MICROBIAL GENOMICS

METHOD
Volk *et al.*, *Microbial Genomics* 2022;8:000850
DOI 10.1099/mgen.0.000850

MICROBIOLOGY SOCIETY
OPEN DATA OPEN ACCESS

Catwalk: identifying closely related sequences in large microbial sequence databases

Denis Volk[1], Fan Yang-Turner[1]†, Xavier Didelot[2,3], Derrick W. Crook[1] and David Wyllie[4,*]

Abstract

There is a need to identify microbial sequences that may form part of transmission chains, or that may represent importations across national boundaries, amidst large numbers of SARS-CoV-2 and other bacterial or viral sequences. Reference-based compression is a sequence analysis technique that allows both a compact storage of sequence data and comparisons between sequences. Published implementations of the approach are being challenged by the large sample collections now being generated. Our aim was to develop a fast software detecting highly similar sequences in large collections of microbial genomes, including millions of SARS-CoV-2 genomes. To do so, we developed Catwalk, a tool that bypasses bottlenecks in the generation, comparison and in-memory storage of microbial genomes generated by reference mapping. It is a compiled solution, coded in Nim to increase performance. It can be accessed via command line, REST API or web server interfaces. We tested Catwalk using both SARS-CoV-2 and *Mycobacterium tuberculosis* genomes generated by prospective public-health sequencing programmes. Pairwise sequence comparisons, using clinically relevant similarity cut-offs, took about 0.39 and 0.66 µs, respectively; in 1 s, between 1 and 2 million sequences can be searched. Catwalk operates about 1700 times faster than, and uses about 8% of the RAM of, a Python reference-based compression and comparison tool in current use for outbreak detection. Catwalk can rapidly identify close relatives of a SARS-CoV-2 or *M. tuberculosis* genome amidst millions of samples.

DATA SUMMARY

1. Data used for benchmarking:

Mycobacterium tuberculosis data was from https://ora.ox.ac.uk/objects/uuid:82ce6500-fa71-496a-8ba5-ba822b6cbb50.

SARS-CoV-2 data was obtained from https://cog-uk.s3.climb.ac.uk/phylogenetics/latest/cog_unmasked_alignment.fasta on 1/12/2021. This file is approximately 60 GB. A 1% random downsample of this is included in https://github.com/dvolk/catwalk.

The simulated dataset is available at https://github.com/dvolk/catwalk/tree/master/benchmark.

2. Software availability:

the code version submitted for review is v.1.0.0, available as https://github.com/dvolk/catwalk/archive/refs/tags/1.0.0.tar.gz

INTRODUCTION

The last few years have highlighted both the potential and the challenges in guiding public-health action using large-scale microbiological genomics. Sequencing of SARS-CoV-2 genomes on an unprecedented scale has allowed us to identify the diversification,

Received 04 January 2022; Accepted 18 May 2022; Published 30 June 2022
Author affiliations: [1]Nuffield Department of Medicine, University of Oxford, Oxford, UK; [2]School of Life Sciences, University of Warwick, Coventry CV4 7AL, UK; [3]Department of Statistics, University of Warwick, Coventry CV4 7AL, UK; [4]UK Health Security Agency, Forvie Site, Addenbrookes' Campus, Robinson Way, Cambridge CB2 0SR, UK.
*Correspondence: David Wyllie, david.wyllie@ukhsa.gov.uk
Keywords: bacterial genomics; microbial relatedness; outbreak detection.
Abbreviations:
Abbreviations: NIHR, National Institute for Health Research; RAM, random access memory; SNV, single nucleotide variant; UKHSA, UK Health Security Agency.
†Present address: UKRI Science and Technologies Facilities Council, Harwell, UK.
Data statement: The authors confirm all supporting data, code and protocols have been provided within the article or through supplementary data files.

Impact Statement

There is a need for fast computational tools to identify similar microbial genomes, since such tools can detect recent transmissions from within databases of sequenced microbes. However, the huge numbers of viral sequences generated during the SARS-CoV-2 pandemic have challenged existing approaches for doing so. In this work, we present a software tool, Catwalk, which performs such comparisons. It uses various optimizations to speed up comparisons and to reduce the memory usage required; it can compare a SARS-CoV-2 genome with over 2 million others in under a second using standard computational hardware. This is much faster than previous approaches. The tool described will assist public-health programmes in identifying potential transmission events using large sequence databases derived from population level pathogen sequencing.

geographical structuring and ongoing transmission of SARS-CoV-2 globally, and has delineated transmission routes locally, nationally and internationally [1, 2]. At the same time, developments in *Mycobacterium tuberculosis* genomics [3] have highlighted the key role of genomic prediction of antibiotic resistance, something that is likely to accelerate use of *M. tuberculosis* genomics for clinical use, thereby providing dense sampling which, at least in developed economies, has already been used successfully to track outbreaks [4].

For both the pathogens above and many others, there is a public-health need to rapidly find sequence pairs compatible with recent transmission. Such an ability underpins rapid detection of *M. tuberculosis* transmission [4], as well as identifying the SARS-CoV-2 importation events that disseminated successful clones widely [1, 2]. In both cases, although indels do evolve and may have important functional implications, single nucleotide variation occurs faster, and is the preferred type of variation from which to undertake fine-scale reconstruction of recent molecular evolution [5–7].

Here, we focus on a generic approach applicable to novel and existing epidemic pathogens where mapping to a reference genome is available. This approach, which relies on storage of the sequence in the form of differences from a reference sequence (reference-based compression), has been previously described [8] and is used by the UK Health Security Agency (UKHSA) for tracking *M. tuberculosis* relatedness [8]. The advantage of this approach is that is does not rely on additional pathogen-specific information, such as the tree required for phylogenetic placement [9], or a core-genome multilocus sequence typing scheme [10]. The disadvantage is a requirement to compare new samples against all existing ones, which is barrier to scalability.

To diminish the impact of the requirement for one-against-all comparisons, we analysed the comparison of reference-based compressed sequences. We present Catwalk, a high-performance, memory-frugal open-source software that addresses two important performance bottlenecks. We compare its performance with that of the findNeighbour2 Python implementation [8], and illustrate algorithm performance using synthetic, *M. tuberculosis* and SARS-CoV-2 genomes.

THEORY AND IMPLEMENTATION

Faster, more memory-efficient sequence compression and comparison

Efficient computation of pairwise single nucleotide variant (SNV) distances between DNA sequences derived from reference mapping can be achieved by first computing the differences relative to the reference sequence, and subsequently deducing the pairwise differences [8]. We analysed the performance of this approach, which was previously implemented in Python. In the previous implementation, differences from the references were represented as sets [8]. We noticed that storing large sets might consume significant amounts of RAM simply because of the requirement to store pointers, which are known as references in Python. Pointers are used to link elements in the hash table data structure used for Python 3 set storage and the linked lists containing the values themselves [11]. Therefore, we explored software systems where lower pointer use might result in RAM savings.

Furthermore, we analysed the set operations themselves. The approach described previously [8] considered a mapped sequence S represented as a series of sets S_A, S_C, S_G, S_T containing positions of non-reference A, C, G and T bases, and S_N containing the positions of bases that could not be called. Similarly, a second sequence R is represented as sets R_A, R_C, R_G, R_T, $R_{N.}$ The difference *d* between S and R can be written as

$$d = \sum_{b \in \{A,C,T,G\}} \left| \left(S_b - R_N\right) \wedge \left(R_b - S_N\right) \right| \qquad (eq.\ 1)$$

where

A – B denotes the difference between sets A and B, defined as the elements found in A and not found in B;

A ∧ B denotes the symmetric difference of sets A and B, defined as the elements found in either A or B but not both; |X| denotes the number of elements in (cardinality of) X.

Table 1. Data sets used for benchmarking

Data set	Description	Nucleotides excluded from comparisons	No. of sequences	Median distance from reference	Median no. of unknown bases
M. tuberculosis	Genomes sequenced by UKHSA, mapped and variation called against NC_000962.2 (4411532 nt)	557291	12832	1119	40369
SARS-CoV-2	Genomes generated by COG-UK consortium, mapped and variation called against MN908947.3 (29903 nt)	386	1529081	17	10.5

Thus, the comparison relies on performing eight set differences, and summing the symmetric differences observed. This is evident because if we were computing only differences from the reference for the base A, we can compute

$$\left|\left(S_A - R_N\right) \wedge \left(R_A - S_N\right)\right| \quad (eq.\ 2)$$

where $S_A - R_N$ represents the positions in sequence S that are non-reference A at positions which are called in sequence R, so that *eq*. 2 reflects non-reference A present in R or S, but not both.

Given the above, and since our objective is to find small distances, compatible with recent transmission from case to case, we identified three optimizations:

(1) Positions containing unknown nucleotides (S_N, R_N; *eq*. 1) are represented as sets of integers while non-reference calls (S_b, R_b; *eq*. 1) are represented as an ordered list of integers. The implementation chosen (see below) avoids the need for pointers, reducing memory usage.

(2) Symmetric differences between S_b and R_b are identified with a modified symmetric difference algorithm, which is a modification of the approach taken in the corresponding C++ algorithm [12]. It iterates simultaneously over both the ordered lists of differences S_b and R_b, detecting variants present in only one of the two sets, excluding positions with unknown bases (S_N, R_N). This has at worst $O(|S_b|+|R_b|-1)$ complexity, assuming lookup of whether an unknown position is present in S_N or R_N has O(1) complexity, as is typically the case for hash-based lookups. However, the modification exits early if distances greater than the prespecified maximum distance reportable are identified. This is acceptable since the distance can only increase (not decrease) during computation. This is an important time-saving optimization, see below.

(3) We implemented the algorithm in Nim (https://nim-lang.org/), generating compiled code.

Components

We built Catwalk as four components, with independent responsibilities in terms of computation and interfaces, allowing users access via command line, REST API and a graphical web browser interface. These components are:

(1) engine – a Nim library that stores sequences in a compressed format relative to the genome reference in memory;
(2) web server – a web server that runs the engine and offers a RESTful API for genome comparison;
(3) web user interface – a web application that connects to a server and allows users to identify and browse those samples that have low single nucleotide distances to an individual genome sequence;
(4) clients – command-line and Python tools that connect to a server via RESTful API.

Benchmarking

We conducted benchmarking on a bare metal desktop computer equipped with an Intel I7 8700 CPU, a solid-state disc and 40 GB of RAM. We used two real data sets (Table 1) and one simulated one representing *in silico* SARS-CoV-2 evolution. We measured time to load a multifasta file into either Catwalk or a reference Python implementation in the *seqComparer* module of

Table 2. Comparison of *M. tuberculosis* dataset load speed and comparison speed for Catwalk versus findNeighbour2 *seqComparer*

Implementation	Time to load data [s per sample]	Time to query neighbours with 20 SNV cut-off, per sample queried [median (25th, 75th centiles)] [µs]	Memory usage [MB per sequence]
findNeighbour2 *seqComparer* (Python)	0.31	1700 (1500, 2100)	1.78
Catwalk	0.0076	0.98 (0.91, 1.04)	0.15
Approximate performance difference	40-fold speed-up	1750-fold speed-up	12-fold lower memory usage

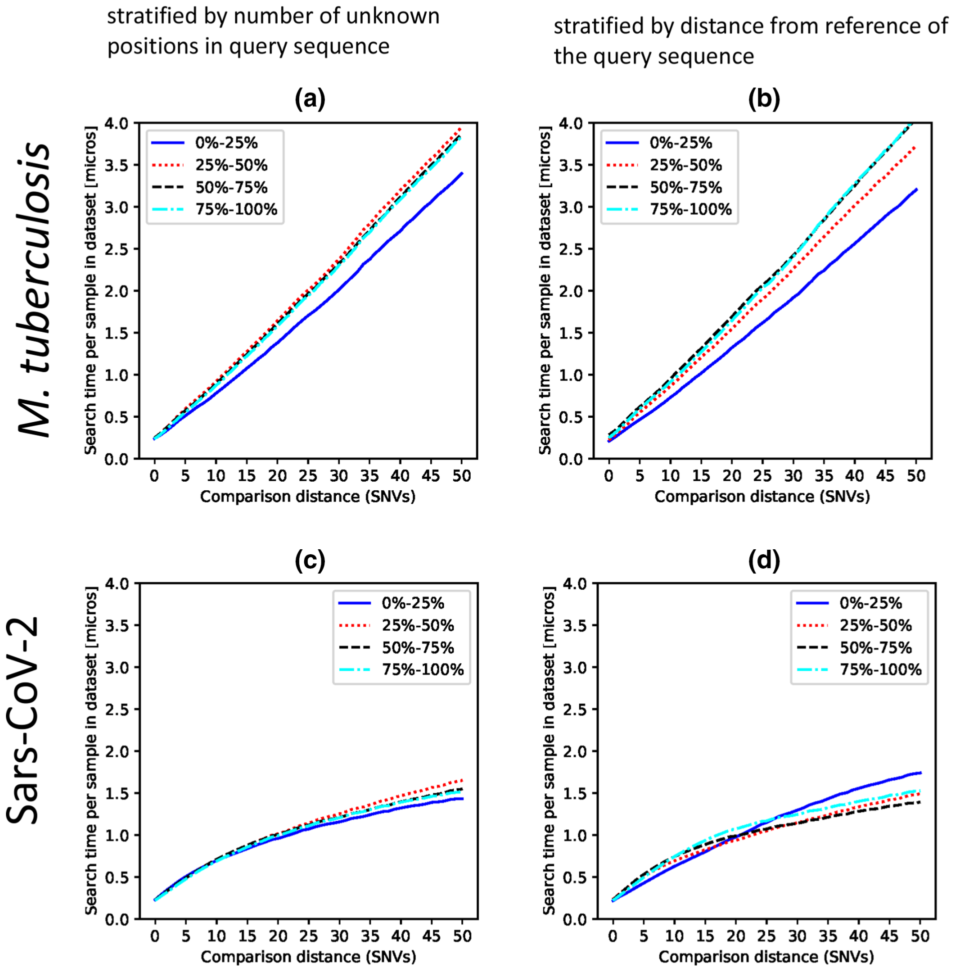

Fig. 1. Catwalk search performance. Time taken to search neighbours of a single sample in the *M. tuberculosis* data set (*n*=12 832) (a, b) or a SARS-CoV-2 data set (*n*=1 529 081) (c, d) using different SNV cut-off values. Results are expressed as search time in µs ('micros') per sample in each data set to allow comparisons between data sets. Standard errors of the mean timing are less than 3% of the plotted values in all cases, and are not shown. In (a) and (c), results are stratified by the number of unknown positions in the sequence used for the query sequence, in quantiles. In (b) and (d), results are stratified by the distance from the reference of the query sequence, in quantiles.

findNeighbour2 [8]. We also measured the time to find the neighbours of a random selection of 100 sequences from within the data set, and assessed the impact of differing single nucleotide variation cut-offs on algorithm performance. Both benchmarks examine the performance of the computational engine (component 1 above); the costs of the web server (component 2 above) processing the request and associated http transport performance were not assessed.

RESULTS

Catwalk performance relative to a reference implementation

To assess the impact of the optimizations engineered into Catwalk, we compared Catwalk performance with the Python implementation in the *seqComparer* class in the reference Python implementation findNeighbour2 [8]. As test data, we used a data set of 12832 *M. tuberculosis* genomes produced as part of the UKHSA's routine prospective sequencing of tuberculosis in England (Table 1). We observed an approximately 40-fold improvement in load speeds, 1750-fold speed-up in comparison rates, with memory usage 12-fold lower than used by the reference implementation (Table 2).

Determinants of Catwalk performance

Next, we considered determinants of Catwalk performance. We compared real data sets from *M. tuberculosis* and SARS-CoV-2 (see Table 1). Considering *M. tuberculosis*, search performance is slower as the SNV cut-off used increases (Fig. 1a, b). Searches are also slower if larger numbers of unknown bases are present (Fig. 1a) or if the distance from the reference is larger (Fig. 1b).

With a 6 SNV cut-off, which is recommended for use in public-health settings [4], the mean search time per sample in the data set was 0.66 µs.

To further explore the relationship between search times, the number of unknown nucleotides and distance from the reference, we also applied Catwalk to a simulated data set in which genomes are mutated based on the SARS-CoV-2 reference genome. This approach showed similar relationships between distance to reference as was seen in the clinical *M. tuberculosis* data set. The number of unknown bases present in the simulated data (up to 20% of the genome) had little impact on performance. Results are available in the project GitHub repository.

Performance of Catwalk in searching UK SARS-CoV-2 genomics data

Finally, we examined performance with a data set of 1.5 million sequences of SARS-CoV-2 generated by the COG-UK consortium (Table 1). The relationship with SNV distances to the reference (Fig. 1c) is similar to that seen with *M. tuberculosis*; the curved shape likely reflects population structuring in the data set used. This is as expected since the virus accrues about 25 mutations per year relative to the ancestral sequence observed in January 2020, with very strong geographical structuring. Additionally, there are less than 25% unknown bases in the data set, which (as in the simulation) has little impact on search speed (Fig. 1d). With a 3 SNV cut-off, which has been used in nosocomial outbreaks [13], the search time was 0.39 µs per sample in the data set. Thus, for two clinically relevant data sets and SNV cut-offs, the Catwalk tool affords search speeds of more than 1 million samples per second.

DISCUSSION

The explosion of genome sequence data seen during the SARS-CoV-2 pandemic has exposed the need for new fast tools for sequence comparison. Catwalk is an example of such: applicable to reference-mapped sequence data, it stores reference compressed sequences in RAM, allowing rapid sequence searches, and is optimized for the detection of the short SNV distances that are of public-health relevance. Catwalk is accessible via a REST web services interface, command line or web server interface. Both speed and memory usage are much improved compared to previous Python implementations. Performance is best where samples are close to the reference genome, and with lower numbers of undetermined sites. We observed search times for both SARS-CoV-2 and *M. tuberculosis* genomes produced during public-health sequencing projects of between one and two million samples per second; this represents an approximate 1750-fold speed-up on a previous technology, while also requiring about 8% of the memory usage of the comparator.

Our tool has several limitations, by design. Firstly, it is applicable to reference-mapped, fixed-length sequence data only. Secondly, it analyses single nucleotide variation, not other classes of genetic variation such as indels. Thirdly, it does not store distances between sequences, although that could be achieved by an external application using Catwalk as a distance computation engine. We do not include this capacity because it can present a significant storage challenge: for the approximately 1.5 million SARS-CoV-2 genomes analysed, there are approximately 9×10^9 pairs of sequences separated by 0 to 3 SNVs. Storing this amount of data is feasible, but for some applications it may be unnecessary if (as is the case for SARS-CoV-2 genomes) the data can be recomputed faster than it can be loaded out of a database. However, findNeighbour4 [14] is a larger open-source component currently under development that stores Catwalk outputs in a database. Finally, in its current form, Catwalk runs on a single machine only. The algorithm could be refactored to run across a grid of redundant machines, but given the fast load rate, small memory footprint and rapid comparisons achieved this is unnecessary for all applications we attempted, including in SARS-CoV-2.

In summary, we have produced an implementation of a reference-based compression algorithm for fixed-length microbial genomes. This can store sequences and allow pairwise SNV comparisons between millions of SARS-CoV-2 genome pairs. We believe this ultrafast algorithm will find widespread use in public genomics, as it is complementary with existing tools based on phylogenetic placement or core-genome multilocus sequence typing for bacteria.

Funding information
This work is supported by the Wellcome Trust (Scalable Pathogen Pipeline for Turning Next Generation Sequencing Pathogen Data into Clinical Results) (215800/Z/19/Z); the National Institute for Health Research (NIHR) Health Protection Research Unit in Healthcare Associated Infections and Antimicrobial Resistance (NIHR200915), a partnership between the UKHSA and the University of Oxford; the NIHR Health Protection Research Unit in Genomics and Enabling Data (NIHR200892), a partnership between the UKHSA and the University of Warwick. The views expressed are those of the authors and not necessarily those of the NIHR, UKHSA or the Department of Health and Social Care (UK).

Acknowledgements
We are grateful to the COG-UK Consortium for the public release of SARS-CoV-2 genome sequences used for software testing.

Author contributions
D.V., D.W.: conceptualization. D.V., D.W.: methodology. D.V.: software. D.W.: validation. D.W.: formal analysis. D.W., D.V., X.D.: investigation. D.W., D.V., F.Y.-T.: writing – original draft preparation. D.X., D.W.: writing – review and editing. F.Y.-T., D.W., D.W.C.: supervision. D.X., D.W.C., D.W.: funding.

Conflicts of interest
The authors declare that there are no conflicts of interest.

References

1. da Silva Filipe A, Shepherd JG, Williams T, Hughes J, Aranday-Cortes E, *et al*. Genomic epidemiology reveals multiple introductions of SARS-CoV-2 from mainland Europe into Scotland. *Nat Microbiol* 2021;6:112–122.
2. Tordoff DM, Greninger AL, Roychoudhury P, Shrestha L, Xie H, *et al*. Phylogenetic estimates of SARS-CoV-2 introductions into Washington State. *Lancet Reg Health Am* 2021;1:100018.
3. Allix-Béguec C, Arandjelovic I, Bi L, Beckert P, Bonnet M, *et al*. Prediction of susceptibility to first-line tuberculosis drugs by DNA sequencing. *N Engl J Med* 2018;379:1403–1415.
4. Nikolayevskyy V, Niemann S, Anthony R, van Soolingen D, Tagliani E, *et al*. Role and value of whole genome sequencing in studying tuberculosis transmission. *Clin Microbiol Infect* 2019;25:1377–1382.
5. Godfroid M, Dagan T, Merker M, Kohl TA, Diel R, *et al*. Insertion and deletion evolution reflects antibiotics selection pressure in a *Mycobacterium tuberculosis* outbreak. *PLoS Pathog* 2020;16:e1008357.
6. Rochman ND, Wolf YI, Faure G, Mutz P, Zhang F, *et al*. Ongoing global and regional adaptive evolution of SARS-CoV-2. *Proc Natl Acad Sci USA* 2021;118:e2104241118.
7. Harvey WT, Carabelli AM, Jackson B, Gupta RK, Thomson EC, *et al*. SARS-CoV-2 variants, spike mutations and immune escape. *Nat Rev Microbiol* 2021;19:409–424.
8. Mazariegos-Canellas O, Do T, Peto T, Eyre DW, Underwood A, *et al*. BugMat and FindNeighbour: command line and server applications for investigating bacterial relatedness. *BMC Bioinformatics* 2017;18:477.
9. Matsen FA, Kodner RB, Armbrust EV. pplacer: linear time maximum-likelihood and Bayesian phylogenetic placement of sequences onto a fixed reference tree. *BMC Bioinformatics* 2010;11:538.
10. Kohl TA, Harmsen D, Rothgänger J, Walker T, Diel R, *et al*. Harmonized genome wide typing of tubercle bacilli using a web-based gene-by-gene nomenclature system. *EBioMedicine* 2018;34:131–138.
11. **Cython documentation and source code:** setobject.c; 2021. https://github.com/python/cpython/blob/main/Objects/setobject.c [accessed 1 December 2021].
12. **C++ symmetric difference set algorithm**; 2021. https://www.cplusplus.com/reference/algorithm/set_symmetric_difference/ [accessed 1 December 2021].
13. Lumley SF, Constantinides B, Sanderson N, Rodger G, Street TL, *et al*. Epidemiological data and genome sequencing reveals that nosocomial transmission of SARS-CoV-2 is underestimated and mostly mediated by a small number of highly infectious individuals. *J Infect* 2021;83:473–482.
14. **FindNeighbour4**: a server application for investigating bacterial relatedness using reference-mapped data; 2021. https://github.com/davidhwyllie/findNeighbour4 [accessed 1 December 2021].