Expanding standards in viromics: *in silico* evaluation of viral identification, taxonomy, and auxiliary metabolic genes (AMGs) curation

Akbar Adjie Pratama [1, 2], Ben Bolduc [1, 2], Ahmed A. Zayed [1, 2], Zhi-Ping Zhong [1, 2,] Jiarong Guo [1, 2], Dean Vik [1, 2], Maria Consuelo Gazitua [3], James Wainaina [1, 2], Simon Roux [*4], Matthew B. Sullivan [* 1, 2, 5]

[1] Department of Microbiology, Ohio State University, Columbus, OH, United States
[2] Center of Microbiome Science, Ohio State University, Columbus, OH, United States
[3] Viromica Consulting, Santiago, Chile
[4] Lawrence Berkeley National Lab, Berkeley, CA, United States
[5] Department of Civil, Environmental and Geodetic Engineering, Ohio State University, Columbus, OH, United States
*Corresponding Authors: Simon Roux, Matthew B. Sullivan
Email address: sroux@lbl.gov, sullivan.948@osu.edu

Project goals: The overarching goal of this project is to establish ecological paradigms for how viruses alter soil microbiomes and nutrient cycles by developing foundational (eco)systems biology approaches for soil viruses. Specifically, we aim to provide recommendations for how best to analyze (dsDNA) viruses in viromes and bulk metagenomic samples. We use *in-silico* datasets that mimic viromes and bulk metagenomes with varied inference from non-virus 'distractor' sequences to evaluate (*i*) options for viral identification, (*ii*) genomic fragment sizes for viral classification via gene-sharing networks, as well as (*iii*) provide guidelines for best practices for the evaluation of candidate auxiliary metabolic genes (AMGs). These analyses and results contribute to the growing set of community-driven benchmarks and guidelines in the field of environmental virology.

Abstract:
Metagenomic approaches have been critical for revealing viral roles across diverse ecosystems, driving force in microbial diversity and nutrient cycling. However, the emergent field of viral ecogenomics would benefit from comparative benchmarking to better standardize and enable comparison across datasets. Here we constructed *in silico*-generated datasets that mimicked features of viromes and bulk metagenomes, and used these to provide guidelines and highlight potential pitfalls of viral metagenomic analyses. We compared the performance of the most commonly-used viral identification tools, evaluate viral taxonomic assignations, and propose guidelines to systematize the evaluation of candidate AMGs.

The *in silico* benchmarking of five commonly-used viral identification tools show that gene-content-based tools consistently performed well for long (≥3 kbp) contigs, while *k*-mer- and blast-based tools were uniquely able to detect viral sequences from short (<3 kbp) contigs. Notably, however, the performance increase of *k*-mer- and blast-based tools for short contigs was obtained at the cost of increased false positives (sometimes up to ~40%), particularly when eukaryotic or mobile genetic element sequences were included in the test datasets. For viral classification, variously sized genome fragments were assessed using gene-sharing network analytics to quantify drop-offs in taxonomic assignments, which revealed correct assignations ranging from ~90% (whole genomes) down to ~50% (3 kbp sized genome fragments). Finally, we highlight how fragmented assemblies can lead to erroneous identification of AMGs, and outline a comprehensive workflow that can be used to curate candidate AMGs in viral genomes assembled from metagenomes. Together these benchmarking experiments provide guidance for researchers seeking to best detect and characterize the myriad viruses 'hidden' in diverse sequence datasets.

Funding statement: This research was supported by the U.S. Department of Energy (#DE-SC0020173 and #248445), and the Gordon and Betty Moore Foundation (#3790) to MBS. The work conducted by the U.S. Department of Energy Joint Genome Institute is supported by the Office of Science of the U.S. Department of Energy under contract no. DE-AC02-05CH11231. An award from the Ohio Supercomputer Center (OSC) to MBS supported computing resources used here.