SOFTWARE TOOL ARTICLE

Lessons learned: overcoming common challenges in reconstructing the SARS-CoV-2 genome from short-read sequencing data via CoVpipe2

[version 1; peer review: 1 approved, 1 approved with reservations]

Marie Lataretu, Oliver Drechsel, René Kmiecinski, Kathrin Trappe, Martin Hölzer*, Stephan Fuchs*

Genome Competence Center (MF1), Robert Koch Institute, Berlin, 13353, Germany

* Equal contributors

v1 **First published:** 01 Sep 2023, **12**:1091
https://doi.org/10.12688/f1000research.136683.1
Latest published: 16 Apr 2024, **12**:1091
https://doi.org/10.12688/f1000research.136683.2

Abstract

Background: Accurate genome sequences form the basis for genomic surveillance programs, the added value of which was impressively demonstrated during the COVID-19 pandemic by tracing transmission chains, discovering new viral lineages and mutations, and assessing them for infectiousness and resistance to available treatments. Amplicon strategies employing Illumina sequencing have become widely established for variant detection and reference-based reconstruction of SARS-CoV-2 genomes, and are routine bioinformatics tasks. Yet, specific challenges arise when analyzing amplicon data, for example, when crucial and even lineage-determining mutations occur near primer sites.

Methods: We present CoVpipe2, a bioinformatics workflow developed at the Public Health Institute of Germany to reconstruct SARS-CoV-2 genomes based on short-read sequencing data accurately. The decisive factor here is the reliable, accurate, and rapid reconstruction of genomes, considering the specifics of the used sequencing protocol. Besides fundamental tasks like quality control, mapping, variant calling, and consensus generation, we also implemented additional features to ease the detection of mixed samples and recombinants.

Results: Here, we highlight common pitfalls in primer clipping, detecting heterozygote variants, and dealing with low-coverage regions and deletions. We introduce CoVpipe2 to address the above challenges and have compared and successfully validated the pipeline against selected publicly available benchmark datasets. CoVpipe2 features high usability, reproducibility, and a modular design that

Open Peer Review

Approval Status ✓ ✓

	1	2
version 2 (revision) 16 Apr 2024		✓ view
version 1 01 Sep 2023	✓ view	? view

1. **Wolfgang Maier**, Albert-Ludwigs-Universitat Freiburg, Freiburg, Germany

2. **Sondes Haddad-Boubaker**, University of Tunis El Manar, Tunis, Tunisia

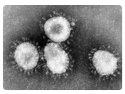
Any reports and responses or comments on the article can be found at the end of the article.

specifically addresses the characteristics of short-read amplicon protocols but can also be used for whole-genome short-read sequencing data.

Conclusions: CoVpipe2 has seen multiple improvement cycles and is continuously maintained alongside frequently updated primer schemes and new developments in the scientific community. Our pipeline is easy to set up and use and can serve as a blueprint for other pathogens in the future due to its flexibility and modularity, providing a long-term perspective for continuous support. CoVpipe2 is written in Nextflow and is freely accessible from https://github.com/rki-mf1/CoVpipe2 under the GPL3 license.

Keywords
SARS-CoV-2, genome reconstruction, whole-genome sequencing, short reads, Illumina, amplicons, WGS, Nextflow pipeline, virus bioinformatics

This article is included in the Bioinformatics gateway.



This article is included in the Coronavirus (COVID-19) collection.

This article is included in the Virus Bioinformatics collection.

Corresponding authors: Marie Lataretu (lataretum@rki.de), Stephan Fuchs (FuchsS@rki.de)

Author roles: Lataretu M: Data Curation, Formal Analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – Original Draft Preparation; **Drechsel O**: Methodology, Software, Writing – Review & Editing; **Kmiecinski R**: Methodology, Software; **Trappe K**: Software, Writing – Review & Editing; **Hölzer M**: Formal Analysis, Investigation, Methodology, Supervision, Validation, Writing – Original Draft Preparation; **Fuchs S**: Conceptualization, Funding Acquisition, Methodology, Project Administration, Supervision, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: M.L. was supported by the European Centre for Disease Control (grant number ECDC GRANT/2021/008 ECD.12222).
The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

How to cite this article: Lataretu M, Drechsel O, Kmiecinski R *et al.* **Lessons learned: overcoming common challenges in reconstructing the SARS-CoV-2 genome from short-read sequencing data via CoVpipe2 [version 1; peer review: 1 approved, 1 approved with reservations]** F1000Research 2023, **12**:1091 https://doi.org/10.12688/f1000research.136683.1

First published: 01 Sep 2023, **12**:1091 https://doi.org/10.12688/f1000research.136683.1

Introduction

Since the publication of the first genome sequence of the novel SARS-CoV-2 virus in January 2020 – just 12 days after the initial report of the virus – the international GISAID database[1–3] now includes more than 15.5 million SARS-CoV-2 whole-genome sequences (accessed May 23, 2023). The genomic data and metadata collected in GISAID and other resources such as EBI's COVID-19 Data Portal[4] are pivotal for the largest worldwide genomic surveillance effort ever undertaken to track the evolution and spread of the virus causing the COVID-19 disease. Important viral genome regions have been monitored for mutations, for example, in the spike gene and other immunologically relevant loci. The reconstruction of accurate SARS-CoV-2 genomic sequences is paramount to detect and track substitutions, insertions, and deletions correctly; interpret them in terms of vaccine development, test the efficiency of target regions and antibody binding sites, detect outbreaks and transmission chains, and finally inform public health authorities to consider adjustment of containment measures.[5]

According to Ewan Birney, director of EMBL-EBI in Cambridge, U.K., "Genome sequencing is routine in the same way the U.S. Navy routinely lands planes on aircraft carriers. Yes, a good, organized crew does this routinely, but it is complex and surprisingly easy to screw up.".[6]

This quote is no less accurate for genome reconstruction, a crucial step in SARS-CoV-2 genomic surveillance. While sequencing efforts were scaling up rapidly around the globe, several pipelines for the reference-based assembly of SARS-CoV-2 genomes were developed in parallel and in an attempt to rapidly generate the necessary genome sequences (Table 1). During the first lockdown in Germany in mid-March 2020, the Bioinformatics unit at the Robert Koch Institute (RKI), Germany's Public Health institute, also started developing a genome reconstruction pipeline, specifically targeting Illumina amplicon sequencing data and amplicon schemes. During the development, the pipeline was extensively tested and has gone through continuous improvement due to adjusted wet lab protocols and primer schemes, to accurately call variants in low-coverage regions and near primer sites, to deal with deletions and low-coverage regions correctly, and to robustly reconstruct high-quality SARS-CoV-2 consensus sequences for downstream analyses and genomic surveillance. Due to these features and tests, CoVpipe1 was successfully used in the past at RKI's sequencing facility and several labs nationwide.[7–10] If not addressed appropriately, genotyping errors can lead to wrong consensus sequences and thus impact downstream analyses such as phylogenetic reconstructions and transmission chain tracking in outbreaks.[11]

While sequencing intensity and turnaround times on variant detection increased in different countries, there are also major disparities between high-, low- and middle-income countries in the SARS-CoV-2 global genomic surveillance efforts.[12] A recent study also found significant differences between bioinformatics approaches that use the same input data but detect different variants in SARS-CoV-2 samples.[13] In addition, each technology and sequencing approach has its own advantages and limitations, challenging a harmonized genomic surveillance of the virus.[14]

In Germany, sequencing efforts increased tremendously in January 2021 following the entry into force of a federal directive (Coronavirus Surveillance Verordnung—CorSurV). Subsequently, a large-scale, decentralized genomic sequencing and data collection system ("Deutscher Elektronischer Sequenzdaten-Hub", DESH)[15] has been established, accompanied by a medium-scale integrated molecular surveillance infrastructure (IMS-SC2) at the RKI.[8] By May 22, 2023, 1,227,036 whole-genome SARS-CoV-2 sequences that met the quality criteria were transmitted to the RKI via DESH. Due to their low cost, sensitivity, flexibility, specificity, and efficiency, amplicon-based sequencing designs are broadly used for SARS-CoV-2 sequencing and reference-based genome reconstruction.[16–20] From the ∼1,2 million DESH genomes (publicly available at github.com/robert-koch-institut/SARS-CoV-2-Sequenzdaten_aus_Deutschland), ∼1,1 million (90.91%) were sequenced with Illumina devices, highlighting the importance of the technology for genomic surveillance (Table 2). Illumina technology has a lower share at the international level (78.57%), while Oxford Nanopore Technologies (ONT) increased to 12.73% (Table 2). However, Illumina remains the most widely used approach among the various sequencing technologies, followed by ONT sequencing by a wide margin. A recent benchmark study also showed the advantages of using Illumina MiSeq compared to ONT GridION for SARS-CoV-2 sequencing, resulting in a higher number of consensus genomes classified by Nextclade[21] as good and mediocre.[22] However, these results are, of course, also dependent on the bioinformatics tool chains and could change as ONT becomes more accurate.[23] In addition, although both technologies require the same computational steps for reference-based genome reconstruction (preprocessing, mapping, variant calling, consensus), they need different tools optimized for either short- or long-read data and the associated error profiles to produce high-quality consensus sequences. Therefore, we developed one pipeline, especially for ONT data,[24] and CoVpipe2, specifically targeting high-accuracy SARS-CoV-2 amplicon data derived from short-read Illumina sequencing. CoVpipe2 is a Nextflow re-implementation of CoVpipe1 (written in Snakemake, gitlab.com/RKIBioinformaticsPipelines/ncov_minipipe) and comes with additional features, simplified installation, full container support, and continuous maintenance.

Table 1. A collection of available software for SARS-CoV-2 genome reconstruction. Here, we mainly focus on open-source pipelines with available source code, specifically targeting the reconstruction of SARS-CoV-2 genomes, also including general pipelines adapted to work with SARS-CoV-2 sequencing data (e.g., V-pipe). Publ. – Publication focusing on the pipeline in a preprint or a peer-reviewed journal, Seq. tech. – focused sequencing technology, Implementation – main software backbone to run the tool (please note that not all pipelines use a workflow management system such as Nextflow[25] or Snakemake[26]), Dependencies – a list of options to handle necessary software dependencies, Latest release – latest available release version as accessed May 23, 2023. mNGS – sequencing approach not based on amplicons but rather sequencing all available RNA/cDNA (metatranscriptomics/genomics).

Tool	Publ.	Code	Seq. tech.	Implementation	Dependencies	Latest release
CoVpipe2	–	github.com/rki-mf1/covpipe2	Illumina	Nextflow	Conda, Mamba, Docker, Singularity	0.4.2 (2023-06-09)
poreCov	[24]	github.com/replikation/poreCov	Nanopore	Nextflow	Docker, Singularity	1.8.3 (2023-05-12)
viralrecon	[27]	github.com/nf-core/viralrecon	Illumina, Nanopore	Nextflow	Conda, Docker, Singularity, Podman, Shifter, Charliecloud	2.6.0 (2023-03-23)
SIGNAL	[28]	github.com/jaleezyy/covid-19-signal	Illumina	Snakemake	Conda, Mamba, Docker (a single container with all dependencies)	1.6.2 (2023-05-20)
V-pipe	[29]	github.com/cbg-ethz/V-pipe	Illumina	Snakemake	Conda, Docker (a single container with all dependencies)	2.99.3 (2022-11-02)
NCBI SC2VC	–	github.com/ncbi/sars2variantcalling	Illumina, Nanopore, PacBio	Snakemake (multiple pipelines, run via wrapper script)	Docker (a single container with all dependencies)	3.3.4 (2022-10-28)
VirPipe	[30]	github.com/KijinKims/VirPipe	Illumina, Nanopore (mNGS)	Python, Nextflow	Conda and Docker	1.0.0 (2022-09-24)
ViralFlow	[31]	github.com/dezordi/ViralFlow	Illumina	Python	Conda, Singularity, Docker (one single container/environment)	v.0.0.6 (2021-04-18)
EDGE COVID-19	[32]	github.com/LANL-Bioinformatics/EDGE/tree/SARS-CoV2	Illumina, Nanopore	Python, Perl, available as web tool	Docker (a single container with all dependencies)	2.4.0 (2020-12-03)
Galaxy-COVID19	[33]	galaxyproject.org/projects/covid19/workflows	Illumina (amplicon & mNGS), Nanopore	Galaxy	Access to a Galaxy instance	different pipelines & versions
HAVoC	[34]	bitbucket.org/auto_cov_pipeline/havoc	Illumina	Shell scripts	Conda, Mamba	no release
PipeCoV	[35]	github.com/alvesrco/pipecov	Illumina (amplicon & mNGS)	Shell scripts	Docker	no release

F1000Research 2023, 12:1091 Last updated: 13 NOV 2025

Table 2. Sequencing technologies used for SARS-CoV-2 sequencing of German and international samples. Data based on 1,2 million and 15,5 million whole-genome sequences submitted to the German DESH portal ("Deutscher Elektronischer Sequenzdaten-Hub") and to the GISAID database, respectively (accessed: May 22, 2023). To the best of our knowledge, we corrected typos such as *Nanonopore*, *Nasnopore*, and *Illunima* in the GISAID metadata and summarized the available terms into the broader categories shown in this table (*e.g.*, *NextSeq500* ⇒ *Illumina*). Entries we could not assign to one of the listed sequencing technologies were added to the *Other/Unknown* category. Please note that most German DESH sequences are also part of the GISAID data set. # – Number of sequences, % – Percentage of sequences on total data set.

Sequencing technology	# DESH	% DESH	# GISAID	% GISAID
Illumina	1,115,501	90.91	12,240,675	78.57
Nanopore	75,358	6.14	1,984,505	12.73
SMRT PacBio	0	0	573,214	3.67
Ion Torrent	29,450	2.40	349,001	2.24
MGI DNBSEQ	0	0	177,480	1.13
Sanger	0	0	57,820	0.37
Other/Unknown	6,727	0.54	195,833	1.25
Total	1,227,036	100	15,578,528	100

Here we present CoVpipe2, our pipeline engineered over nearly three years of pandemic genome sequencing that accurately reconstructs SARS-CoV-2 consensus sequences from Illumina short-read sequencing data, focusing on challenges associated with amplicon sequencing on a large scale. Besides implementation details, we also highlight pitfalls we discovered and solved during the pipeline development.

Methods

Implementation

CoVpipe2 is implemented using the workflow management system Nextflow[25] to achieve high reproducibility and performance on various platforms. The user can choose to use CoVpipe2 with Conda or Mamba support,[36] or containers (Docker,[37] Singularity[38]) to handle all software dependencies. The Conda/Mamba environments and the container images are preconfigured and have fixed versions of the incorporated tools. The precompiled Docker containers are stored on hub.docker.com/u/rkimf1. Containers and environments are downloaded and cached automatically when executing CoVpipe2. If needed, the Docker images can also be converted into Singularity images by the pipeline. CovPipe2 includes Nextclade[39] and pangolin[40] for lineage assignment. Both tools rely on their latest code and database versions. To address this, we implemented the --update option inspired by poreCov,[24] which triggers an update to the latest available version from anaconda.org or hub.docker.com/u/rkimf1, respectively. --update is disabled by default; the tool versions can also be pinned manually.

When CoVpipe2 is running on a high-performance computing (HPC) cluster (*e.g.*, SLURM, LSF) or in the cloud (*e.g.*, AWS, GCP, Azure), all resources (CPUs, RAM) are pre-configured for all processes but can be customized via a user-specific configuration file. We use complete version control for CoVpipe2, from the workflow itself (releases) to each tool to guarantee reproducible results. To this end, all Conda/Mamba environments and containers use fixed versions. In addition, each CoVpipe2 release can be invoked and executed individually, and the tool versions used during genome reconstruction and analysis are listed in Nextflow report files.

CoVpipe2 is publicly available under a GPL-3.0 license at github.com/rki-mf1/covpipe2, where details about the implementation and different executions of CoVpipe2 can be found. We use GitHub's CI for various pipeline tests, particularly a dry-run to check for integrity and an end-to-end test with special attention to the called variants to ensure continuous code quality and robust results.

Operation

As a minimal setup, Nextflow (minimal version 22.10.1, nextflow.io) and either Conda, Mamba, Docker, or Singularity need to be installed for CoVpipe2. Nextflow can be used on any POSIX-compatible system, *e.g.*, Linux, OS X, and on Windows via the Windows Subsystem for Linux (WSL). Nextflow requires Bash 3.2 (or later) and Java 11 (or later, up to 18) to be installed. Initial installation and further updates to the workflow and included tools can be performed with simple commands:

```
# install (or update) the pipeline
nextflow pull rki-mf1/CoVpipe2

# check available pipeline versions
nextflow info rki-mf1/CoVpipe2

# run a certain release version
nextflow run rki-mf1/CoVpipe2 -r v0.4.1 --help

# test the installation with local execution and Conda
nextflow run rki-mf1/CoVpipe2 -r v0.4.1 -profile local,conda,test --cores 2 --max_cores 4
```

The pipeline can be executed on various platforms controlled via the Nextflow `-profile` parameter, which makes it easily scalable, *e.g.*, for execution on an HPC. Each run profile is created via combining different *Executors* (local, slurm) and *Engines* (conda, mamba, docker, singularity); the default execution-engine combination (profile) is `-profile local,conda`.

An overview of the workflow is given in Figure 1. FASTQ files and a reference genome sequence (FASTA) are the minimum required pipeline inputs. If no reference genome sequence is provided, the SARS-CoV-2 index case reference genome with accession number MN908947.3 (identical to NC_045512.2) is used by default (as well as the corresponding annotation GFF), and then only FASTQ files are required. All files can be provided via file paths or defined via a comma-separated sample sheet (CSV); thus, CoVpipe2 can run in batch mode and analyze multiple samples in one run. Optionally, raw reads are checked for mixed samples with the tool LCS.[41] Next, raw reads are quality-filtered and trimmed using fastp[42] and optionally filtered taxonomically by Kraken2.[43] We provide an automated download of a precalculated Kraken2 database composed of SARS-CoV-2 and human genomes from Zenodo. However, the user is free to use a custom database. The reads are aligned to the reference genome using BWA,[44] and the genome coverage is calculated by BEDTools genomecov.[45] Primers can be optionally clipped after mapping with BAMclipper,[46] which is essential to avoid contaminating primer sequences in amplicon data. To locate the primer sequences, a browser extensible data paired-end (BEDPE) file containing all primer coordinates is required as input. If only a BED file is provided, CoVpipe2 can automatically convert it to a BEDPE file. The user can also choose from provided popular VarSkip (github.com/nebiolabs/VarSkip) and ARTIC primer schemes.[47] Next, FreeBayes[48] calls variants (default thresholds: minimum alternate count of 10, minimum alternate fraction of 0.1, and minimum coverage of 20), which are normalized with BCFtools norm.[49] Resulting variants are analyzed and annotated with SnpEff,[50] filtered by QUAL (default 10), INFO/

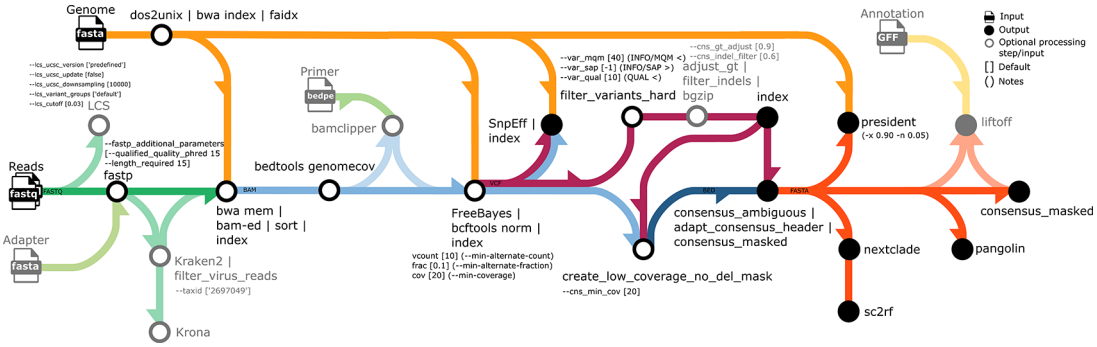


Figure 1. Overview of the CoVpipe2 workflow. The illustration shows all input (🖴) and output (●) files as well as optional processing steps and optional input (○). For each computational step, the used parameters and default values (in brackets [...]) are provided, as well as additional comments ((...)). The arrows connect all steps and are colored to distinguish different data processing steps: green – read (FASTQ) quality control and taxonomy filtering, yellow – reference genome (FASTA) and reference annotation (GFF) for lift-over, blue – mapping files (BAM) and low coverage filter (BED), purple – variant calls (VCF), orange – consensus sequence (FASTA). The icons and diagram components that make up the schematic figure were originally designed by James A. Fellow Yates and nf-core under a CCO license (public domain).

SAP (default disabled), and INFO/MQM (default 40) values, and optionally adjusted for the genotype (default enabled with minimum variant frequency 0.9). Thus, in the case of mixed variants, the predominant variant can be defined as a homozygous genotype if its frequency reaches a certain threshold. With these settings, alternate variants with a frequency of more than 90% are set as alternative nucleotides. Furthermore, CoVpipe2 filters out indels below a certain allele frequency, which is, by default, enabled with a minimum allele frequency of 0.6. We create a low coverage mask, without deletions, from the mapping file and the adjusted and filtered VCF file (default coverage cutoff 20), which is then used in the consensus calling with BCFtools consensus. We output an ambiguous consensus sequence with IUPAC characters[51] (RYMKSWHBVDN) and a masked one only containing ACTG and Ns. Then, PRESIDENT (GitHub) assesses the quality of each reconstructed genome via pairwise alignment to the SARS-CoV-2 index case (NC_045512.2) using pblat[52] with an identity threshold of 0.9 and N threshold of 0.05 per default. pangolin assigns a lineage and Nextclade annotates the mutations; the Nextclade alignment serves as input for sc2rf (original repository from github.com/lenaschimmel/sc2rf, with updates from github.com/ktmeaton/ncov-recombinant) for detection of recombinants. If an annotation file is provided, the reference annotation is mapped to the reconstructed genome with Liftoff.[53]

All results are summarized via an R Markdown template. The resulting HTML-based report summarizes different quality measures and mapping statistics for each input sample, thus allowing the user to spot low-performing samples even in extensive sequencing runs with many samples. A conditional notification warns the user if samples identified as negative controls (by matching the string 'NK') show high reference genome coverage (threshold over 0.2).

Common challenges in amplicon-based genome reconstruction and solutions in CoVpipe2

Here, we highlight some implementation decisions that prevent common pitfalls and thus improve the quality of reconstructed SARS-CoV-2 genomes. Although amplicon-based approaches are widely used, these technologies are associated with flaws and limitations that must be considered to ensure that the genotypes obtained, and thus the resulting genome sequences, are reliable.[13,54] While some of these implementations might seem obvious and easy to fix, we frequently observed specific errors in the vast amount of consensus genome sequences sent to us via the DESH system.

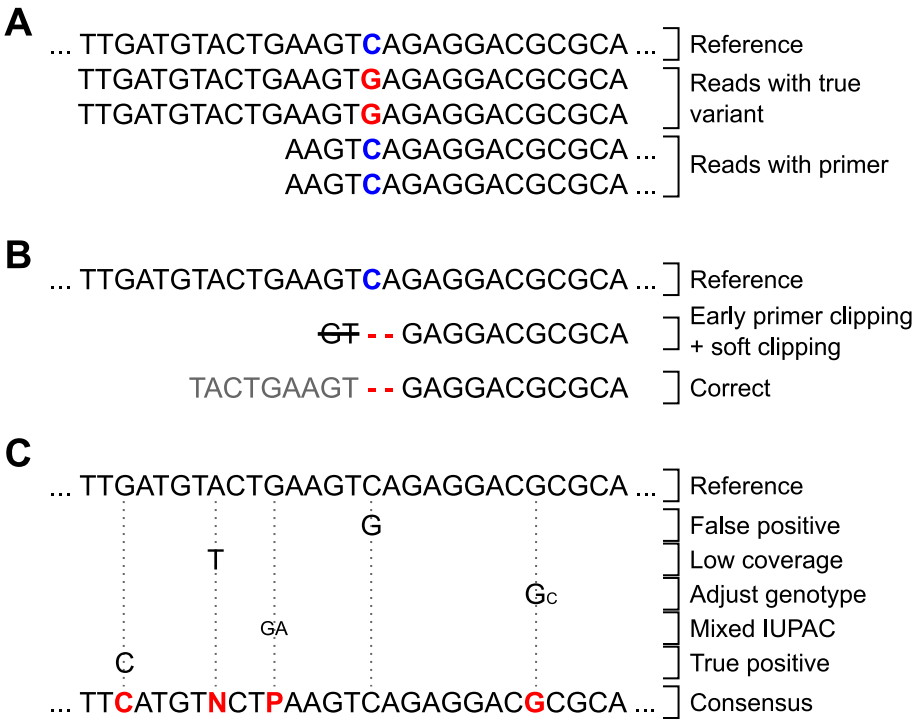


Figure 2. Exemplary cases that should be considered during genotyping. (A) Primer sequences need clipping to call true variants (red) and not mask them via reference bases (blue). (B) Early primer clipping may result in missed deletions due to algorithmic soft clipping (primer sequence in light gray). (C) Genotyping parameters must be carefully set to reliably call different variant cases and represent them in the final consensus. CoVpipe2 puts primer clipping after the mapping step to prevent B) and implements carefully chosen default parameters for robust genotyping also of mixed variants, C).

Accurate primer clipping to avoid dilution and edge effects Primer clipping is an essential step for amplicon sequencing data because primers are inherent to the reference sequence and can disguise true variants in the sample.[13] However, removing primers before the mapping step can result in unwanted edge effects.[55] For example, deletions located close to the end of amplicons may be soft-clipped by the mapping software and hence can not be called as variants subsequently (Figure 2). Therefore, primer clipping should be performed after mapping to prevent any soft clipping of variants close to the amplicon ends.

As an example and worst-case scenario, clipping primer sequences before mapping bears the risk of missing a critical deletion used to define the previous VOC B.1.1.7, namely deletion HV69/70 in the spike gene (S:H69-, S:V70-). We observed such a misclassification using Paragon CleanPlex amplicon-based sequencing. The kit uses primers similar to the ARTIC protocol V3, where the deletion S:HV69/70 is close to the end of an amplicon. If primer clipping is performed before mapping, the mapping tool might soft clip the amplicon end rather than opening a gap, which is more expensive than masking a few nucleotides (Figure 2). We checked 151,565 B.1.1.7 sequences obtained via DESH for the characteristic S:H69, S:V70 deletions and found that 139,891 (92.3%) included the deletion. The remaining sequences contained the deletion only partially or lacked it completely. It is unlikely that these B.1.1.7 sequences lost this characteristic deletion. Thus we assume that some of the reconstructed Alpha sequences sent to the RKI from different laboratories in Germany do not account for the described effect and thus miss the detection of this deletion. Unfortunately, we don't know which bioinformatics pipelines the submitting laboratories were using and can only assume an error because of missing or incorrect primer clipping. To avoid this problem, we shift primer clipping after the read mapping step in CoVpipe2; otherwise, a vital feature of an emerging virus lineage might be missed if mutations accumulate close to amplicon ends.

Genotype adjustment to exclude sporadic variant calls As an additional feature, we provide the option to adjust the genotype for sites where the vast majority of reads support a variant call, but the variant was called heterozygous. By default, the genotype of these locations is set to homozygous if a particular variant call is supported by 90% of the aligned reads.

Deletion-aware masking of low-coverage regions We ensure that only low-coverage positions that are not deletions are masked. Several pipelines implement a feature to mask low-coverage regions. However, deletions are basically genomic regions with no coverage, and if not appropriately implemented, a pipeline might accidentally mask deletions as low-coverage regions. To prevent this, CoVpipe2 creates a low coverage mask (default minimum coverage 20) from the BAM file with BEDtools genomecov. In the second step, BEDtools subtract removes all deleted sites from the low-coverage mask. Finally, the mask is used in the consensus generation with BCFtools consensus.

IUPAC consensus generation with indel filter CoVpipe2 generates different consensus sequences based on the IUPAC code. First, an explicit consensus is generated where all ambiguous sites and low-covered regions are hard-masked. Second, a consensus where only low-covered regions are hard-masked. The pipeline includes as much information as possible in the unambiguous consensus sequence by adding low-frequency variants with the respective IUPAC symbol. However, no symbol represents "indel or nucleotide", so indels are always incorporated into the consensus sequence. As a result, low-frequency or heterozygote indels, often false positives, can introduce frameshifts into the consensus sequence. We overcome this with an indel filter based on allele frequency before consensus generation. Thus, indels below a defined threshold (per default 0.6) are not incorporated in the consensus sequences but can still be looked up in the VCF file.

Additional features beyond genome reconstruction

Over time, we added features beyond genome reconstruction to CoVpipe2 to answer newly emerging questions during the pandemic. The modular design of our implementation makes this seamless.

Mixed infections and recombinants We included LCS for raw reads. LCS was originally developed for the SARS-CoV-2 lineage decomposition of mixed samples, such as wastewater or environmental samples.[41] In the context of amplicon-based SARS-CoV-2 sequencing, we use LCS results for indications for potential new recombinants, and possible mixed infections (co-infections with different SARS-CoV-2 lineages).

Further, we added sc2rf to detect potential new recombinants via screening the consensus genome sequences. Like other tools and scripts, sc2rf depends on up-to-date lineage and mutation information, specifically, on a manually curated JSON file. We switched to another repository (JSON GitHub project) than the original one that includes the sc2rf scripts because of more frequent updates on the JSON file.

Keeping up to date with lineage and clade assignments We implemented an update feature for pangolin and Nextclade. Both tools, especially pangolin, rely on the latest datasets for the lineage assignment of newly designated Pango lineages.[56] Depending on the selected engine, Conda/Mamba or container execution, CoVpipe2 checks for the latest available version from Anaconda or our DockerHub, respectively. The tool versions can also be pinned manually.

Prediction of mutation effects The called variants are annotated and classified based on predicted effects on annotated genes with SnpEff. SnpEff reports different effects, including synonymous or non-synonymous SNPs, start codon gains or losses, stop codon gains or losses; and classifies them based on their genomic locations. Lastly, CoVpipe2 uses Liftoff to generate an annotation for each sample if a reference annotation is provided.

Results

Selection of benchmark datasets

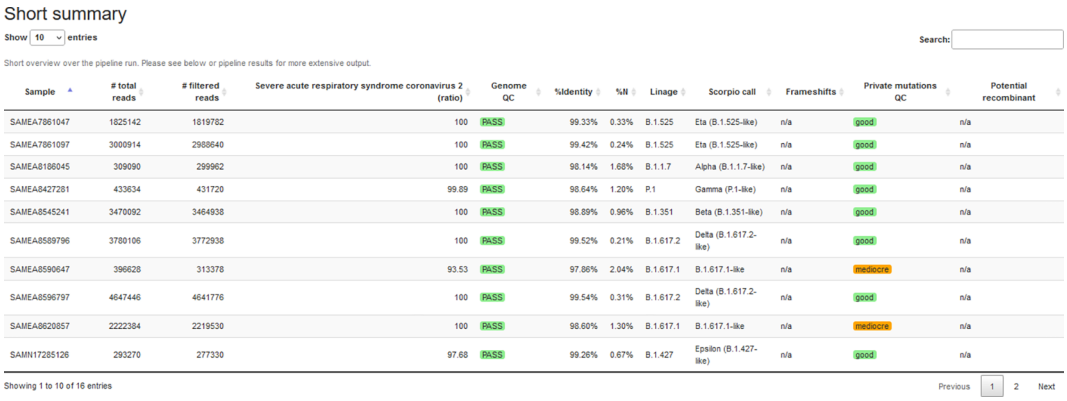
We compared the results of CoVpipe2 (v0.4.0) with publicly available benchmark datasets for SARS-CoV-2 surveillance[57] (GitHub CDC data). For our study, we selected from the available benchmark datasets all samples that were sequenced with the ARTIC V3 primer set (according to CDC data, release v0.7.2), ending up with in total 54 samples from three datasets:

- i) 16 samples from VOI/VOC lineages (dataset 4)
- ii) 33 samples from non-VOI/VOC lineages (dataset 5)
- iii) 5 samples from failed QC (dataset 6)

We stick to the naming scheme (dataset 4, 5, and 6) as declared in the original study.[57] For all 54 samples, we downloaded the raw reads from ENA with nf-core/fetchngs (v1.9).[58] For 49 samples, we downloaded the available consensus sequences from GISAID[1–3] (samples from dataset 4 and 5). We run CoVpipe2 with Singularity, species filtering (`--kraken`) and the latest pangolin and Nextclade versions at that point (containers: `rkimf1/pangolin:4.2-1.19--dec5681`, `rkimf1/nextclade2:2.13.1--ddb9e60`). We further examined the consensus sequences from dataset 4 and 5, comparing CoVpipe2 and GISAID sequences: We compared lineages information of the reconstructed genomes assigned by pangolin with the indicated lineages from CDC data. Note that the pangolin versions and datasets differ. Also, we run Nextclade (same container version as noted above) to compare the mutation profile, and MAFFT,[59] v7.515 (2023/Jan/15), for comparison on sequence level of the unambiguous IUPAC consensus of CoVpipe2.

Reporting

The HTML report consists of several sections and summarizes different results. The first table aggregates critical features for each sample: the number of reads, genome QC, the assigned lineage, and recombination potential, see Figure 3. Furthermore, read properties such as the number of bases and the length of reads (before/after trimming) are summarized. An optional table lists the species filtering results emitted by Kraken2.[43] The report summarizes reads mapped to the reference genome (number and fraction of input) and the median/standard deviation of fragment sizes, shown as a histogram plot. Genome-wide coverage plots allow users to observe low-coverage regions and potential amplicon


Short summary

Show 10 entries

Search:

Short overview over the pipeline run. Please see below or pipeline results for more extensive output.

Sample	# total reads	# filtered reads	Severe acute respiratory syndrome coronavirus 2 (ratio)	Genome QC	%Identity	%N	Linage	Scorpio call	Frameshifts	Private mutations QC	Potential recombinant
SAMEA7861047	1825142	1819782	100	PASS	99.33%	0.33%	B.1.525	Eta (B.1.525-like)	n/a	good	n/a
SAMEA7861097	3000914	2988640	100	PASS	99.42%	0.24%	B.1.525	Eta (B.1.525-like)	n/a	good	n/a
SAMEA8186045	309090	299962	100	PASS	98.14%	1.68%	B.1.1.7	Alpha (B.1.1.7-like)	n/a	good	n/a
SAMEA8427281	433634	431720	99.89	PASS	98.64%	1.20%	P.1	Gamma (P.1-like)	n/a	good	n/a
SAMEA8545241	3470092	3464938	100	PASS	98.89%	0.96%	B.1.351	Beta (B.1.351-like)	n/a	good	n/a
SAMEA8589796	3780106	3772938	100	PASS	99.52%	0.21%	B.1.617.2	Delta (B.1.617.2-like)	n/a	good	n/a
SAMEA8590647	396628	313378	93.53	PASS	97.86%	2.04%	B.1.617.1	B.1.617.1-like	n/a	mediocre	n/a
SAMEA8596797	4647446	4641776	100	PASS	99.54%	0.31%	B.1.617.2	Delta (B.1.617.2-like)	n/a	good	n/a
SAMEA8620857	2222384	2219530	100	PASS	98.60%	1.30%	B.1.617.1	B.1.617.1-like	n/a	mediocre	n/a
SAMN17285126	293270	277330	97.68	PASS	99.26%	0.67%	B.1.427	Epsilon (B.1.427-like)	n/a	good	n/a

Showing 1 to 10 of 16 entries

Previous 1 2 Next

Figure 3. Extract of CoVpipe2's summary report for dataset 4. The standalone HTML report summarizes different quality measures and tool results. The overview table can include a conditional notification for negative controls with high reference genome coverage (not shown in this example). All tables are searchable and sortable.

drop-outs to optimize primers. We summarize the output for all samples in different tables: i) genome quality from PRESIDENT output, ii) lineage assignments from pangolin output, and iii) variants in amino acid coordinates and detected frameshifts from Nextclade output.

Discussion

CovPipe2 reconstructs consensus genomes matching previously reported SARS-CoV-2 lineages

Here, we compare the results of CoVpipe2 against a selection of available benchmark datasets[57] and their respective consensus genome sequences available from GISAID.[1–3] We discuss the observed differences. No software is perfect, and CoVpipe2 may have problems with certain combinations of amplicon schemes and sequencing designs, leading to specific borderline cases in variant detection, which we also highlight and discuss. In addition, in conflicting cases, the real sequence often stays unknown until further sequencing efforts are performed. This makes continuous development and testing of bioinformatics pipelines all the more critical.

Dataset 4, VOI/VOC lineages

Lineages Despite the different pangolin tool versions and lineage definitions that changed over time, all pangolin lineages from CoVpipe2 match the corresponding lineages reported at github.com/CDCgov/datasets-sars-cov-2, see *Extended data*, Table S1.

Pairwise alignment The sequence identity ranges from 98.44 % to 99.79 % (including Ns) between the corresponding GISAID and CoVpipe2 genome pairs (*Extended data*, Table S2). Nine out of 16 sequences are identical when mismatches resulting from gaps are not considered. For one sample, the corresponding GISAID and CoVpipe2 sequences contain three ACGT-nucleotide mismatches. Seven out of 16 GISAID consensus sequences do not contain any Ns, possibly indicating that low coverage regions have been not masked. The respective reconstructed genomes from CoVpipe2 contain Ns located in the first or last 150 nucleotides of the genome sequences, thus masking low coverage regions (*Extended data*, Table S3). Due to tiled PCR amplicons, 5′ and 3′ ends of the genome usually have too little coverage or are not sequenced. The genome ends containing Ns seem to be trimmed in 14 GISAID genomes. Overall, the number of Ns in the GISAID and CoVpipe2 genomes is comparable, with CoVpipe2 genomes tending to have more Ns due to the low coverage filter resulting in Ns at genome ends. In addition, CoVpipe2 does not trim Ns from the genome ends and might be more conservative with its default settings, as positions below 20 X coverage (`--cns_min_cov`) will be masked with ambiguous N bases in the consensus sequence.

Mutations Although all genomes reconstructed from CoVpipe2 were assigned to the same lineage as previously reported (CDC data), there are minor differences in Nextclade's mutation profile, *Extended data*, Table S4. Three of 16 reconstructed genomes have one or two nucleotide substitutions more than the respective GISAID genome.

Dataset 5, non-VOI/VOC lineages

Lineages Pangolin lineages from CoVpipe2 exactly match the reported lineages at CDC data in 27 of 33 samples, see *Extended data*, Table S5. For five samples, the pangolin lineage based on the genome sequence reconstructed by Covpipe2 is the parent lineage of the expected sub-lineage. For example, the genome sequence of sample SAMN15919634 was assigned to the B.1.1 lineage after CoVpipe2 reconstruction, whereas the corresponding GISAID sequence is assigned to B.1.1.431 (CDC data). Since the lineage assignments of Nextclade exactly match for each CoVpip2-GISAD pair, the different pangolin and pangolin data versions used in CoVpipe2 and Xiaoli and Hagey *et al.*[57] are most likely the reason for this discrepancy. Different parameter thresholds can also lead to different lineage assignments. For example, the genome sequence of SAMN17571193 was assigned to B.1.1.450 by CoVpipe2 compared to B.1.1.391 in Xiaoli and Hagey *et al.*[57] The mutation profile differs by two additional SNPs (genome coordinates: C3037T and C3787T) constituting one mutation on amino acid level (ORF1a:D1962E) in the GISAID consensus sequence. Both positions (3037 and 3787) have a read coverage below 20, so they are not eligible for variant calling with default settings in CoVpipe2 and are masked with N. Therefore, this difference in lineage assignment is due to CoVpipe2's more restrictive approach to integrating the called variants into the consensus.

Pairwise alignment The pairwise sequence identity ranges from 95.96 % to 99.80 % (including Ns) between the GISAID and CoVpipe2's reconstructed genome sequences (*Extended data*, Table S6). Ignoring gap mismatches, 13 out of 33 sequences are identical. No sample contains ACGT mismatches. 18 out of 33 GISAID consensus sequences do not contain any Ns. Sixteen of the respective reconstructed genomes have Ns, all located in the first or last 150 nucleotides (*Extended data*, Table S7). Because of tiled PCR amplicons, the 5′ and 3′ ends typically have too little coverage or are not sequenced. The 5′ and 3′ genome ends containing Ns seem trimmed in 30 GISAID genomes. Overall, the number of Ns in

the GISAID and CoVpipe2 genomes is comparable, with CoVpipe2 genomes tending to have more Ns. CoVpipe2 does not trim Ns from the genome ends and might be more conservative with its default settings, as positions below 20 (`--cns_min_cov`) will be masked with N in the consensus sequence.

Mutations There are minor differences in Nextclade's mutation profile (*Extended data*, Table S8). Five of the 33 reconstructed genomes have one or two more nucleotide substitutions compared to the GISAID genome.

Dataset 6, samples failing quality control

For the five samples from the failed QC dataset by Xiaoli and Hagey *et al.*,[57] CoVpipe2 correctly labeled four samples with failed genome QC. Three samples contain at least one frameshift, whereas two samples, SAMN17486862 and SAMN17822806, were reconstructed by CoVpipe2 without frameshifts (*Extended data*, Table S9). The consensus sequence of sample SAMN17486862 passed QC according to CoVpipe2's genome QC criteria.

All the selected samples from this dataset have originally failed QC because of a VADR[60] alert number greater than one.[57] VADR is part of the TheiaCoV (formerly 'Titan') 1.4.4 pipeline,[61] which was used to analyze the samples in Xiaoli and Hagey *et al*.[57] Among other things, VADR considers frameshifts, which do not occur in two genomes reconstructed with CoVpipe2. However, one of these two consensus genomes (SAMN17822806) contains too many Ns to pass CoVpipe2's genome QC.

Limitations

The nature of a computational pipeline is that it is a chain of existing individual tools. Especially given the rapid evolution of SARS-CoV-2 during the pandemic, many reference-based tools rely on up-to-date databases and resources to reflect the current situation. For example, LCS depends on a variant marker table and user-defined variant groups. Similarly, sc2rf relies on a list of common variants for each lineage. In addition, Nextclade and pangolin periodically publish up-to-date datasets. Therefore, it is critical for a pipeline, especially in the context of a surveillance tool for rapidly evolving pathogens such as SARS-CoV-2, to allow for regular updates to the underlying data structures. While we have implemented some functionality to update tools such as Nextclade and pangolin automatically, this is not possible for all resources and can only be achieved through the continuous development and maintenance of a pipeline. Furthermore, our default settings may not fit all input data and must be selected carefully.

Finally, there must be enough good-quality input reads to reconstruct a genome successfully. In particular, with amplicon sequencing data, some regions might have a lower coverage due to amplicon dropouts. Thus, the genome as a whole can be reconstructed with acceptable quality. However, some essential mutations can be missing due to low coverage or variant-calling quality.

Conclusions

Accurate and high-throughput genotyping and genome reconstruction methods are central for monitoring SARS-CoV-2 transmission and evolution. CoVpipe2 provides a fully automated, flexible, modular, and reproducible workflow for reference-based variant calling and genome reconstruction from short-read sequencing data, emphasizing amplicon-based sequencing schemes. Due to the implementation in the Nextflow framework, the setup and automatic installation of the required tools and dependencies is simple and allows the execution on different computing platforms. The comparison with a benchmark dataset showed comparable results where differences could be pinned down to different parameters, filtering thresholds, and tool versions used for lineage assignments. The pipeline is optimized for SARS-CoV-2 amplicon data but can also be used for other viruses and whole-genome sequencing protocols. Amplicon-optimized default parameters and the ability to customize critical parameters, combined with comprehensive reporting, ensure the quality of reported genomes and prevent the inclusion of low-quality sequences in downstream analyses and public repositories. Furthermore, CoVpipe2 will form the basis for additional genomic surveillance programs at the Public Health Institute of Germany that will extend to other viruses.

Data availability

Underlying data

All SRA accession IDs of raw reads and GISAID IDs of consensus sequences are listed at[57] and github.com/CDCgov/datasets-sars-cov-2; and in our Open Science Framework repository osf.io/26hyx (*Extended data*: https://doi.org/10.17605/OSF.IO/MJ6EQ).[62] We used ARTIC V3 samples from dataset 4 (VOI/VOC lineages, 16 samples), dataset 5 (non-VOI/VOC lineages, 33 samples), and dataset 6 (failedQC, 5 samples) from the original study of Xiaoli and Hagey *et al*.[57]

Extended data

Open Science Framework. Lessons learned: overcoming common challenges in reconstructing the SARS-CoV-2 genome from short-read sequencing data via CoVpipe2, https://doi.org/10.17605/OSF.IO/MJ6EQ.[62]

This project contains the following extended data:

- Data folder Accession ID. (SRA and GISAID accession ID lists)
- Data folder Comparison. (Scripts and results of the benchmark comparison)
- Data folder CoVpipe2 results. (Results of CoVpipe2 for each benchmark dataset)
- Data folder Extended data. (Supplementary tables S1-S9)

Software availability

- Software and source code available from: https://github.com/rki-mf1/covpipe2
- Archived source code at time of publication: https://doi.org/10.5281/zenodo.8082695.[63]
- License: GNU General Public License v3.0 (GPL3)

Acknowledgements

We gratefully acknowledge all data contributors, i.e., the Authors and their originating laboratories responsible for obtaining the specimens, and their submitting laboratories for generating the genetic sequence and metadata and sharing via the GISAID Initiative, on which this research is based. We also thank all German Electronic Sequence Data Hub contributors, the IMS-SC2 laboratory network, and all data providers, i.e., the originating laboratories responsible for obtaining samples and the submitting laboratories where genetic sequence data were generated and shared. We are incredibly grateful to Petra Kurzendörfer, Tanja Pilz, Aleksandar Radonić, and Aaron Houterman for outstanding sequencing support. We thank Marianne Wedde for manually checking and validating consensus sequences and called variants and Thorsten Wolff and Max von Kleist for fruitful discussions. We thank Ben Wulf, Fabian Rost, and Alexander Seitz – early users of the first version of CoVpipe for valuable feedback and reporting issues.

References

1. Shu Y, McCauley J: **GISAID: Global initiative on sharing all influenza data – from vision to reality.** *Eurosurveillance.* 2017; **22**(13): 30494.
2. Elbe S, Buckland-Merrett G: **Data, disease and diplomacy: Gisaid's innovative contribution to global health.** *Global Chall.* 2017; **1**(1): 33–46.
PubMed Abstract | Publisher Full Text | Free Full Text
3. Khare S, Gurry C, Freitas L, *et al.*: **GISAID Core Curation Team, and Sebastian Maurer-Stroh. Gisaid's role in pandemic response.** 2021. 2096-7071.
Reference Source
4. Harrison PW, Lopez R, Rahman N, *et al.*: **The COVID-19 Data Portal: accelerating SARS-CoV-2 and COVID-19 research through rapid open access data sharing.** *Nucleic Acids Res.* 2021; **49**(W1): W619–W623.
PubMed Abstract | Publisher Full Text | Free Full Text
5. Robishaw JD, Alter SM, Solano JJ, *et al.*: **Genomic surveillance to combat COVID-19: challenges and opportunities.** *Lancet Microbe.* 2021; **2**(9): e481–e484.
PubMed Abstract | Publisher Full Text | Free Full Text
6. Julianna LeMieux Genetic Engineering & Biotechnology News: **All Aboard the Genome Express: Is a new generation of DNA sequencing technology about to hit the fast track?** last accessed December 01, 2022.
Reference Source
7. Hufsky F, Lamkiewicz K, Almeida A, *et al.*: **Computational strategies to combat COVID-19: useful tools to accelerate SARS-CoV-2 and coronavirus research.** *Brief. Bioinform.* 2021; **22**(2): 642–663.
PubMed Abstract | Publisher Full Text | Free Full Text
8. Djin Ye O, Hölzer M, Paraskevopoulou S, *et al.*: **Advancing precision vaccinology by molecular and genomic surveillance of Severe Acute Respiratory Syndrome Coronavirus 2 in Germany, 2021.** *Clin. Infect. Dis.* 2022; **75**(Supplement_1): S110–S120.
Publisher Full Text
9. Baumgarte S, Hartkopf F, Hölzer M, *et al.*: **Investigation of a limited but explosive COVID-19 outbreak in a German secondary school.** *Viruses.* 2022; **14**(1): 87.
PubMed Abstract | Publisher Full Text | Free Full Text
10. Loss J, Wurm J, Varnaccia G, *et al.*: **Transmission of sars-cov-2 among children and staff in german daycare centres.** *Epidemiol. Infect.* 2022; **150**: e141.
PubMed Abstract | Publisher Full Text | Free Full Text
11. De Maio N, Walker C, Borges R, *et al.*: **Issues with SARS-CoV-2 sequencing data.** last accessed November 25, 2022.
Reference Source
12. Brito AF, Semenova E, Dudas G, *et al.*: **Global disparities in SARS-CoV-2 genomic surveillance.** *Nat. Commun.* 2022; **13**(1): 1–13.
13. Connor R, Yarmosh DA, Maier W, *et al.*: **Towards increased accuracy and reproducibility in SARS-CoV-2 next generation sequence analysis for public health surveillance.** *bioRxiv.* 2022.
14. Chiara M, D'Erchia AM, Gissi C, *et al.*: **Next generation sequencing of SARS-CoV-2 genomes: challenges, applications and opportunities.** *Brief. Bioinform.* 2021; **22**(2): 616–630.
PubMed Abstract | Publisher Full Text | Free Full Text

15. Robert Koch Institute: **Deutscher Elektronischer Sequenzdaten-Hub (DESH).** last accessed November 25, 2022.
Reference Source

16. Grubaugh ND, Gangavarapu K, Quick J, *et al.*: **An amplicon-based sequencing framework for accurately measuring intrahost virus diversity using PrimalSeq and iVar.** *Genome Biol.* 2019; **20**(1): 1–19.
Publisher Full Text

17. Resende PC, Motta FC, Roy S, *et al.*: **SARS-CoV-2 genomes recovered by long amplicon tiling multiplex approach using nanopore sequencing and applicable to other sequencing platforms.** *BioRxiv.* 2020.

18. Brinkmann A, Ulm S-L, Uddin S, *et al.*: **Amplicov: Rapid whole-genome sequencing using multiplex PCR amplification and real-time Oxford Nanopore MinION sequencing enables rapid variant identification of SARS-CoV-2.** *Front. Microbiol.* 2021; **12**: 1703.
Publisher Full Text

19. Hilaire BGS, Durand NC, Mitra N, *et al.*: **A rapid, low cost, and highly sensitive SARS-CoV-2 diagnostic based on whole genome sequencing.** *BioRxiv.* 2020.

20. Gohl DM, Garbe J, Grady P, *et al.*: **A rapid, cost-effective tailed amplicon method for sequencing SARS-CoV-2.** *BMC Genomics.* 2020; **21**(1): 1–10.
Publisher Full Text

21. Hadfield J, Megill C, Bell SM, *et al.*: **Nextstrain: real-time tracking of pathogen evolution.** *Bioinformatics.* 2018; **34**(23): 4121–4123.
PubMed Abstract | Publisher Full Text | Free Full Text

22. Tshiabuila D, Giandhari J, Pillay S, *et al.*: **Comparison of SARS-CoV-2 sequencing using the ONT GridION and the Illumina MiSeq.** *BMC Genomics.* 2022; **23**(1): 1–17.
Publisher Full Text

23. Luo J, Meng Z, Xingyu X, *et al.*: **Systematic benchmarking of nanopore Q20+ kit in SARS-CoV-2 whole genome sequencing.** *Front. Microbiol.* 2022; 4059.

24. Brandt C, Krautwurst S, Spott R, *et al.*: **poreCov – an easy to use, fast, and robust workflow for SARS-CoV-2 genome reconstruction via nanopore sequencing.** *Front. Genet.* 2021; 1397.

25. Di Tommaso P, Chatzou M, Floden EW, *et al.*: **Nextflow enables reproducible computational workflows.** *Nat. Biotechnol.* 2017; **35**(4): 316–319.
PubMed Abstract | Publisher Full Text

26. Köster J, Rahmann S: **Snakemake – a scalable bioinformatics workflow engine.** *Bioinformatics.* 2012; **28**(19): 2520–2522.
PubMed Abstract | Publisher Full Text

27. Patel H, Monzón S, Varona S, *et al.*: **nf-core/viralrecon: nf-core/viralrecon v2.6.0 - Rhodium Raccoon.** March 2023.
Publisher Full Text

28. Nasir JA, Kozak RA, Aftanas P, *et al.*: **A comparison of whole genome sequencing of SARS-CoV-2 using amplicon-based sequencing, random hexamers, and bait capture.** *Viruses.* 2020; **12**(8): 895.
PubMed Abstract | Publisher Full Text | Free Full Text

29. Posada-Céspedes S, Seifert D, Topolsky I, *et al.*: **V-pipe: a computational pipeline for assessing viral genetic diversity from high-throughput data.** *Bioinformatics.* 2021; **37**(12): 1673–1680.
PubMed Abstract | Publisher Full Text | Free Full Text

30. Kim K, Park K, Lee S, *et al.*: **Virpipe: an easy and robust pipeline for detecting customized viral genomes obtained by nanopore sequencing.** *Bioinformatics.* 2023; **39**: btad293.
PubMed Abstract | Publisher Full Text | Free Full Text

31. Dezordi FZ, da Silva Neto AM, de Lima Campos T, *et al.*: **Viralflow: a versatile automated workflow for sars-cov-2 genome assembly, lineage assignment, mutations and intrahost variant detection.** *Viruses.* 2022; **14**(2): 217.
PubMed Abstract | Publisher Full Text | Free Full Text

32. Lo C-C, Shakya M, Connor R, *et al.*: **EDGE COVID-19: a web platform to generate submission-ready genomes from SARS-CoV-2 sequencing efforts.** *Bioinformatics.* 2022; **38**(10): 2700–2704.
PubMed Abstract | Publisher Full Text | Free Full Text

33. Maier W, Bray S, van den Beek M, *et al.*: **Ready-to-use public infrastructure for global SARS-CoV-2 monitoring.** *Nat. Biotechnol.* 2021; **39**(10): 1178–1179.
PubMed Abstract | Publisher Full Text | Free Full Text

34. Nguyen PTT, Plyusnin I, Sironen T, *et al.*: **HAVoC, a bioinformatic pipeline for reference-based consensus assembly and lineage assignment for SARS-CoV-2 sequences.** *BMC Bioinformat.* 2021; **22**(1): 1–8.
Publisher Full Text

35. Oliveira RRM, Negri TC, Nunes G, *et al.*: **PipeCov: a pipeline for SARS-CoV-2 genome assembly, annotation and variant identification.** *PeerJ.* 2022; **10**: e13300.
PubMed Abstract | Publisher Full Text | Free Full Text

36. Grüning B, Dale R, Sjödin A, *et al.*: **Bioconda: sustainable and comprehensive software distribution for the life sciences.** *Nat. Methods.* 2018; **15**(7): 475–476.
Publisher Full Text

37. Boettiger C: **An introduction to Docker for reproducible research.** *Oper. Syst. Rev.* 2015; **49**(1): 71–79.
Publisher Full Text

38. Kurtzer GM, Sochat V, Bauer MW: **Singularity: Scientific containers for mobility of compute.** *PLoS One.* 2017; **12**(5): e0177459.
PubMed Abstract | Publisher Full Text | Free Full Text

39. Aksamentov I, Roemer C, Hodcroft EB, *et al.*: **Nextclade: clade assignment, mutation calling and quality control for viral genomes.** *J. Open Source Softw.* 2021; **6**(67): 3773.
Publisher Full Text

40. O'Toole Á, Scher E, Underwood A, *et al.*: **Assignment of epidemiological lineages in an emerging pandemic using the pangolin tool.** *Virus Evol.* 2021; **7**(2): veab064.
PubMed Abstract | Publisher Full Text | Free Full Text

41. Valieris R, Drummond RD, Defelicibus A, *et al.*: **A mixture model for determining SARS-Cov-2 variant composition in pooled samples.** *Bioinformatics.* 2022; **38**(7): 1809–1815.
PubMed Abstract | Publisher Full Text

42. Chen S, Zhou Y, Chen Y, *et al.*: **fastp: an ultra-fast all-in-one FASTQ preprocessor.** *Bioinformatics.* 2018; **34**(17): i884–i890.
PubMed Abstract | Publisher Full Text | Free Full Text

43. Wood DE, Jennifer L, Langmead B: **Improved metagenomic analysis with Kraken 2.** *Genome Biol.* 2019; **20**: 1–13.
Publisher Full Text

44. Li H: **Aligning sequence reads, clone sequences and assembly contigs with bwa-mem.** 2013.

45. Quinlan AR: **BEDTools: the Swiss-army tool for genome feature analysis.** *Curr. Protoc. Bioinformat.* 2014; **47**(1): 11–12.

46. Chun Hang A, Ho DN, Kwong A, *et al.*: **BAMClipper: removing primers from alignments to minimize false-negative mutations in amplicon next-generation sequencing.** *Sci. Rep.* 2017; **7**(1): 1–7.

47. Tyson JR, James P, Stoddart D, *et al.*: **Improvements to the ARTIC multiplex PCR method for SARS-CoV-2 genome sequencing using nanopore.** *BioRxiv.* 2020.

48. Garrison E, Marth G: **Haplotype-based variant detection from short-read sequencing.** *arXiv preprint arXiv:1207.3907.* 2012.

49. Danecek P, Bonfield JK, Liddle J, *et al.*: **Twelve years of SAMtools and BCFtools.** *Gigascience.* 2021; **10**(2): giab008.
PubMed Abstract | Publisher Full Text | Free Full Text

50. Cingolani P, Platts A, Coon M, *et al.*: **A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of *Drosophila melanogaster* strain w1118; iso-2; iso-3.** *Fly.* 2012; **6**(2): 80–92.
PubMed Abstract | Publisher Full Text | Free Full Text

51. Cornish-Bowden A: **Nomenclature for incompletely specified bases in nucleic acid sequences: recommendations 1984.** *Nucleic Acids Res.* 1985; **13**(9): 3021–3030.
PubMed Abstract | Publisher Full Text | Free Full Text

52. Wang M, Kong L: **pblat: a multithread blat algorithm speeding up aligning sequences to genomes.** *BMC Bioinformat.* 2019; **20**(1): 1–4.

53. Shumate A, Salzberg SL: **Liftoff: accurate mapping of gene annotations.** *Bioinformatics.* 2021; **37**(12): 1639–1643.
PubMed Abstract | Publisher Full Text | Free Full Text

54. Kubik S, Marques AC, Xing X, *et al.*: **Recommendations for accurate genotyping of SARS-CoV-2 using amplicon-based sequencing of clinical samples.** *Clin. Microbiol. Infect.* 2021; **27**(7): 1036.e1–1036.e8.
Publisher Full Text

55. Satya RV, DiCarlo J: **Edge effects in calling variants from targeted amplicon sequencing.** *BMC Genomics.* 2014; **15**(1): 1073–1077.
Publisher Full Text

56. Rambaut A, Holmes EC, O'Toole Á, *et al.*: **A dynamic nomenclature proposal for sars-cov-2 lineages to assist genomic epidemiology.** *Nat. Microbiol.* 2020; **5**(11): 1403–1407.
PubMed Abstract | Publisher Full Text | Free Full Text

57. Xiaoli L, Hagey JV, Park DJ, *et al.*: **Benchmark datasets for sars-cov-2 surveillance bioinformatics.** *PeerJ.* 2022; **10**: e13821.
Publisher Full Text

58. Ewels PA, Peltzer A, Fillinger S, *et al.*: **The nf-core framework for community-curated bioinformatics pipelines.** *Nat. Biotechnol.* 2020; **38**(3): 276–278.
PubMed Abstract | Publisher Full Text

59. Katoh K, Standley DM: **MAFFT multiple sequence alignment software version 7: improvements in performance and usability.** *Mol. Biol. Evol.* 2013; **30**(4): 772–780.
PubMed Abstract | Publisher Full Text | Free Full Text

60. Schäffer AA, Hatcher EL, Yankie L, *et al.*: **Vadr: validation and annotation of virus sequence submissions to genbank.** *BMC Bioinformat.* 2020; **21**: 1–23.

61. Libuit K, RA P III, Ambrosio F, *et al.*: **Public health viral genomics: bioinformatics workflows for genomic characterization, submission preparation, and genomic epidemiology of viral pathogens, especially the sars-cov-2 virus.** 2022.

62. Lataretu M, Hölzer M: Lessons learned: overcoming common challenges in reconstructing the SARS-CoV-2 genome from short-read sequencing data via CoVpipe2. [Dataset]. 2023, June 27.
Publisher Full Text

63. MarieLataretu: rki-mf1/CoVpipe2: Version v0.4.3 (v0.4.3). *Zenodo.* 2023.
Publisher Full Text

Open Peer Review

Current Peer Review Status: ✔ ?

Version 1

Reviewer Report 17 January 2024

https://doi.org/10.5256/f1000research.149827.r233668

?

Sondes Haddad-Boubaker
University of Tunis El Manar, Tunis, Tunisia

This paper presents a comprehensive bioinformatics workflow designed for the reconstruction of SARS-CoV-2 genomes using short-read sequencing data. The workflow description offers a detailed overview of the processes involved; however, there are opportunities for improvement to enhance the overall quality of the paper.

1-Organization:
While the overall structure is clear, introducing numbering at both the heading and sub-heading levels would enhance clarity and facilitate better distinction between various steps in the workflow.

2-Technical Terminology:
Given the diverse audience, including virologists and other biologists, it is crucial to ensure accessibility by providing clear definitions for all technical terms and acronyms. This will make the paper more easy to readers who may not be familiar with specific bioinformatics or genomics terminology.

3- References and Sources:
Include proper references and sources for the tools and databases mentioned by providing specific URLs for easy access to external resources.

4-Methods and Results:
- Introduce a dedicated "Pipeline Evaluation" subsection in both the "Methods" and "Results" sections and change the abstract section accordingly.
-Include details on the investigated sequences, with a thorough description of sample types, cycle threshold (ct) values, and sample categories.
-Consider expanding the evaluation by incorporating additional samples/sequences, especially failed sequences from samples with varying ct values (especially high ct values). The Investigation of the contribution of the pipeline in obtaining reliable sequences from samples with high ct values may be helpfull for virologist who are dealing with such challenges especially in samples

obtained from long-term excretors.
- Please Illustrate "common challenges" with a graph or figure for better presentation and understanding.
5- Discussion:
- Emphasize and discuss the added value of the pipeline in obtaining reliable sequences from samples with high ct values. Compare the results obtained using this pipeline with those from other existing pipelines to highlight its superiority.
- Provide detailed insights into how this pipeline can be applied to study other viruses,

Is the rationale for developing the new software tool clearly explained?
Yes

Is the description of the software tool technically sound?
Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?
Partly

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?
Partly

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?
Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Virology

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 10 Apr 2024

Martin Hölzer

(I) - ***Organization:***
[1] Reviewer Concern: *While the overall structure is clear, introducing numbering at both the heading and sub-heading levels would enhance clarity and facilitate better distinction between various steps in the workflow.*

Author Response: Thanks for the suggestion. Numbering (sub)sections helps to structure the text better and distinguish which parts belong together semantically. However, there is little we can do about it, as this is the journal's style. Nevertheless, we will ask the

editor/typesetting team if that's possible.

(II) - ***Technical Terminology:***

[2] Reviewer Concern: *Given the diverse audience, including virologists and other biologists, it is crucial to ensure accessibility by providing clear definitions for all technical terms and acronyms. This will make the paper more easy to readers who may not be familiar with specific bioinformatics or genomics terminology.*

Author Response: Thanks for the comment. We agreed and added a list of Abbreviations to the manuscript to make it easier for readers to follow the story.

- COVID-19 - Coronavirus disease 2019
- SARS-CoV-2 - Severe acute respiratory syndrome coronavirus 2
- GPL3 license - GNU General Public License
- GISAID - Global Initiative on Sharing All Influenza Data
- EBI - European Bioinformatics Institute
- EMBL - European Molecular Biology Laboratory
- RKI - Robert Koch Institute
- CorSurV - Coronavirus Surveillance Verordnung (eng., Coronavirus Surveillance Regulation)
- DESH - Deutscher Elektronischer Sequenzdaten-Hub (eng., German Electronic Sequence Data Hub)
- IMS-SC2 - Integrated Molecular Surveillance for SARS-CoV-2
- ONT - Oxford Nanopore Technologies
- NGS - Next-Generation Sequencing
- HPC - High-Performance Computing
- WSL - Windows Subsystem for Linux
- CSV file - Comma-Separated Values file
- GFF file - General Feature Format file
- BEDPE file - Browser Extensible Data Paired-End file
- VCF file - Variant Call Format file
- HTML - Hypertext Markup Language
- BAM file - Binary Alignment and Map file
- BED file - Browser Extensible Data file
- CCO license - Creative Commons Zero license
- VOC - Variants of Concern
- VOI - Variants of Interest
- IUPAC - International Union of Pure and Applied Chemistry
- indel - Insertion/Deletion Variant
- JSON file - JavaScript Object Notation file
- CDC - Centers for Disease Control and Prevention

- ENA - European Nucleotide Archive
- QC - Quality Control
- PCR - Polymerase Chain Reaction
- ID - Identifier
- SRA - Sequence Read Archive

(III) - ***References and Sources***
[3] **Reviewer Concern:** *Include proper references and sources for the tools and databases mentioned by providing specific URLs for easy access to external resources.*

Author Response: We agree that it's crucial to acknowledge all tools and resources properly. When there is an original publication for a tool or database, we cite the publication. If not, we cite the code repository or the URL to the resource (such as GitHub or Zenodo). We carefully checked the text again and added citations/URLs if they were missing and necessary. For example, Rev #1 also commented that the specific URL to the custom Kraken 2 database on Zenodo was missing. We added that.

(IV) - ***Methods and Results***
[4] **Reviewer Concern**: *Introduce a dedicated "Pipeline Evaluation" subsection in both the "Methods" and "Results" sections and change the abstract section accordingly.*

Author Response: We fully agree that presenting the implementation of a pipeline together with its evaluation with additional analysis can be confusing. So what are the "Methods" and the "Results" parts, then? This is often a problem when simultaneously presenting and evaluating a new software implementation. But we also need to adhere to the style guidelines for journals.
We wrote a "Software Tool Articles" and have to follow this structure: https://f1000research.com/for-authors/article-guidelines/software-tool-articles.
As you can see in these guidelines, "Software Tool Articles typically contain the following sections: Introduction, Methods, Results (Optional), Use Cases (Optional), Conclusions/Discussion." In the first version, we skipped the "Use Cases" section to discuss our example data sets for pipeline evaluation directly in the "Results" section. However, thanks to your comment, we believe that also skipping the "Results" section entirely makes our manuscript clearer. We deleted the "Results" section and added the subsections "Selection of benchmark datasets and pipeline evaluation" and "Reporting" at the end of the "Methods". We changed the subsection "Selection of benchmark datasets" to "Selection of benchmark datasets and pipeline evaluation" to include your suggestion.
We think that the structure is now clearer because we first describe in the "Methods" the implementation of the pipeline and how to operate it, according to the journal guidelines for "Software Tool Articles": The Methods should "Include a subsection on **Implementation** describing how the tool works and any relevant technical details required for implementation; and a subsection on **Operation**, which should include the minimal system requirements needed to run the software and an overview of the workflow."

Then, we describe some specific implementation decisions followed by the example data sets for pipeline evaluation and, finally, the report structure we implemented.
According to the journal guideline for "Software Tool Articles": "Abstracts are structured into Background, Methods, Results, and Conclusions", thus, we can not change sections in the abstract.
Although we generally agree that different subheadings would help follow the story, we must also stick to the journal guidelines.
---

[5] **Reviewer Concern**: *Include details on the investigated sequences, with a thorough description of sample types, cycle threshold (ct) values, and sample categories.*

Author Response: We only use publicly available data sets and reference the original sources (publication, GitHub, and ENA repositories). We suggest that readers should refer to the original sources for further details. However, the description of the benchmark datasets is now also part of the "Methods". Here, we describe:
"We compared the results of CoVpipe2 (v0.4.0) with publicly available benchmark datasets for SARS-CoV-2 surveillance 57 (GitHub CDC data)."
[57] Xiaoli L, Hagey JV, Park DJ, et al.: Benchmark datasets for sars-cov-2 surveillance bioinformatics. PeerJ. 2022;10:e13821. 10.7717/peerj.13821
We do not want to mirror the details of the benchmark data sets that are described in the original sources. In addition, we can not provide additional information, such as Ct values, because, to the best of our knowledge, this information is not available in the original publication or in the data source (GitHub, ENA).
---

[6] **Reviewer Concern**: *Consider expanding the evaluation by incorporating additional samples/sequences, especially failed sequences from samples with varying ct values (especially high ct values). The Investigation of the contribution of the pipeline in obtaining reliable sequences from samples with high ct values may be helpful for virologist who are dealing with such challenges especially in samples obtained from long-term excretors.*

Author Response: Thanks for the comment. We agree that investigating challenging samples is especially interesting for users of CoVpipe2. As described above (Q [5]), we selected a publicly available benchmark dataset for SARS-CoV-2 surveillance (Xiaoli et al. 2022) to compare our results directly with previous calculations. In addition, this data set also includes difficult samples that should not withstand automatic quality control (QC) and could mimic high Ct values.
CoVpipe2 was developed as a robust and standardized workflow to support genome reconstruction in genomic surveillance programs. Thus, our main goal in developing CoVpipe2 was to provide a robust bioinformatics pipeline for short-read sequencing data that recognizes important mutations with decent allele frequency and automatically identifies and masks ambiguous positions. We implemented parameters (20X coverage to consider a position for variant calling, 90% ACGT nucleotide identity to the reference) to discover low-quality samples that might originate from high Ct values. Running the pipeline on amplicon sequencing data from samples with high Ct can result in "read stacks" with high sequence depth for certain well-amplified amplicons, but it could also lead to low

horizontal genome coverage due to low input RNA quantity. CoVpipe2 will report such samples as “failed” in the QC report. Thus, only samples with a decent vertical (sequencing depth) and horizontal genome coverage should be used for downstream genomic surveillance and trustworthy lineage assignment.
Nevertheless, CoVpipe2 also reports the full intermediate results, such as BAM files and unfiltered VCF files. Experienced users can investigate all variant calls and their respective allele frequencies - also for QC-failed samples. Thus, it is also possible to investigate mixed variant calls (co-infection, recombinants) and low-frequency variants with the help of CoVpipe2. However, for routine genomic surveillance applications, such challenging samples will be automatically flagged as QC-failed in the pipeline, supporting non-expert users in decision-making and selecting suitable samples for surveillance. Obtaining reliable consensus genomes from high Ct samples is generally difficult. In our experience, it is better to flag such samples with a warning and inform the user that those are of lower quality and probably not suited for further downstream analysis. Thus, CoVpipe2 helps virologists identify such challenging samples so that they can be selected for re-sequencing or exclusion from downstream analysis.

---

[7] **Reviewer Concern**: *Please Illustrate "common challenges" with a graph or figure for better presentation and understanding.*

Author Response: Here, we present a bioinformatics pipeline with the specific objective of reconstructing robust SARS-CoV-2 consensus genomes from patient samples and short-read data, which is also reflected in the title of our paper. We present CoVpipe2 as a solution to overcome such challenges in reconstructing robust SARS-CoV-2 genomes from short-read (amplicon) data. In Figure 2, we explicitly illustrate common challenges regarding variant calling, which is one of the main obstacles in many reference-based virus bioinformatics pipelines, and where we specifically integrated solutions in CoVpipe2 to overcome such challenges. Besides the dedicated figure for variant calling challenges, we examine other relevant challenges in the context of the CoVpipe2 implementation, such as amplicon drop-outs, in the text.
For a general overview, we think that common challenges in the context of amplicon sequencing and virus bioinformatics need to be more broadly addressed in dedicated benchmark studies such as those already available from Beerenwinkel et al. 2012; Murray et al. 2015; Fitzpatrick et al. 2022; Liu et al. 2021.

(V) - ***Discussion:***
[8] **Reviewer Concern**: *Emphasize and discuss the added value of the pipeline in obtaining reliable sequences from samples with high ct values. Compare the results obtained using this pipeline with those from other existing pipelines to highlight its superiority.*

Author Response: As described in [6], we developed CoVpipe2 as a robust and modular surveillance pipeline focusing on amplification protocols and short-read data. Thus, for samples with high Ct values and where amplification can not yield enough output, CoVpipe2 will mark them as “failed” in the reporting. High Ct samples will usually result in regions

(amplicons) with low coverage. Such regions are then automatically masked by “N” bases in the final consensus. We don't think a bioinformatics pipeline should construct any “reliable” consensus genome sequence when the data is insufficient. Thus, it is more important to identify such low-quality samples and flag them with a user warning. Our filtering and reporting aims to fit the needs of large-scale surveillance programs with detailed QC information and provide a quick overview of sample results, to identify such challenging samples easily. Thus, CoVpipe2 helps virologists to identify such problematic samples so that they can be selected for re-sequencing or excluded from downstream analysis.
Regarding the comparison to other pipelines, we implicitly did that by selecting the test data sets. Those come from another independent benchmark study (Xiaoli et al. 2022), and we compare our CoVpipe2 results against those from the original benchmark paper. The original authors wrote in their publication:
> The datasets presented here were generated to help public health laboratories build sequencing and bioinformatics capacity, benchmark different workflows and pipelines, and calibrate QC thresholds to ensure sequencing quality.
All available pipelines (Tab. 1 in manuscript) excel in various properties. While some strive to have high detection rates for minor variants for research settings, CoVpipe2 was developed to be easily extendable and adjustable to new requirements in surveillance or other viruses [see 9]. Thus, we would like to stick to our decision of utilizing a publicly available and carefully constructed, independent benchmark data set instead of including more samples and pipelines. Our study focuses on presenting the CoVpipe2 implementation and highlighting various implementation decisions in the context of reconstructing robust genome sequences for surveillance tasks. Nevertheless, we agree that another large-scale and up-to-date benchmark study comparing all available pipelines (Tab. 1), including CoVpipe2, would be interesting but is beyond the scope of our Software article.

---

[9] **Reviewer Concern**: *Provide detailed insights into how this pipeline can be applied to study other viruses*

Author Response: The predecessor of CoVpipe2, the snakemake pipeline CoVpipe1, was used to create adapted pipelines for RSV and Influenza. In this context, we discovered that other viruses might need other tools and parameters to reflect their characteristics (genome size, segmentation, reference selection). Also, the needs for final reporting may differ depending on the virus under investigation, and changes in the pipeline may be necessary. Besides, we successfully used CoVpipe2 on Polio and Measles viruses for genome reconstruction and variant calling from short-read sequencing data. In short, for Polio viruses, we sequenced the same 24 samples with Sanger, Illumina, and Nanopore, and CoVpipe2 was able to identify the same variants compared to the other sequencing technologies and associated bioinformatic steps (unpublished preliminary data).
In general, the basic software framework - the generic sub-processes of raw data quality control, read alignment, variant calling, and consensus building - and the bioinformatic challenges for data derived from amplicon sequencing are equally applicable to other viruses. CoVpipe2 can serve as a blueprint for other pathogens, especially other unsegmented viruses, and provide first insights into the variants and consensus sequences. Tools, parameters, and thresholds might need careful adjustments depending on the pathogen. Similarly, the downstream analysis might be pathogen-specific, e.g., require

pathogen-specific datasets (such as reference sequences for Influenza from Nextclade). We are currently working on a harmonized multi-pathogen pipeline with different profiles (tools, parameter settings) tailored towards specific viruses. Besides, interested users can already run CoVpipe2 on other (non-segmented) viruses by simply switching to another reference genome as we did before successfully for analyzing Polio virus amplicon data (parameter --ref_genome).
We extend the "Conclusion" accordingly:
"We used CoVpipe1 and CoVpipe2, which were initially developed for SARS-CoV-2, to reconstruct the genomes of other viruses. By selecting different reference genomes for polio, measles, RSV, and influenza viruses and modifying the analysis processes for the latter two viruses, we were able to demonstrate the workflow's potential as a universal blueprint for viral genome analysis. This adaptability has been demonstrated by the ability to identify consistent variants and the successful application to different viral characteristics, such as varying genome size and segmentation. However, it should also be noted that for segmented viruses, more customization is required than simply replacing the (non-segmented) reference genome. However, our experience suggests that customization of tools, parameters, and reporting requirements is needed for each virus, pointing to developing a harmonized pipeline for multiple pathogens. CoVpipe2 thus proves to be a robust tool for SARS-CoV-2 and paves the way for broad application in virology research by highlighting its ability to serve as a fundamental framework for building customized pipelines for a wide range of pathogens."

Competing Interests: No competing interests were disclosed.

Reviewer Report 20 September 2023

https://doi.org/10.5256/f1000research.149827.r203328

Wolfgang Maier
Bioinformatics Group, Department of Computer Science, Albert-Ludwigs-Universitat Freiburg, Freiburg, Baden-Württemberg, Germany

The manuscript by Lataretu et al. describes CoVpipe2, a bioinformatics pipeline for constructing viral consensus sequences from SARS-CoV-2 short sequenced reads obtained using the Illumina platform. It discusses the current state of the pipeline, various design decisions that have led to that state, and typical analysis pitfalls that the authors hope to overcome with their pipeline design.

The pipeline itself is implemented as a Nextflow pipeline and comes under a free and open-source license, which means that its exact steps and parameters can be explored down to any desired level of detail. Still the authors provide a very helpful overview in the manuscript text and in Figure

1 through both of which the reader can gain a good understanding of the pipeline layout and its components.

The authors point out the existence of multiple alternative analysis pipelines with similar scope as CoVpipe2 and list many of them in Table 1. I would expect most of these alternative pipelines to produce consensus genomes of similar quality as CoVpipe2, and most of the steps that CoVpipe2 is composed of are relatively standard in the field. From this perspective, the manuscript could be said to lack novelty. Beyond the core steps shared in similar form with many other pipelines, there are, however, some smart extra steps built into CoVpipe2 that are innovative ideas, like screening steps for mixed-infection and recombinant samples and consensus genome annotation with Liftoff.

More importantly, however, the authors are not just advertising yet another pipeline for SARS-CoV-2 genome analysis, but their manuscript is the kind of documentation that you wish every such pipeline came with: it explains not only individual analysis steps, but also their purpose, special analysis tweaks found to be necessary, and provides links to all relevant resources.

In summary, the manuscript describes a robust and mature resource for reproducible data analysis and does an excellent job at that. I enjoyed reading it and even though I've spent a considerable amount of time on developing similar pipelines I still picked up a few new ideas from it.

I have no concerns regarding publication of this valuable manuscript, just one comment that the authors may wish to address in the manuscript directly or in a separate reply:
For some steps in CoVpipe2 it seems there would have been several tools to choose from, and I'm wondering whether the authors of the pipeline have evaluated alternatives. In particular, I'd be interested in learning why freebayes was chosen as the variant caller and why primer trimming is done with BAMclipper, as I don't think these two tools are used by many other comparable pipelines. If the authors had specific reasons to prefer these tools over alternatives, it might add to the value of the manuscript if these were added to the text.

Beyond that, I have found a small number of inconsistencies and typos that I think should be fixed before publication:

1. the authors have done a very careful citation job, in general, but I think the following resources/specifications also deserve links/citations:
   - Zenodo
   - The "precalculated Kraken2 database" deposited at Zenodo
   - The BEDPE format (https://bedtools.readthedocs.io/en/latest/content/general-usage.html#bedpe-format)
   - Instead of providing a general anaconda.org link (as done twice in the Methods section), it would be more helpful to provide direct links to the latest versions of pangolin and nextclade (https://anaconda.org/bioconda/pangolin and https://anaconda.org/bioconda/nextclade), which also includes channel information.

2. In the introduction "*While sequencing intensity and turnaround times on variant detection increased in different countries*" is probably intended to mean increasing sequencing intensity, but *decreasing* turnaround times?

3. In the introduction, when DESH genomes statistics are given, the numbers should be 1.2 million and 1.1 million (period instead of comma), and in the discussion of Dataset 5 -> Lineages "*for each CoVpip2-GISAD pair*" has a typo in the pipeline name.

Is the rationale for developing the new software tool clearly explained?

Yes

Is the description of the software tool technically sound?

Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

Yes

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Bioinformatics, pathogen genomics, genetics

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Author Response 10 Apr 2024

Martin Hölzer

[1]
Reviewer Question: *For some steps in CoVpipe2 it seems there would have been several tools to choose from, and I'm wondering whether the authors of the pipeline have evaluated alternatives. In particular, I'd be interested in learning why freebayes was chosen as the variant caller and why primer trimming is done with BAMclipper, as I don't think these two tools are used by many other comparable pipelines. If the authors had specific reasons to prefer these tools over alternatives, it might add to the value of the manuscript if these were added to the text.*

Author Response: Thanks for the question. You are absolutely right; as with almost any bioinformatics pipeline, there are different options when choosing tools for steps such as

quality control, mapping, and variant calling. Our final selection of tools is based on our experience in analyzing sequencing data throughout the SARS-CoV-2 pandemic. Especially in the early days, we performed various internal benchmarks on SARS-CoV-2 Illumina data and manually investigated mapping results and variant calls together with colleagues from our expert unit for respiratory viruses.

Thus, our main objective in CoVpipe2 was to reliably detect variants with high allelic depth and good read support. Low-frequency variants were not the primary focus of the pipeline, as the tool is intended to reconstruct robust consensus genomes from patient samples that can be used for genomic surveillance. However, if a user wants to use CoVpipe2 for different research questions, the implementation allows full customization of the necessary parameters (allele frequency, genotype adjustment, ...) By screening the literature and examining other pipelines and community standards, we carefully selected the tools that performed best in our internal benchmarks for SARS-CoV-2 short-read data.

Regarding variant calling, we first tested LoFreq (Wilm *et al.* 2012). Although very sensitive, LoFreq lacks a strong genotyping module that was crucial for our downstream processing of the called variants. Furthermore, the output files were hard to process (non-standard VCF formats). We implemented GATK as a second choice, which is a standard tool for eukaryotic genomic variant calling (McKenna *et al*. (2010), Van der Auwera & O'Connor (2020)) but was also shown to perform well on non-human targets (Lefouili *et al.* 2022). Performance and output standards were excellent, but it turned out that GATK misses a low amount of viral genomic variants in some samples, although multisample calling was employed. Single false negative variants, which we identified via a comprehensive investigation of the BAM files, were deemed to be too important to stick with the tool. Finally, we chose freebayes (Garrison *et al*. 2012), which excelled with high performance, high precision, and output files that were straightforward to process in downstream steps of the pipeline.

In addition to the variant callers, there is indeed a large selection of quality processing tools. We opted for fastp rather than Trimmomatic because the processing speed is much faster, and the output quality is at least as good. It was shown, that “data filtered by Trimmomatic, SOAPNuke, Cutadapt and fastp were detected with 7174, 7040, 6942 and 6708 false positive variants respectively” (Chen *et al*. 2018), highlighting that fastp preprocessing can even improve the specificity of downstream analysis. In addition, fastp is equipped with a broader portfolio of parameter options.

Another crucial step in read processing is primer clipping, which may promote mapping artifacts and dilution of variant calls if done before read alignment. If InDels occur close to the end of amplicons, a gap open penalty is more expensive than a few mismatches in mapping. For example, this caused trouble with the Spike DEL69/70 in several amplicon kits and made it necessary to use the artificial primer as a mapping anchor. Hence we replaced cutadapt, which clips adapters before mapping, with BAMclipper, which removes adapters after mapping. In our opinion, primer clipping should generally be performed after mapping in reference-based analysis. We, therefore, also discuss late primer clipping more prominently in connection with CoVpipe2.

Furthermore, we fully agree that continuous benchmarking is a necessary process to adapt pipelines to changing wet lab procedures, adapted priming schemes, and pathogen evolution. If we find problems, we will also adapt CoVpipe2 accordingly and release new stable versions for reproducible research.

We added the following text to the manuscript:

“We carefully selected the bioinformatics tools integrated into CoVpipe2 based on internal

benchmarks and in-depth manual reviews of sequencing data, mapping results, and called variants. Based on our hands-on experience with SARS-CoV-2 sequencing datasets during the pandemic, this approach ensured that tools that can robustly detect high allele depth and well-covered variants are used for detection. Despite the primary goal of CoVpipe2 to identify high-confidence variants to reconstruct robust consensus genomes for genomic surveillance, the pipeline also provides flexibility for adaptation to different research environments."

---

[2]
Reviewer Question: *the authors have done a very careful citation job, in general, but I think the following resources/specifications also deserve links/citations]:*

Author Response: Thanks for the comment. We fully agree and added more precise references for various sources such as Zenodo, the custom Kraken 2 database, BEDPE format, and Nextcalde/pangolin on anaconda.org.

---

[3]
Reviewer Question: *In the introduction "While sequencing intensity and turnaround times on variant detection increased in different countries" is probably intended to mean increasing sequencing intensity, but *decreasing* turnaround times?*
Author Response: Yes, you are right; thanks for catching this. We changed the text accordingly:
"While sequencing intensity increased and turnaround times on variant detection decreased in different countries, there are also major disparities between high-, low- and middle-income countries in the SARS-CoV-2 global genomic surveillance efforts."

---

[4]
Reviewer Question: *In the introduction, when DESH genomes statistics are given, the numbers should be 1.2 million and 1.1 million (period instead of comma), and in the discussion of Dataset 5 -> Lineages "for each CoVpip2-GISAD pair" has a typo in the pipeline name.*

Author Response: Thanks, we corrected that.

Competing Interests: No competing interests were disclosed.