Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

Genome analysis

metaviralSPAdes: assembly of viruses from metagenomic data

Dmitry Antipov [1,*], Mikhail Raiko [1], Alla Lapidus [1] and Pavel A. Pevzner [1, 2]

[1]Center for Algorithmic Biotechnology, Institute for Translational Biomedicine, St. Petersburg State University, St. Petersburg, Russia
[2]Department of Computer Science and Engineering, University of California, San Diego, California, USA.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Although the set of currently known viruses have been steadily increasing, only a tiny fraction of the Earth's virome has been sequenced so far. Shotgun metagenomic sequencing provides an excellent opportunity to reveal novel viruses but faces the computational challenge of identifying viral genomes that are often difficult to detect in metagenomic assemblies.
Results: We describe a metaviralSPAdes tool for identifying viral genomes in metagenomic assembly graphs that is based on analyzing variations in the coverage depth between viruses and bacterial chromosomes. We benchmarked metaviralSPAdes on diverse metagenomic datasets, verified our predictions using a set of virus-specific Hidden Markov Models, and demonstrated that it improves on the state-of-the-art viral identification pipelines.
Availability: metaviralSPAdes includes viralAssembly, viralVerify, and viralComplete modules. viralAssembly, viralVerify and viralComplete are available as standalone packages: https://github.com/ablab/spades/tree/metaviral_publication, https://github.com/ablab/viralVerify/ and https://github.com/ablab/viralComplete/
Contact: d.antipov@spbu.ru
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

In the last few years, metagenomic sequencing greatly expanded our knowledge of the Earth's virome (Paez-Espino *et al.*, 2016; Roux *et al.*, 2016). However, since extracting complete sequences of viral genomes from metagenomic assemblies remains challenging, many viruses evade identification even though metagenomic datasets contain reads sampled from these viruses (Dutilh *et al.*, 2014).

Previous studies, aimed at the discovery of novel viruses, often focused on viral contigs in metagenomic assemblies and thus missed an opportunity to sequence complete viral genomes by switching from the contig-based to the assembly graph-based analysis. Since a recent study (Roux *et al.*, 2017) reported that metaSPAdes (Nurk *et al.*, 2017) resulted in the most contiguous viral assemblies, we extended metaSPAdes into metaviralSPAdes that attempts to sequence complete viral genomes rather than fragmented viral contigs.

Identifying viral genomes in metagenomic datasets is not unlike identifying plasmids since both viruses and plasmids form small subgraphs of the metagenomic assembly graphs. However, in difference from plasmid sequencing where multiple plasmid identification tools have been developed (Antipov *et al.*, 2016, 2019; Rozov *et al.*, 2017), there is still no specialized viral assembler. metaviralSPAdes modifies various steps of the metaplasmidSPAdes tool (Antipov *et al.*, 2019) to make it applicable to viral sequencing. Below we describe the metaviralSPAdes pipeline and apply it for virus discovery in diverse metagenomic datasets.

2 Methods

metaviralSPAdes pipeline consists of three independent steps - viralAssembly for finding putative viral subgraphs in a metagenomic assembly graph and generating contigs in these graphs, viralVerify for checking whether the resulting contigs have viral origin, and viralComplete for checking whether these contigs represent complete viral genomes.

2.1 Assembling viral sequences (viralAssembly)

To assemble viral sequences, metaviralSPAdes modifies approaches implemented in metaSPAdes (Nurk *et al.*, 2017) and metaplasmidSPAdes (Antipov *et al.*, 2019). First, it uses metaSPAdes to construct the assembly graph. Since various viral strains are often highly variable (Shapiro and Putonti, 2018), and since we focus on species-level viral assembly, viralAssembly modifies the bulge removal procedure as compared to metaSPAdes (Nurk *et al.*, 2017). Specifically, it collapses long and similar (with respect to the edit distance) parallel edges in the assembly graph that are shorter than $maxBulgeSize$ (the default value 1000 nucleotides)

and that differ from each other by less than $maxDivergence$ (the default value 0.2). The divergence between two sequences is defined as the edit distance between them divided by the length of the shorter sequence.

Since the vast majority of plasmids are circular, metaplasmidSPAdes is based on identifying high-coverage cycles in the assembly graph, i.e., cycles with coverage by reads exceeding the coverage of neighboring edges in the assembly graph. In contrast, since many viruses are linear (50% of DNA viruses in the RefSeq (O'Leary *et al.*, 2016) database), metaviralSPAdes searches for both high coverage cycles and high coverage paths that start from a vertex of in-degree 0 (*source* vertices) and end in a vertex of out-degree 0 (*sink* vertices) of the assembly graph. We classify such a path as *long* if its length exceeds a threshold $Length$ (the default value 1000 nucleotides) and *high coverage* if its coverage exceeds a threshold $Coverage$ (the default value 5x). Long high-coverage paths represent putative sequences of linear viruses.

Many linear DNA viruses have terminal repeats (Deng *et al.*, 2012; Casjens and Gilcrease, 2009), and thus correspond to small subgraphs rather than isolated paths in the assembly graphs. Sequences of 377 out of 2584 linear DNA viruses in the RefSeq database have terminal repeats with a length exceeding the typical length of k-mers used for constructing assembly graphs (the default value $k = 55$ for metaSPAdes). 168 of such viruses can be represented as sequences $ARBR$ or RAR', where R is a terminal repeat of length > 55 bp and R' is its complement (Supplementary Figure 1S).

To identify linear viruses with terminal repeats metaviralSPAdes consider small (less than 6 edges) connected components of the assembly graph. We refer to a path in a graph as a *postman tour* if it visits each edge of the graph and the total length of its unique edges (i.e., edges that are visited just once) exceeds half of the total path length. For components of type $ARBR$ and RAR', since there exists a unique postman tour in such components, viralAssembly outputs complete sequence instead of the corresponding subgraph.

To give users an option to examine both complete viral sequences (identified based on analyzing small subgraphs of the metaSPAdes assembly graphs) and partial viral sequences (corresponding to metaSPAdes contigs), the viralAssembly output is combined with the regular metaSPAdes output.

2.2 Viral verification (viralVerify)

The viralVerify module checks whether contigs found by viralAssembly indeed represent viruses. The popular virus identification tools, such as the HMM-based VirSorter (Roux *et al.*, 2015) and the *k*-mer based VirFinder (Ren *et al.*, 2017) have limitations: they are sometimes confuse phages with plasmids and are rather conservative, thus missing putative novel viruses. We thus developed the viralVerify tool that examines the gene content of a contig and classifies it as viral/bacterial/uncertain using a Naive Bayesian classifier. It can be used as a standalone tool to predict contigs of viral origin in any assembled metagenome.

The viralVerify step in metaviralSPAdes is designed similarly to the plasmidVerify step in metaplasmidSPAdes (Antipov *et al.*, 2019). To construct a set of viral HMMs, we selected all 10,544 viruses from the RefSeq database and split them into the training and validation datasets (7,381 and 3,163 viruses, respectively). We predicted genes with Prodigal v2.6.3 (Hyatt *et al.*, 2010) and ran hmmsearch (part of HMMER 3.1b2, http://hmmerorg/) using Pfam-A database v. 30.0 (El-Gebali *et al.*, 2018). Afterwards, we counted the frequencies of matches to the training dataset, and used them to train a Naive Bayesian classifier (Friedman *et al.*, 2001) along with frequencies from *nonViralDatabase* (combined *PlasmidDatabase* and *nonPlasmidDatabase* from Antipov *et al.*, 2019). Supplemental Table 1 lists the HMM frequencies in the training dataset.

Given a contig, viralVerify predicts genes in this contig using Prodigal in the metagenomic mode, runs hmmsearch on the predicted proteins, and calculates the score as the ratio of log probabilities. If the absolute score is less than a $scoreThreshold$ (the default value is 3), a contig is classified as "uncertain", otherwise it is classified as "viral" ($score > scoreThreshold$) or "bacterial" ($score < -scoreThreshold$).

To help analyze the rapidly growing amount of novel data, we have added a script that allows users to construct their own training database from a set of viral, chromosomal and plasmid contigs, as well as custom HMM database.

2.3 Viral completeness verification (viralComplete)

If a newly constructed viral contig is complete and belongs to a known family of viruses then its gene content is likely to be similar to the gene content of a known virus. We thus compute the "similarity" of a given contig (based on the Naive Bayesian Classifier) to each known virus from the RefSeq database, and check whether the most similar known virus have length similar to the contig length. This comparison includes the following steps:

1. Predict genes and proteins in a given contig using Prodigal.
2. Match each predicted protein P against all N viral proteins from the RefSeq database using BLAST (with e-value cutoff = 1e-6) and define $number(P)$ as the number of viral proteins matching P. We say that a virus V matches a protein P if one of the proteins in this virus matches P.
3. If a virus V matches a protein P, we define $Prob(V|P) = 1/number(P) - \epsilon$, where ϵ is a small number (equal to 1e-6 in implementation). If a virus V does not match P, we define $Prob(V|P) = \epsilon \cdot number(P)/(N - number(P))$. Thus, each virus that matches P has the same (large) probability $Prob(V|P)$ and each virus that does not match P has the same small probability.
4. If a given contig has predicted proteins $P_1, P_2, \ldots, P_k$, we assume that they all are pairwise conditionally independent and define $Prob(V|P_1, P_2, \ldots, P_k)$ as $Prob(V|P_1) \cdot Prob(V|P_2) \cdot \ldots \cdot Prob(V|P_k)$. A most probable virus V^* is defined as a virus maximizing this probability.
5. Check whether the given contig and the virus V^* have similar lengths, i.e., if the length of V^* falls in the range $(0.9 \cdot length(contig), length(contig)/0.9)$.

3 Results

3.1 Datasets

We used both simulated metagenomes and real metagenomes/metaviromes to benchmark metaviralSPAdes:

3.1.1 Simulated metagenomes

We simulated 5 metagenomic datasets using CAMISIM (Fritz *et al.*, 2019). For each metagenome, 15 bacterial and 15 viral genomes were drawn from the test datasets. The abundance distribution followed the log-normal distribution with $\mu = 1$ and $\sigma = 0.5$. Total abundance of the viral genomes was set to be 10 times higher than the abundance of microbial genomes, to model high abundances of viruses in real metagenomic datasets (see Supplemental Table 2).

3.1.2 Real metagenomes

We selected 18 diverse metagenomic datasets described in Supplemental Table 3 to benchmark metaviralSPAdes. Two out of these 18 datasets represent metavirome datasets originating from marine samples that were size-selected for viruses. Additionally, we used sequences of known origin from the RefSeq database for benchmarking viralVerify and viralComplete. As the true negative test datasets, we used the PlasmidDatabase dataset (2,387 plasmids) and 9,890 randomly selected fragments from

nonPlasmidContigs dataset (80,681 chromosome fragments) described in Antipov *et al.*, 2019. Since VirSorter and VirFinder are designed for DNA viruses, for a fair comparison we selected only double-stranded DNA viruses (total 1,368) from the viralVerify validation dataset as the true positive test dataset. Additionally, we trained and checked viralVerify's performance on small RNA viruses (Supplemental Table 7). The same true positive test dataset was used to benchmark viralComplete.

3.2 viralAssembly benchmarking

Since there are still no specialized assembly tools that identify viral genomes in metagenomic datasets, we compared viralAssembly against metaSPAdes on 18 real datasets described in Supplemental Table 3. We analyzed only complete (i.e., circular contigs or linear contigs starting in sources and ending in sinks) and high-coverage (>5x) sequences for benchmarking (viralAssembly and metaSPAdes report the same set of partial contigs). We used three different contig length cutoffs (0.5 kb, 3 kb, and 10 kb) and checked the viral origin of the contigs using viralVerify. Since the contigs number may reflect an increase in the number of fragmented contigs (rather than complete viruses), we also checked the completeness of the predicted viral contigs using viralComplete. viralAssembly outperformed metaSPAdes in the number of assembled viral contigs on 12 out of these 18 samples (Supplemental Table 6).

3.3 viralVerify benchmarking

We benchmarked viralVerify versus VirSorter (Roux *et al.*, 2015) VirFinder (Ren *et al.*, 2017) and virMine (Garretto *et al.*, 2019) on 1368 dsDNA viruses from Refseq database as the true positive dataset, and two true negative datasets - plasmids from the RefSeq database and the set of 10kb-long randomly selected fragments of bacterial chromosomes, to mimic a real output of a metagenomic assembly (see Table 1). Researchers are usually interested in viruses distantly related to known ones, or in the contigs of unknown origin, referred to as the "dark matter" of metagenome. We thus took into account contigs of interest that cannot be certainly attributed as viruses or chromosomes but deserve manual inspection (category 3 in the VirSorter output, "Uncertain" category in the viralVerify output and "Unknown" in virMine). Since BLASTN and BLASTX (Altschul *et al.*, 1990) are popular as virus detection tools, we also included them in our benchmarking analysis. Although BLASTX and viralVerify showed similar results, it is two orders of magnitude slower than viralVerify. For the true negative dataset (9,890 chromosomal fragments), the running time of viralVerify and BLASTX was 279 and 36,364 minutes, respectively. Also, since we randomly split the entire dataset into the training and testing datasets, the training dataset is likely to contain viruses from the same taxonomic groups as the testing dataset. To test the performance in the case of novel taxonomic groups, we excluded the entire viral family of Podoviridae from the training dataset and compared viralVerify and BLASTX results on the members of this family. Afterwards, to compare performance on higher taxonomic level, we trained classifier on Caudovirales order (tailed bacteriophages), and compared viralVerify with BLASTX on the non-Caudovirales phages. Supplemental Table 8 illustrates that viralVerify improves on BLASTX in this more difficult test, likely because the HMM-based approach is more sensitive than the local alignment approach.

Additionally, we benchmarked VirSorter, VirFinder, virMine and viralVerify on simulated metagenomes. Since the bacterial chromosomes we use for simulation may carry prophages, we needed to separate contigs that belong to reference viruses from those of prophage origin.

To identify contigs of viral origin (true positives), we compared them with the reference viruses using minimap2 (Li, 2018) and considered sequences with nucleotide identity $> 95\%$ as viral. To account for possible prophage sequences in the reference chromosomes, we aligned all contigs that were unaligned on the previous step against the viral RefSeq database.

Table 1. Benchmarking various viral detection approaches.

	True positive, dsDNA viruses	True negative, 10k chunks	True negative, plasmids
Total	1,368	9,890	2,387
VirSorter (1-2 categories)	758 55.4%	151 15.5%	441 18.5%
VirSorter (1-2-3 categories)	766 56%	200 15.5%	677 28.4%
VirFinder	866 63.3%	117 1.1%	205 8.6%
viralVerify (Virus only)	1,277 93.3%	118 1.2%	79 3.3%
viralVerify (Virus+Uncertain)	**1,319** **96.4%**	245 2.5%	149 6.2%
virMine (Virus only)	1,176 85.9%	**39** **0.4%**	14 0.6%
virMine (Virus+Unknown)	1,229 89.8%	42 0.4%	15 0.6%
BLASTN	1,069 78.1%	42 0.4%	**8** **0.3%**
BLASTX	1,258 91.9 %	55 0.6%	17 0.7%

Results of the viral detection benchmarking on the true positive and the true negative test datasets. The numbers and percentages represent sequences identified as viral. VirFinder was launched with the score at least 0.7 and p-value below 0.05, BLASTN and BLASTX were launched against the database from the viralVerify training dataset, with the E-value threshold 0.001 (top hit was selected).

All contigs that mapped to any virus not used for simulation with identity $> 80\%$ and span $> 50\%$ were removed from comparison (see Supplemental Table 2).

Although all tools except virMine showed similar precision , viralVerify improved on all tools in terms of recall. Relatively low precision of all tools except virMine ($< 64\%$) can be explained by many identified prophage sequences that are absent in the viral RefSeq database. Fig. 1 and Supplemental Table 4 illustrate that performance of all tools increases with the increase in the contig lengths.

For the real datasets, we analyzed the results of the metaSPAdes assembly for the 18 metagenomic samples. We compared viralVerify with VirSorter, VirFinder and the results of BLAST alignment to viral RefSeq and metagenomic viral contigs (mVCs) from Paez-Espino *et al.* (2016). Although the ground truth in this computational experiment is unknown, VirFinder and viralVerify predicted significantly more sequences than VirSorter for most samples (Supplemental Table 5).

3.4 viralComplete benchmarking

To benchmark viralComplete, we randomly split the test dataset of 1368 dsDNA viruses from RefSeq in two equal parts, and cut one of these parts into fragments of size $x\%$ of their original length, resulting in a true negative dataset ($x\%$ is selected uniformly at random between 10% and 90%). Table 2 presents viralComplete results. viralComplete shows 12.1% completeness (83 out of 684 viral fragments) for the true negative dataset

(fragmented viruses) and 86.8% completeness (594 out of 684 complete viruses) for the true positive dataset.

Table 2. Benchmarking viralComplete on true positive and true negative datasets.

	684 fragmented	684 complete
	dsDNA viruses	dsDNA viruses
Complete	83 (12.1%)	594 (86.8%)
Partial	601 (87.9%)	90 (13.2%)

3.5 Exploring novel viruses assembled by metaviralSPAdes

We checked whether some of the assembled viral contigs represent crAssphages, wide-spread and abundant phages in the human microbiome that however evaded all virus detection tools until recently (Dutilh *et al.*, 2014). Yutin *et al.* (2018) revealed a previously unknown family of crAssphage-like viruses, represented in many genomic and metagenomic databases as misclassified bacterial contigs or uncultured viruses. These crAssphage contigs avoided detection because over 80% of the predicted proteins in these contigs showed no significant similarity to known protein sequences. Also, even though the length of the previously known crAssphage genomes is 90-100 kbp, the lengths of these contigs were significantly shorter, likely representing incomplete phage genomes. However, based on a conserved gene content, Yutin *et al.* (2018) identified a distinct crAssphage group and a group of similar crAssphage-like viruses.

metaviralSPAdes assembled seven complete or near-complete phages from the crAssphage family, including members of the crAssphage group, in various metagenomes (Supplemental Table 3). Supplemental Figure 2 presents a phylogenetic tree of major capsid proteins of the fully assembled viruses from the crAssphage family.

4 Discussion

We demonstrated that metaviralSPAdes improves identification of complete viruses from metagenomic datasets. Our analysis of newly sequenced phages from the crAssphage family illustrates that metaviralSPAdes has a potential to transform metagenomics-based assembly of novel viruses from a challenging task into a routine procedure.

However, many viruses still remain undetected or incomplete, indicating that we may be close to reaching the limits of viral sequencing using short-read technologies. Kolmogorov *et al.*, 2019 recently demonstrated that long-read technologies recover more viruses from metagenomic datasets than short-read technologies. However, the accuracy of viral sequences recovered from long-read metagenomic datasets is often inferior, especially for viral genomes with coverage below 30x. We thus argue that integration of long-read and short-read metagenomic datasets is a promising approach for recovering many new viruses.

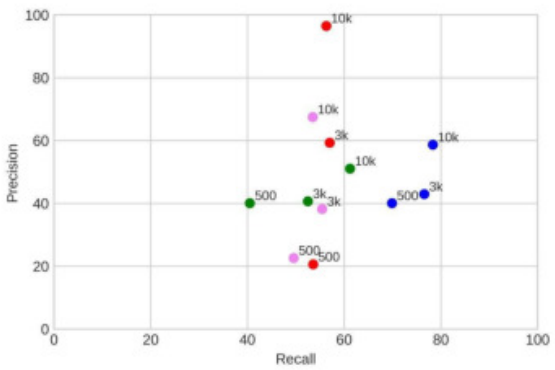

Figure 1. Average precision and recall on five simulated datasets for viralVerify (blue), VirFinder (violet) virMine (red) and VirSorter (green). Precision and recall were calculated separately for contigs longer than 0.5 kb, 3 kb, and 10 kb.

Acknowledgements

We wish to thank Natalia Yutin and Eugene Koonin for helpful comments and discussion.

Funding

This work was supported by the Russian Science Foundation (grants 19-14-00172 and 19-16-00049).

References

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic local alignment search tool. *Journal of molecular biology*, **215**(3), 403–410.

Antipov, D., Hartwick, N., Shen, M., Raiko, M., Lapidus, A., and Pevzner, P. (2016). plasmidspades: assembling plasmids from whole genome sequencing data. *bioRxiv*, page 048942.

Antipov, D., Raiko, M., Lapidus, A., and Pevzner, P. A. (2019). Plasmid detection and assembly in genomic and metagenomic data sets. *Genome research*, **29**(6), 961–968.

Casjens, S. R. and Gilcrease, E. B. (2009). Determining dna packaging strategy by analysis of the termini of the chromosomes in tailed-bacteriophage virions. *Bacteriophages*, pages 91–111.

Deng, Z., Wang, Z., and Lieberman, P. M. (2012). Telomeres and viruses: common themes of genome maintenance. *Frontiers in oncology*, **2**, 201.

Dutilh, B. E., Cassman, N., McNair, K., Sanchez, S. E., Silva, G. G., Boling, L., Barr, J. J., Speth, D. R., Seguritan, V., Aziz, R. K., *et al.* (2014). A highly abundant bacteriophage discovered in the unknown sequences of human faecal metagenomes. *Nature communications*, **5**, 4498.

El-Gebali, S., Mistry, J., Bateman, A., Eddy, S. R., Luciani, A., Potter, S. C., Qureshi, M., Richardson, L. J., Salazar, G. A., Smart, A., *et al.* (2018). The pfam protein families database in 2019. *Nucleic acids research*, **47**(D1), D427–D432.

Friedman, J., Hastie, T., and Tibshirani, R. (2001). *The elements of statistical learning*, volume 1. Springer series in statistics New York.

Fritz, A., Hofmann, P., Majda, S., Dahms, E., Dröge, J., Fiedler, J., Lesker, T. R., Belmann, P., DeMaere, M. Z., Darling, A. E., *et al.* (2019). Camisim: simulating metagenomes and microbial communities. *Microbiome*, **7**(1), 17.

Garretto, A., Hatzopoulos, T., and Putonti, C. (2019). virmine: automated detection of viral sequences from complex metagenomic samples. *PeerJ*, **7**, e6695.

Hyatt, D., Chen, G.-L., LoCascio, P. F., Land, M. L., Larimer, F. W., and Hauser, L. J. (2010). Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC bioinformatics*, **11**(1), 119.

Kolmogorov, M., Rayko, M., Yuan, J., Polevikov, E., and Pevzner, P. (2019). metaflye: scalable long-read metagenome assembly using repeat graphs. *bioRxiv*, page 637637.

Li, H. (2018). Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*, **34**(18), 3094–3100.

Nurk, S., Meleshko, D., Korobeynikov, A., and Pevzner, P. A. (2017). metaspades: a new versatile metagenomic assembler. *Genome research*, **27**(5), 824–834.

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

Paez-Espino, D., Eloe-Fadrosh, E. A., Pavlopoulos, G. A., Thomas, A. D., Huntemann, M., Mikhailova, N., Rubin, E., Ivanova, N. N., and Kyrpides, N. C. (2016). Uncovering earth's virome. *Nature*, **536**(7617), 425.

Ren, J., Ahlgren, N. A., Lu, Y. Y., Fuhrman, J. A., and Sun, F. (2017). Virfinder: a novel k-mer based tool for identifying viral sequences from assembled metagenomic data. *Microbiome*, **5**(1), 69.

Roux, S., Enault, F., Hurwitz, B. L., and Sullivan, M. B. (2015). Virsorter: mining viral signal from microbial genomic data. *PeerJ*, **3**, e985.

Roux, S., Brum, J. R., Dutilh, B. E., Sunagawa, S., Duhaime, M. B., Loy, A., Poulos, B. T., Solonenko, N., Lara, E., Poulain, J., *et al.* (2016). Ecogenomics and potential biogeochemical impacts of globally abundant ocean viruses. *Nature*, **537**(7622), 689.

Roux, S., Emerson, J. B., Eloe-Fadrosh, E. A., and Sullivan, M. B. (2017). Benchmarking viromics: an in silico evaluation of metagenome-enabled estimates of viral community composition and diversity. *PeerJ*, **5**, e3817.

Rozov, R., Brown Kav, A., Bogumil, D., Shterzer, N., Halperin, E., Mizrahi, I., and Shamir, R. (2017). Recycler: an algorithm for detecting plasmids from de novo assembly graphs. *Bioinformatics*, **33**(4), 475–482.

Shapiro, J. W. and Putonti, C. (2018). Gene co-occurrence networks reflect bacteriophage ecology and evolution. *MBio*, **9**(2), e01870–17.

Yutin, N., Makarova, K. S., Gussow, A. B., Krupovic, M., Segall, A., Edwards, R. A., and Koonin, E. V. (2018). Discovery of an expansive bacteriophage family that includes the most abundant viruses from the human gut. *Nature microbiology*, **3**(1), 38.