













1 phylo2vec: a library for vector-based phylogenetic tree 2 manipulation

3 Neil Scheidwasser ^{1*}, Ayush Nag ^{2*}, Matthew J. Penn ¹, Anthony M.
4 V. Jakob ⁴, Frederik M. Andersen ¹, Mark P. Khurana ¹, Landung
5 Setiawan ², David A. Duchêne ¹, and Samir Bhatt ^{1,3¶}

6 ¹ Section of Health Data Science and AI, University of Copenhagen, Copenhagen, Denmark ² eScience
7 Institute, University of Washington, Seattle, United States ³ MRC Centre for Global Infectious Disease
8 Analysis, Imperial College London, London, United Kingdom ⁴ Independent researcher ¶ Corresponding
9 author * These authors contributed equally.

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Open Journals](#) 

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

License

Authors of papers retain copyright
and release the work under a
Creative Commons Attribution 4.0
International License ([CC BY 4.0](#)).

Summary

Phylogenetics is a fundamental component of evolutionary analysis frameworks in biology (Yang, 2014) and linguistics (Atkinson & Gray, 2005). Recently, the advent of large-scale genomics and the SARS-CoV-2 pandemic has highlighted the necessity for phylogenetic software to handle large datasets (Attwood et al., 2022; Kapli et al., 2020; Khurana et al., 2024; Kraemer et al., 2025). While significant efforts have focused on scaling optimisation algorithms (De Maio et al., 2023; Sanderson, 2021; Turakhia et al., 2021), visualization (Sanderson, 2022), and lineage identification (McBroome et al., 2024), an emerging body of research has been dedicated to efficient representations of data for genomes (Deorowicz et al., 2023) and phylogenetic trees (Chauve et al., 2025; Penn et al., 2024; Richman et al., 2025). Compared to the traditional Newick format which represents trees using strings of nested parentheses (Felsenstein, 2004), modern tree representations utilize integer vectors to define the tree topology traversal. This approach offers several advantages, including easier manipulation, increased memory efficiency, and applicability to machine learning.

Here, we present the latest release of phylo2vec (or Phylo2Vec), a high-performance software package for encoding, manipulating, and analysing binary phylogenetic trees. At its core, the package is based on the phylo2vec (Penn et al., 2024) representation of binary trees, and is designed to enable fast sampling and tree comparison. This release features a core implementation in Rust for improved performance and memory efficiency (Figure 1), with wrappers in R and Python (superseding the original release (Penn et al., 2024)), making it accessible to a broad audience in the bioinformatics community.

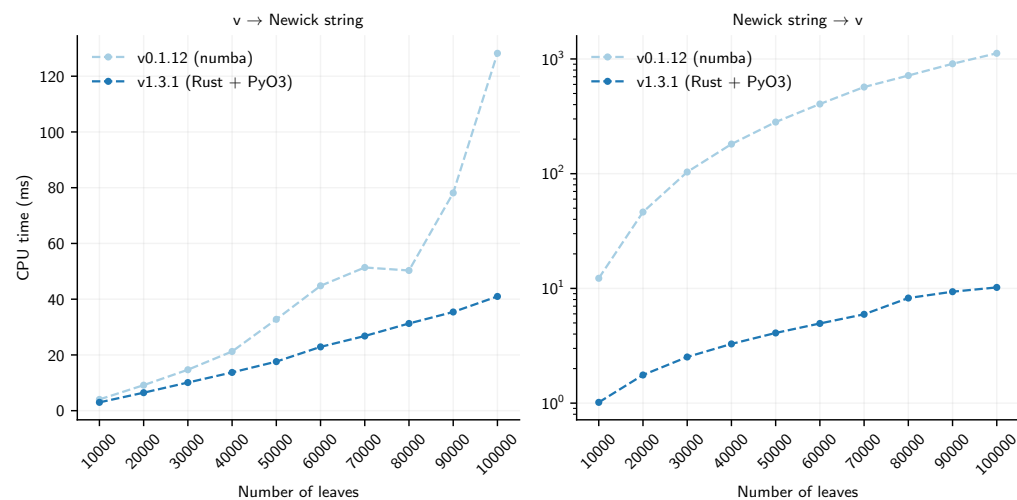


Figure 1: Benchmark times for converting a phylo2vec vector to a Newick string (left) and vice versa (right). Execution time was measured over at least 20 runs per size, comparing Python functions in the latest release (via Rust bindings with [PyO3](#)) against the previous release ([Penn et al., 2024](#)) based on Numba ([Lam et al., 2015](#)). All benchmarks were performed on a workstation equipped with an AMD Ryzen Threadripper PRO 5995WX (64 cores, 2.7 GHz) and 256 GB of RAM.

Statement of need

The purpose of the phylo2vec library is threefold. First, it provides robust phylogenetic tree manipulation in Rust, complementing other efforts such as `light_phylogeny` ([Duchemin et al., 2018](#)) for reconciled phylogenies ([Nakhleh, 2013](#)), and `rust-bio` ([Köster, 2016](#)), which does not yet cover phylogenetics. Second, it complements existing libraries such as `ape` ([Paradis & Schliep, 2019](#)) in R, and `ete3` ([Huerta-Cepas et al., 2016](#)) and `DendroPy` ([Moreno et al., 2024](#)) in Python, by providing fast tree sampling, fast tree comparison and efficient tree data compression ([Penn et al., 2024](#)). Third, the phylo2vec representation offers a pathway to using new optimisation frameworks for phylogenetic inference. A notable example is `GradME` ([Penn et al., 2023](#)), a gradient descent-based algorithm which uses a continuous relaxation of the phylo2vec representation.

Features

The presented release of phylo2vec addresses optimisations limitations of ([Penn et al., 2024](#)) with $\mathcal{O}(n \log n)$ implementations for vector-to-Newick and Newick-to-vector conversions, leveraging Adelson-Velsky and Landis (AVL) trees ([Adelson-Velsky & Landis, 1962](#)) and Fenwick trees ([Fenwick, 1994](#)), respectively.

New features in the latest release include an extension of the vector representation to support branch length annotations, leaf-level operations (pruning, placement, and MRCA identification), fast cophenetic distance matrix calculation, and various optimisation schemes based on phylo2vec tree representations, notably hill-climbing ([Penn et al., 2024](#)) and `GradME` ([Penn et al., 2023](#)). We also propose a likelihood function for Bayesian MCMC inference that leverages tree representation similarities with `BEAGLE` ([Ayres et al., 2012](#); [Suchard & Rambaut, 2009](#)). Finally, user-friendliness is enhanced with step-by-step demos of phylo2vec's representations and core functions.

Maintenance

A strong focus of this release is to support long-term maintenance through implementing recommended software practices into its project structure and development workflow. The project is structured with a Rust API containing core algorithms with language bindings to avoid tight coupling and enable easy language additions. Additionally, we have established a robust continuous integration (CI) pipeline using GitHub Actions, which features:

- Unit test frameworks for Rust ([cargo](#)), Python ([pytest](#)), and R ([testthat](#) ([Wickham, 2011](#)))
- Benchmarking on the Rust code ([criterion](#)) and its Python bindings ([pytest-benchmark](#))

Lastly, to complement Jupyter Notebook demos, comprehensive documentation is provided using [Jupyter Book](#) and [Read the Docs](#).

Acknowledgements

S.B. acknowledges funding from the MRC Centre for Global Infectious Disease Analysis (reference MR/X020258/1), funded by the UK Medical Research Council (MRC). This UK funded award is carried out in the frame of the Global Health EDCTP3 Joint Undertaking. S.B. acknowledges support from the Danish National Research Foundation via a chair grant (DNRF160, also supporting N.S. and M.P.K.), The Eric and Wendy Schmidt Fund For Strategic Innovation via the Schmidt Polymath Award (G-22-63345, also supporting M.J.P. and F.M.A.), and the Novo Nordisk Foundation via The Novo Nordisk Young Investigator Award (NNF20OC0059309). D.A.D. is supported by a Data Science - Emerging researcher award from Novo Nordisk Fonden (NNF23OC0084647). The authors also thank Madeline Gordon for her help in implementing functions related to the matrix format.

References

- Adelson-Velsky, Georgii, & Landis, E. (1962). An algorithm for the organization of information. *Proc. USSR Acad. Sci.*, 146, 263–266.
- Atkinson, Q. D., & Gray, R. D. (2005). Curious parallels and curious connections—phylogenetic thinking in biology and historical linguistics. *Syst. Biol.*, 54(4), 513–526. <https://doi.org/10.1080/10635150590950317>
- Attwood, S. W., Hill, S. C., Aanensen, D. M., Connor, T. R., & Pybus, O. G. (2022). Phylogenetic and phylodynamic approaches to understanding and combating the early SARS-CoV-2 pandemic. *Nat. Rev. Genet.*, 23(9), 547–562. <https://doi.org/10.1038/s41576-022-00483-8>
- Ayres, D. L., Darling, A., Zwickl, D. J., Beerli, P., Holder, M. T., Lewis, P. O., Huelsenbeck, J. P., Ronquist, F., Swofford, D. L., Cummings, M. P., & others. (2012). BEAGLE: An application programming interface and high-performance computing library for statistical phylogenetics. *Syst. Biol.*, 61(1), 170–173. <https://doi.org/10.1093/sysbio/syr100>
- Chauve, C., Colijn, C., & Zhang, L. (2025). A vector representation for phylogenetic trees. *Philos. Trans. R. Soc. B*, 380(1919), 20240226. <https://doi.org/10.1098/rstb.2024.0226>
- De Maio, N., Kalaghatgi, P., Turakhia, Y., Corbett-Detig, R., Minh, B. Q., & Goldman, N. (2023). Maximum likelihood pandemic-scale phylogenetics. *Nat. Genet.*, 55(5), 746–752. <https://doi.org/10.1038/s41588-023-01368-0>
- Deorowicz, S., Danek, A., & Li, H. (2023). AGC: Compact representation of assembled genomes with fast queries and updates. *Bioinformatics*, 39(3), btad097. <https://doi.org/10.1093/bioinformatics/btad097>

- 99 Duchemin, W., Gence, G., Arigon Chifolleau, A.-M., Arvestad, L., Bansal, M. S., Berry, V.,
100 Boussau, B., Chevenet, F., Comte, N., Davin, A. A., & others. (2018). RecPhyloXML: A
101 format for reconciled gene trees. *Bioinformatics*, 34(21), 3646–3652. <https://doi.org/10.1093/bioinformatics/bty389>
102
- 103 Felsenstein, J. (2004). *Inferring phylogenies*. Sinauer Associates.
- 104 Fenwick, P. M. (1994). A new data structure for cumulative frequency tables. *Softw. Pract.*
105 *Exp.*, 24(3), 327–336. <https://doi.org/10.1002/spe.4380240306>
- 106 Huerta-Cepas, J., Serra, F., & Bork, P. (2016). ETE 3: Reconstruction, analysis, and
107 visualization of phylogenomic data. *Mol. Biol. Evol.*, 33(6), 1635–1638. <https://doi.org/10.1093/molbev/msw046>
108
- 109 Kapli, P., Yang, Z., & Telford, M. J. (2020). Phylogenetic tree building in the genomic age.
110 *Nat. Rev. Genet.*, 21(7), 428–444. <https://doi.org/10.1038/s41576-020-0233-0>
- 111 Khurana, M. P., Curran-Sebastian, J., Scheidwasser, N., Morgenstern, C., Rasmussen, M., Fon-
112 ager, J., Stegger, M., Tang, M.-H. E., Juul, J. L., Escobar-Herrera, L. A., & others. (2024).
113 High-resolution epidemiological landscape from~ 290,000 SARS-CoV-2 genomes from
114 Denmark. *Nat. Commun.*, 15(1), 7123. <https://doi.org/10.1038/s41467-024-51371-0>
- 115 Köster, J. (2016). Rust-bio: A fast and safe bioinformatics library. *Bioinformatics*, 32(3),
116 444–446. <https://doi.org/10.1093/bioinformatics/btv573>
- 117 Kraemer, M. U., Tsui, J. L.-H., Chang, S. Y., Lytras, S., Khurana, M. P., Vanderslott, S., Bajaj,
118 S., Scheidwasser, N., Curran-Sebastian, J. L., Semenova, E., & others. (2025). Artificial
119 intelligence for modelling infectious disease epidemics. *Nature*, 638(8051), 623–635.
120 <https://doi.org/10.1038/s41586-024-08564-w>
- 121 Lam, S. K., Pitrou, A., & Seibert, S. (2015). Numba: A LLVM-based Python JIT compiler.
122 *Proc. 2nd Workshop on the LLVM Compiler Infrastructure in HPC*, 1–6. <https://doi.org/10.1145/2833157.2833162>
123
- 124 McBroome, J., Bernardi Schneider, A. de, Roemer, C., Wolfinger, M. T., Hinrichs, A. S.,
125 O'Toole, A. N., Ruis, C., Turakhia, Y., Rambaut, A., & Corbett-Detig, R. (2024). A
126 framework for automated scalable designation of viral pathogen lineages from genomic
127 data. *Nat. Microbiol.*, 9(2), 550–560. <https://doi.org/10.1038/s41564-023-01587-5>
- 128 Moreno, M. A., Holder, M. T., & Sukumaran, J. (2024). DendroPy 5: a mature Python library
129 for phylogenetic computing. *J. Open Source Softw.*, 9(101), 6943. <https://doi.org/10.21105/joss.06943>
130
- 131 Nakhleh, L. (2013). Computational approaches to species phylogeny inference and gene tree
132 reconciliation. *Trends Ecol. Evol.*, 28(12), 719–728. <https://doi.org/10.1016/j.tree.2013.09.004>
133
- 134 Paradis, E., & Schliep, K. (2019). Ape 5.0: An environment for modern phylogenetics
135 and evolutionary analyses in R. *Bioinformatics*, 35, 526–528. <https://doi.org/10.1093/bioinformatics/bty633>
136
- 137 Penn, M. J., Scheidwasser, N., Khurana, M. P., Duchêne, D. A., Donnelly, C. A., & Bhatt,
138 S. (2024). Phylo2Vec: A vector representation for binary trees. *Syst. Biol.*, syae030.
139 <https://doi.org/10.1093/sysbio/syae030>
- 140 Penn, M. J., Scheidwasser, N., Penn, J., Donnelly, C. A., Duchêne, D. A., & Bhatt, S. (2023).
141 Leaping through tree space: Continuous phylogenetic inference for rooted and unrooted
142 trees. *Genome Biol. Evol.*, 15(12), evad213. <https://doi.org/10.1093/gbe/evad213>
- 143 Richman, H., Zhang, C., & IV, F. A. M. (2025). Vector encoding of phylogenetic trees by
144 ordered leaf attachment. In *arXiv*. <https://doi.org/10.48550/arXiv.2503.10169>
- 145 Sanderson, T. (2021). Chronumetal: Time tree estimation from very large phylogenies. In

- 146 *bioRxiv*. Cold Spring Harbor Laboratory. <https://doi.org/10.1101/2021.10.27.465994>
- 147 Sanderson, T. (2022). Taxonium, a web-based tool for exploring large phylogenetic trees.
148 *eLife*, 11. <https://doi.org/10.7554/eLife.82392>
- 149 Suchard, M. A., & Rambaut, A. (2009). Many-core algorithms for statistical phylogenetics.
150 *Bioinformatics*, 25(11), 1370–1376. <https://doi.org/10.1093/bioinformatics/btp244>
- 151 Turakhia, Y., Thornlow, B., Hinrichs, A. S., De Maio, N., Gozashti, L., Lanfear, R., Haussler,
152 D., & Corbett-Detig, R. (2021). Ultrafast sample placement on existing tRees (USHER)
153 enables real-time phylogenetics for the SARS-CoV-2 pandemic. *Nat. Genet.*, 53(6),
154 809–816. <https://doi.org/10.1038/s41588-021-00862-7>
- 155 Wickham, H. (2011). Testthat: Get started with testing. *The R Journal*, 3(1), 5–10.
156 <https://doi.org/10.32614/rj-2011-002>
- 157 Yang, Z. (2014). *Molecular evolution: A statistical approach*. Oxford University Press.
158 <https://doi.org/10.1093/acprof:oso/9780199602605.001.0001>

DRAFT