of different levels of quality, published in Arumugam et al. (2019). We also show how this differs from the phylogenetic analyses that one can perform using GTDB-tk.

Results

Assume that you have sequenced and assembled one or more bacterial genomes, or have calculated a metagenomic binning of contigs. There are a number of command-line pipelines that can be used to determine closely related genomes, ranging from very fast, *k*-mer based heuristics such as Mash (Ondov et al. 2016), Sourmash (Pierce et al. 2019), or marker-gene based phylogenetic placement methods such as GTDB-tk (Parks et al. 2020), to more thorough, but slower protein-alignment based approaches such as DIAMOND+MEGAN (Buchfink et al. 2015; Huson et al. 2016) or HUMAnN2 (Franzosa et al. 2018). These methods all require scripting to go from an input file containing one or more sequences of interest to a visualization of the phylogenetic context of the input sequences. Moreover, the visual representation of the context is often performed using a phylogenetic or taxonomic tree, which presents a definite clustering of taxa with little indication of uncertainty or alternative groupings.

The shortcomings of using a single phylogenetic tree to represent uncertain data are well known and have been addressed in number of different approaches, such as consensus networks (Holland et al. 2004), DensiTree visualizations (Bouckaert 2010), or the "branch parsimony score" that aims at quantifying uncertainty in sample placements (Turakhia et al. 2021), to name a few.

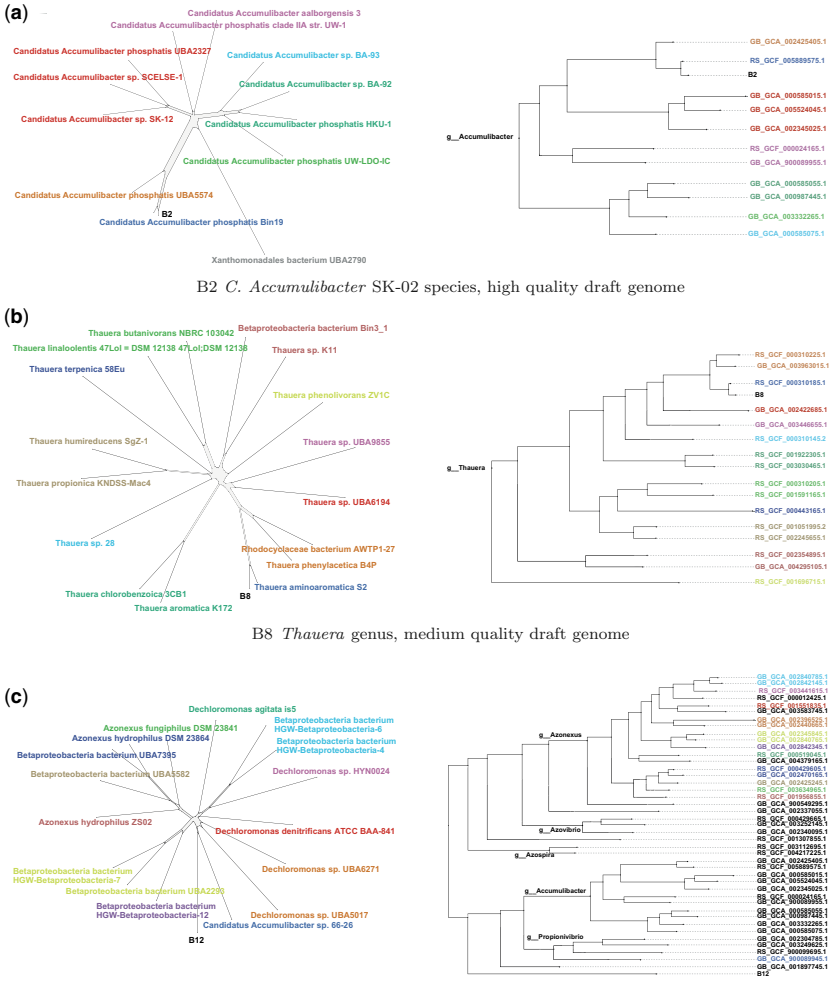
We provide a fast and interactive implementation for exploring phylogenetic context of a set of microbial sequences of interest. The user loads one or more files of query DNA sequences and then requests that all similar reference genomes are determined. Then a threshold is set for the maximum distance of reference genomes, or number of reference genomes, to be considered. These are downloaded and a Mash comparison of the query sequences and all similar reference genomes is performed, the neighbor-net method is run, and the result is presented as a phylogenetic outline. (The user can also choose to use a tree-building method such as neighbor-joining; Saitou and Nei 1987).

To illustrate our method, we applied it to a number of "draft genomes" reported in Arumugam et al. (2019). These draft genomes contain assembled contigs of long-read microbiome sequences obtained from a bio-reactor enriched for polyphosphate accumulation (Each such draft genome is a "metagenomic assembly bin" that consists of one or more contigs that are deemed to belong to the same genome.). The paper reports a taxonomic assignment for each bin that is based on an analysis of the contained protein-coding genes and confirmed using 16S rRNA sequences, when present. For each of the 14 reported draft genomes, we calculated a phylogenetic outline to display the phylogenetic context of the closest reference genomes below a certain distance.

On the left of figure 2, we show one "high-quality" draft genome (that is, with $> 90\%$ completeness and $< 5\%$ contamination), one "medium-quality" draft genome (with $\geq 50\%$ completeness and $< 10\%$ contamination), and one "low-quality" draft genome (with $< 50\%$ completeness and $< 10\%$ contamination), respectively. See Bowers et al. (2017) for the definition of the three quality levels in terms of completeness and contamination. The other 11 bins are shown in the Supplementary Material online.

Generally speaking, in all three cases, the phylogenetic context is compatible with the taxonomic assignment reported in Arumugam et al. (2019). In the case of draft genome B2, all (but one) reference genomes displayed in the phylogenetic context are members of the genus *Candidatus Accumulibacter*. This is in agreement with the classification presented in Arumugam et al. (2019), which assigned B2 to the species *Candidatus Accumulibacter* sp. SK-02. The closest species in the phylogenetic context analysis is *Candidatus Accumulibacter phosphatis* Bin19, Mash distance 0.01, a genome that was not available to Arumugam et al. (2019). The second closest, *Candidatus Accumulibacter phosphatis* UBA5574, Mash distance 0.07, is not represented by protein sequences in NCBI and thus was not part of the database used by Arumugam et al. (2019).

Unexpectedly, the species *Xanthomonadales bacterium* UBA2790, which comes from a different taxonomic class, also appears in the phylogenetic context of B2, with a Mash distance of 0.2. Note that this metagenome-assembled genome (MAG) comes from a sample of granular sludge that also gave rise to two *Candidatus Accumulibacter* reference genomes (Parks et al. 2017) and we suspect that it might



B2 *C. Accumulibacter* SK-02 species, high quality draft genome

B8 *Thauera* genus, medium quality draft genome

B12 Betaproteobacteria class, low quality draft genome

FIG. 2.—Phylogenetic context and placement. For three metagenomic draft genomes B2, B8, and B12, we report the taxonomic assignment and genome quality (Arumugam et al. 2019), and display both the phylogenetic outline computed by SplitsTree5 and a tree representing the phylogenetic placement computed using GTDB-Tk.

be contaminated with *Candidatus Accumulibacter* contigs or sequence.

In the case of draft genome B8, all references genomes displayed in the phylogenetic context are *Thauera* species, except one unclassified Betaproteobacteria bacterium, which is, however, a member of the genus *Thauera*, and this supports the assignment to the genus *Thauera*. The closest reference genome *Thauera aminoaromatica* S2 has a Mash distance of 0.2.

Finally, in Arumugam et al. (2019), the draft genome B12 was classified as a member of the Betaproteobacteria class, suggesting that there did not exist a closely related reference at the time of the publication of the data set. In the phylogenetic context computed by SplitsTree, B12 is placed closest to

Candidatus Accumulibacter sp. 66-26 with a Mash distance of 0.14. In addition, some species of the genera *Azonexus and Dechloromonas* can be found that are similar to B12, with Mash distances below 0.18. These two genera belong to the family of Azonexaceae in the NCBI taxonomy, whereas *Candidatus Accumulibacter* does not have a defined family or order. All three genera belong to the family Rhodocyclaceae in the GTDB taxonomy. Although B12 does not have many closely related reference genomes, the phylogenetic outline produced by SplitsTree5 suggests that B12 belongs to the Rhodocyclaceae family, which is more specific that the assignment suggested in Arumugam et al. (2019).

In each of the three examples, it took between 1 and 3 min to determine all reference genomes whose sketches have a distance of at most 0.3 to the sketch of the draft genome, and then to compute and display the phylogenetic outlines for the ten most similar references. Computations were carried out on a laptop with eight cores (at 2.4 GHz) and 32 GB of memory. Reference genomes are downloaded (and cached) on demand, which takes additional time. The distance thresholds used to select the closest reference genomes for each bin were chosen interactively and are reported in the Supplementary Material online.

To illustrate the improved practical performance of the outline algorithm on a larger data set, we computed the phylogenetic context for draft genome B12 using the 1,000 closest reference genomes. Running the neighbor-net algorithm on this data takes 90 s and results in 4,516 splits. The equal-angle algorithm (Dress and Huson 2004) produces a splits network with 108,640 nodes and 212,762 edges, and requires about 7 min to compute and show the network. In contrast, our new outline algorithm produces a splits network with 8,028 nodes and 8,028 edges and requires only 2 s for this (not shown here).

For the purpose of comparison, we applied GTDB-Tk (Chaumeil et al. 2019) in phylogenetic placement mode (classify_wf workflow) to the draft genomes B2, B8, and B12. GTDB-Tk uses GTDB accessions to label reference genomes, whereas SplitsTree uses the strain names associated with the assemblies in the assembly reports of NCBI. In figure 2, we show relevant part of the placement tree computed by GTDB-tk and use colors to indicate corresponding GTDB accessions and NCBI strain names.

In the case of draft genome B2, the tree computed by GTDB-Tk agrees very well with the phylogenetic context computed using SplitsTree, placing B2 next to *Candidatus Accumulibacter phosphatis* Bin19, and to other reference genomes shown in the phylogenetic outline (fig. 2*a*). The distances computed by SplitsTree5 were also similar to those reported by GTDB-Tk.

In the case of draft genome B8, the phylogenetic context included all members of the genus *Thauera* from GTDB-Tk, and placed the query next to *Thauera aminoaromatica* S2. The tree produced by GTDB-Tk contains the same references and has a similar topology (fig. 2*b*). This suggests that, if distances between genomes are small enough, then a Mash-based analysis, as in SplitsTree5, may perform very similar to a marker-gene and ANI-based analysis, as in GTDB-Tk.

Finally, in the case of B12, this draft genome is further away from any reference genome than the two draft genomes just discussed (fig. 2*c*). GTDB-Tk places B12 outside of the boundaries of any genera, but closer to the genus *Accumulibacter*, and closest to the species *Candidatus Accumulibacter* sp. 66-26.

SplitsTree5 also places B12 closest to the species *Candidatus Accumulibacter* sp. 66-26; however, the rest of the references shown in the phylogenetic context are from the genus *Azonexus* instead of *Accumulibacter*. Although the distances within the genus are in agreement with those computed by GTDB, here, we see a difference in the phylogeny outside genus boundaries. This is reflects the fact that ANI values are known to provide only a poor estimation of evolutionary distance across different genera (Qin et al. 2014).

We also determined the phylogenetic context and GTDB-Tk placement for all other 11 MAGs reported in Arumugam et al. (2019) and report these in the Supplementary Material online. For those draft genomes for which very similar reference genomes can be found, the phylogenetic context computed by SplitsTree is similar to the phylogenetic placement computed by GTDB-Tk. In the other cases, either the phylogenetic context contains only very few references, or it contains a wide range of different references and disagrees with the phylogenetic placement computed by GTDB-Tk (see B4 and B6 in the Supplementary Material online). These disagreements persist even if one uses a more accurate calculation of ANI (not shown here), indicating that they are due to a fundamental difference between ANI analysis and marker-gene analysis.

Discussion

Here, we bring together a number of different ideas, using the GTDB database to represent the taxonomy of bacterial and archaeal genomes; Mash sketches and Bloom filters for fast sequence comparison; and the neighbor-net method and our new concept of phylogenetic outlines for visualization. We thus provide a fast heuristic for establishing the phylogenetic context for one or more prokaryotic genomes or DNA sequences. We demonstrated that our approach can be applied to usefully determine and visualize the phylogenetic context of bacterial draft genomes at different levels of assembly quality.

We believe that the use of a phylogenetic outline, rather than a phylogenetic tree, to represent phylogenetic context is more suitable because outlines can express vagueness in the placement of taxa with respect to each other, whereas trees suggest a specific branching pattern. For example, in figure 4*a*, we show the unrooted, resolved phylogenetic tree

computed using the neighbor-joining algorithm (Saitou and Nei 1987). Both the splits network (fig. 4*b*) and the phylogenetic outline (fig. 4*c*) place *Competibacteraceae bacterium UBA2788* halfway between *Candidatus Contendobacter odensis Run B J11* and the draft genome B11. This ambiguity of placement is not evident in the tree representation.

The mash-based calculation of phylogenetic outlines presented here is, on the one hand, a form of ab-initio phylogenetic analysis, in which we infer evolutionary relationships from data. On the other hand, the aim is to visualize the phylogenetic context of sequences, which is similar to the goal of phylogenetic placement. We provide a graphical user interface that allows the user to interactively compute and explore the context of sequences. Although GTDB-tk provides methods for both phylogenetic placement and ab initio phylogenetic analysis, these calculations are performed using a script and the output is presented as text files. Although the resulting trees can be viewed using third party tools, the leaves of the tree are labeled by GTDB accessions, whereas our approach provides the option of labeling leaves by the associated NCBI names.

Here, the focus is on prokaryotic sequences. It would be straight-forward to adopt this approach to eukaryotes with small genomes, such as *Phytophthora* or certain insects, say. Virus genomes such as HIV or SARS-2-COVID, are too small to benefit from a naïve sketching approach, whereas mammalian genomes are probably too big to handle with our current code base. Using mash to screen sequencing data for the presence of certain genomes is an attractive idea (Ondov et al. 2019), but it exceeds the envisioned scope of this software.

Phylogenetic outlines are not limited to depicting phylogenetic context and we envision them becoming the preferred visualization of the output of the neighbor-net algorithm in other types of analysis, too.

Using a phylogenetic outline to represent phylogenetic context does not replace careful alignment and sophisticated phylogenetic analysis when the goal is to understand the evolutionary history of a set of taxa in detail. In addition, contamination of metagenomic assembly bins may cause difficulties. Nevertheless, we believe that our approach will prove to be a useful addition to the biologists' computational toolbox.

Materials and Methods

Preprocessing the Reference Database

We downloaded the GTDB taxonomy (Parks et al. 2020) in July 2020. The taxonomy has 240,103 nodes, of which 194,600 are leaves. GTDB identifies 31,910 genomes representative genomes. These are available from the GTDB download page https://data.gtdb.ecogenomic.org/releases/latest/genomic_files_reps/ (last accessed September 16, 2021). Links to the other (nonrepresentative) genomes are contained in the GenBank or RefSeq (Pruitt et al. 2009) assembly summary reports on the NCBI genomes FTP site ftp://ftp.ncbi.nlm.nih.gov/genomes/ASSEMBLY_REPORTS/.

In a processing step, we computed a Mash sketch (Ondov et al. 2016) for each of the 31,910 representative genomes, using a word size of $k = 21$ and sketch size of $s = 10{,}000$. Multipart genome sequences were concatenated. For each internal node of the GTDB taxonomy, we computed a Bloom filter (Bloom 1970) representing all k-mers contained in all sketches associated with genomes below the node, using a false positive probability of 0.0001. For these calculations, we used our own implementations of the Mash algorithm, mash-sketches, and Bloom filters, bfilter-tool, which we provide as a part of our SplitsTree5 package.

All taxa, Mash sketches, Bloom filters, and genome URL's were loaded into an SQLITE database file gtdb-rep-k21-s10000-May2021.db. In addition, the file contains an explicit representation of the GTDB taxonomy using a node-to-parent mapping. The database schema is shown in figure 3. The database file is 12.4 GB in size and does not contain the actual genome sequences; these are downloaded (and cached) by our implementation on demand.

The Outline Algorithm

For a given distance matrix D on a set of n taxa $\mathcal{X}$, the neighbor-net algorithm (Bryant and Moulton 2004) computes a set of weighted splits Σ of $\mathcal{X}$, that is, a set of bipartitions of the form $S = A|B$, where $A \neq \emptyset$, $B \neq \emptyset$, $A \cap B = \emptyset$ and $A \cup B = \mathcal{X}$. The set of splits computed by neighbor-net has quadratic size $O(n^2)$. The set of splits is "circular," which implies that they can be represented by an "outer-labeled

Fig. 3.—Database schema. The info table contains general information, such as version and size of the database. The primary key for all other tables is the taxon ID. For each reference species, the genomes table contains the genome accession, genome size, and the URL of a FastA file containing the genome sequence. The mash_sketches table contains a mash sketch for each reference species, whereas the bloom_filters tables contains a Bloom filter for each higher-rank taxon. The taxa table contains the ID and name, the parent ID, and the rank, for each taxon.

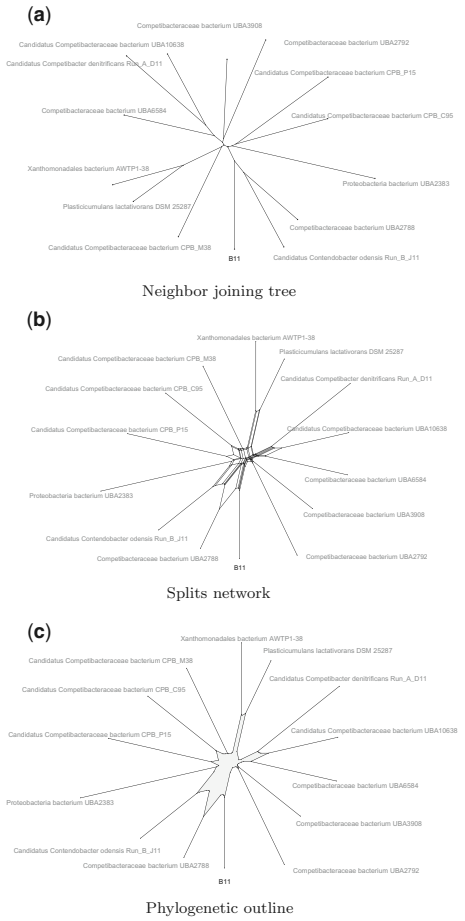


FIG. 4.—Tree and networks. For a low-quality draft genome "B11" from Arumugam et al. (2019), we display its calculated phylogenetic context, using (*a*) a neighbor-joining tree with 26 nodes and 25 edges, (*b*) a splits network with 120 nodes and 197 edges, and (*c*) a phylogenetic outline with 68 nodes and 68 edges, respectively.

planar" splits network (Dress and Huson 2004) (fig. 4*b*), using $O(n^4)$ nodes and edges, in the worst case.

Here, we describe the computation of a phylogenetic outline that requires only $O(n^2)$ nodes and edges (fig. 4*c*). In a phylogenetic outline, each split $S = A|B$ is represented by a single edge, or two parallel edges, that separate all taxa in A from all taxa in B, and thus a phylogenetic outline fulfills the definition of a splits network (Dress and Huson 2004).

Consider a set Σ of m splits on $\mathcal{X}$, each split S with a positive weight $\omega(S)$. Assume, without loss of generality, that Σ contains all trivial splits on $\mathcal{X}$, that is, all splits that separate exactly one taxon from all others. We will assume that the splits are circular, that is, that there exists an ordering $x_1, x_2, \ldots, x_n$ of the taxon set $\mathcal{X}$ such that each split $S \in \Sigma$ can be written as $S = \{x_i, \ldots, x_j\}|\mathcal{X} - \{x_i, \ldots, x_j\}$, with $1 < i \leq j \leq n$, in other words, as an interval of elements of $\mathcal{X}$, which does not contain the first taxon, versus all others. This condition is always satisfied by the output of neighbor-net (Bryant and Moulton 2002).

To illustrate this, consider the set of splits $\mathcal{S} = \{S_1, \ldots, S_5, S_a, S_b, S_c\}$ on $\mathcal{X} = \{x_1, \ldots, x_5\}$, where $S_a = \{x_2, x_3\}|\{x_1, x_4, x_5\}$, $S_b = \{x_3, x_4, x_5\}|\{x_1, x_2\}$ and $S_c = \{x_3, x_4\}|\{x_1, x_2, x_5\}$. Moreover, for $i = 1, \ldots 5$, let S_i be the trivial split separating x_i from all other taxa. This set of splits is circular, as illustrated in figure 5*a*.

Circularity implies that, for each split $S \in \Sigma$, the split part not containing x_1 is an interval of the form $I(S) = \{x_i, \ldots, x_j\}$ with $1 < i \leq j \leq n$. We will use $i(S)$ and $j(S)$ to refer to the two interval bounds.

Our new "outline algorithm" for computing a phylogenetic outline proceeds in three steps. In summary, first, we define two "events" per split. Second, we sort all events. Third, we process all events in sorted order, constructing either 0 or 1 new nodes and/or edges, per event.

For each split S, we define two events, an "outbound event" S^+, crossing over to the other side of S from the side that contains x_1, and an "inbound event" S^-, returning back to the side of S that contains x_1. We will sort these events and then use them to construct the phylogenetic outline.

We define a total ordering on all events as follows (fig. 5*b and c*):

- For two outbound events S^+ and T^+, set $S^+ < T^+$, if either $i(S) < i(T)$ or both $i(S) = i(T)$ and $j(S) > j(T)$.
- For two inbound events S^- and T^-, set $S^- < T^-$, if either $j(S) < j(T)$ or both $j(S) = j(T)$ and $i(S) > i(T)$.
- For an outbound event S^+ and an inbound event T^-, set $S^+ < T^-$, if $i(S) < j(T) + 1$, and set $S^+ > T^-$, otherwise.

The ordering of all $O(n^2)$ events can be computed in $O(n^2)$ steps: Use radix sort to first sort all outbound events S^+ in decreasing order of $j(S)$, and then in increasing order of $i(S)$. Similarly, use radix sort to first sort all inbound events S^- in decreasing order of $i(S)$ and then in increasing order of $j(S)$. Finally merge the two lists of events observing the relative ordering of outbound and inbound events.

We now describe how to create the nodes and edges of the outline (fig. 5*d*).

We will use p to denote the current location, initially set to (0, 0). Place taxon x_1 on a new node v_1 at location $\pi(v_1) = p$. We will use $\Sigma(v)$ to denote the set of splits that separates a node v from the node v_1.

For each split S, we define an angle:

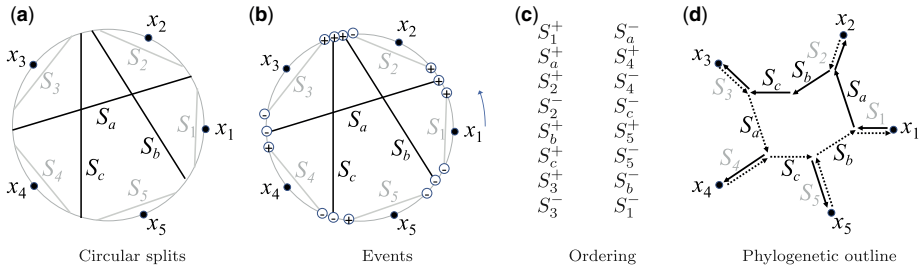


Fig. 5.—Circular splits and outline. (*a*) A set of splits $\{S_1, \ldots, S_5, S_a, S_b, S_c\}$ that is circular, that is, for which the taxa can be placed around a circle such all splits correspond to chords of the circle. (*b*) Traveling around the circle in positive orientation, each split S is encountered twice, first where the interval that does not contain taxon x_1 starts (outbound event S^+ marked ⊕) and then again where that interval ends (inbound event S^- marked ⊖). (*c*) The ordered events, listed in two columns. (*d*) Starting at x_1, in the order of the events, when encountering an outbound event S^+ move perpendicularly to the chord for split S by a distance of $\omega(S)$, as indicated by solid arrows. When encountering an inbound event S^-, move in the opposite direction by the same distance, as indicated by dotted arrows.

$$\alpha(S) = \frac{i(S) + j(S) - 2}{2n} 360^\circ.$$

In the example in figure 5*b*, this is perpendicular to the chord associated with S.

We process all events as described in the following two paragraphs. Let v be the current node, initially set to v_1.

To process an outbound event for a split S, move the current location p in the direction of $\alpha(S)$ by a distance of $\omega(S)$, the given positive weight of S. Create a new node w and connect v to w by a new edge. Set $\Sigma(w) = \Sigma(v) \cup \{S\}$. Update the current node, setting $v = w$.

To process an inbound event for a split S, move the current location p in the opposite direction of $\alpha(S)$ by a distance of $\omega(S)$. Consider the set of splits $\Sigma' = \Sigma(v) - \{S\}$. We set $w = u$, if there exists a node u with $\Sigma(u) = \Sigma'$. Else, we create a new node w, and set $\pi(w) = p$ and $\Sigma(w) = \Sigma'$. We connect v *and* w by an edge, if they are not already connected by an edge. Update the current node, setting $v = w$.

After processing all events, we arrive back at the starting point (0, 0); this is due to the fact that translations are commutative and so, for each split, the effect of processing its outbound event and the effect of later processing its inbound event cancel each other out.

The number m of circular splits on n taxa is bounded by $O(n^2)$. As discussed above, there will be at most $2m$ nodes and $2m$ edges in the network, and therefore the size is bounded by $O(n^2)$. The events are sorted using radix sort, in time linear in the number of events, and thus in $O(n^2)$ time. The construction of nodes and edges also requires only $O(n^2)$ steps. Hence, the outline algorithm requires at most $O(n^2)$ in total. The network size and time requirement compare favorably to the $O(n^4)$ network size and time worst-case requirements of the equal angle algorithm (Dress and Huson 2004), which is currently used to visualize the output of the neighbor-net algorithm in SplitsTree4 (Huson and Bryant 2006).

We now discuss how to compute a rooted phylogenetic outline. For midpoint rooting, we proceed as follows. We first determine two taxa, *a and b*, that maximize the split distance $d_\Sigma(a, b) = \sum_{S \in \Sigma(a,b)} \omega(S)$, where the sum is taken over the set $\Sigma(a, b)$ of all splits S that separate *a and b*. The set $\Sigma(a, b)$ is then sorted by increasing cardinality of the split part $S(a)$ containing a, and then by increasing size of the intersection of $S(a)$ with the interval of all taxa that lie between *a and b* in the cycle. The root is then positioned in the first split for which the accumulated sum of weights is at least half of $d_\Sigma(a, b)$. For rooting by outgroup, the root is placed in the middle of a split that separates the outgroup from the rest of the taxa and is minimal with respect to that property.

Graphical User Interface

We have implemented the approach described here in our program SplitsTree5. To compute a phylogenetic outline displaying the phylogenetic context for one or more prokaryotic sequences, select the File → Analyze Genomes. . . menu item. This will open a dialog with three tabs. The first tab is used to select the input file(s) and output file, and to determine whether all sequences in a given file are to be concatenated or to be treated separately (fig. 6*a*). In addition, one can set a minimum sequence length (here set to 100,000 bp). The second tab is used to edit the names for the sequences (fig. 6*b*). The third tab is used to perform a Mash-based search in the GTDB database and to select which reference genomes should be included in the phylogenetic outline, based on their distances to the input sequences (fig. 6*c*).

The example presented is a medium-quality draft genome consisting of 25 contigs assembled from long-read sequences, designated bin B8 in Arumugam et al. (2019) with taxonomic

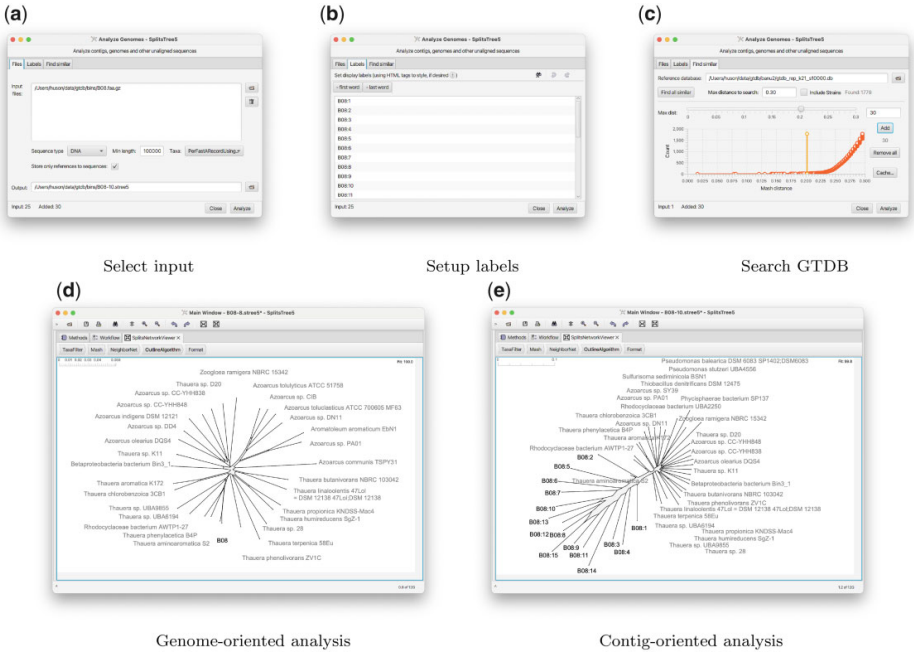


FIG. 6.—Phylogenetic context analysis. (*a*) The user selects the input file(s) and decides whether to analyze on a "per file" (complete genome) or "per FastA record" (individual contigs) basis. (*b*) The labels are set for the input sequences. (*c*) A search against the GTDB database is initiated and a threshold for the maximum distance is set. Once completed, a phylogenetic outline is drawn. (*d*) In a genome-oriented analysis, the phylogenetic outline shows the context of the concatenated input sequences. (*e*) Alternatively, in a contig-oriented analysis, the different sequences in the input file are represented individually in the phylogenetic outline.

assignment to the genus *Thauera*. In figure 6*d*, we use a phylogenetic outline to show the phylogenetic context of the draft genome, involving the 30 closest reference genomes.

In figure 6*e*, we show the phylogenetic context for the 15 (of 25) input contigs whose length achieves the set threshold of 100,000 bp. The contigs are numbered by decreasing length, ranging from B08:1 with length 770,679 bp to B08:15 with length 109,403 bp, respectively. Although the outline indicates that the longest contig B08:1 is very similar to the shown reference genomes, the similarity between contigs and references decreases with decreasing contig length.

Running GTDB-Tk

The frame-shift corrected bins from Arumugam et al. (2019) were classified using the phylogenetic-placement mode of GTDB-Tk (Chaumeil et al. 2019), using the GTDB database R95 version (Parks et al. 2020). We ran the classify_wf workflow with the default settings, using 32 cores both for the main pipeline and for the pplacer program. GTDB-Tk completed the phylogenetic placement of all bins in 26 min. In order to visualize the resulting phylogenetic placements, we opened the Newick-formatted gtdbtk.bac120.classify.tree output file in Dendroscope (Huson and Scornavacca 2012) and manually extracted the relevant subtrees for the bins shown in figure 2 and the Supplementary Material online.

Supplementary Material

Supplementary data are available at *Genome Biology and Evolution* online.

Acknowledgments

D.H.H. acknowledges Catalyst: Leaders funding provided by the New Zealand Ministry of Business, Innovation and Employment and administered by the Royal Society Te Apārangi. The authors acknowledge infrastructural support by the cluster of Excellence EXC2124 Controlling Microbes to Fight Infection (CMFI), project ID 390838134.

Author Contributions

D.B. and D.H.H. conceptualized the project. D.B. and D.H.H. developed the outline algorithm. D.H.H. designed and implemented the software. C.B. and B.C. designed and populated the database. C.B. performed all data analysis and comparisons. D.H.H. and C.B. wrote the original draft of the manuscript and all authors edited the manuscript.

Data Availability

The algorithms are implemented in Java in our program SplitsTree5. Installers for SplitsTree5, and the current database file gtdb-rep-k21-s10000-May2021.db, are freely available here: https://software-ab.informatik.uni-tuebingen.de/download/splitstree5 (last accessed September 16, 2021). The open source is available here: http://github.com/huson-lab/splitstree5 (last accessed September 16, 2021). In addition, we provide a Python implementation of neighbor-net and phylogenetic outlines here: https://github.com/huson-lab/SplitsPy (last accessed September 16, 2021). No new data were generated or analysed in support of this research.

Literature Cited

Arumugam K, et al. 2019. Annotated bacterial chromosomes from frameshift-corrected long read metagenomic data. Microbiome 7(1):61.

Bloom BH. 1970. Space/time trade-offs in hash coding with allowable errors. Commun ACM. 13(7):422–426.

Bouckaert RR. 2010. DensiTree: making sense of sets of phylogenetic trees. Bioinformatics 26(10):1372–1373.

Bowers RM, et al. 2017. Minimum information about a single amplified genome (MISAG) and a metagenome-assembled genome (MIMAG) of bacteria and archaea. Nat Biotechnol. 35(8):725–731.

Bryant D, Moulton V. 2002. NeighborNet: an agglomerative method for the construction of planar phylogenetic networks. In: Guigó R, Gusfield D, editors. Algorithms in bioinformatics. WABI 2002. Vol. LNCS 2452. Berlin, Heidelberg: Springer. p. 375–391.

Bryant D, Moulton V. 2004. Neighbor-net: an agglomerative method for the construction of phylogenetic networks. Mol Biol Evol. 21(2):255–265.

Buchfink B, Xie C, Huson DH. 2015. Fast and sensitive protein alignment using DIAMOND. Nat Methods. 12(1):59–60.

Chaumeil P-A, Mussig AJ, Hugenholtz P, Parks DH. 2019. GTDB-Tk: a toolkit to classify genomes with the genome taxonomy database. Bioinformatics 36(6):1925–1927.

Dress AWM, Huson DH. 2004. Constructing splits graphs. IEEE/ACM Trans Comput Biol Bioinform. 1(3):109–115.

Franzosa EA, et al. 2018. Species-level functional profiling of metagenomes and metatranscriptomes. Nat Methods. 15(11):962–968.

Holland B, Huber K, Moulton V, Lockhart PJ. 2004. Using consensus networks to visualize contradictory evidence for species phylogeny. Mol Biol Evol. 21(7):1459–1461.

Huson DH, Bryant D. 2006. Application of phylogenetic networks in evolutionary studies. Mol Biol Evol. 23(2):254–267.

Huson DH, et al. 2016. MEGAN Community Edition – interactive exploration and analysis of large-scale microbiome sequencing data. PLoS Comput Biol. 12(6):e1004957.

Huson DH, Rupp R, Scornavacca C. 2010. Phylogenetic networks. Cambridge: Cambridge University Press.

Huson DH, Scornavacca C. 2012. Dendroscope 3 – a program for computing and drawing rooted phylogenetic trees and networks. Syst Biol. 61(6):1061–1067.

Kitts PA, et al. 2016. Assembly: a resource for assembled genomes at ncbi. Nucleic Acids Res. 44(D1):D73–D80.

Matsen FA, Kodner RB, Armbrust EV. 2010. pplacer: linear time maximum-likelihood and bayesian phylogenetic placement of sequences onto a fixed reference tree. BMC Bioinformatics 11(1):538.

Ondov BD, et al. 2016. Mash: fast genome and metagenome distance estimation using MinHash. Genome Biol. 17(1):132.

Ondov BD, et al. 2019. Mash screen: high-throughput sequence containment estimation for genome discovery. Genome Biol. 20(1):232.

Parks DH, et al. 2017. Recovery of nearly 8,000 metagenome-assembled genomes substantially expands the tree of life. Nat Microbiol. 2(11):1533–1542.

Parks DH, et al. 2020. A complete domain-to-species taxonomy for bacteria and archaea. Nat Biotechnol. 38(9):1079–1086.

Pierce NT, Irber L, Reiter T, Brooks P, Brown CT. 2019. Large-scale sequence comparisons with sourmash. F1000Res. 8:1006.

Pruitt KD, Tatusova T, Klimke W, Maglott DR. 2009. NCBI reference sequences: current status, policy and new initiatives. Nucleic Acids Res. 37(Database issue):D32–D36.

Qin Q-L, et al. 2014. A proposed genus boundary for the prokaryotes based on genomic insights. J Bacteriol. 196(12):2210–2215.

Saitou N, Nei M. 1987. The neighbor-joining method: a new method for reconstructing phylogenetic trees. Mol Biol Evol. 4(4):406–425.

Solomon B, Kingsford C. 2016. Fast search of thousands of short-read sequencing experiments. Nat Biotechnol. 34(3):300–302.

Turakhia Y, et al. 2021. Ultrafast sample placement on existing trees (usher) enables real-time phylogenetics for the sars-cov-2 pandemic. Nat Genet. 53(6):809–816.

Associate editor: Barbara Holland

Appendix V

AMERICAN SOCIETY FOR MICROBIOLOGY | mSystems®

RESEARCH ARTICLE

Using AnnoTree to Get More Assignments, Faster, in DIAMOND+MEGAN Microbiome Analysis

Anupam Gautam,[a,b] Hendrik Felderhoff,[a] Caner Bağci,[a,b] Daniel H. Huson[a,b,c]

[a]Institute for Bioinformatics and Medical Informatics, University of Tübingen, Tübingen, Germany
[b]International Max Planck Research School "From Molecules to Organisms," Max Planck Institute for Biology Tübingen, Tübingen, Germany
[c]Cluster of Excellence: Controlling Microbes to Fight Infection, Tübingen, Germany

ABSTRACT In microbiome analysis, one main approach is to align metagenomic sequencing reads against a protein reference database, such as NCBI-nr, and then to perform taxonomic and functional binning based on the alignments. This approach is embodied, for example, in the standard DIAMOND+MEGAN analysis pipeline, which first aligns reads against NCBI-nr using DIAMOND and then performs taxonomic and functional binning using MEGAN. Here, we propose the use of the AnnoTree protein database, rather than NCBI-nr, in such alignment-based analyses to determine the prokaryotic content of metagenomic samples. We demonstrate a 2-fold speedup over the usage of the prokaryotic part of NCBI-nr and increased assignment rates, in particular assigning twice as many reads to KEGG. In addition to binning to the NCBI taxonomy, MEGAN now also bins to the GTDB taxonomy.

IMPORTANCE The NCBI-nr database is not explicitly designed for the purpose of microbiome analysis, and its increasing size makes its unwieldy and computationally expensive for this purpose. The AnnoTree protein database is only one-quarter the size of the full NCBI-nr database and is explicitly designed for metagenomic analysis, so it should be supported by alignment-based pipelines.

KEYWORDS microbiome analysis, taxonomy, functional analysis, alignment, protein sequences, NCBI-nr, AnnoTree, function, software

Next-generation sequencing (NGS) has revolutionized many areas of biological research (1, 2), providing ever-more data at an ever-decreasing cost. One such area is microbiome research, the study of microbes in their theater of activity using metagenomic sequencing (3). Here, deep short-read sequencing, and improving performance of long-read sequencing, are helping to explore the roles and interactions of microbiomes in different environments.

The two initial tasks for any microbiome sequencing data set are (i) taxonomic analysis, that is, to determine which organisms are present in a microbiome sample, and (ii) functional analysis to determine which genes and pathways are present. Task i can be addressed, to a degree, using amplicon sequencing (targeting 16S rRNA, 18S rRNA, or internal transcribed sequencing, for example). However, a species- or strain-level taxonomic analysis, and task ii, both are best addressed using whole-genome shotgun sequencing, that is, metagenomics proper.

Metagenomic shotgun sequencing reads can be analyzed using a number of different approaches, such as alignment-free *k*-mer analysis (4–6), alignment against DNA references (7, 8), or translated alignment against protein references (9–13).

The DIAMOND+MEGAN approach uses DIAMOND (14) to perform translated alignment (short-read sequencing) or "frameshift-aware" translated alignment (long-read sequencing) of metagenomic reads or contigs against the NCBI-nr database (15). The resulting alignment files are then "meganized" or analyzed using MEGAN (16) so as to

Editor Naseer Sangwan, Cleveland Clinic

Address correspondence to Daniel H. Huson, daniel.huson@uni-tuebingen.de.
The authors declare no conflict of interest.

Received 23 November 2021
Accepted 24 January 2022
Published 22 February 2022

perform taxonomic and functional binning. A detailed protocol of this simple two-step pipeline is presented in reference 17.

During meganization, reads are assigned to taxonomic classes in the NCBI taxonomy (18) and to the GTDB taxonomy (19). Reads are also assigned to functional entities in EC (20), EggNOG (21), InterPro families (22), KEGG (23), and SEED (24, 25).

The DIAMOND+MEGAN pipeline was originally designed for use with the NCBI-nr database (10). The NCBI-nr database contains nonidentical protein sequences from GenBank CDS translations, PDB (26), Swiss-Prot (27), PIR, and PRF, covering all domains of life and viruses. In August 2021, NCBI-nr comprised over 420 million sequences, of which just over half, ≈213 million, belong to bacteria or archaea. The NCBI-nr database is not explicitly designed for use in metagenomic analysis, and its rapidly increasing size is making it unwieldy for this purpose.

To perform taxonomic binning, MEGAN assigns reads to the NCBI taxonomy, which contains more than 2.2 million nodes and is designed for "structuring communication concerning all forms of life on Earth" (18, 28). More recently, the GTDB taxonomy (19) was developed for the explicit purpose of taxonomic analysis of microbiome sequencing data. Here, we introduce support for the GTDB taxonomy in MEGAN. The DIAMOND+MEGAN pipeline now performs taxonomic binning of metagenomic reads according to both the NCBI taxonomy and the GTDB taxonomy.

AnnoTree (29) provides functional annotations from over 27,000 bacterial and 1,500 archaeal genomes obtained from the GTDB database (19). The total number of protein sequences contained in the AnnoTree database is 106,052,079. The AnnoTree database is approximately one-quarter the size of NCBI-nr and only half the size of the prokaryotic part of NCBI-nr. However, AnnoTree contains protein sequences from more metagenome assembled genomes (MAGs) than NCBI-nr, and these are more likely to be found in microbiome samples.

In this paper, we describe and provide the necessary files for performing DIAMOND+MEGAN analysis using the AnnoTree protein reference database, as an alternative to NCBI-nr, to analyze the prokaryotic content of metagenomic sequencing samples. To illustrate the utility of this approach, first we compare the performance of DIAMOND+MEGAN using NCBI-nr and AnnoTree protein reference sequences on a published mock community of 23 bacterial and 3 archaeal strains (30). Using 10 published data sets from a range of different environments and obtained using both short- and long-read sequencing techniques, we then demonstrate that AnnoTree-based analysis is twice as fast and has a higher assignment rate than NCBI-nr-based analysis when using the DIAMOND+MEGAN pipeline. We also compare examples of both the NCBI and GTDB taxonomic binning.

RESULTS

Using the standard DIAMOND+MEGAN analysis pipeline (16), metagenomic sequencing reads are first aligned against the NCBI-nr database using DIAMOND, and then the resulting alignments are processed by MEGAN (or the command-line tool daa-meganizer) so as to perform taxonomic and functional binning. We will refer to this as an NCBI-nr run of the pipeline.

In this paper, we describe a new application of the DIAMOND+MEGAN pipeline in which DIAMOND alignment is performed against the AnnoTree protein database, and we refer to this as an AnnoTree run. To enable the pipeline to be run in this way, we provide (i) a FastA file containing all ≈106 million AnnoTree protein sequences and (ii) an SQLite database (https://www.sqlite.org) that contains mappings of AnnoTree protein accessions to taxonomic and functional classes in all supported classifications. In addition, we provide a set of Python scripts that can be used to update both the FastA file and the mapping database.

To allow a fair comparison between the two runs, when aligning against the NCBI-nr database, throughout this study, we only aligned against the prokaryotic entries of the NCBI-nr database.

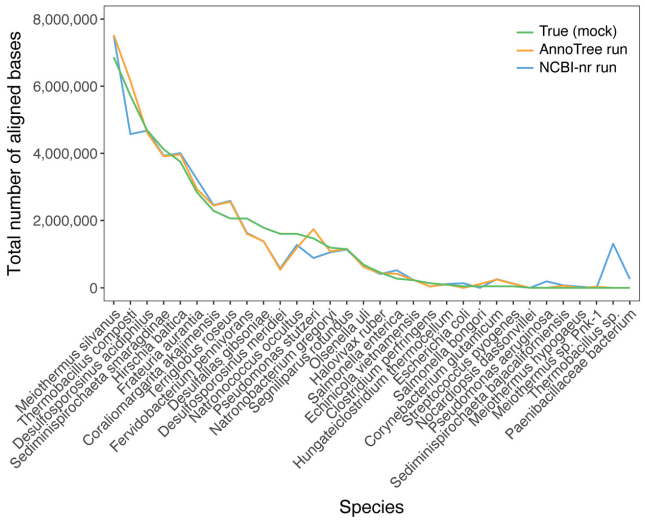


FIG 1 For prokaryotic species detected during analysis, we report the approximate relative abundance of organisms in the mock community (green) and the number of bases aligned and assigned by DIAMOND+MEGAN in an NCBI-nr run (blue) and in an AnnoTree run (orange).

Performance on a mock community. For an initial comparison of the two ways of running the DIAMOND+MEGAN pipeline, we ran both on a set of 53,654 PacBio shotgun reads from the MBARC-26 mock community of 23 bacterial and 3 archaeal strains (30). In Fig. 1, we compare the two taxonomic profiles obtained from the NCBI-nr and AnnoTree runs with the published profile. Both computed profiles follow the published one quite closely, with some low-abundance false-negative assignments, namely, to *Salmonella bongori* and *Escherichia coli* for the former and latter run, respectively, and to *Nocardiopsis dassonvillei* for both. The NCBI-nr run gives rise to five false-positive bacterial assignments, namely, to *Paenibacillaceae bacterium*, *Pseudomonas aeruginosa*, *Sediminispirochaeta bajacaliforniensis*, *Meiothermus hypogaeus*, and *Thermobacillus* species. The latter has a large count that bleeds over from the high-abundance true-positive *Thermobacillus composti*. The AnnoTree run gives rise to two false-positive bacteria, *Sediminispirochaeta bajacaliforniensis* and *Meiothermus* sp. strain *Pnk-1*. There are no false positives or negatives in the archaea.

These results indicate that the use of the AnnoTree protein database, in place of the NCBI-nr protein database, is worth pursuing when the goal is to determine the prokaryotic content of a sample.

Performance on 10 different data sets. For a more detailed comparison of NCBI-nr and AnnoTree runs of the DIAMOND+MEGAN pipeline, we applied the two variants to a set of 10 different published samples that cover a range of different environments. Nine of the 10 data sets consist of short reads, and one consists of long reads. There are ≈580 million reads in total. The number of reads per data set is listed in Table 1. For each data set, we also report the number and percentage of reads that obtained a DIAMOND alignment against AnnoTree and NCBI-nr, respectively (Spearman's correlation $\rho = 1$, $P = 2.2e-16$). In most cases, the ratio of the former number to the latter is ≈1, except in the case of the soil sample, where ≈50% more reads have an alignment against AnnoTree.

Out of the total of ≈580 million reads, DIAMOND aligns ≈307 million (≈52.9%) reads against the AnnoTree database and ≈302 million (≈52%) reads against the NCBI-nr database, that is, nearly 1% more reads against the former database, although the latter database contains more than twice as many entries.

TABLE 1 Accession numbers and total number of reads for each data set[a]

Data set	Accession no.	Total no. of reads	Reads with DIAMOND alignments				
			AnnoTree (no.)	%	NCBI-nr (no.)	%	Ratio
River1	ERR466320	646,178	410,118	63.5	406,913	63.0	1.0
River2	SRR8859111	129,753,222	90,535,941	69.8	88,403,713	68.1	1.0
Seagrass	SRR6350025	98,260,754	36,053,215	36.7	33,717,202	34.3	1.1
Skin	ERR2538467	22,827,626	13,403,495	58.7	14,122,490	61.9	0.9
Stool	ERR2641811	33,214,614	29,132,562	87.7	30,101,313	90.6	1.0
Soil	SRR7521491	97,595,185	10,992,188	11.3	7,264,223	7.4	1.5
Thermal Pools	SRR6344961	52,908,626	15,751,382	29.8	16,625,446	31.4	0.9
Bioreactor1	SRR9831403	99,998,110	73,151,916	73.1	72,806,515	72.8	1.0
Bioreactor2	SRR8313048	44,258,996	36,608,649	82.7	37,477,641	84.7	1.0
Bioreactor3[b]	SRR8305972	694,827	613,958	88.4	616,536	88.7	1.0
Total		580,158,138	306,653,424	52.86	301,541,992	51.98	1.02

[a]For both the AnnoTree and NCBI-nr protein databases, we report the number and percentage of reads that obtained an alignment using DIAMOND. We also report the ratio between the two numbers.
[b]Long-read data set.

Taxonomic binning. The DIAMOND+MEGAN pipeline assigns aligned reads both to the NCBI taxonomy (18) and, now, the GTDB taxonomy (19).

In the case of the NCBI taxonomy, performing AnnoTree runs on all data sets assigns 99.5% of all aligned reads to a taxonomic node, whereas the assignment rate when performing NCBI-nr runs is only 98.7% (Table 2). In total, using AnnoTree rather than NCBI-nr leads to the taxonomic assignment of ≈7.6 million additional reads (≈1.3% of all reads). However, in more detail, the number of assigned reads is a few percentage points higher for the NCBI-nr runs on the two human-associated samples, skin and stool, and a few percentage points lower for the seagrass and soil samples.

For each of the 10 data sets, in Fig. 2 we present a more detailed comparison of the assignment of reads to the NCBI taxonomy for the AnnoTree and NCBI-nr runs. The results for both runs agree for 30 to 60% of all reads. For most data sets, the percentage of assigned reads that are assigned by only one of the two runs is similar, with the biggest exception being the soil sample, where approximately 40% of all assigned reads are assigned by AnnoTree only. For most data sets, the percentage of reads that are assigned incompatibly to different lineages is below or around 10%, with the exception of a bioreactor sample, where this value is around 30%.

In the case of the GTDB taxonomy, performing AnnoTree runs on all data sets assigns ≈99% of all aligned reads to a taxonomic node, whereas the assignment rate when performing NCBI-nr runs is only 93.6% (Table 2). In total, using AnnoTree rather than NCBI-nr leads to an assignment of ≈21.5 million additional reads (≈3.7% of all reads). On the stool sample, the assignment rate for the NCBI-nr run is 1% higher than that for the AnnoTree run, but in all other cases, the assignment rate for the AnnoTree runs is a couple of percentage points higher than that for the NCBI-nr runs.

TABLE 2 Assigned reads[a]

Classification	AnnoTree run			NCBI-nr run			Ratio
	Assigned	% of R	% of AI	Assigned	% of R	% of AI	
NCBI taxonomy	305,150,157	52.6	99.5	297,539,333	51.3	98.7	1.0
GTDB taxonomy	303,770,449	52.4	99.1	282,269,816	48.6	93.6	1.1
EC	78,874,545	13.6	25.7	76,552,285	13.2	25.4	1.0
eggNOG	95,932,149	16.5	31.3	87,131,284	15.0	28.9	1.1
InterPro	142,250,858	24.5	46.4	143,885,580	24.8	47.7	1.0
KEGG	209,371,499	36.1	68.3	123,130,673	21.2	40.8	1.7
SEED	102,452,692	17.7	33.4	100,615,086	17.3	33.4	1.0

[a]For each of the classifications provided by MEGAN and summarized over all AnnoTree runs and NCBI-nr runs of the DIAMOND+MEGAN pipeline on the 10 data sets listed in Table 1, we report the number of assigned reads (assigned), the percentage of all reads (% of R), and the percentage of all aligned reads (% of AI). In the last column, we report the ratio of the reads assigned using AnnoTree and NCBI-nr, respectively.

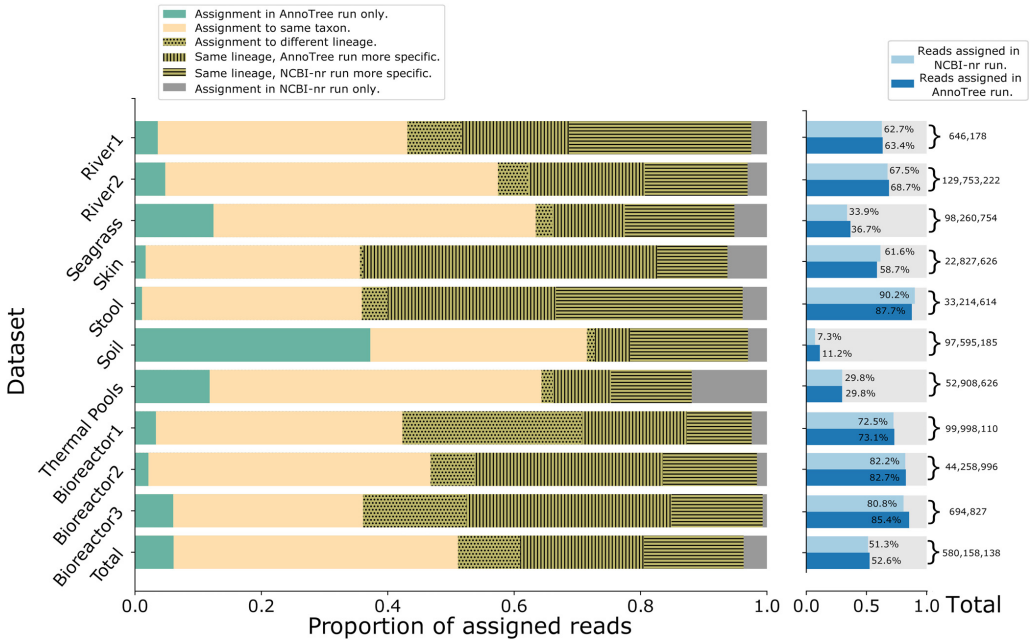


FIG 2 Details of the assignment of reads to the NCBI taxonomy. For each of the 10 data sets, for the total set of reads assigned by either an AnnoTree run or NCBI run of the DIAMOND+MEGAN pipeline, we show the proportion of reads only assigned by the AnnoTree run (green), assigned by both runs to the same taxon (yellow), or only assigned by the NCBI run (gray). For reads with differing assignments, we show the proportion assigned to incompatible lineages (dotted) or two compatible lineages with either the AnnoTree assignment being more specific (vertical stripes) or the NCBI-nr assignment being more specific (horizontal stripes). On the right, we indicate the total number of reads and the number of reads assigned by either the AnnoTree or NCBI-nr run.

In Fig. 3, we present more details on the assignment of reads to the GTDB taxonomy. The results are similar to those for the assignment to the NCBI taxonomy but with a somewhat decreased level of conflicting assignments for most data sets.

Functional binning. The DIAMOND+MEGAN pipeline assigns aligned reads to a number of different functional classifications, currently EC numbers (20), eggNOG (21), InterPro families (22), KEGG (23), and SEED (24). For most functional classifications, the assignment rates for the AnnoTree runs are slightly higher than those for the NCBI-nr runs.

In the case of KEGG, performing AnnoTree runs on all data sets assigns ≈68.3% of all aligned reads to a KEGG node, whereas the assignment rate when performing NCBI-nr runs is only ≈40.8% (Table 2). AnnoTree runs make ≈70% more assignments of reads to KEGG nodes than the NCBI-nr runs do.

The number of read assignments to KEGG is particularly low for the soil sample, with AnnoTree-based assignment of only ≈8% and NCBI-based assignment of only ≈1.6% of all reads (Fig. 4).

Running time. As the AnnoTree protein database is less than one-quarter the size of the full NCBI-nr protein database, using the former during the alignment step of the DIAMOND+MEGAN pipeline will speed up the analysis. This will be offset, very slightly, by the fact that the number of aligned reads will be slightly higher. In the analysis performed here, we only aligned against the prokaryotic content of NCBI-nr, which contains approximately twice as many sequences as the AnnoTree protein database.

In Table 3, summarizing all 10 data sets, we report the CPU time used by DIAMOND, Meganizer, and both combined during both an NCBI-nr run and an AnnoTree run of the DIAMOND+MEGAN pipeline. On all data sets, DIAMOND alignment against the

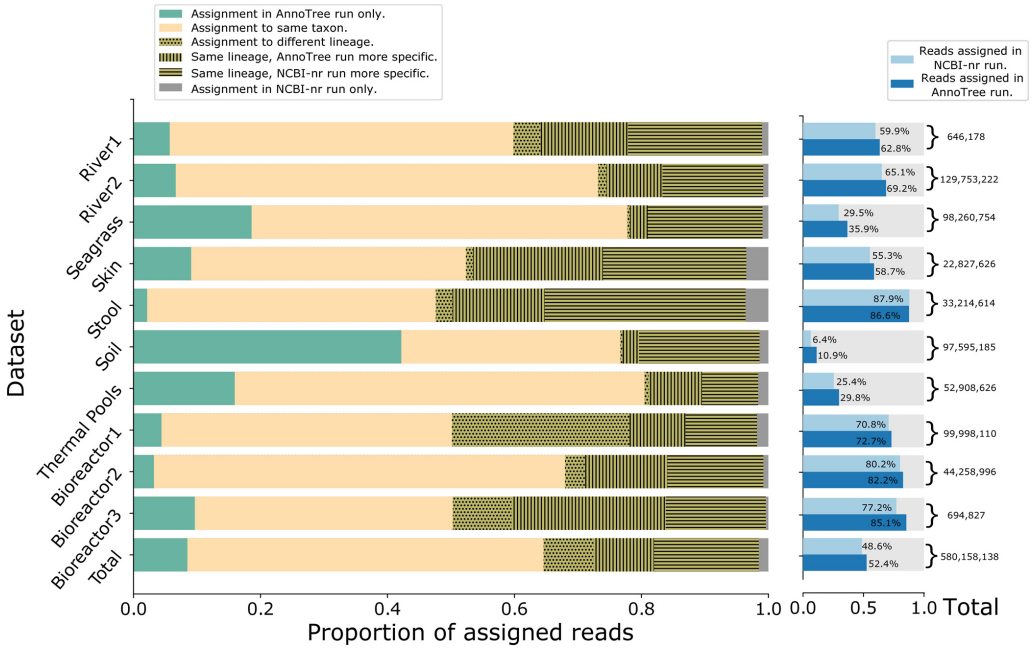


FIG 3 Details of the assignment of reads to the GTDB taxonomy, using the same colors as those in Fig. 2.

prokaryotic proteins in NCBI-nr takes about twice as long as alignment against the AnnoTree database, while meganization usually takes slightly longer in AnnoTree runs. In total, an AnnoTree run of the DIAMOND+MEGAN pipeline is twice as fast as an NCBI-nr run.

Performing an NCBI run using the full NCBI-nr database, not just the prokaryotic part, on each of the 10 data sets takes ≈3 times as long as an AnnoTree run, with an ≈4% increase of assignments to the NCBI taxonomy, on average.

DISCUSSION

When first published in 2007 (10), MEGAN was run together with BLASTX (9) on data sets containing hundreds of thousands of short reads against the NCBI-nr protein database, which then contained around 3 million sequences. The DIAMOND alignment program (14) was later designed to allow the alignment of much larger data sets against a much larger NCBI-nr database. As the NCBI-nr database continues to grow, alignment against the full database presents an increasingly severe computational bottleneck.

As an alternative, projects focusing on the prokaryotic content of microbiome samples can make use of the GTDB taxonomy and the AnnoTree protein database, which are both explicitly designed for this. As the AnnoTree protein database is only 1/4 the size of the NCBI-nr database, DIAMOND alignment of reads against the AnnoTree protein database takes at most only half as much time while providing a superior assignment rate.

In this study, we illustrated the performance of AnnoTree runs using 10 example data sets from different environments. The results on these examples confirm that using AnnoTree is a useful alternative to using NCBI-nr.

The soil sample stands out. Its DIAMOND alignment rates against AnnoTree and NCBI-nr are very low at only 11.3% and 7.4%, respectively, in contrast to the alignment rates for

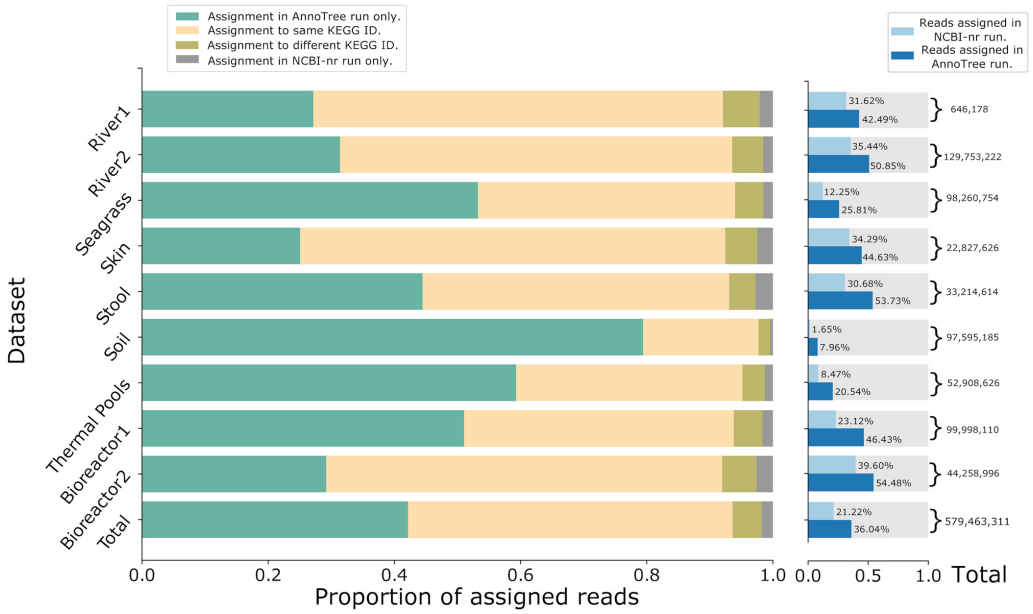


FIG 4 Details of the assignment of reads to KEGG. For each of the 10 data sets, for the total set of reads assigned by either an AnnoTree run or NCBI run of the DIAMOND+MEGAN pipeline, we show the proportion of reads only assigned by the AnnoTree run (green), assigned by both runs to the same class (yellow) or different classes (olive), or only assigned by the NCBI run (gray).

the sool sample, which are very high at 87.7% and 90.6%, respectively. This illustrates that the diversity of the soil environment is only poorly represented in current databases (31, 32), while human stool samples and human-associated microbes have been studied in detail (33). The fact that the AnnoTree run has a higher assignment rate than the NCBI-nr run on the soil sample may be because the AnnoTree database recruits more sequences from metagenomic assembled genomes than NCBI-nr does.

To extend the AnnoTree-based approach to the detection and analysis of viral sequences, one could extend the AnnoTree protein database using either the virus subdivision of NCBI-nr or another dedicated resource (34, 35).

MATERIALS AND METHODS

Data sets. We downloaded 53,654 PacBio shotgun reads (Sequence Read Archive [SRA] accession no. SRR3656744) from the MBARC-26 (Mock Bacteria ARchaea Community) data set (30), with a length of 11 to 16,403 and mean of 1,643.5. The true community profile reported in Fig. 1 was estimated from Fig. 2a of reference 36.

The 10 example data sets, listed in Table 4, were downloaded in FASTA format from the NCBI SRA using the NCBI SRA toolkit's fastq-dump program: fastq-dump --split-spot --fasta 80 -I accession. Data sets with paired-end reads were concatenated into a single file. No additional preprocessing was performed.

In more detail, we used two data sets from rivers, River1 (https://www.ncbi.nlm.nih.gov/sra/?term=ERR466320) and River2 (37), one from the seagrass rhizosphere (38), one from the skin (39), one from

TABLE 3 CPU time used for running DIAMOND, Meganizer, and both combined[a]

DIAMOND			Meganizer			DIAMOND+MEGAN		
NCBI-nr	AnnoTree	Ratio	NCBI-nr	AnnoTree	Ratio	NCBI-nr	AnnoTree	Ratio
125,288 min	61,443 min	2.0	2,241 min	2,404 min	0.9	127,529 min	63,847 min	2.0

[a]Summarizing all 10 data sets, we show the CPU time used for running DIAMOND, Meganizer, and both combined during either an NCBI-nr run (restricted to prokaryotic sequences) or an AnnoTree run of the DIAMOND+MEGAN pipeline.

TABLE 4 SRA run ID, sequencing platform, read layout, and total number of reads

Data set	SRA run ID	Platform	Layout	Total no. of reads
River1	ERR466320	LS454	Single	646,178
River2	SRR8859111	Illumina	Paired	129,753,222
Seagrass	SRR6350025	Illumina	Paired	98,260,754
Skin	ERR2538467	Illumina	Single	22,827,626
Stool	ERR2641811	Illumina	Paired	33,214,614
Soil	SRR7521491	Illumina	Paired	97,595,185
Thermal pools	SRR6344961	Illumina	Paired	52,908,626
Bioreactor1	SRR9831403	Illumina	Paired	99,998,110
Bioreactor2	SRR8313048	Illumina	Paired	44,258,996
Bioreactor3	SRR8305972	ONT	Single	694,827

the stool (40), one from the soil (41), one from thermal pools (42), and three from bioreactors (Bioreactor1 [43], Bioreactor2 [44], and Bioreactor3 [44]). Nine of the 10 data sets consist of short reads, whereas the last data set consists of ONT MinION long reads.

Protein reference databases. The NCBI-nr protein database was downloaded in January 2021 from the NCBI FTP site using the link ftp://ftp.ncbi.nlm.nih.gov/blast/db/FASTA/nr.gz. We also downloaded the two files prot.accession2taxid.gz (ftp://ftp.ncbi.nlm.nih.gov/pub/taxonomy/accession2taxid/prot.accession2taxid.gz) and nodes.dmp (ftp://ftp.ncbi.nlm.nih.gov/pub/taxonomy/taxdmp.zip).

A DIAMOND index was then generated using the following command (requiring 150 CPU minutes): diamond makedb --in nr.gz -d nr --taxonmap prot.accession2taxid.gz --taxonnodes nodes.dmp.

For both the AnnoTree Bacteria database and the AnnoTree Archaea database, we downloaded MySQL dump files (version of 25 August 2020) from the AnnoTree Bitbucket repository (https://bitbucket.org/doxeylabcrew/annotree-database/src/master/). These files were then imported into a MySQL server (version 5.7.35). The databases each have 21 tables, which hold information on the AnnoTree hierarchy, the GTDB taxonomy, the NCBI taxonomy, protein sequences, and additional mappings to Pfam, TIGRFAMs, and KEGG.

For each sequence in the "protein_sequences" table, we constructed a unique two-part accession string by concatenating its "gene_id" and "gtdb_id" values. For example, the protein sequence with gene identifier (ID) AE009439_1_1 and GTDB genome ID GB_GCA_000007185_1 was given the two-part accession AE009439_1_1__GB_GCA_000007185_1.

These accessions and the corresponding protein sequences (from both databases) were written to a FastA file, annotree.fasta, which we make available at https://software-ab.informatik.uni-tuebingen.de/download/megan-annotree.

A DIAMOND index was then generated using the following command (requiring 33 CPU minutes): diamond makedb --in annotree.fasta.gz -d annotree.

MEGAN mapping databases. We use the term "meganization" to refer to the process of analyzing the alignments of a set of sequences so as to perform taxonomic binning (for example, using the naive LCA algorithm for short reads or the interval-union LCA for long reads) and functional binning (usually using a best-hit approach). Meganization of DIAMOND alignments is performed either interactively using MEGAN or in a command-line fashion using the daa-meganizer program, which is bundled with MEGAN.

To perform meganization, MEGAN requires a so-called mapping database. This is an SQLite database file that contains a mapping of protein sequence accessions to all used taxonomical and functional classifications, namely, the NCBI taxonomy, the GTDB taxonomy, EC, EGGNOG, INTERPRO families, KEGG (MEGAN Ultimate Edition), and SEED. For NCBI-nr runs, we used the mapping database megan-map-Jul2020-2-ue.db, which we downloaded from https://software-ab.informatik.uni-tuebingen.de/download/megan6.

For AnnoTree runs, we created a new mapping database, called megan-mapping-annotree-June-2021.db, in SQLite format. This file is available at https://software-ab.informatik.uni-tuebingen.de/download/megan-annotree.

We used the above-described two-part accessions as the primary key for the mapping table. We determined the other entries of the mapping table as follows. The value for the GTDB and NCBI taxonomies were obtained from the "node_tax" tables of the two MySQL databases described above, using the gtdb_id part of the two-part accession.

The value for the KEGG classification was obtained from the "kegg_top_hits" tables of the two MySQL databases, using the gene_id part of the two-part accession. In the case that there is more than one possible KEGG assignment for a given protein, we randomly selected one. This was necessary because MEGAN allows at most one assignment per reference sequence. Both GTDB and KEGG IDs were additionally formatted to match the format required by MEGAN.

We calculated entries for the other classifications supported by MEGAN (EC, EGGNOG, INTERPRO, and SEED) by performing a join on the MD5 hash values of the protein sequences in the NCBI-nr and AnnoTree protein databases, in other words, by copying the classifications of an NCBI-nr accession over to an AnnoTree two-part accession whenever the two accessions correspond to the same protein sequence. We list the number of accessions that have assignments in the different classifications in Table 5.

For most functional classifications, the number of AnnoTree proteins with assignments is smaller than that for NCBI-nr proteins, which is because the AnnoTree assignments are copied from NCBI-nr

TABLE 5 Number of prokaryotic accessions in NCBI-nr or AnnoTree that have map to a class in the classification systems[a]

Classification	NCBI-nr (no.)	AnnoTree (no.)	Ratio
NCBI taxonomy	182,329,414	106,052,079	1.72
GTDB taxonomy	126,956,422	106,052,079	1.2
EC	4,501,593	2,962,187	1.51
eggNOG	4,274,800	3,506,041	1.21
InterPro	19,748,423	11,069,757	1.78
KEGG	8,218,708[b]	56,577,432	0.15
SEED	31,117,272	16,183,436	1.92

[a]For two different taxonomical classifications (NCBI and GTDB) and for five different functional classifications (EC, EGG, InterPro, KEGG, and SEED) supported by MEGAN, we report the number of prokaryotic accessions in NCBI-nr or AnnoTree that have a mapping to a class in the classification and the corresponding ratio.
[b]MEGAN ultimate edition.

assignments. In the case of KEGG, the assignments are obtained from the AnnoTree database, which appears to maintain a much richer mapping than previously provided by MEGAN.

Data set processing. We ran DIAMOND (v2.0.4.142) on each short-read data set as diamond blastx -d ⟨index⟩ -q ⟨input⟩ -o ⟨output⟩ -f 100 -b24 -c1. Here, ⟨index⟩ is the index file, either annotree or nr, computed as described above. The input file, in (compressed) FastA or FastQ format, is specified by ⟨input⟩, and the output file is specified by ⟨output⟩. We used -f 100 to specify output in DAA format. The two remaining options, -b24 -c1, were used in an attempt to tune performance.

In addition, for purposes of comparison against the AnnoTree database, when running DIAMOND on NCBI-nr, we used the option -taxonlist 2,2157 to restrict alignment to bacteria (taxon ID 2) and archaea (taxon ID 2157).

When processing long reads, we also specified the -long-reads option.

The resulting DAA files were meganized using the daa-meganizer program MEGAN (version 6.21.5, ultimate edition, built 5 May 2021) as tools/daa-meganizer -i ⟨input⟩ -mdb ⟨mapping⟩. Here, the input file ⟨input⟩ is a DAA file produced by DIAMOND, and the mapping file ⟨mapping⟩ was either megan-map-Jul2020-2-ue.db or megan-mapping-annotree-June-2021.db, depending on whether the DIAMOND run was against the NCBI-nr or AnnTree protein database, respectively. When processing long reads, we also specified the -lg option.

Data comparison. We used the MEGAN tool daa2info to extract the mapping of reads to taxonomic and functional classes obtained in both the NCBI-nr and AnnoTree runs of the DIAMOND+MEGAN pipeline.

The following command was used to extract the mapping of reads to classes for all classifications: tools/daa2info -i ⟨input⟩ -o ⟨output⟩ -l -m -r2c Taxonomy GTDB KEGG EC EGGNOG INTERPRO2GO SEED. Here, the input file ⟨input⟩ is a meganized DAA file and the output file ⟨output⟩ is a text file. The output was used to determine the assignment rates for different classifications.

Similarly, the following command was used to extract the mapping of reads to taxonomic paths in the NCBI taxonomy: tools/daa2info -i ⟨input⟩ -o ⟨output⟩ -l -m -r2c Taxonomy -p true -r true. The output was used to generate Fig. 2.

The following command was used to extract the mapping of reads to taxonomic paths in the GTDB taxonomy: tools/daa2info -i ⟨input⟩ -o ⟨output⟩ -l -m -r2c GTDB -p true -r true. The output was used to generate Fig. 3.

Computational resources. The DIAMOND+MEGAN pipeline was run on a Linux virtual machine (provided to us by de.NBI Cloud, highmem xlarge) with Ubuntu 18.04.4 LTS operating system, Intel Xeon Gold 6140 CPU, 2.30 GHz (processor model name), 28 sockets, 1 core per socket, 1 thread per core, 28 CPUs (on-line CPU list: 0 to 27), 504 GB of RAM, and 6 TB of hard disk space. Reported run times are "user time" as calculated using the Linux "time" command. Furthermore, for MEGAN, the RAM size was set to 500 GB. All other calculations were undertaken on a MacBookPro laptop with a 2.6-GHz 6-core (12 threads) Intel Core i7 processor and 16 GB 2400-MHz DDR4 RAM.

Statistical analysis. Spearman's correlations were computed using ggplot2 (45).

Data availability. All data sets analyzed here are publicly available from NCBI SRA, using the accession numbers listed in Table 4. The AnnoTree protein FastA file and mapping database both can be downloaded from https://software-ab.informatik.uni-tuebingen.de/download/megan-annotree.

ACKNOWLEDGMENTS

This work was supported by the BMBF-funded de.NBI Cloud within the German Network for Bioinformatics Infrastructure (de.NBI) (031A532B, 031A533A, 031A533B, 031A534A, 031A535A, 031A537A, 031A537B, 031A537C, 031A537D, and 031A538A). We also acknowledge support by the High Performance and Cloud Computing Group at the Zentrum für Datenverarbeitung of the University of Tübingen, the state of Baden-Württemberg through bwHPC, and the German Research Foundation (DFG) through grant no. INST 37/935-1 FUGG. We acknowledge infrastructural support by the cluster of Excellence EXC2124 Controlling Microbes to Fight Infection (CMFI), project ID

390838134. C.B. was supported by the German Research Foundation (DFG) through grant no. HU 566/12-1. Further, we acknowledge support by the Open Access Publishing Fund of University of Tübingen.

D.H.H. and C.B. conceptualized the project. H.F. and C.B. performed the computations. A.G. and H.F. analyzed the results. A.G. and D.H.H. wrote the manuscript. All authors edited the manuscript.

REFERENCES

1. Wuyts S, Segata N. 2019. At the forefront of the sequencing revolution—notes from the RNGS19 conference. Genome Biol 20:93. https://doi.org/10.1186/s13059-019-1714-3.
2. Koboldt DC, Steinberg KM, Larson DE, Wilson RK, Mardis ER. 2013. The next-generation sequencing revolution and its impact on genomics. Cell 155:27–38. https://doi.org/10.1016/j.cell.2013.09.006.
3. Handelsman J. 2004. Metagenomics: application of genomics to uncultured microorganisms. Microbiol Mol Biol Rev 68:669–685. https://doi.org/10.1128/MMBR.68.4.669-685.2004.
4. Ounit R, Wanamaker S, Close TJ, Lonardi S. 2015. CLARK: fast and accurate classification of metagenomic and genomic sequences using discriminative k-mers. BMC Genomics 16:1–13.
5. Breitwieser FP, Baker DN, Salzberg SL. 2018. KrakenUniq: confident and fast metagenomics classification using unique k-mer counts. Genome Biol 19:1–10.
6. Storato D, Comin M. 2020. Improving metagenomic classification using discriminative k-mers from sequencing data, p 68–81. *In* International symposium on bioinformatics research and applications. Springer, Cham, Switzerland.
7. Segata N, Waldron L, Ballarini A, Narasimhan V, Jousson O, Huttenhower C. 2012. Metagenomic microbial community profiling using unique clade-specific marker genes. Nat Methods 9:811–814. https://doi.org/10.1038/nmeth.2066.
8. Herbig A, Maixner F, Bos KI, Zink A, Krause J, Huson DH. 2016. MALT: fast alignment and analysis of metagenomic DNA sequence data applied to the Tyrolean iceman. BioRxiv https://doi.org/10.1101/050559.
9. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. 1990. Basic local alignment search tool. J Mol Biol 215:403–410. https://doi.org/10.1016/S0022-2836(05)80360-2.
10. Huson DH, Auch AF, Qi J, Schuster SC. 2007. MEGAN analysis of metagenomic data. Genome Res 17:377–386. https://doi.org/10.1101/gr.5969107.
11. Huson DH, Mitra S, Ruscheweyh HJ, Weber N, Schuster SC. 2011. Integrative analysis of environmental sequences using MEGAN4. Genome Res 21:1552–1560. https://doi.org/10.1101/gr.120618.111.
12. Keegan KP, Glass EM, Meyer F. 2016. MG-RAST, a metagenomics service for analysis of microbial community structure and function. Methods Mol Biol 1399:207–233. https://doi.org/10.1007/978-1-4939-3369-3_13.
13. Franzosa EA, McIver LJ, Rahnavard G, Thompson LR, Schirmer M, Weingart G, Lipson KS, Knight R, Caporaso JG, Segata N, Huttenhower C. 2018. Species-level functional profiling of metagenomes and metatranscriptomes. Nat Methods 15:962–968. https://doi.org/10.1038/s41592-018-0176-y.
14. Buchfink B, Xie C, Huson DH. 2015. Fast and sensitive protein alignment using DIAMOND. Nat Methods 12:59–60. https://doi.org/10.1038/nmeth.3176.
15. Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Wheeler DL. 2005. GenBank. Nucleic Acids Res 33:D34–D38.
16. Huson DH, Beier S, Flade I, Górska A, El-Hadidi M, Mitra S, Ruscheweyh HJ, Tappu R. 2016. MEGAN community edition-interactive exploration and analysis of large-scale microbiome sequencing data. PLoS Comput Biol 12:e1004957. https://doi.org/10.1371/journal.pcbi.1004957.
17. Bağcı C, Patz S, Huson DH. 2021. DIAMOND+MEGAN: fast and easy taxonomic and functional analysis of short and long microbiome sequences. Curr Protoc 1:e59. https://doi.org/10.1002/cpz1.59.
18. Schoch CL, Ciufo S, Domrachev M, Hotton CL, Kannan S, Khovanskaya R, Leipe D, Mcveigh R, O'Neill K, Robbertse B, Sharma S, Soussov V, Sullivan JP, Sun L, Turner S, Karsch-Mizrachi I. 2020. NCBI taxonomy: a comprehensive update on curation, resources and tools. Database 2020:baaa062. https://doi.org/10.1093/database/baaa062.
19. Parks DH, Chuvochina M, Waite DW, Rinke C, Skarshewski A, Chaumeil PA, Hugenholtz P. 2018. A standardized bacterial taxonomy based on genome phylogeny substantially revises the tree of life. Nat Biotechnol 36:996–1004. https://doi.org/10.1038/nbt.4229.
20. Webb EC. 1992. Enzyme nomenclature 1992: recommendations of the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology on the Nomenclature and Classification of Enzymes. Academic Press, Cambridge, MA.
21. Huerta-Cepas J, Szklarczyk D, Heller D, Hernández-Plaza A, Forslund SK, Cook H, Mende DR, Letunic I, Rattei T, Jensen LJ, von Mering C, Bork P. 2019. eggNOG 5.0: a hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses. Nucleic Acids Res 47:D309–D314. https://doi.org/10.1093/nar/gky1085.
22. Mitchell AL, Attwood TK, Babbitt PC, Blum M, Bork P, Bridge A, Brown SD, Chang H-Y, El-Gebali S, Fraser MI, Gough J, Haft DR, Huang H, Letunic I, Lopez R, Luciani A, Madeira F, Marchler-Bauer A, Mi H, Natale DA, Necci M, Nuka G, Orengo C, Pandurangan AP, Paysan-Lafosse T, Pesseat S, Potter SC, Qureshi MA, Rawlings ND, Redaschi N, Richardson LJ, Rivoire C, Salazar GA, Sangrador-Vegas A, Sigrist CJA, Sillitoe I, Sutton GG, Thanki N, Thomas PD, Tosatto SCE, Yong S-Y, Finn RD. 2019. InterPro in 2019: improving coverage, classification and access to protein sequence annotations. Nucleic Acids Res 47:D351–D360. https://doi.org/10.1093/nar/gky1100.
23. Kanehisa M, Goto S. 2000. KEGG: Kyoto encyclopedia of genes and genomes. Nucleic Acids Res 28:27–30. https://doi.org/10.1093/nar/28.1.27.
24. Overbeek R, Olson R, Pusch GD, Olsen GJ, Davis JJ, Disz T, Edwards RA, Gerdes S, Parrello B, Shukla M, Vonstein V, Wattam AR, Xia F, Stevens R. 2014. The SEED and the rapid annotation of microbial genomes using subsystems technology (RAST). Nucleic Acids Res 42:D206–D214. https://doi.org/10.1093/nar/gkt1226.
25. Wattam AR, Abraham D, Dalay O, Disz TL, Driscoll T, Gabbard JL, Gillespie JJ, Gough R, Hix D, Kenyon R, Machi D, Mao C, Nordberg EK, Olson R, Overbeek R, Pusch GD, Shukla M, Schulman J, Stevens RL, Sullivan DE, Vonstein V, Warren A, Will R, Wilson MJC, Yoo HS, Zhang C, Zhang Y, Sobral BW. 2014. PATRIC, the bacterial bioinformatics database and analysis resource. Nucleic Acids Res 42:D581–D591. https://doi.org/10.1093/nar/gkt1099.
26. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE. 2000. The protein data bank. Nucleic Acids Res 28:235–242. https://doi.org/10.1093/nar/28.1.235.
27. Bairoch A, Apweiler R. 2000. The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. Nucleic Acids Res 28:45–48. https://doi.org/10.1093/nar/28.1.45.
28. Federhen S. 2012. The NCBI taxonomy database. Nucleic Acids Res 40:D136–D143. https://doi.org/10.1093/nar/gkr1178.
29. Mendler K, Chen H, Parks DH, Lobb B, Hug LA, Doxey AC. 2019. AnnoTree: visualization and exploration of a functionally annotated microbial tree of life. Nucleic Acids Res 47:4442–4448. https://doi.org/10.1093/nar/gkz246.
30. Singer E, Andreopoulos B, Bowers RM, Lee J, Deshpande S, Chiniquy J, Ciobanu D, Klenk H-P, Zane M, Daum C, Clum A, Cheng J-F, Copeland A, Woyke T. 2016. Next generation sequencing data of a defined microbial mock community. Sci Data 3:1–8. https://doi.org/10.1038/sdata.2016.81.
31. Mocali S, Benedetti A. 2010. Exploring research frontiers in microbiology: the challenge of metagenomics in soil microbiology. Res Microbiol 161:497–505. https://doi.org/10.1016/j.resmic.2010.04.010.
32. Choi J, Yang F, Stepanauskas R, Cardenas E, Garoutte A, Williams R, Flater J, Tiedje JM, Hofmockel KS, Gelder B, Howe A. 2017. Strategies to improve reference databases for soil microbiomes. ISME J 11:829–834. https://doi.org/10.1038/ismej.2016.168.
33. Human Microbiome Project Consortium. 2012. Structure, function and diversity of the healthy human microbiome. Nature 486:207–214. https://doi.org/10.1038/nature11234.
34. Paez-Espino D, Chen IMA, Palaniappan K, Ratner A, Chu K, Szeto E, Pillay M, Huang J, Markowitz VM, Nielsen T, Huntemann M, Reddy TB, Pavlopoulos GA, Sullivan MB, Campbell BJ, Chen F, McMahon K, Hallam SJ, Denef V, Cavicchioli R, Caffrey SM, Streit WR, Webster J, Handley KM, Salekdeh GH, Tsesmetzis N, Setubal JC, Pope PB, Liu WT, Rivers AR, Ivanova NN, Kyrpides

NC. 2016. IMG/VR: a database of cultured and uncultured DNA viruses and retroviruses. Nucleic Acids Res 45:D457–D465. https://doi.org/10.1093/nar/gkw1030.
35. Nayfach S, Páez-Espino D, Call L, Low SJ, Sberro H, Ivanova NN, Proal AD, Fischbach MA, Bhatt AS, Hugenholtz P, Kyrpides NC. 2021. Metagenomic compendium of 189,680 DNA viruses from the human gut microbiome. Nat Microbiol 6:960–970. https://doi.org/10.1038/s41564-021-00928-6.
36. Singer E, Bushnell B, Coleman-Derr D, Bowman B, Bowers RM, Levy A, Gies EA, Cheng J-F, Copeland A, Klenk H-P, Hallam SJ, Hugenholtz P, Tringe SG, Woyke T. 2016. High-resolution phylogenetic microbial community profiling. ISME J 10:2020–2032. https://doi.org/10.1038/ismej.2015.249.
37. Behera BK, Patra B, Chakraborty HJ, Sahu P, Rout AK, Sarkar DJ, Parida PK, Raman RK, Rao AR, Rai A, Das BK, Jena J, Mohapatra T. 2020. Metagenome analysis from the sediment of river Ganga and Yamuna: in search of beneficial microbiome. PLoS One 15:e0239594. https://doi.org/10.1371/journal.pone.0239594.
38. Cucio C, Overmars L, Engelen AH, Muyzer G. 2018. Metagenomic analysis shows the presence of bacteria related to free-living forms of sulfur-oxidizing chemolithoautotrophic symbionts in the rhizosphere of the seagrass Zostera marina. Front Mar Sci 5:171. https://doi.org/10.3389/fmars.2018.00171.
39. Lam TH, Verzotto D, Brahma P, Ng AHQ, Hu P, Schnell D, Tiesman J, Kong R, Ton TMU, Li J, Ong M, Lu Y, Swaile D, Liu P, Liu J, Nagarajan N. 2018. Understanding the microbial basis of body odor in pre-pubescent children and teenagers. Microbiome 6:1–14. https://doi.org/10.1186/s40168-018-0588-z.
40. Poole AC, Goodrich JK, Youngblut ND, Luque GG, Ruaud A, Sutter JL, Waters JL, Shi Q, El-Hadidi M, Johnson LM, Bar HY, Huson DH, Booth JG, Ley RE. 2019. Human salivary amylase gene copy number impacts oral and gut microbiomes. Cell Host Microbe 25:553–564. https://doi.org/10.1016/j.chom.2019.03.001.
41. Gastauer M, Vera MPO, De Souza KP, Pires ES, Alves R, Caldeira CF, Ramos SJ, Oliveira G. 2019. A metagenomic survey of soil microbial communities along a rehabilitation chronosequence after iron ore mining. Sci Data 6:1–10. https://doi.org/10.1038/sdata.2019.8.
42. Wilkins LG, Ettinger CL, Jospin G, Eisen JA. 2019. Metagenome-assembled genomes provide new insight into the microbial diversity of two thermal pools in Kamchatka, Russia. Sci Rep 9:1–15. https://doi.org/10.1038/s41598-019-39576-6.
43. Mardanov AV, Kotlyarov RV, Beletsky AV, Nikolaev YA, Kallistova AY, Grachev VA, Berestovskaya YY, Pimenov NV, Ravin NV. 2019. Metagenomic data of the microbial community of lab-scale nitritation-anammox sequencing-batch bioreactor performing nitrogen removal from synthetic wastewater. Data Brief 27:104722. https://doi.org/10.1016/j.dib.2019.104722.
44. Arumugam K, Bağcı C, Bessarab I, Beier S, Buchfink B, Gorska A, Qiu G, Huson DH, Williams RB. 2019. Annotated bacterial chromosomes from frame-shift-corrected long-read metagenomic data. Microbiome 7:1–13.
45. Wickham H. 2016. ggplot2: elegant graphics for data analysis. Springer-Verlag, New York, NY.