Bioinformatics, 38(3), 2022, 631–647
https://doi.org/10.1093/bioinformatics/btab703
Advance Access Publication Date: 12 October 2021
Original Paper

Sequence analysis

MCRL: using a reference library to compress a metagenome into a non-redundant list of sequences, considering viruses as a case study

Arbel D. Tadmor [1,2,*] and Rob Phillips[3,4]

[1]TRON - Translational Oncology at the University Medical Center of Johannes Gutenberg University, 55131 Mainz, Germany, [2]Department of Biochemistry and Molecular Biophysics, California Institute of Technology, Pasadena, CA 91125, USA, [3]Department of Bioengineering, California Institute of Technology, Pasadena, CA 91125, USA and [4]Department of Applied Physics, California Institute of Technology, Pasadena, CA 91125, USA

*To whom correspondence should be addressed.

Associate Editor: Pier Luigi Martelli

Received on October 29, 2019; revised on October 3, 2021; editorial decision on October 5, 2021; accepted on October 7, 2021

Abstract

Motivation: Metagenomes offer a glimpse into the total genomic diversity contained within a sample. Currently, however, there is no straightforward way to obtain a non-redundant list of all putative homologs of a set of reference sequences present in a metagenome.

Results: To address this problem, we developed a novel clustering approach called ‘metagenomic clustering by reference library’ (MCRL), where a reference library containing a set of reference genes is clustered with respect to an assembled metagenome. According to our proposed approach, reference genes homologous to similar sets of metagenomic sequences, termed ‘signatures’, are iteratively clustered in a greedy fashion, retaining at each step the reference genes yielding the lowest *E* values, and terminating when signatures of remaining reference genes have a minimal overlap. The outcome of this computation is a non-redundant list of reference genes homologous to minimally overlapping sets of contigs, representing potential candidates for gene families present in the metagenome. Unlike metagenomic clustering methods, there is no need for contigs to overlap to be associated with a cluster, enabling MCRL to draw on more information encoded in the metagenome when computing tentative gene families. We demonstrate how MCRL can be used to extract candidate viral gene families from an oral metagenome and an oral virome that otherwise could not be determined using standard approaches. We evaluate the sensitivity, accuracy and robustness of our proposed method for the viral case study and compare it with existing analysis approaches.

Availability and implementation: https://github.com/a-tadmor/MCRL.

Contact: arbel.tadmor@tron-mainz.de

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

In recent years, the field of metagenomics has revolutionized our understanding of uncultured bacteria and viruses in Earth’s biosphere. Deciphering the vast amounts of genomic data encoded in metagenomes is challenging and requires novel data mining approaches. Interpretation of metagenomes typically involves an annotation phase performed, e.g. by alignment of reads against public sequence databases, such as the RefSeq database (Pruitt *et al.*, 2007) or the National Center for Biotechnology Information (NCBI) nr database (Sayers *et al.*, 2020), or by homology-based searches of query sequences against databases containing known functional or taxonomic information (Huntemann *et al.*, 2016; Meyer *et al.*, 2008; Oulas *et al.*, 2015). For example, in PATRIC (Brettin *et al.*, 2015) annotation is based on alignment of *k*-mers (Edwards *et al.*, 2012) against protein families in SEED (Meyer *et al.*, 2009), and in the case of the DOE-JGI Microbial Genome Annotation Pipeline (Mavromatis *et al.*, 2009), protein coding genes are compared to protein families using databases, such as Pfam (Bateman *et al.*, 2004) and TIGRFAM (Haft *et al.*, 2003), functional annotation databases, such as KEGG (Kanehisa and Goto, 2000) and COG/KOG (Tatusov *et al.*, 2000), and composite protein domain databases, such as Interpro (Hunter *et al.*, 2008). Conventional annotation approaches, however, do not address the question of whether a metagenome contains homologs of a pre-defined group of genes of interest, or provide a non-redundant list of all sequences present in a metagenome that are homologous to such a pre-defined group of genes.

For well-studied environments, reference catalogs can be established and reads can be screened directly against those catalogs. For example, for studies focusing on the human gut, a human gut microbial integrated gene catalog was established as part of the MetaHIT project (Li *et al.*, 2014) and studies of the human gut microbiome can use this database to screen reads. Indeed, for certain applications dedicated bioinformatic pipelines have been established. For example, to infer community function of human-associated microbial communities directly from reads, HUMAnN (Abubucker *et al.*, 2012) was developed as part of the Human Microbiome Project (Methé *et al.*, 2012). For analysis of viral communities, many dedicated tools for detection of viral sequences and/or assembly of viral genomes using reference databases have been developed, including, e.g. VIP (Li *et al.*, 2016), SURPI (Naccache *et al.*, 2014), VirFinder (Ho and Tzanetakis, 2014) and VirusFinder (Wang *et al.*, 2013). However, such methods (further discussed in Section 5) do not provide information regarding gene redundancy, nor are they designed to remove such redundancy. For example, closely related genes or non-overlapping sequences stemming from a common genomic source would be annotated and reported separately. Furthermore, methods, such as those described above are not expected to work well when metagenomic sequences considerably diverge from reference sequences or when applied to novel poorly studied communities that do not have conventional gene catalogs. An alternative approach is to map reads to specific gene collections or gene families using targeted assembly approaches, such as implemented in MEGAN (Huson *et al.*, 2017), or methods, such as Xander (Wang *et al.*, 2015), and SAT-assembler (Zhang *et al.*, 2014). However, such approaches are expected to have limited sensitivity for highly divergent sequences or when searching for novel diversity. Furthermore, targeted sequencing approaches may result in fragmented assembly of sequences and do not provide information about gene redundancy or remove such redundancy (further discussed in Section 5). Thus, a more generalized approach is needed that is not specific to certain environments or gene sets, and which is capable of providing a non-redundant list of all sequences present in a given metagenome that are homologous to a pre-defined set of genes.

Mining metagenomes for homologs of a pre-defined set of reference sequences is particularly challenging for viruses. Viral genes present in a given sample will often be highly divergent from reference sequences, making reference-based assembly challenging if not impractical for many viral sequences. Furthermore, viral genes can exist as ensembles of closely related sequences due to the high mutation rate of viruses. Such ensembles can therefore contain many closely related variants, resulting in a high degree of sequence redundancy. In addition, local populations of viral genomes often contain a high degree of mosaic diversity due to horizontal gene transfer (Hendrix 2003), potentially adding another layer of redundancy. Assembly-based methods for viral identification are not designed to account for these forms of redundancy and will report each genomic variant as a novel gene or genome even if the entire ensemble of genes or genomes reflects a single cohesive group of closely related genes or viral genomes.

A reverse annotation approach for data mining metagenomes

To circumvent such challenges, we propose a 'reverse annotation' approach called Metagenomic Clustering by Reference Library (MCRL) for extracting non-redundant diversity from a metagenome based on diversity defined in a reference library. MCRL screens an assembled metagenome for the presence of known (i.e. annotated) reference genes and removes redundant reference genes by applying an iterative clustering algorithm to the reference sequences with respect to the given metagenome. Each reference gene is characterized by its signature in the metagenome, defined as the list of contigs that are putative homologs of that reference gene. MCRL takes advantage of the fact that reference genes with similar sequences will have similar signatures in the metagenome in order to group these reference genes together, retaining the reference gene yielding the lowest E value to represent that cluster. This process is repeated iteratively in a greedy fashion, terminating when all reported reference genes have distinct signatures with only residual overlap. The list of reference genes reported by MCRL provides insight into the existence of different putative gene families present in the metagenome that cannot be easily extracted based on standard metagenomic annotation, targeted assembly or by conventional metagenomic clustering approaches (Kopylova *et al.*, 2016; Li *et al.*, 2012).

MCRL analysis focuses on contigs obtained from preassembled metagenomes as opposed to reads because homology can be determined with greater sensitivity and precision when using larger assembled fragments. Thus, by considering *de novo* assembled contigs from unfiltered reads and using a homology metric to score alignments as opposed to aligning reads against reference sequences, MCRL is capable of detecting more divergent sequences from the reference library. Furthermore, by considering contigs, MCRL can effectively remove spurious alignments that otherwise would add noise to signatures and potentially bias clustering. Finally, from a practical perspective, analysis of reads could drastically increase the computation time required by MCRL (in particular the clustering phase), potentially rendering implementation impractical.

Contrasting MCRL with metagenomic clustering approaches

Since MCRL performs clustering with the goal of data reduction, it is useful to contrast MCRL with conventional methods for metagenomic clustering to understand in what way these approaches differ. Conventional metagenomic clustering programs 'compress' a metagenome by grouping similar metagenomic sequences into groups or clusters, and then replace each cluster with a representative metagenomic sequence. Metagenomic clustering can be performed by one of three methods (Kopylova *et al.*, 2016; Navas-Molina *et al.*, 2013): (i) *de novo* clustering, where input sequences are grouped based on pairwise similarity. *De novo* clustering programs include, e.g. mothur (Schloss *et al.*, 2009), CD-HIT (Fu *et al.*, 2012), DNAclust (Ghodsi *et al.*, 2011), Swarm (Mahé *et al.*, 2015), OTUCLUST (Albanese *et al.*, 2015), SUMACLUST (Mercier *et al.*, 2013), UCLUST and USEARCH (Edgar, 2010), (ii) closed-reference clustering, supported e.g. by UCLUST and USEARCH (Kopylova *et al.*, 2016), where sequences that match a reference sequence [e.g. based on BLAST alignments (17)] are clustered and remaining sequences are discarded and (iii) open-reference clustering methods, supported by programs, such as UCLUST (Kopylova *et al.*, 2016), which combine both categories: sequences that match a reference sequence are clustered, and remaining sequences are clustered *de novo* (Kopylova *et al.*, 2016; Navas-Molina *et al.*, 2013). Most metagenomic clustering methods, such as CD-HIT, UCLUST, DNAclust, OTUCLUST and SUMACLUST use a greedy strategy for clustering. Although both MCRL and metagenomic clustering methods have a similar goal, which is to remove redundancy from a metagenome, these approaches are fundamentally different. Whereas the tools described above cluster metagenomic sequences, MCRL clusters metagenomic signatures. We therefore found it insightful to compare results obtained using both methods in order to highlight the unique features of MCRL.

In the present report, we describe the MCRL algorithm and characterize it terms of its sensitivity, accuracy and robustness, contrasting it where possible with metagenomic clustering. Our primary focus in this report is on analysis of viral genes; however, we also discuss additional applications for which MCRL could be useful.

2 Materials and methods

2.1 Definitions

MCRL requires as input a 'reference library', which is FASTA file containing a list of annotated amino acid sequences, termed 'reference genes'. Any group of genes or sequences can be used as a reference library. A reference gene is said to have a *signature* in the metagenome, defined as the list of contigs in the metagenome yielding E values below a given threshold, E_0 (0.001 by default), when aligning the reference gene against the metagenome. Two *signatures* S_i and S_j are then said to overlap if the following condition is satisfied:

$$M(|S_i \cap S_j|/|S_i|,\ |S_i \cap S_j|/|S_j|\ ) \geq T \qquad (1)$$

where $|s|$ is the number of elements in the set s, T is a pre-specified threshold (0.5 by default) and M represents either the *maximum* function for an 'inclusive' overlap condition, or the *minimum* function for a 'stringent' overlap condition. If the *signatures* of two reference genes overlap based on Equation (1), the reference genes are said to be *related*. Note that according to this definition, *related* reference genes can also be distant homologs.

2.2 The MCRL algorithm

2.2.1 Clustering

Initially, MCRL retains only reference genes yielding a minimal E value below a pre-specified E value threshold (E_{th}) when BLASTed against the metagenome (10^{-7} by default). This filtering step is performed in order to retain only reference genes that yield reasonable alignments to contigs. Next, of the remaining reference genes, MCRL proceeds to find for each reference gene all reference genes to which it is *related*, and *elects* from this group the reference gene yielding the minimal E value as the *delegate* to represent the given reference gene in the current iteration. Thus, if a given reference gene is not *elected* to be a *delegate* by any reference gene including itself it will be left out of the next iteration. At the end of a given iteration, some retained reference genes can still be *related*. Therefore, this filtering step is repeated iteratively until no two reference genes reported by MCRL are *related* (see Supplementary Fig. S1a for a diagram summarizing the MCRL algorithm). The final set of reference genes reported by MCRL is non-redundant in the sense that none of the reported reference genes are *related*. Each reference gene reported by MCRL is paired with a 'representative contig', which is the contig yielding the lowest E value in the *signature* of the reported reference gene. Figure 1a and b demonstrates the MCRL algorithm on a simple example.

Reference gene clusters and networks

Each reported reference gene represents a 'reference gene cluster', which comprises all related reference genes yielding higher E values that elected the reported reference gene. Reference genes, which are not part of the reference gene cluster, but elect delegates that eventually lead to the reported reference gene are part of a larger 'reference gene network' associated with the reported reference gene (illustrated in Fig. 1c). The reported reference gene lies at the epicenter of this reference gene network, with all branches leading to it. Supplementary Figure S1b shows examples of reference gene clusters and networks.

2.2.2 Symmetric versus asymmetric clustering

MCRL supports two definitions for the overlap of *signatures*: a symmetric overlap condition, termed stringent overlap, where both reference genes must meet the overlap threshold T in Equation (1) in order to be *related* (and are therefore treated symmetrically by the algorithm), and an asymmetric overlap condition termed inclusive overlap, where it is sufficient for only one of the two reference genes to meet the overlap threshold T in order for the two reference genes to be *related* (i.e. overlap is determined asymmetrically by the algorithm). Symmetric clustering will result in more reference genes being reported compared to asymmetric clustering because a stringent overlap condition is more restrictive in determining overlap between *signatures* compared to an inclusive overlap condition. Consequently, symmetric clustering will result in greater sensitivity to detect gene families residing in the metagenome and hence result in greater granularity but may be less effective in removing redundant reference genes. Asymmetric clustering, in contrast, which is based on a more permissive overlap condition, leads to more agglomerative clustering and hence to fewer reference genes being reported, but with a smaller inter-signature overlap. Asymmetric clustering will therefore lead to a less redundant list of reported reference genes. The choice for the overlap condition depends on the desired application. To maximize sensitivity a stringent overlap condition should be selected, and to minimize redundancy of reported reference genes an inclusive overlap condition should be selected. Both clustering approaches will be demonstrated below on real-world examples.

2.2.3 Sensitivity of clustering to input parameters

Clustering by MCRL is controlled by three input parameters: (i) the overlap criterion, M, (ii) the E value threshold defining signatures, E_0 (the minimal E value that all contigs in a signature must pass) and (iii) the threshold for signature overlap, T. The E value threshold for detection, E_{th}, is a filtering parameter and not a clustering parameter: all reference genes are initially screened and only reference genes yielding a minimal E value below E_{th} are retained for analysis. E_{th} therefore sets the desired sensitivity for homology for the initial selection phase of reference genes and does not affect the clustering process itself. It is therefore also possible to apply this filter post-clustering. The default value for E_{th} is relatively low (10^{-7}) to ensure accuracy of reference gene assignment while recovering the majority of reported reference genes with a signature size of 10 or higher (Supplementary Fig. S2). Once the overlap criterion is selected and the desired filtering threshold E_{th} is set, the list of reported reference genes is relatively robust to the choice of E_0 and T. Supplementary Table S1 summarizes the impact of changing MCRL input parameters on predictions for case examples. This topic as well as the choice for default parameters is discussed in greater detail in Supplementary Discussion S1.1.

2.2.4 Implementation

The metagenome provided to MCRL should be assembled into contigs and can be provided in either amino acid format or nucleotide format as a FASTA file. Metagenomes do not need to be annotated, but if annotation is provided the annotation of the representative contig will be presented in the final output along with the annotation of the reported reference genes. The reference library itself is simply a FASTA file of annotated amino acid sequences. However, MCRL is designed to accept as input also reference libraries based on the RefSeq database, which include in addition to a FASTA file a GenPept file containing additional annotation. MCRL will then present the annotation from both files for each reported reference gene. This is a useful feature because RefSeq FASTA file annotation is often insufficient to define the gene. All alignments in MCRL are performed on amino acid sequences using either the gapped BLAST algorithm (Altschul *et al.*, 1997; Camacho *et al.*, 2009), or DIAMOND (Buchfink *et al.*, 2015), which is faster when using large reference libraries.

Since MCRL is computationally intensive due to the large number of alignments and pairings that need to be performed, both the alignment phase and the clustering phase can be run using multiple threads. A runtime benchmark of MCRL is provided in Supplementary Table S2, and a more detailed discussion regarding the complexity of MCRL and software operation is provided in Supplementary Discussion S1.2.

2.3 Evaluation of MCRL

2.3.1 *In silico* spike-in experiments

As a case study, we focused on the problem of mapping viral genes in metagenomes. To accomplish this, we provided MCRL with the viral RefSeq database as a reference library. When MCRL is applied to a metagenome with the viral RefSeq database as an input reference library, MCRL will return a list of non-redundant viral gene families present in the analyzed metagenome with homologous counterparts in the viral RefSeq database. To determine the sensitivity and accuracy of MCRL, we performed in Section 3 a series of *in silico* experiments where we spiked a baseline metagenome with controlled viral reference genes at tuned mutation rates. For the spike-in experiments, we selected the large terminase subunit (TerL) gene, a component of the DNA packaging and cleaving mechanism of double-stranded DNA phages (Rao and Feiss, 2008). The TerL gene is considered to be one of the most universally conserved phage genes in nature (Casjens, 2003) due to the presence of certain universally conserved motifs in this gene (Rao and Feiss, 2008). TerL genes

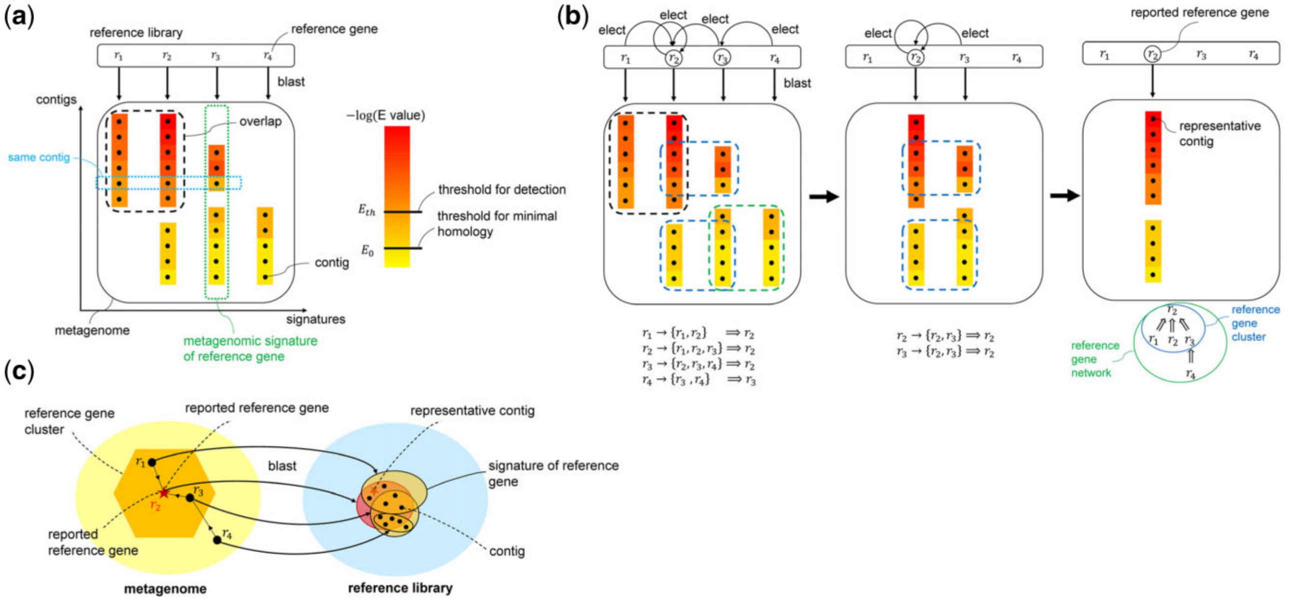

Fig. 1. Schematic illustration of the MCRL algorithm. (**a**) Each reference gene r_i is aligned against the given metagenome yielding a metagenomic signature. Metagenomic signatures are depicted as a set of vertical black dots inside rectangles (e.g. the dotted vertical green rectangle is the signature of reference gene r_3), color-coded according to the E value yielded by each contig. Each horizontal line in this diagram represents a single contig, showing to which signatures the given contig belongs (e.g. the dotted horizontal blue rectangle represents a contig that is part of the signatures of r_1, r_2 and r_3). In this example, the signatures of r_1 and r_2 have a 60% overlap based on Equation (1) using the stringent definition of overlap and a 100% overlap using the inclusive definition of overlap (overlap is indicated by the dashed black rectangle). The color bar shows the range of E values, indicating the threshold for detection of a reference gene (E_{th}), and the minimal threshold for defining homology (E_0). (**b**) Illustration of the MCRL clustering algorithm using a stringent overlap condition. In this example, there are four reference genes. The diagram indicates which reference genes are *related*, and which reference genes were *elected* at each iteration. For example, r_1 is *related* to itself and to r_2 (denoted as $r_1 \rightarrow \{r_1,\ r_2\}$). Since the E value of r_2 was lower than r_1, r_1 *elected* r_2 as the *delegate* (denoted $r_1 \Longrightarrow r_2$). In this manner, in the first iteration, r_2 and r_3 were *elected*, and in the second iteration, r_3 *elected* r_2. Therefore r_2 was the reported reference gene for this cluster. Also shown are the reference gene cluster and the reference gene network that result from this set of reference genes. (**c**) An alternative representation of the diagram shown in panel (b). Each reference gene reported by MCRL lies at the epicenter (red star) of a network of reference genes (black dots). Each node in the network represents a reference gene, and each edge is directed and connects a reference gene to its *delegate*, forming tracks leading to the epicenter. A reference gene cluster is defined as the collection of nodes one edge away from the epicenter, and therefore by definition *related* to the reported reference gene (illustrated by overlapping signatures in the metagenome). The representative contig is denoted in this illustration as a red star in the metagenomic signature of the reported reference gene

in nature fall into at least eight distinct families based on their end-generation function, forming eight robust phylogenetic groups (Casjens *et al.*, 2005). We therefore selected for each of these eight TerL gene families a representative gene (summarized in Supplementary Table S3), extracted a random fragment from this representative gene spanning the mean contig length of the baseline metagenome (N) and spiked the baseline metagenome with a single fragment per gene family as illustrated in Supplementary Figure S3. This process was repeated 10 times to generate 10 spike-in metagenomes. To simulate TerL sequence diversity, we inserted mutations in each fragment at a specified mutation rate by substituting a fixed percent of amino acids, P, with mutations, as illustrated in Supplementary Figure S3, such that $P = 100(1 - n/N)$ with N=124 amino acids, and n=0, 12, 25, 37, 50, 62, 74 and 87 amino acids. To simulate a realistic substitution model for our spike-in experiments, random sites were substituted based on transition probabilities calculated from an alignment of TerL alleles comprising 130 unambiguous amino acid spanning 53 TerL genes that reproduce the TerL gene phylogeny determined by Casjens *et al.* (2005) for 7 of the TerL gene families. Transition probabilities were determined in a similar way to those calculated for protein blocks by the BLAST algorithm (Henikoff and Henikoff, 1992). Since the local alignment performed by MCRL is not position sensitive, mutations were introduced at random positions in the gene. As a baseline metagenome, we used a whole community oral metagenome obtained from a periodontally healthy human subject (Xie *et al.*, 2010) with a mean contig length of 372 ± 127 (SD) nucleotides, and a median contig length of 411 nt comprising in total ∼80 000 contigs (MG-RAST (Glass *et al.*, 2010) identifier mgm4446622.3), using the translated assembled metagenome provided by the authors. In Supplementary Analysis S2, we extend our benchmark using the same spike-in methodology to capsid and nucleocapsid protein families organized based on their architectural classes (Krupovic and Koonin, 2017). For TerL sequences MCRL was used with the BLAST aligner. For capsid and nucleocapsid sequences MCRL was used with both BLAST and DIAMOND and results are compared in Supplementary Analysis S2.

2.3.2 Evaluation of MCRL on experimental datasets

To demonstrate MCRL predictions for real experimental datasets, in Section 4, we applied MCRL to two oral datasets: (i) a whole community oral metagenome obtained from a periodontally healthy human subject (Belda-Ferre *et al.*, 2012) with a mean contig length of 354 ± 114 (SD) nucleotides and a median contig length of 390 nt comprising in total ∼180 000 contigs (mgm4447192.3), and (ii) an oral virome of a periodontally healthy human subject (Pride *et al.*, 2012) with a mean contig length of 357 ± 135 (SD) nucleotides and a median contig length of 393 nt comprising in total ∼140 000 contigs (mgm4446120.3). The oral virome is a translated metagenome of viral particles obtained by filtering a saliva sample through a 0.2 μm filter followed by cesium chloride gradient purification (Pride *et al.*, 2012). In both cases, we used the translated assembled metagenome provided by the authors. For all analyses, we used the viral RefSeq database (Pruitt *et al.*, 2007) release 95 as an input reference library for MCRL.

2.4 Benchmarking MCRL against a metagenomic clustering program

MCRL was benchmarked against the metagenomic clustering program CD-HIT (Fu *et al.*, 2012; Li and Godzik, 2006). All CD-HIT parameters except for the sequence identity threshold were set to their default values as used by the webserver application. For a sequence identity threshold of 30% the benchmark was performed against PSI-CD-HIT, which is the algorithm selected by the webserver for this sequence identity threshold.

3 Performance evaluation of MCRL

The sensitivity of MCRL to detect and resolve different gene families depends on several factors, including the diversity present in the metagenome, the ability of the reference library to resolve this diversity, and the coverage of the metagenome. In a series of *in silico* spike-in experiments, we determined the sensitivity and accuracy of MCRL to detect and resolve TerL gene fragments, averaging results across the eight primary families of the TerL gene (see Section 2).

3.1 Estimation of MCRL sensitivity using *in silico* spike-in experiments

Detection sensitivity for a given mutation rate (given by 100 – *P*) was calculated as the average fraction of spiked TerL fragments captured by signatures of reported reference genes, averaged across 10 spike-in metagenomes (in total 10 trials × 8 mutation rates = 80 metagenomes). This analysis was repeated twice, once using a stringent overlap condition and once using an inclusive overlap condition, resulting in 160 spike-in metagenomic analyses in total. We found that for mutation rates corresponding to >∼70% identity at the amino acid level, sensitivity exceeded 80–95% for the inclusive and stringent overlap conditions, respectively, and below ∼70% identity sensitivity rapidly decreased (solid lines in Fig. 2). Sensitivity per reference gene *cluster*, in comparison, decreased more gradually with mutation rate (dot-dashed lines in Fig. 2), reflecting the larger signature span of reference gene clusters. For both overlap conditions, signatures were nearly always unique to a single TerL gene family regardless of the mutation rate (Supplementary Table S4, dashed lines in Fig. 2). Hence, the capacity of MCRL to discriminate TerL gene families was high and independent of mutation rate or overlap condition. Figure 2 further shows that a stringent overlap condition resulted in higher sensitivity compared to an inclusive overlap condition, as expected. Moreover, for >∼70% identity, ∼75% of spiked TerL fragments were reported as representative contigs when using a stringent overlap condition, rapidly decreasing below ∼70% identity. On the other hand, with a stringent overlap condition, often two or more reported reference genes corresponded to the same spiked TerL fragment, whereas with an inclusive overlap condition this did not occur: each spiked TerL fragment always corresponded to a single reported reference gene (Supplementary Table S4), consistent with the inclusive overlap condition being less redundant.

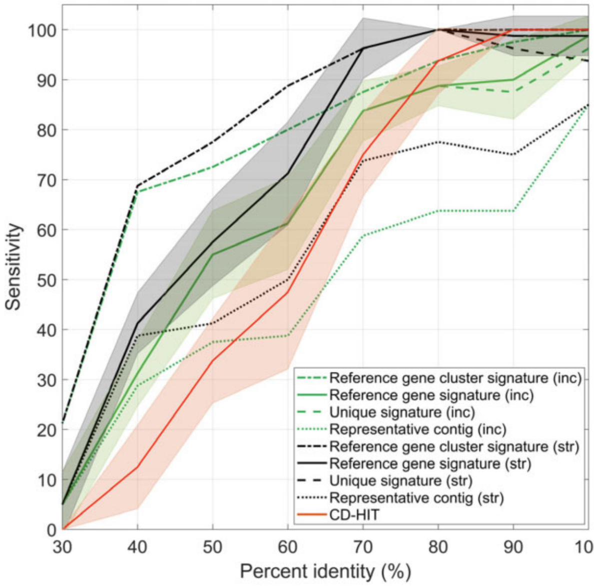

Fig. 2. MCRL and CD-HIT sensitivity determined by *in silico* spike-in experiments. Solid black and green lines show MCRL sensitivity to detect spiked TerL fragments in the signature of reported reference genes as a function of the simulated mutation rate. Shaded areas correspond to 1 SD. Dashed lines show sensitivity when additionally requiring uniqueness: i.e. requiring that signatures positive for a given spiked TerL fragment do not contain spiked fragments from other TerL gene families. Dotted lines further require that the spiked TerL fragments were the representative contigs. Dash-dotted lines show the sensitivity to detect spiked TerL fragments in the signature of reference gene clusters. The solid red line shows CD-HIT sensitivity to detect spiked TerL fragments when clustering the spiked metagenome together with the viral reference library using a sequence identity threshold of 30%. inc, inclusive overlap; str, stringent overlap

Inspecting the individual contribution of each TerL gene family to sensitivity, we found that certain TerL gene families were more dependent on the applied overlap condition than others (Table 1). For example, sensitivity for detection of HK97 fragments was significantly lower compared to other TerL gene families when applying the inclusive overlap condition. This drop in sensitivity occurs because with asymmetric clustering, a reference gene with a short signature can be elected to replace a reference gene with a long signature leading to potential loss of information encoded in the long signature. This form of information loss does not occur with symmetric clustering, and indeed detection sensitivity for HK97 fragments was restored when using the stringent overlap condition (Table 1).

Interestingly, both the Mu-like and RhodoGTA representatives, despite having only remote homologs in the reference library performed on par with other TerL representatives that had exact matches in the reference library, such as Lambda and P2 (Supplementary Table S3). This result suggests that MCRL can attain high sensitivity also when the reference library contains only distant homologs of the genomic source, possibly indicating that homology is better preserved by natural evolution than it was preserved in our simulation, in which case our simulation may reflect a worst-case scenario.

3.2 Accuracy of reference gene assignment

To evaluate the accuracy of assigning contigs to candidate gene families, we checked whether reference genes reported by MCRL that corresponded to spiked TerL fragments (listed in Supplementary Table S5) had the correct phylogenic placement, i.e. matched the TerL family to which the spiked TerL fragment belonged. A maximum likelihood phylogenetic analysis showed that for the stringent overlap condition 41 of 44 reported reference genes grouped with the correct TerL gene family, and for the inclusive overlap condition 37 of 38 reported reference genes grouped with the correct TerL gene family (Supplementary Fig. S4a), demonstrating that MCRL associated the spiked TerL fragments with reference genes belonging to the correct TerL gene family with high accuracy. This conclusion was further corroborated by the classification of the phage genomes harboring the reference genes (Supplementary Table S5).

3.3 Clustering spiked metagenomes with CD-HIT

To compare MCRL with metagenomic clustering, we selected a representative program for this computational approach and performed an in-depth comparison between both methods. For our comparison, we selected CD-HIT, a popular method for metagenomic clustering that has been used in large-scale sequencing projects, such as UniProt (Suzek *et al.*, 2007), SWISS-MODEL (Arnold *et al.*, 2006) and CAMERA (Sun *et al.*, 2010), and based on citations continues to be frequently used. When clustering spiked metagenomes with CD-HIT (Fu *et al.*, 2012; Li and Godzik, 2006) using default input parameters, we found that regardless of the mutation rate, each spiked TerL fragment always mapped to a cluster of one element comprising only that given TerL fragment, indicating that these spike-in fragments were indeed foreign to the baseline oral metagenome and therefore did not cluster with any other contig. We next clustered the spiked baseline metagenome together with the viral reference library using different values for the sequence identity threshold, which is a global clustering threshold parameter used by CD-

Table 1. MCRL and CD-HIT sensitivity to detect spiked TerL fragments for different TerL gene families: the table is calculated for MCRL with an inclusive overlap condition, MCRL with a stringent overlap condition and CD-HIT with a sequence identity threshold of 30%

		TerL spike in (n=10 metagenomes)								
		T4	T7	Mu-like	P22	P2	Lambda	RhodoGTA	HK97	
Percent identity	100%	100	100	100	100	100	100	90	100	(a) Inclusive overlap
	90%	100	100	100	100	100	100	90	30	
	80%	100	100	100	100	100	100	100	10	
	70%	100	100	100	100	100	100	70	0	
	60%	100	100	70	100	100	10	10	0	
	50%	100	100	70	100	70	0	0	0	
	40%	30	100	20	100	0	0	0	0	
	30%	10	0	0	30	0	0	0	0	
	100%	100	100	100	100	100	100	90	100	(b) Stringent overlap
	90%	100	100	100	100	100	100	90	100	
	80%	100	100	100	100	100	100	100	100	
	70%	100	100	100	100	100	100	70	100	
	60%	100	100	70	100	100	10	10	80	
	50%	100	100	70	100	70	0	0	20	
	40%	100	100	20	100	10	0	0	0	
	30%	10	0	0	30	0	0	0	0	
	100%	100	100	100	100	100	100	100	100	(c) CD-HIT
	90%	100	100	100	100	100	100	100	100	
	80%	100	90	100	100	100	80	80	100	
	70%	100	70	50	100	90	60	30	100	
	60%	100	20	40	100	40	30	0	50	
	50%	80	0	10	100	30	10	0	40	
	40%	20	0	0	80	0	0	0	0	
	30%	0	0	0	0	0	0	0	0	

Note: The table shows for each simulated mutation rate the percent of metagenomes (out of 10) for which a spiked TerL fragment was included in a signature of a reported reference gene (MCRL) or clustered with a reference gene (CD-HIT). The graded shading corresponds to the values in the table such that the lower the percent value in the table the lighter the shade.

HIT (defined as the threshold for the number of identical amino acids in an alignment divided by the length of the shorter sequence). We then determined for each sequence identity threshold the sensitivity of CD-HIT to detect spiked TerL fragments. Sensitivity was determined per given mutation rate by calculating the fraction of spiked TerL fragments that clustered together with at least one reference gene, averaged across 10 trials. This calculation was repeated for eight different mutation rates and four CD-HIT sequence identity thresholds (30%, 50%, 70% and 90%), resulting in a total of 320 clustered metagenomes. We found that CD-HIT sensitivity was highest when using a sequence identify threshold of 30% (Supplementary Fig. S5), where CD-HIT applies the PSI-CD-HIT algorithm using BLAST alignments (see Section 2.4). Although the individual contribution of each TerL gene family to CD-HIT sensitivity displayed a comparable distribution to MCRL using a stringent overlap condition (Table 1), MCRL sensitivity using the stringent overlap condition was on average higher (Fig. 2). For example, for mutation rates corresponding to 40–80% identity, MCRL sensitivity was higher on average by 21%. When considering capsid and nucleocapsid sequences, we found that the gap in sensitivity increased by ~14%, and, in addition, CD-HIT sensitivity was less uniform across different capsid families compared to MCRL (Supplementary Analysis S2). These results reflect the fact that despite including the reference library in the CD-HIT clustering process, contigs do not necessarily cluster with reference genes. MCRL, in contrast, is designed to align a reference library against a metagenome.

We next evaluated the accuracy of assigning contigs to CD-HIT clusters in a similar manner to our analysis in Section 3.2. Since the spiked TerL gene fragments were foreign to the baseline metagenome, when CD-HIT cluster representatives were not the spiked TerL fragments themselves, they were the reference library genes. Furthermore, since the CD-HIT algorithm is constructed such that query sequences are compared to longer representatives (Holm and Sander, 1998; Li *et al.*, 2001), when a spiked fragment clustered with a reference gene, the representative sequence was the longer reference gene. We therefore checked whether each representative (reference) sequence of a CD-HIT cluster containing a spiked TerL fragment had the same phylogenetic placement as the TerL family to which the spiked TerL fragment belonged (see Supplementary Table S5 for a list of all representative reference sequences). A phylogenetic analysis of these representative sequences showed that CD-HIT cluster assignment was accurate for all 25 representative sequences (Supplementary Fig. S4b). When considering capsid and nucleocapsid sequences, however, we found that although CD-HIT was accurate, it was less effective compared to MCRL in detecting reference genes that were closely related to the representatives of the capsid and nucleocapsid protein families used as the templates for the spike-in experiments. This limitation reflects the fact that representatives in CD-HIT are prioritized for length and not homology (Supplementary Analysis S2).

3.4 Detecting contigs originating from a common genomic source

In our simulation in Section 3.1, we spiked each metagenome with a single contig for each TerL gene family. However, a more realistic scenario would be one where for each TerL gene family the metagenome contains multiple contigs mapping to random regions of the source TerL gene. Such a situation can arise when coverage is sufficiently high, and the length of the source gene is long compared to the length distribution of contigs in the metagenome, resulting in contigs with no overlap. In our baseline metagenome, e.g. the mean contig length corresponded to 124 ± 42 (SD) amino acids, which is significantly shorter than the typical length of TerL genes (on average ~550 amino acids). Furthermore, when sequence diversity is present, as is often the case with viral sequences, different alleles encoding different variants of the gene family can co-exist in the sample, resulting in a collection of contigs encoding different variants that can be partially overlapping.

We therefore performed a second series of *in silico* spike-in experiments where we simulated a distribution of contigs originating from a single common genomic source belonging to a specific TerL gene family, with each fragment encoding a slightly different variant. The baseline metagenome was then spiked with ten contigs per TerL gene family per baseline mutation rate (Supplementary Fig. S6). Our simulation showed that nearly all spiked TerL fragments were associated with a single reported reference gene (Supplementary Table S6). For example, for the inclusive overlap condition, even at 40% identity, for six out of eight TerL gene families at least 90% of fragments were associated with a single reported reference gene.

3.5 Detecting a common genomic source with CD-HIT

When clustering spiked metagenomes with CD-HIT using a sequence identity threshold of 70% at the amino acid level, which should capture the low level of sequence diversity that we introduced into TerL fragments, CD-HIT was capable of grouping only about 25% of gene fragments across all mutation rates (Supplementary Table S6). This result is expected for metagenomic clustering methods because contigs originating from different regions of a gene may end up in different clusters if the gene is long enough compared to the mean contig length. MCRL, however, leverages the reference gene to act as a pseudo scaffold to collect as many homologous contigs as possible emanating from a common genomic source, given the limits of homology. For the same reason, when adding the reference library to the CD-HIT clustering process, CD-HIT performance improved (Supplementary Table S6 and Supplementary Fig. S7). For example, when applying a 30% sequence identity threshold CD-HIT was capable of grouping on average 65% of gene fragments (range: 46.3–90%) across all mutation rates compared to an average success rate of 47% ± 1.6% when the reference library was not included in the clustering process. However, since there is no mechanism in CD-HIT to ensure that contigs cluster with reference genes, MCRL with a stringent overlap condition outperformed CD-HIT for percent identities ≥40%, and was able to cluster on average 88% ± 11.2% of fragments compared to 67.5% ± 15.6% for CD-HIT, a gain of ~20% (Supplementary Fig. S7). In Supplementary Analysis S2, we further show that for capsid gene families MCRL was able to cluster on average 98.1% of fragments down to 40% identity, whereas CD-HIT was able to cluster on average only 69.8% of fragments in the same range (with the reference library included in the clustering process), and with less uniformity across different capsid families. In Section 5, we provide a more detailed comparison between MCRL and CD-HIT.

3.6 Sensitivity of MCRL to stochastic perturbations

The final set of contigs comprising a metagenome can be impacted by various factors, such as coverage, the specific program used for assembly and assembly parameters. We therefore explored the degree to which stochastic perturbations applied to a metagenome impacted MCRL results. To determine this impact, we first evaluated the effect of stochastic perturbations on the overlap of signatures as defined by Equation (1). To this end, we performed a simulation where for each combination of signature lengths L_1 and L_2, we introduced a perturbation such that each contig in each signature has a probability of p to be either discarded (50% chance) or duplicated (50% chance), setting p such that for L_i=3 the length of perturbed signatures has a standard deviation of 1. We then calculated the degree of concordance between the overlap decision before and after the perturbation, averaging over all possible overlap scores between 0 and 1. We found that overlap decisions were reproducible for both overlap conditions over all signature length ranges (Supplementary Fig. S8a). For example, the average concordance for the stringent and inclusive overlap conditions was 97%±2.9% and 94%±2.7%, respectively, when calculated up to a signature length of 50. When comparing short versus long signatures, we found that concordance was ~97% when both signatures were 40 or longer, and ~91% when both signatures were 10 or shorter, irrespective of the overlap condition. Thus, for both overlap conditions the overlap metric was quite robust to perturbations, also when signatures were short.

To check the impact of stochastic perturbations on MCRL performance, we performed a simulation where we perturbed a baseline oral metagenome by randomly selecting P percent of all contigs in the metagenome, and then each selected contig was either discarded or duplicated with 50% chance. This experiment was repeated 10 times for each selected value of P, testing values of P between 5% and 40% in 5% jumps (80 metagenomes in total). We then determined the average concordance C between MCRL predictions for the baseline (unperturbed) metagenome and the perturbed metagenomes. We found that MCRL predictions for both the stringent and inclusive overlap prediction were indeed robust to the stochastic perturbations that were introduced (Supplementary Fig. S8b). For example, when 20% of contigs were randomly perturbed (on average 10% duplicated and 10% discarded), MCRL was still capable of detecting ~90% of the original reported reference genes, with long signatures ($L \geq 40$) exhibiting a concordance of ~95% for both overlap conditions, and short signatures ($L \leq 10$) exhibiting a concordance between ~88% (stringent overlap) and ~92% (inclusive overlap). Overall, the degree of discordance ($100 - C$) was proportional to the degree of perturbation (P), with the inclusive overlap condition exhibiting slightly more robustness to perturbations (Supplementary Fig. S8c), suggesting that MCRL behaved in a predictable manner when perturbed.

Finally, we explored the impact of perturbations as a function of signature length (L) by calculating the average percent of reported reference genes with signature length L in the baseline (unperturbed) metagenome whose signatures were perturbed and were still reported by MCRL in the perturbed metagenome (Supplementary Fig. S8d). Our simulation shows that reference genes with short signatures were generally robust to perturbations, with the inclusive overlap condition performing slightly better. For example, for signature lengths between 2 and 5, the concordance with the unperturbed metagenome was on average 86.5% for the inclusive overlap condition, and 80.7% for the stringent overlap condition, and for signature lengths between 6 and 10 concordance increased to 95.1% for the inclusive overlap condition, and 92.2% for the stringent overlap condition (for L=1 concordance drops to 50% because there is a 50% chance that the single contig comprising the signature is discarded).

3.7 Impact of chimeras

Chimeric DNA sequences occur when DNA polymerases incompletely extend a segment of DNA resulting in a partially amplified sequence, which can then hybridize to an alternative form of the template strand (Bradley and Hillis, 1997). As a result, PCR products that are a fusion of two original template sequences are generated (Bradley and Hillis, 1997). Since chimeric contigs can confound analysis by being incorrectly interpreted as novel sequences, we explored how MCRL classifies chimeric sequences. To check the impact of chimeras on cluster assignment, we spiked TerL fragments corresponding to the eight TerL gene families into a baseline oral metagenome once as controls (no chimeras), and once as chimeric fragments. Chimeric fragments were generated by fusing a random fragment originating from the i-th TerL gene family at a proportion of $P \geq 0.5$ (major parent) with a random fragment originating from the j-th TerL gene family at a proportion of $1 - P$ (minor parent). In total, each metagenome was spiked with 56 chimeras and 8 template controls, repeating this experiment 10 times for various values of P, as illustrated in Supplementary Figure S9a. To assess the impact of chimeras on MCRL clustering, we calculated the percent of chimeras with a given major parent that were assigned to the same reference gene as the control corresponding to this major parent. Likewise, we calculated the percent of chimeras with a given minor parent that were assigned to the same reference gene as the control corresponding to this minor parent. We found that for both overlap conditions, chimeric contigs were always assigned to the reference gene corresponding to the major parent control (solid lines in Supplementary Fig. S9b). However, because contigs in MCRL have a ‘soft’ assignment, where a given contig can belong to more than one signature (see example in Section 5.3), our simulation showed that for $P < 0.85$ chimeric contigs were also assigned to the reference gene corresponding to the minor parent with a probability that scaled linearly with $1 - P$ (dashed lines in Supplementary Fig. S9b), otherwise the minor parent was not detected (Supplementary Fig. S9c). When contigs contained equal contribution from both parents (P=0.5), both parents were accuracy assigned in all cases. MCRL was therefore robust to chimeras in the sense that the major parent was always robustly detected and accurately assigned, and the minor parent, when detected, was also accurately assigned.

3.8 Runtime benchmark

The most appropriate setting to compare MCRL and CD-HIT runtimes is to use for CD-HIT a sequence identity threshold of 30%, which is the threshold required in order for the sensitivity of CD-HIT to approach the sensitivity of MCRL (Supplementary Fig. S5). Under these conditions, using MCRL with the BLAST aligner and including the reference library in the CD-HIT clustering process, MCRL was about twice as fast as CD-HIT (Supplementary Table S2). When running MCRL with DIAMOND, MCRL was 8.5 times faster than CD-HIT when using one thread per program, and ~4 times faster when using 4 or more threads per program, as shown in Supplementary Table S2 (see Supplementary Discussion S1.2 for further details).

4 Probing viral diversity in metagenomes

4.1 The challenge of mapping viral diversity

Mapping viral diversity is a challenging problem because viruses occupy a large sequence space, with the majority of viral sequences displaying no similarity to proteins from known isolate viruses (Edwards and Rohwer, 2005; Hurwitz *et al.*, 2016; Paez-Espino *et al.*, 2016). Furthermore, local populations of viruses can contain many variants (Hendrix, 2003; Paez-Espino *et al.*, 2016; G. Mahmoudabadi *et al.*, *BioRxiv, doi: https://doi.org/10.1101/516864*, 2021, *In prep.*) existing as quasi-species. Such variability needs to be clustered in order not to overestimate viral diversity (Paez-Espino *et al.*, 2016). In addition, ~60% of sequenced bacterial genomes are predicted to encode at least one integrated phage genetic element (Casjens, 2003; Edwards and Rohwer, 2005). Since lysogenic phages can be functional but not expressed, not all lysogenic phages can be detected in viromes. Therefore, there is also a need for tools capable of extracting viral diversity from whole community shotgun metagenomes where viral and bacterial sequences are intermixed. Existing tools for data reduction, such as metagenomic clustering methods, are not ideally suited to address this problem because only overlapping contigs are clustered, which can result in fragmentation of sequence information (as demonstrated in Section 3.5). Furthermore, determining Operational Taxonomical Units (OTUs) for viral sequences is non-trivial due to the high mutation rate of viruses, which can also be system dependent. More specific methods, such as targeted assembly and assembly-based viral detection tools, are also not ideally suited for this task because these methods do not deal with the problem of sequence redundancy, will have limited sensitivity to detect sequences that are highly divergent from reference sequences and are also susceptible to fragmented assembly.

MCRL can help mitigate some of these challenges. By providing MCRL with a comprehensive viral reference library, MCRL will output the set of non-redundant viral gene families present in the given metagenome (whether whole community or a virome), targeting bacterial and/or eukaryotic viruses, depending on the reference library used. Alternatively, MCRL can be used to focus on specific classes of viral genes, such as certain structural genes, certain genes involved in virion assembly, or focus on certain groups of viruses, such as pathogenic viruses. The advantage of MCRL is that it provides a non-redundant list of viral gene families such that local genetic variants at the individual gene level are grouped together—even if the contigs themselves do not overlap but are homologous to the same reference gene. Another advantage of MCRL is that gene families naturally emerge from the algorithm without the need to prespecify a threshold for OTUs or determine OTUs on the fly (see Section 7).

4.2 Mapping viral gene families in oral samples

Mapping viral gene families in an oral metagenome

We applied MCRL to a translated whole community shotgun oral metagenome obtained from a plaque sample of a periodontally healthy human subject (Belda-Ferre *et al.*, 2012). This metagenome comprised ~180 000 contigs with a median contig length corresponding to 130 amino acids when translated. Of the ~370 000 viral reference genes included in the viral RefSeq database that we used, ~26 000 genes yielded E values below the default E value detection threshold of E_{th}=10^{-7}. Of these, MCRL reported 1152 viral reference genes when applying an inclusive overlap condition compared to 1880 viral reference genes when applying a stringent overlap condition (Table 2, raw MCRL output files for all metagenomes are provided in Supplementary Table S7), with the former being a subset of the latter (Supplementary Table S1). Overall, MCRL provided a data reduction factor between 0.6% and 1.1% (Table 2). For both overlap conditions, the combined signatures of all reference genes in all reference gene clusters covered ~13% of all contigs in the metagenome (Table 2), suggesting that ~87% of contigs were not of viral origin or reflect completely novel viral diversity. When applying an inclusive overlap condition, the average overlap between signatures of reported reference genes was only 3%, compared to 36% when applying a stringent overlap condition (Table 2). Similarly, when applying an inclusive overlap condition, 99.6% of representative contigs were unique (i.e. non-duplicate entries), compared to 87.5% when applying a stringent overlap condition. The differences we observed between the two overlap conditions are consistent with our expectation that a stringent overlap condition will be more sensitive compared to an inclusive overlap condition, but less effective at removing redundant reference genes.

Mapping viral gene families in an oral virome

We next applied MCRL to an oral virome obtained from a filtered saliva sample of a second periodontally healthy human subject (Pride *et al.*, 2012) comprising ~140 000 contigs with a median contig length corresponding to 131 amino acids when translated. Of the ~370 000 viral reference genes, ~27 000 genes yielded E values below the E value detection threshold, E_{th}. Of these, MCRL reported 903 and 1877 viral reference genes when applying inclusive and stringent overlap conditions, respectively, with the former being a subset of the latter (Supplementary Table S1). Here too, MCRL achieved similar data reduction factors as in the oral metagenome (between 0.7% and 1.4%). For both overlap conditions, ~32% of all contigs in the metagenome belonged to the combined signatures of all reference genes in all reference gene clusters (Table 2), 2.5 times more than in the oral metagenome. This difference in coverage stems from the fact that viromes are comprised of viral particles, which are the target entities for the (viral) reference library used is this analysis.

Increasing the E value threshold for signatures, E_0, from 0.001 to 0.01 increased the coverage by only 2.7%, and increasing the E value threshold for detection, E_{th}, from 10^{-7} to 0.001 increased the coverage by only 1.1%. Thus, about two-thirds of contigs in the oral virome could not be characterized in terms of currently known viral diversity, in accordance with the conclusions of the oral virome study (Pride *et al.*, 2012), as well as other studies that show that the majority of natural genomic diversity of viruses remains uncharted (Bench *et al.*, 2007; Edwards and Rohwer, 2005; Paez-Espino *et al.*, 2016). Here too, we found similar trends in redundancy of reported reference genes as observed in the oral metagenome case study: when applying an inclusive overlap condition, the average overlap

Table 2. MCRL and CD-HIT clustering results for the oral metagenome and oral virome

	MCRL (viruses)		CD-HIT				
	Stringent	Inclusive	Ref. library	90%	70%	50%	30%
			Oral metagenome				
No. of reported clusters	1880	1152	without	158 472	124 003	93 596	77 567
			with	158 465	123 975	95 585	77 782
Data reduction factor (%)	1.1	0.6	without	89.2	69.8	52.7	43.7
			with	89.2	69.8	53.8	43.8
Percent of metagenome covered	12.8	12.5	n/a	100	100	100	100
Overlap between metagenomic clusters (%)	36	3.2	n/a	0	0	0	0
			Oral virome				
No. of reported clusters	1877	903	without	64 646	26 594	15 861	15 358
			with	64 619	26 540	16 271	15 669
Data reduction factor (%)	1.4	0.7	without	47.3	19.5	11.6	11.2
			with	47.3	19.4	11.9	11.5
Percent of metagenome covered	32.2	31.4	n/a	100	100	100	100
Overlap between metagenomic clusters (%)	49	7	n/a	0	0	0	0

Note: Data reduction factors were calculated as the ratio of the number of reported reference genes (or CD-HIT clusters) and the total number of contigs in the metagenome. For CD-HIT, results are shown for different sequence identity thresholds with and without combining the metagenome with the reference library. When the reference library was included in the CD-HIT clustering process, clusters containing only reference genes were excluded from the analysis. The 'percent of metagenome covered' in the case of MCRL is calculated as the percent of contigs in all signatures of all reference gene clusters. The 'overlap between metagenomic clusters' in the case of MCRL was calculated as $\frac{1}{N}\sum_{i=1}^{N} \max_{j,j\neq i}\{S_i \cap S_j / S_i\}$, where S_i is the signature of the i-th reported reference gene, and N is the total number of reported reference genes.

between signatures of reported reference genes was 7%, compared to 49% when applying a stringent overlap condition (Table 2), and when applying an inclusive overlap condition, 99.8% of representative contigs were unique compared to 80.5% when applying a stringent overlap condition.

4.3 Viral reference gene networks in the oral cavity

Each reported reference gene is part of a network of reference genes determined by the clustering algorithm. The reference gene cluster is part of this network, and is comprised of all nodes one edge away from the reported reference gene, which forms the epicenter of this network. Figure 3a shows examples of different types of reference gene networks computed for the oral metagenome. Taking a bird's eye view of all reference gene networks computed for the oral metagenome helps to highlight the differences between symmetric (Fig. 3b) and asymmetric (Fig. 3c) clustering in terms of number, size and branch lengths of resulting networks. For example, comparing panels b and c shows that asymmetric clustering is indeed more agglomerative, resulting in fewer, more compact networks, as indicated by the distribution of the longest branch length (Supplementary Fig. S10), and with reference gene clusters containing more nodes (on average about twice as many compared to symmetric clustering).

Overall, between ~15% and ~30% of reported reference genes did not have any related reference genes (Supplementary Fig. S11). Moreover, when color-coding nodes according to the minimal E value yielded by each reference gene (Supplementary Fig. S12), we see that although reported reference genes often yielded significantly better alignments compared to other nodes, in many cases homology was limited for all nodes, including the epicenter. Numerically, 36% and 46% of all reported reference genes resulted in alignments yielding <40% identity for asymmetric and symmetric clustering, respectively. These findings further indicate that the viral RefSeq database was stretched to account for the breadth of viral diversity observed in this sample.

5 Comparison to alternative methods

5.1 Targeted assemblers

An alternative strategy to extract information about specific gene families from metagenomes is to use targeted assemblers. For example, Huson *et al.* (2017) implemented within the metagenomic analysis tool MEGAN (Huson *et al.*, 2016), a method for protein-alignment-guided assembly of orthologous gene families. According to this method a user selects a gene family and all reads binned to that gene family are assembled (Huson *et al.*, 2017). Other methods for targeted assembly include, e.g. Xander (Wang *et al.*, 2015) and the SAT-assembler (Zhang *et al.*, 2014). Xander performs gene-targeted metagenomic assembly using a Hidden Markov Model (Eddy, 2004) for the gene of interest to guide assembly. The SAT-assembler aligns reads to input gene families using HMMER (Eddy, 2009), a profile-based homology search tool, and reads corresponding to specific gene families are subsequently assembled.

Although targeted assemblers address the need to mine metagenomes for specific gene families, assembly-based solutions have certain intrinsic limitations. First, results based on targeted assembly will be redundant when many variants of a given gene or gene set are present in the metagenome, as is often the case with viral genes. Second, an intrinsic limitation of assembly-based approaches is that non-overlapping contigs will not be assembled together, resulting in further redundancy. Such a situation can occur when the gene of interest is significantly longer compared to the length distribution of contigs in the metagenome. For example, in the case of the MEGAN assembler, only contigs that span known protein domains are constructed leading to contig fragmentation (Huson *et al.*, 2017) and hence further redundancy. Furthermore, since methods, such as the SAT-assembler and MEGAN require reads to align to reference protein sequences, these methods will have difficulty detecting novel genes that are highly divergent from the reference gene set. Thus, such an approach may not be ideally suited for investigating novel diversity or sequences that are expected to be highly divergent from the reference genes. Finally, gene-centric assemblers, such as MEGAN and Xander focus on gene families, however, for certain applications it is desired to define reference libraries encompassing larger gene collections. For example, in Section 4, our gene set included all ~370 000 genes comprising the viral RefSeq database.

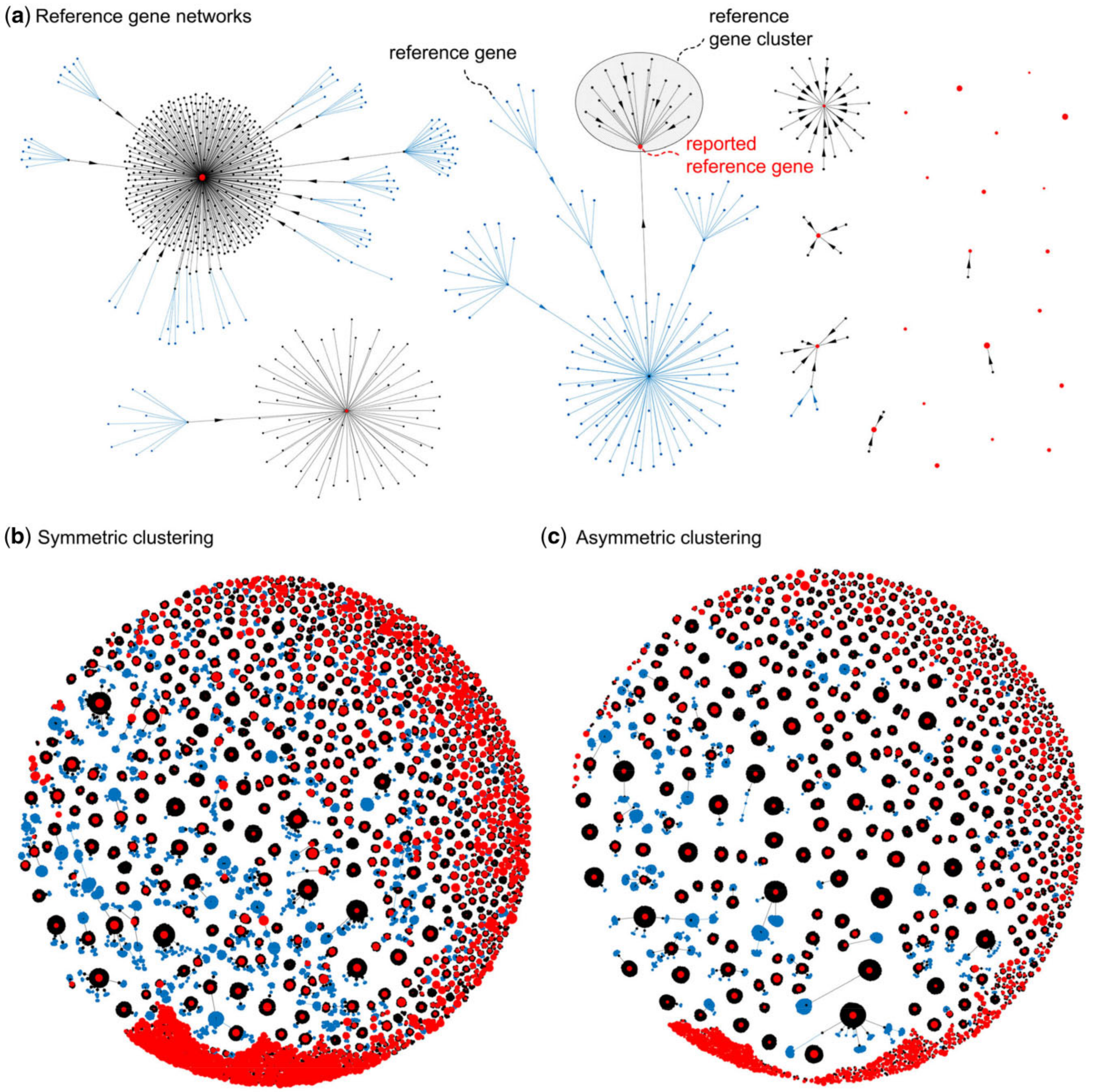

Fig. 3. Viral reference gene networks computed for the oral metagenome. (**a**) Examples of viral reference gene networks computed for the oral metagenome. Each node represents a reference gene. The red node at the epicenter of each network is the reported reference gene, drawn to be proportional to the logarithm of its signature size in the metagenome. Edges of reference gene clusters are drawn in black, and edges leading to reference gene clusters are drawn in blue. For clarity, the directionally of selected edges is shown. Viral reference gene networks computed for the oral metagenome are shown for (**b**) symmetric clustering, and (**c**) asymmetric clustering. The figure shows all genes in the viral RefSeq database passing the *E* value threshold for detection, E_{th} (10^{-7}), comprising in total ~26 000 genes

5.2 Tools for virus discovery

MCRL can also be contrasted with tools for discovery of viruses from metagenomes, such as VIP, SURPI, VirFinder and VirusFinder to name a few. The general strategy of these tools is subtraction of host-related reads followed by alignment to a reference database and assembly of viral genomes. In the case of VIP, reads corresponding to the host are first subtracted, and then remaining reads are either aligned against viral pathogen databases [ViPR (Pickett *et al.*, 2012) and IRD (Zhang *et al.*, 2017)] for detection, or in a second mode, remaining reads are aligned against publicly available viral genomes (e.g. the viral RefSeq database), and matching reads are *de novo* assembled (Li *et al.*, 2016). SURPI similarly first subtracts human reads and then either aligns remaining reads to bacterial and viral databases to identify bacterial and viral related genes, or in a second mode, remaining reads are aligned against NCBI's nr nucleotide database and matching reads are assembled enabling to identify different classes of organisms (bacterial, fungal, parasitic and viral). In the case of SURPI, non-matching reads are further aligned against a protein databases to identify divergent organisms (Naccache *et al.*, 2014). VirFinder similarly subtracts host reads, assembles all remaining reads and then BLASTs the resulting contigs against viral versus non-viral databases to identify viral sequences, further BLASTing contigs that did not yield significant hits against NCBI's Conserved Domain Database (Lu *et al.*, 2020). VirusFinder, in contrast, after subtraction of host reads assembles only reads mapping to a database of viral sequences (Wang *et al.*, 2013).

Conventional viral detection tools, however, do not address the problem of gene redundancy, and assembly-based viral detection methods are also susceptible to fragmented assembly, which in turn leads to further redundancy. In contrast, MCRL provides a non-redundant list of viral gene families present in a given metagenome that is robust to allelic diversity even when contigs do not overlap. Furthermore, given the large sequence space that viruses occupy (Edwards and Rohwer, 2005; Hurwitz *et al.*, 2016; Paez-Espino *et al.*, 2016), viral genes will typically be highly divergent compared to reference sequences, and therefore methods that detect viral genes by requiring reads to align against reference sequences will likely have limited sensitivity. MCRL, on the other hand, can identify viral genes present in a given sample even if they are significantly divergent from the reference library since MCRL detection is based on protein homology at the contig level using contigs assembled *de novo* from unfiltered reads.

5.3 Metagenomic clustering

Both MCRL and metagenomic clustering programs aim to remove metagenomic redundancy by grouping interrelated contigs, although qualitatively these approaches are distinct. Contigs included in a metagenomic cluster obtained by programs such as CD-HIT are interrelated by virtue of having similar sequences. In MCRL, contigs included in a metagenomic signature are also interrelated, the difference being that they are not similar to each other, but are similar to the corresponding reference gene. By quantitatively comparing results obtained by both methods, we can highlight the features that are unique to MCRL. We therefore applied CD-HIT to the oral metagenome analyzed in Section 4 using different sequence identity thresholds. Table 2 shows that CD-HIT reported about one hundred times more clusters than MCRL, reducing the data content of the metagenome only marginally, with data reduction factors ranging from 89.2% when using a 90% identity threshold (the default in CD-HIT) to 43.7% when using a 30% sequence identity threshold compared to 1.1% for MCRL (Table 2). For the oral virome analyzed in Section 4, CD-HIT achieved at most an 11.2% reduction in sequence data, compared to 1.4% for MCRL. Clustering the metagenome with the reference library did not impact these results (Table 2).

The reason MCRL achieves significantly greater data reduction compared to CD-HIT is twofold: first, MCRL is limited by the diversity included in the reference library. The first filtering step performed by MCRL retains only reference genes yielding E values below E_{th} when aligned against the metagenome. This filtering step lead to a 14.7% and 19.9% data reduction in the oral metagenome and oral virome, respectively (calculated as the maximum number of potential reported reference genes, i.e. reference genes retained after E value filtering, divided by the total number of contigs). The remaining data reduction stems from clustering reference genes. CD-HIT clusters tend to be significantly smaller than MCRL signatures resulting in many more clusters being reported compared to MCRL (Supplementary Table S8). This occurs because CD-HIT clusters are comprised of only similar overlapping contigs (based on the global sequence identity threshold), whereas MCRL signatures encompass all contigs that are putative homologs of the given reference gene across the entire gene, including also non-overlapping contigs. MCRL therefore has the potential to achieve significantly greater data reduction because, under ideal conditions, one MCRL signature would correspond to all CD-HIT clusters emanating from a common genomic source.

Another difference between MCRL and CD-HIT is that CD-HIT partitions contigs into non-overlapping clusters, whereas MCRL signatures can partly overlap as defined by Equation (1) (Table 2). This type of 'soft' assignment of contigs, where a contig can belong to more than one signature, is required because different reference genes can share certain homologous domains. For example, in the case of the oral metagenome, MCRL reported two lysins, YP_009639755.1 and YP_009626536.1, corresponding to two different strains of a *Corynebacterium* phage (phi674 and Poushou, respectively). Both lysins share a peptidase domain in the N-terminal region of this gene, but otherwise did not share other domains (Supplementary Fig. S13). Consequently, the signatures of these lysins contained a certain fraction of overlapping contigs that mapped to the shared peptidase domain but were largely differentiated by MCRL due to contigs mapping to non-homologous regions of these two genes (Supplementary Fig. S13).

Correspondence between CD-HIT clusters and MCRL signatures

The relationship between reported reference genes and reference gene clusters determined by MCRL and CD-HIT clusters is illustrated in Figure 4a. A more detailed examination of the correspondence between CD-HIT clusters and MCRL signatures showed that when using the inclusive overlap condition there was a virtually unique mapping between representative contigs and CD-HIT clusters (i.e. no other representative contig mapped to the same CD-HIT cluster) across all sequence identity thresholds (Fig. 4b). This result suggests that a unique mapping exists between MCRL signatures and CD-HIT clusters. To understand more generally how MCRL signatures correspond to CD-HIT clusters, we mapped all contigs in the signature of each reported reference gene to CD-HIT clusters, and then calculated for each corresponding CD-HIT cluster the percent of contigs in that cluster that overlapped with the original MCRL signature, as depicted in Figure 4a. We found that each MCRL signature corresponded to multiple CD-HIT clusters (mean number ranging between 5 and 20 CD-HIT clusters, Supplementary Table S9), with the number of clusters depending on the sequence identity threshold used by CD-HIT. As the sequence identity threshold was reduced, MCRL signatures mapped to a smaller number of larger CD-HIT clusters, encompassing a greater degree of diversity (Supplementary Table S9). Across all sequence identity thresholds, the majority of contigs in CD-HIT clusters mapped back to the corresponding signature, with a mean overlap ranging from 72% to 98% (Fig. 4c). Moreover, the vast majority of contigs in a given CD-HIT cluster (>93%) did not map to signatures of other reference genes (Fig. 4c) suggesting that the correspondence between MCRL reference gene clusters and matching CD-HIT clusters was largely a one-to-many relationship, as illustrated in Figure 4a.

Furthermore, at high sequence identity thresholds, MCRL signature sizes were nearly identical to the combined size of all corresponding CD-HIT clusters (Supplementary Fig. S14), indicating very little overhead. At low sequence identity thresholds, however, the relative size of corresponding CD-HIT clusters was larger (Supplementary Fig. S14). This difference occurs because at low sequence identity thresholds CD-HIT clusters encompass more sequence diversity, including contigs that are not homologous to the reference gene (e.g. contigs mapping to a region of the source gene that is not homologous to the reference gene), and therefore these spurious contigs are not included in MCRL signatures.

Finally, we found that by-and-large, long reference genes (e.g. >1000 amino acids) corresponded to more CD-HIT clusters compared to short reference genes (e.g. <200 amino acids, Supplementary Fig. S15). This result is expected given that the median contig length in these metagenomes was 130 amino acids when translated. For example, the signature of the reference gene YP_009626515.1 reported by MCRL for the oral virome, which encodes a TerL gene spanning 545 amino acids, was spread across five CD-HIT clusters mapping to different regions of this gene (Supplementary Fig. S16).

6 Mapping viral genes families in an oral virome

To illustrate a real-world application of MCRL, we demonstrate how MCRL can be used to obtain a non-redundant list of candidates belonging to a specific viral gene family. As a viral gene, we chose the highly conserved TerL gene (Casjens, 2003, 2005; Rao and Feiss, 2008), which can be used as a specific marker for phages and prophages (Casjens, 2003). To obtain a non-redundant list of TerL genes present in a given metagenome, we first used MCRL in conjunction with the viral RefSeq library to obtain a non-redundant list of viral genes present in the metagenome. To obtain a non-redundant list of TerL genes, we screened the RefSeq-derived annotation provided by MCRL (including both FASTA and GenPept annotation) for genes annotated as TerL genes. Table 3 shows the resulting

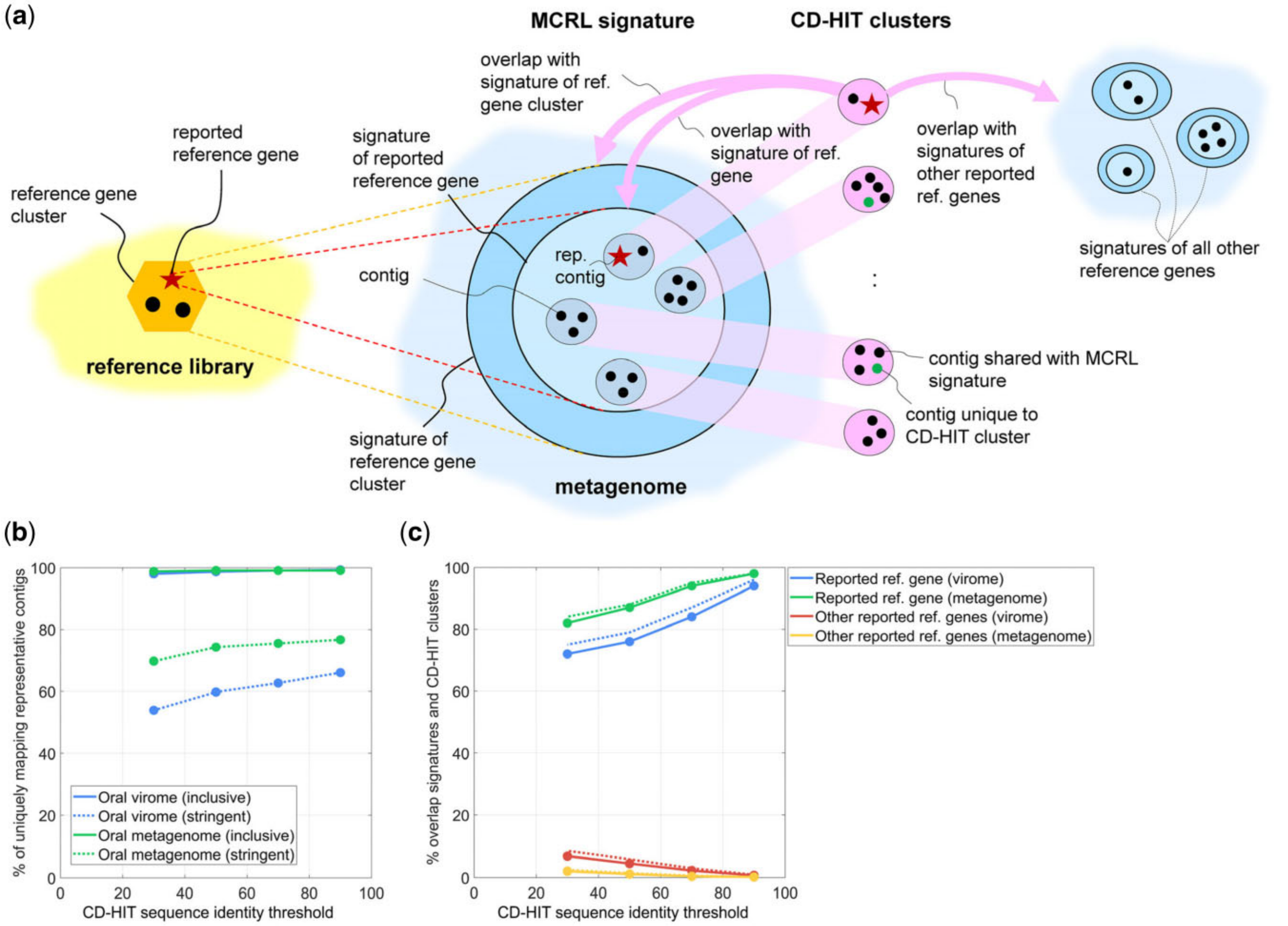

Fig. 4. Correspondence between MCRL and CD-HIT clusters. (**a**) For each reference gene reported by MCRL (red star in reference gene cluster) a mapping can be established between all contigs belonging to the signature of the reported reference gene and corresponding CD-HIT clusters. Black dots in CD-HIT clusters represent shared contigs, and green dots represent contigs that were unique to CD-HIT clusters. To establish the correspondence between MCRL and CD-HIT clusters, we determined the overlap between the signature of each reported reference gene (or reference gene cluster) and the corresponding CD-HIT clusters (arrows pointing left). Likewise, we determined the overlap between the corresponding CD-HIT clusters and signatures of other reported reference genes (arrow pointing right). (**b**) Percent of reported reference genes for which the representative contig uniquely mapped to a CD-HIT cluster, i.e. no other representative contig mapped to the same CD-HIT cluster. Results are shown for different CD-HIT sequence identity thresholds. (**c**) Mean overlap between CD-HIT clusters corresponding to a given reported reference gene and the signature of that reported reference gene (blue—oral virome, green—oral metagenome). Also shown is the mean overlap between CD-HIT clusters corresponding to a given reported reference gene and signatures of other reported reference genes (red—oral virome, orange—oral metagenome). Dotted lines correspond to signatures of reference gene clusters. Including the viral reference library in the CD-HIT clustering process did not impact results

22 candidate TerL gene families identified by MCRL to be present in the oral virome out of 903 reported (viral) reference genes.

To see how closely related these candidate TerL gene families are to reference genes in the viral RefSeq database, we plotted the reference gene networks corresponding to these reported reference genes as heat maps, with nodes color-coded according to the minimal *E* value in their signatures (Fig. 5). This analysis showed that even when networks contained many nodes, typically there were few, if any, nodes competing with the epicenter in terms of homology, suggesting that the viral RefSeq database may be undersampling TerL diversity. Conversely, only 9% of reported reference genes encoding TerL genes yielded alignments with <40% identity compared to 36% when considering all reported reference genes for the viral RefSeq reference library, reflecting the conserved nature of TerL genes compared to other viral genes (Casjens, 2003).

Identifying these 22 candidate TerL gene families using standard annotation or by BLASTing contigs against the NCBI-nr database would be challenging because these 22 candidate TerL gene families corresponded to 1550 contigs (given by the union of all signatures of these 22 TerL reference genes), and to 2455 contigs when expanding signatures to include all genes in all TerL reference gene clusters. Metagenomic clustering would not solve this problem because a single TerL gene family can be spread across multiple CD-HIT clusters as discussed above (e.g. the TerL gene shown in Supplementary Fig. S16). To demonstrate this point, when using a 90% sequence identity threshold the signatures of these 22 TerL reference genes were spread across 833 CD-HIT clusters, with 92% of contigs in these clusters mapping back to the signatures of the TerL reference gene clusters. Even when reducing the CD-HIT sequence identity threshold to 30%, these 22 TerL reference genes were still spread across 161 CD-HIT clusters, with only 73% of contigs in these clusters mapping back to the signatures of TerL reference gene clusters, suggesting presence of spurious contigs. Including the viral reference library in the CD-HIT clustering process had no significant impact on the number of corresponding CD-HIT clusters. Moreover, even if annotation would be applied to the ~15 000 CD-HIT clusters found using a 30% sequence identity threshold, identifying these 161 TerL-related clusters and sorting them into 22 families would be challenging because annotation of contigs emanating from the same genomic source can be different.

To demonstrate the application of MCRL on another viral gene family, we next show how MCRL can be used to obtain a non-redundant list of major capsid protein genes in the oral virome. Screening once again the RefSeq-derived annotation provided by MCRL, we were able to identify 21 reported reference genes encoding major capsid proteins, which spanned in total nine different

Table 3. TerL-related reference genes reported by MCRL for the oral virome using an inclusive overlap condition

GenPept source	Viral classification	Signature size	RefSeq gene	*E* value	% Identity (aa)	TerL domain
Geobacillus virus E3	Caudovirales; Siphoviridae	540	YP_009223720.1	2.0E-43	46	n/a
Clostridium phage phiCD111	Caudovirales; Siphoviridae	169	YP_009208355.1	2.0E-75	79	Terminase_3
Streptococcus virus ALQ132	Caudovirales; Siphoviridae	165	YP_003344848.1	2.0E-75	79	Terminase_3
Streptococcus phage SM1	Caudovirales; Siphoviridae	150	NP_862877.1	3.0E-91	98	Terminase_1
Streptococcus phage phiARI0460-1	Caudovirales; Siphoviridae	128	YP_009321976.1	5.0E-93	85	Terminase_3
Methanobacterium phage psiM2	Caudovirales; Siphoviridae	114	NP_046964.1	3.0E-34	47	Terminase_6C
Enterobacter phage Tyrion	Caudovirales; Podoviridae	90	YP_009287734.1	1.0E-58	57	n/a
Streptomyces phage Scap1	Caudovirales; Siphoviridae	56	YP_009615353.1	5.0E-74	73	Terminase_1
Clostridium phage phiCDHM19	Caudovirales; Myoviridae	43	YP_009216852.1	2.0E-66	72	Terminase_6C
Rhodovulum phage vB_RhkS_P1	Caudovirales; Siphoviridae	39	YP_009285918.1	3.0E-67	75	COG4373
Pelagibacter phage HTVC010P	Caudovirales; Podoviridae	38	YP_007517700.1	2.0E-35	50	17 super family
Bacillus phage BtCS33	Caudovirales; Siphoviridae	28	YP_006488672.1	3.0E-50	64	COG4626
Corynebacterium phage Poushou	Caudovirales; Siphoviridae	19	YP_009626515.1	7.0E-62	65	Terminase_1
Listeria phage LP-101	Caudovirales; Siphoviridae	10	YP_009044803.1	5.0E-48	55	COG4626
Delftia phage IME-DE1	Caudovirales; Podoviridae	9	YP_009191792.1	7.0E-63	70	Terminase_6
Haemophilus phage SuMu	Caudovirales; Myoviridae	9	YP_007002934.1	1.0E-57	72	Terminase_6C
Bacteriophage Lily	Caudovirales; Siphoviridae	5	YP_009202208.1	1.0E-30	42	Terminase_GpA
Arthrobacter phage Decurro	Caudovirales; Siphoviridae	4	YP_009191297.1	3.0E-32	47	Terminase_1
Burkholderia phage Bcep176	Caudovirales; Siphoviridae	3	YP_355415.1	6.0E-51	62	COG4626
Polaribacter phage P12002S	Caudovirales; Siphoviridae	2	YP_009195687.1	2.0E-19	35	Terminase_3
Bacillus phage AR9	Caudovirales; Myoviridae	2	YP_009283025.1	2.0E-26	38	17 super family
Enterococcus phage phiFL4A	Caudovirales; Siphoviridae	1	YP_003347385.1	1.0E-14	53	n/a

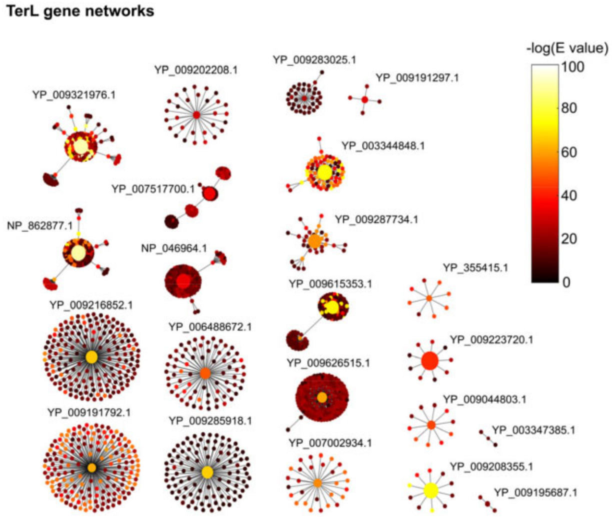

Fig. 5. Heat map of reference gene networks corresponding to TerL genes in the oral virome. Each node is color-coded according to the minimal *E* value yielded by the given reference gene, shown in logarithmic scale. Nodes corresponding to reported reference genes (yielding the minimal overall *E* value) are drawn proportional to the logarithm of the signature size

capsid domains and four unknown domains (Supplementary Table S10). The reference gene networks corresponding to these 21 reported major capsid proteins tended to have just one branch reflecting the general lack of homology between capsid genes (Supplementary Fig. S17). The fact that the number of non-redundant putative gene families detected by MCRL for the TerL gene and the major capsid protein gene were similar (21 versus 22, respectively) potentially hints at the underlying degree of diversity present in the given oral virome.

7 Discussion

We presented a novel strategy for data mining metagenomes for diversity represented in a pre-defined reference library with the goal of quantizing this diversity into bins based on known diversity. According to our proposed approach, a user provided reference library is compressed with respect to a given metagenome by determining for each reference gene all related reference genes, selecting the reference gene yielding the minimal *E* value to represent that group, and repeating this process until no two remaining reference genes are related. This 'compression' procedure is lossy in the sense that diversity that is not spanned by the reference library is lost (by design). However, diversity that is spanned by the reference library preserves information in the sense that each discarded reference gene is represented by a *related* reference gene yielding a lower *E* value. The resulting list of non-redundant reference genes reported by MCRL should therefore capture the diversity retained after initial filtering.

Candidate gene families

The non-redundant list of reference genes reported by MCRL can be regarded as a potential or tentative set of gene families present in the metagenome. We showed that the list of reference genes representing these 'candidate gene families' was relatively robust to input parameters affecting clustering (E_0 and T) and was also robust to stochastic perturbations applied to the metagenome. In terms of uniqueness of results, the list of reference genes reported by MCRL does not depend on the order of genes in the reference library or the order of contigs in the metagenome. In contrast, for some metagenomic clustering methods such USEARCH, the order of records in the metagenome influences the outcome (Mahé *et al.*, 2014, https://drive5.com/usearch/manual/uclust_algo.html).

Signature overlap versus homology

The criterion for determining redundancy [Equation (1)] cannot be substituted by a homology metric between reference genes because in practice, reference genes corresponding to the same set of contigs can be distant homologs. Therefore, there is no clear cutoff for determining redundancy within the reference library. To illustrate this point, Supplementary Figure S18 shows an example of a reference gene cluster corresponding to portal protein gp29 of Mannheimia phage vB_MhM_3927AP2 determined for the oral virome discussed in Section 4 (*E* value 10^{-70}). Despite the fact that all reference genes in this cluster have a high signature overlap with gp29 (median 76.5%), most portal proteins comprising this reference gene cluster yielded low percent identifies when aligned at the amino acid level against gp29 (median 40% identity at the amino acid level). Thus, reference genes corresponding to the same diversity in the metagenome can be remote homologs of each other. The decision whether reference genes that are distant homologs of each other are redundant or not depends, in the end, on the diversity present in the metagenome.

Furthermore, not all reference genes that are homologous are redundant because homologous reference genes can display significant homology to different swaths of diversity within the metagenome despite being homologous to each other. This can occur, e.g. when two reference genes have both shared and unique domains. In this case, despite being homologous to each other, both reference genes can display homology to different sets of contigs. Such an example was provided in Supplementary Figure S13: the lysins of Corynebacterium phage phi674 and Corynebacterium phage Poushou are homologous due to a shared peptidase domain, yielding 52% identify at the amino acid level (E value $= 2{\cdot}10^{-35}$). Yet, these reference genes were reported separately by MCRL because each of these reference genes contained certain unique domains (a lysozyme-like domain in the case of phi674 compared to a mutamidase domain in the case of Poushou) that were homologous to different sets of contigs indicating that both lysin families may be present in the metagenome. A redundancy metric based on homology of reference genes would have led both lysins to be grouped together and hence loss of information. Thus, the domains or motifs contributing to homology can change based on the specific diversity present in the metagenome, and the decision whether two homologous reference genes are redundant or not depends on this diversity. The redundancy of reference genes is therefore not an intrinsic property of reference genes that can be determined purely based on homology considerations post-filtering, but needs to be dynamically determined with respect to the diversity present in the metagenome.

Circumventing the need to define OTUs

The list of reference genes reported by MCRL was obtained without having to invoke an arbitrary global clustering threshold, which metagenomic clustering methods often require in order to define OTUs (Navas-Molina *et al.*, 2013). A global clustering threshold, such as the sequence identity threshold used by CD-HIT, is problematic because different gene families can require different thresholds, and therefore need to be dynamically determined. In MCRL, these challenges are circumvented because candidate gene families naturally emerge from the metagenome by the algorithm, being empirically defined for a given reference library and metagenome. In this respect, MCRL circumvents the need to define artificial units of diversity, such as OTUs, with diversity instead being empirically quantized using the reference library as a specific prism.

More advanced metagenomic clustering methods, such as Swarm, that dynamically breaks OTUs into sub-OTUs (Kopylova *et al.*, 2016; Mahé *et al.*, 2015) have addressed the problem of a global clustering threshold and sequence dependency. However, all metagenomic clustering methods invariably depend on pairwise alignment of sequences to create OTUs. In MCRL, contigs are not required to overlap to be included in signatures, enabling MCRL to draw on more information encoded in the metagenome when computing candidate gene families. Due to this fact, MCRL can achieve greater data reduction for metagenomes compared to metagenomic clustering methods. Since the objective of MCRL and metagenomic clustering methods is different, these methods should not be viewed as competing but complementary.

Limitations of MCRL

MCRL has certain intrinsic limitations that cannot be circumvented. For example, MCRL is limited by the diversity included in the reference library. MCRL is also potentially susceptible to mixed signals originating from different genomics sources, described in greater detail in Supplementary Discussion S1.3. Since MCRL is applied to assembled metagenomes, signatures can depend on the details of the metagenome assembly tools and the specific parameters that were used to assemble the metagenome. MCRL predictions can also depend on the algorithm and settings used to generate the local alignments. However, we have shown that MCRL behaves in a predictable and controlled manner when stochastic perturbations are introduce to signatures and metagenomes. For example, we showed that when 20% of contigs were randomly perturbed (either duplicated or discarded), MCRL was still capable of detecting ~90% of the original reported reference genes, with the impact of perturbation scaling proportionally to the degree of perturbation (Supplementary Fig. S8c). Furthermore, when switching from BLAST to DIAMOND, MCRL sensitivity was still high (Supplementary Figs S18 and S19), and it retained 78% of reported reference genes when using the inclusive overlap condition. We also showed that MCRL is robust to chimeras, with the major parent always detected (Supplementary Fig. S9b).

In terms of performance, we showed that the sensitivity of MCRL to detect a given genomic source can diminishes quite rapidly when the genomic source diverges too far from diversity spanned by the reference library, reflecting a practical limit on the extent of novel diversity that can be detected by MCRL. However, metagenomic signatures can be constructed using any algorithm for determining homology, including more sensitive methods, such as PSI-BLAST (Altschul *et al.*, 1997) and HMMER (Eddy, 2009) that could improve sensitivity to detect more remote homologs. Furthermore, the sensitivity of MCRL to detect novel gene families is expected to continuously improve as reference databases continue to expand at a rapid pace.

8 Conclusions

We presented a novel data mining method for probing a metagenome for homologs of a pre-defined set of reference sequences. By performing iterative clustering on metagenomic signatures of reference genes, MCRL provides as output a non-redundant list of reference genes that share the property that their signatures do not overlap [as defined by Equation (1)]. Using a series of *in silico* spike-in experiments, we showed that MCRL was able, across a wide range of simulated mutation rates, to accurately discriminate different viral gene families, identify close homologs of the templates used for spike-in experiments and effectively group closely related variants. CD-HIT, on the other hand, exhibited between 21% to 35% less sensitivity compared to MCRL, was less effective at identifying the templates used for spike-in experiments and was less effective at clustering closely related variants. We further showed that MCRL reference gene clusters and CD-HIT clusters have a one-to-many correspondence, with CD-HIT clusters becoming fewer and larger as the sequence identity threshold is reduced. Nevertheless, we showed that CD-HIT clusters do not converge to signatures as the CD-HIT sequence identity threshold is reduced because reducing the sequence identity threshold generally results in inclusion of non-specific homologs. In addition, contigs originating from long genes will generally be fragmented across multiple CD-HIT clusters.

Data reduction

In the case of the viral gene set, we showed that MCRL achieved a data reduction factor of ~1% compared to ~10–90% for CD-HIT (depending on the sequence identity threshold). The data reduction factors achieved by CD-HIT and MCRL are, however, qualitatively different in terms of their end goal, the type of redundancy being removed and the method by which clustering is performed. Metagenomic clustering programs aim to reduce redundancy inherent to the metagenome by clustering similar metagenomic sequences. MCRL, in contrast, clusters similar metagenomic signatures with the goal of removing redundant sequences from the reference library. MCRL hence does not attempt to preserve genomic diversity like standard metagenomic clustering methods, but to capture diversity reflected in the reference library and report this diversity with minimal redundancy. In this sense, MCRL is similar to closed-reference clustering methods, but whereas closed-reference clustering methods achieve further data reduction by discarding contigs that do not align with reference sequences, MCRL does the reverse and discards reference sequences that do not align with contigs.

The clustering that MCRL performs to remove redundancy from reference genes is effective because metagenomic signatures encompass all metagenomic sequences homologous to a given reference gene. In contrast, metagenomic clustering methods require metagenomic sequences to have a certain degree of overlap to define a cluster, e.g. by requiring overlap with the representative of the cluster

(Li *et al.*, 2001). Therefore, contigs emanating from a given genomic source can be scattered across multiple clusters. Moreover, standard metagenomic clustering methods use an artificial sequence similarity cutoff to artificially limit diversity included in a given cluster, further limiting their ability to collect all redundant sequences. In this sense, MCRL uses the full-length reference genes as pseudo scaffolds for alignment and clustering. Depending on the extent of homology between a common genomic source in the sample and the closest homolog in the reference library, MCRL can potentially (ideally) group all contigs emanating from a common genomic source.

Another important difference between MCRL and metagenomic clustering is that the data reduction achieved by programs such as CD-HIT is simply correlated with the global sequence identity threshold used to define OTUs. MCRL, however, does not have an equivalent global threshold parameter and the data reduction factor achieved by MCRL reflects an intrinsic redundancy in the reference library with respect to the given metagenome: each reference gene that was not reported by MCRL either did not meet the minimal homology threshold for detection set by the user (E_{th}), or was represented instead by another reference gene that had an overlapping signature and yielded a lower E value. It is in this sense that MCRL's compression preserves information: any reference gene that was not reported by MCRL either did not have significant homologs in the metagenome, or the homology was better captured by another reference gene that was reported by MCRL.

Using MCRL to explore viral diversity

One problem that MCRL is particularly suited for is mapping viral diversity in metagenomes because viral sequences tend to be both divergent and redundant. By selecting as a reference library the viral RefSeq database, we demonstrated how MCRL can be used to provide a non-redundant list of putative viral gene families in a metagenome. This list, in turn, can serve as a starting point for further investigation. For example, we previously used an earlier version of the MCRL algorithm to screen a metagenome from a hindgut of a higher termite for different TerL gene families (Tadmor *et al.*, 2011). This analysis enabled us to identify a highly conserved family of TerL genes that was ubiquitous across different termite species and could therefore serve as a universal marker for phages in this environment (Tadmor *et al.*, 2011). More recently, following a similar approach, we used MCRL to identify a group of highly conserved TerL gene families ubiquitous in the human population (A. D. Tadmor *et al.*, Ubiquitous Phage Markers in Humans, 2021, *In prep.*). In Section 6, we showed that extracting such a list of TerL gene families using a program like CD-HIT would be impractical owing to the large number of clusters involved.

Examples of other potential gene sets

Although we focused our attention on analysis of viral gene sets, MCRL can be used to probe metagenomes using any reference library provided by the user. By constructing different reference libraries MCRL can be used to explore different hypotheses about a given environment. Reference libraries can span, e.g. a certain taxonomical group of organisms, or be more focused, targeting a certain group of genes. One example would be antibiotic resistance genes. Current methods for identifying antibiotic resistance genes range from basic approaches using BLAST alignment, such as ARGs-OAP (Yang *et al.*, 2016), to more sophisticated tools using, for example, targeted assembly, such as ARIBA (Hunt *et al.*, 2017), AmrPlusPlus (Lakin *et al.*, 2017) and fARGene (Berglund *et al.*, 2019), tools using hidden Markov models or deep learning to increase sensitivity, such as Resfams (Gibson *et al.*, 2015) and DeepARG (Arango-Argoty *et al.*, 2018), and tools using machine learning to increase specificity, such as PCM (Ruppé *et al.*, 2018). These tools, however, will generally output all detected/assembled genes, including all variants of a given gene, even if these variants belong to the same gene family. Furthermore, since antibiotic resistance genes tend to be long (944 ± 390 nt, $n = 2631$, based on the CARD database), results may be fragment if contigs in a given metagenome tend to be short. MCRL could contribute by minimizing redundancy and grouping detected genes into distinct gene families, enabling to detect both known and novel antibiotic resistance genes in human and environmental samples, including remote homologs. For this purpose, a reference library could be constructed to span all known antimicrobial resistance genes based on databases, such as CARD (McArthur *et al.*, 2013), ResFinder (Zankari *et al.*, 2012 and MEGARes (Lakin *et al.*, 2017). In the same manner, MCRL could be used to map families of antibiotic genes with similar advantages.

Another potential application for MCRL is to identify gene families associated with virulence and pathogenicity using databases, such as the virulence factor database (Liu *et al.*, 2018), Victors—a database of virulence factors in human and animal pathogens (Sayers *et al.*, 2019), the Pathogenicity Island Database (Yoon *et al.*, 2014), the National Microbial Pathogen Database Resource (McNeil *et al.*, 2006) and the GeneDB database for pathogens (Logan-Klumpler *et al.*, 2011). Alternatively, MCRL could be used to identify genes associated with viral pathogens based on reference databases such ViPR (Pickett *et al.*, 2012) and IRD (Zhang *et al.*, 2017). Here too, the advantage of MCRL would be, on the one hand, reduction of redundancy and organization of detected genes into putative gene families, and on the other hand improved sensitivity to capture more distant homologs.

Reference libraries can also be constructed to target enzymes performing specific functions, such as DNA replication, reverse transcription, or proteins that are part of certain metabolic pathways. For example, reference libraries can be constructed to span all genes involved in carbon, nitrogen and methane cycling (Mackelprang *et al.*, 2011; Zhang *et al.*, 2014), cellulose degradation (Berlemont and Martiny, 2013; Pereyra *et al.*, 2010; Warnecke *et al.*, 2007) or any other phylogenetically diverse biological pathway of interest. Given the large number of annotated genes available today, with over 100 million sequences in the RefSeq database alone, high-resolution reference libraries focusing on specific swaths of genetic diversity can easily be constructed addressing nearly every facet of biology.

Acknowledgment

We wish to thank our reviewers for their thoughtful comments.

Funding

This work was supported by the National Health Institute Director's Pioneer Award and the National Health Institute's Eureka [R01-GM098465].

Conflict of Interest: none declared.

References

Abubucker,S. *et al.* (2012) Metabolic reconstruction for metagenomic data and its application to the human microbiome. *PLoS comput. Biol.*, **8**, e1002358.

Albanese,D. *et al.* (2015) MICCA: a complete and accurate software for taxonomic profiling of metagenomic data. *Sci. Rep.*, **5**, 9743.

Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Arango-Argoty,G. *et al.* (2018) DeepARG: a deep learning approach for predicting antibiotic resistance genes from metagenomic data. *Microbiome*, **6**, 1–15.

Arnold,K. *et al.* (2006) The SWISS-MODEL workspace: a web-based environment for protein structure homology modelling. *Bioinformatics*, **22**, 195–201.

Bateman,A. *et al.* (2004) The Pfam protein families database. *Nucleic Acids Res.*, **32** (Suppl. 1), D138–D141.

Belda-Ferre,P. *et al.* (2012) The oral metagenome in health and disease. *ISME J.*, **6**, 46–56.

Bench,S.R. *et al.* (2007) Metagenomic characterization of Chesapeake Bay virioplankton. *Appl. Environ. Microbiol.*, **73**, 7629–7641.

Berglund,F. *et al.* (2019) Identification and reconstruction of novel antibiotic resistance genes from metagenomes. *Microbiome*, **7**, 52.

Berlemont,R. and Martiny,A.C. (2013) Phylogenetic distribution of potential cellulases in bacteria. *Appl. Environ. Microbiol.*, **79**, 1545–1554.

Bradley,R.D. and Hillis,D.M. (1997) Recombinant DNA sequences generated by PCR amplification. *Mol. Biol. Evol.*, **14**, 592–593.

Brettin,T. *et al.* (2015) RASTtk: a modular and extensible implementation of the RAST algorithm for building custom annotation pipelines and annotating batches of genomes. *Sci. Rep.*, **5**, 8365.

Buchfink,B. *et al.* (2015) Fast and sensitive protein alignment using DIAMOND. *Nat. Methods*, **12**, 59.

Camacho,C. *et al.* (2009) BLAST+: architecture and applications. *BMC Bioinformatics*, **10**, 421.

Casjens,S. (2003) Prophages and bacterial genomics: what have we learned so far? *Mol. Microbiol.*, **49**, 277–300.

Casjens,S. *et al.* (2005) The generalized transducing *Salmonella* bacteriophage ES18: complete genome sequence and DNA packaging strategy. *J. Bacteriol.*, **187**, 1091.

Eddy,S.R. (2004) What is a hidden Markov model? *Nat. Biotechnol.*, **22**, 1315–1316.

Eddy,S.R. (2009) A new generation of homology search tools based on probabilistic inference. *Genome Inform.*, **23**, 205–211.

Edgar,R.C. (2010) Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, **26**, 2460–2461.

Edwards,R.A. *et al.* (2012) Real time metagenomics: using k-mers to annotate metagenomes. *Bioinformatics*, **28**, 3316–3317.

Edwards,R.A. and Rohwer,F. (2005) Viral metagenomics. *Nat. Rev. Microbiol.*, **3**, 504–510.

Fu,L. *et al.* (2012) CD-HIT: accelerated for clustering the next-generation sequencing data. *Bioinformatics*, **28**, 3150–3152.

Ghodsi,M. *et al.* (2011) DNACLUST: accurate and efficient clustering of phylogenetic marker genes. *BMC Bioinformatics*, **12**, 271.

Gibson,M.K. (2015) Improved annotation of antibiotic resistance determinants reveals microbial resistomes cluster by ecology. *ISME J.*, **9**, 207–216.

Glass,E.M. *et al.* (2010) Using the metagenomics RAST server (MG-RAST) for analyzing shotgun metagenomes. *Cold Spring Harb. Protoc.*, **2010**, prot5368.

Haft,D.H. *et al.* (2003) The TIGRFAMs database of protein families. *Nucleic Acids Res.*, **31**, 371–373.

Hendrix,R. (2003) Bacteriophage genomics. *Curr. Opin. Microbiol.*, **6**, 506–511.

Henikoff,S. and Henikoff,J.G. (1992) Amino acid substitution matrices from protein blocks. *Proc. Natl. Acad. Sci. USA*, **89**, 10915–10919.

Ho,T. and Tzanetakis,I.E. (2014) Development of a virus detection and discovery pipeline using next generation sequencing. *Virology*, **471**, 54–60.

Holm,L. and Sander,C. (1998) Removing near-neighbour redundancy from large protein sequence collections. *Bioinformatics*, **14**, 423–429.

Hunt,M. *et al.* (2017) ARIBA: rapid antimicrobial resistance genotyping directly from sequencing reads. *Microb. Genom.*, **3**, e000131.

Huntemann,M. *et al.* (2016) The standard operating procedure of the DOE-JGI Metagenome Annotation Pipeline (MAP v. 4). *Stand. Genomic Sci.*, **11**, 17.

Hunter,S. *et al.* (2008) InterPro: the integrative protein signature database. *Nucleic Acids Res.*, **37** (**Suppl. 1**), D211–D215.

Hurwitz,B.L. *et al.* (2016) Computational prospecting the great viral unknown. *FEMS Microbiol. Lett.*, **363**, fnw077.

Huson,D.H. *et al.* (2016) MEGAN community edition-interactive exploration and analysis of large-scale microbiome sequencing data. *PLoS Comput. Biol.*, **12**, e1004957.

Huson,D.H. *et al.* (2017) Fast and simple protein-alignment-guided assembly of orthologous gene families from microbiome sequencing reads. *Microbiome*, **5**, 11.

Kanehisa,M. and Goto,S. (2000) KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Res.*, **28**, 27–30.

Kopylova,E. *et al.* (2016) Open-source sequence clustering methods improve the state of the art. *MSystems*, **1**, e00003–e00015.

Krupovic,M. and Koonin,E.V. (2017) Multiple origins of viral capsid proteins from cellular ancestors. *Proc. Natl. Acad. Sci. USA*, **114**, E2401–E2410.

Lakin,S.M. *et al.* (2017) MEGARes: an antimicrobial resistance database for high throughput sequencing. *Nucleic Acids Res.*, **45**, D574–D580.

Li,J. *et al.* (2014) An integrated catalog of reference genes in the human gut microbiome. *Nat. Biotechnol.*, **32**, 834.

Li,W. *et al.* (2001) Clustering of highly homologous sequences to reduce the size of large protein databases. *Bioinformatics*, **17**, 282–283.

Li,W. *et al.* (2012) Ultrafast clustering algorithms for metagenomic sequence analysis. *Brief. Bioinform.*, **13**, 656–668.

Li,Y. *et al.* (2016) VIP: an integrated pipeline for metagenomics of virus identification and discovery. *Sci. Rep.*, **6**, 1–10.

Li,W. and Godzik,A. (2006) Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, **22**, 1658–1659.

Liu,B. *et al.* (2018) VFDB 2019: a comparative pathogenomic platform with an interactive web interface. *Nucleic Acids Res.*, **47**, D687–D692.

Logan-Klumpler,F.J. *et al.* (2011) GeneDB—an annotation database for pathogens. *Nucleic Acids Res.*, **40**, D98–D108.

Lu,S. *et al.* (2020) CDD/SPARCLE: the conserved domain database in 2020. *Nucleic Acids Res.*, **48**, D265–D268.

Mackelprang,R. *et al.* (2011) Metagenomic analysis of a permafrost microbial community reveals a rapid response to thaw. *Nature*, **480**, 368–371.

Mahé,F. *et al.* (2014) Swarm: robust and fast clustering method for amplicon-based studies. *PeerJ*, **2**, e593.

Mahé,F. *et al.* (2015) Swarm v2: highly-scalable and high-resolution amplicon clustering. *PeerJ*, **3**, e1420.

Mavromatis,K. *et al.* (2009) The DOE-JGI Standard operating procedure for the annotations of microbial genomes. *Stand. Genomic Sci.*, **1**, 63.

McArthur,A.G. *et al.* (2013) The comprehensive antibiotic resistance database. *Antimicrob. Agents Chemother.*, **57**, 3348–3357.

McNeil,L.K. *et al.* (2006) The National Microbial Pathogen Database Resource (NMPDR): a genomics platform based on subsystem annotation. *Nucleic Acids Res.*, **35** (**Suppl. 1**), D347–D353.

Mercier,C. *et al.* (2013) SUMATRA and SUMACLUST: fast and exact comparison and clustering of sequences. In: *Programs and Abstracts of the SeqBio 2013 workshop (25-26th Nov)*, pp. 27–29.

Methé,B.A. *et al.* (2012) A framework for human microbiome research. *Nature*, **486**, 215–221.

Meyer, F. *et al.* (2008) The metagenomics RAST server–a public resource for the automatic phylogenetic and functional analysis of metagenomes. *BMC Bioinformatics*, **9**, 386.

Meyer,F. *et al.* (2009) FIGfams: yet another set of protein families. *Nucleic Acids Res.*, **37**, 6643–6654.

Naccache,S.N. *et al.* (2014) A cloud-compatible bioinformatics pipeline for ultrarapid pathogen identification from next-generation sequencing of clinical samples. *Genome Res.*, **24**, 1180–1192.

Navas-Molina,J.A. *et al.* (2013) Advancing our understanding of the human microbiome using QIIME. *Methods Enzymol.*, **531**, 371–444.

Oulas,A. *et al.* (2015) Metagenomics: tools and insights for analyzing next-generation sequencing data derived from biodiversity studies. *Bioinform. Biol. Insights*, **9**, 75–88.

Paez-Espino,D. *et al.* (2016) Uncovering Earth's virome. *Nature*, **536**, 425.

Pereyra,L. *et al.* (2010) Detection and quantification of functional genes of cellulose-degrading, fermentative, and sulfate-reducing bacteria and methanogenic archaea. *Appl. Environ. Microbiol.*, **76**, 2192–2202.

Pickett,B.E. *et al.* (2012) ViPR: an open bioinformatics database and analysis resource for virology research. *Nucleic Acids Res.*, **40**, D593–D598.

Pride,D.T. *et al.* (2012) Evidence of a robust resident bacteriophage population revealed through analysis of the human salivary virome. *ISME J.*, **6**, 915–926.

Pruitt,K.D. *et al.* (2007) NCBI reference sequences (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Res.*, **35** (**Suppl. 1**), D61–D65.

Rao,V.B. and Feiss,M. (2008) The bacteriophage DNA packaging motor. *Annu. Rev. Genet.*, **42**, 647–681.

Ruppé,E. *et al.* (2018) Prediction of the intestinal resistome by a three-dimensional structure-based method. *Nat. Microbiol.*, **4**, 112–123.

Sayers,E.W. *et al.* (2020) Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res.*, **48**, D1–D9.

Sayers,S. *et al.* (2019)Victors: a web-based knowledge base of virulence factors in human and animal pathogens. *Nucleic acids Res.*, **47**, D693–D700.

Schloss,P.D. *et al.* (2009) Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. *Appl. Environ. Microbiol.*, **75**, 7537–7541.

Sun,S. *et al.* (2010) Community cyberinfrastructure for advanced microbial ecology research and analysis: the CAMERA resource. *Nucleic Acids Res.*, **39** (**Suppl. 1**), D546–D551.

Suzek,B.E. *et al.* (2007) UniRef: comprehensive and non-redundant UniProt reference clusters. *Bioinformatics*, **23**, 1282–1288.

Tadmor,A.D. *et al.* (2011) Probing individual environmental bacteria for viruses by using microfluidic digital PCR. *Science*, **333**, 58–62.

Tatusov,R.L. *et al.* (2000) The COG database: a tool for genome-scale analysis of protein functions and evolution. *Nucleic Acids Res.*, **8**, 33–36.

Wang,Q. *et al.* (2013) VirusFinder: software for efficient and accurate detection of viruses and their integration sites in host genomes through next generation sequencing data. *PLoS One*, **8**, e64465.

Wang,Q. *et al.* (2015) Xander: employing a novel method for efficient gene-targeted metagenomic assembly. *Microbiome*, **3**, 32.

Warnecke,F. *et al.* (2007) Metagenomic and functional analysis of hindgut microbiota of a wood-feeding higher termite. *Nature*, **450**, 560–565.

Xie,G. *et al.* (2010) Community and gene composition of a human dental plaque microbiota obtained by metagenomic sequencing. *Mol. Microbiol.*, **25**, 391–405.

Yang,Y. *et al.* (2016) ARGs-OAP: online analysis pipeline for antibiotic resistance genes detection from metagenomic data using an integrated structured ARG-database. *Bioinformatics*, **32**, 2346–2351.

Yoon,S.H. *et al.* (2014) PAIDB v2. 0: exploration and analysis of pathogenicity and resistance islands. *Nucleic Acids Res.*, **43**, D624–D630.

Zankari,E. *et al.* (2012) Identification of acquired antimicrobial resistance genes. *J. Antimicrob. Chemother.*, **67**, 2640–2644.

Zhang,Y. *et al.* (2014) A scalable and accurate targeted gene assembly tool (SAT-Assembler) for next-generation sequencing data. *PLoS Comput. Biol.*, **10**, e1003737.

Zhang,Y. *et al.* (2017) Influenza Research Database: an integrated bioinformatics resource for influenza virus research. *Nucleic Acids Res.*, **45**, D466–D474.