# The accuracy of phylogeny reconstruction on simulated whole genome sequences from Evolver

Junilda Spirollari[1], Samantha Leong[2], and Usman Roshan[1]

[1] Department of Computer Science, New Jersey Institute of Technology, Newark, NJ 07102
[2] Department of Medicine, Rutgers New Jersey Medical School, Newark, NJ 07101

**Abstract.** The rise of genome sequencing has produced new methods and studies for phylogeny reconstruction from whole genome data. In the popular gene concatenation method individual genes are aligned separately and then alignments are concatenated on which a phylogeny is determined. In the tree consensus method (also widely used) trees are constructed on individual genes and combined to form a consensus. These methods are likely to miss evolutionary changes in non-coding regions of the genome that contribute to the phylogeny. Other less common methods such as Huson's Genome BLAST Distance Phylogeny (GBDP) use the entire genome sequence and have been previously compared to conserved RNA phylogenies. In this paper we evaluate for the first time the accuracy of whole genome phylogenies on large-scale simulated data produced from the Evolver whole genome evolutionary model. We also explore several variants of GBDP and show that the method can be improved with a whole genome alignment tool instead of BLAST, Jukes-Cantor evolutionary distances, and using just the top few largest aligned blocks in BLAST. We show that our variants give trees with lower Robinsons-Foulds distance than gene concatenation, tree consensus, and the original GBDP. We also evaluate our proposed method on real data with a reference (gene-based) mammalian phylogeny and show that while our tree has higher error than a simple gene-concatenated one it recovers many major clades. Our work here suggests that whole genome sequence phylogeny reconstruction methods that account for changes in non-coding regions may give more accurate trees than existing gene-based methods and that existing whole genome methods can be improved. We make our data freely available at `https://web.njit.edu/~usman/genomephylogeny`.

## 1 Introduction

As genome sequencing becomes mainstream we see studies evaluating the accuracy of phylogeny reconstruction from whole genome sequences [1–9]. Previous studies have considered real data only and gauge the accuracy of their methods by comparing against phylogenies from conserved RNA genes. The two popular methods for constructing phylogenies from whole genome sequences are gene concatenation and tree consensus [10]. In the first method individual gene alignments are concatenated into a single alignment from which typically a maximum likelihood (ML) [11] tree is inferred. In the second method we construct trees on individually aligned genes which are then combined to produce a final tree.

An alternative approach is to consider whole genome sequences for building a tree. One of the earliest such methods is the Genome BLAST Distance Phylogeny (GBDP) by Huson *et. al.* [3]. In this method high scoring segment pairs (HSPs) between two genomes are first determined with the blastn program [12] after which a distance matrix is computing from the HSPs. We explore a similar approach in that we create a distance matrix on which a neighbor joining phylogeny is constructed, except that we determine evolutionary distances from aligned blocks given by the popular LASTZ genome alignment program [13]. While the GBDP method uses a distance calculation based on the number of nucleotides in non-overlapping high scoring alignments we use the Jukes Cantor distance [14] that is determined directly from our aligned blocks.

More importantly, we consider for the first time simulated data to evaluate phylogenies constructed from whole genome sequences. The Evolver program [15] makes it possible to simulate the evolution of whole genome sequences under a sophisticated model that accounts for known genome evolutionary events. Recent large-scale comparisons of genome assemblers and alignment programs such as Assemblathon [16] and Alignathon [17] use simulated data from Evolver to compare different programs. Several recent studies on genome assembly and alignment have also used simulated data from Evolver for evaluation [18–23]. However, we are unaware of previous studies that use Evolver data and model trees for evaluating whole genome phylogeny reconstruction.

In our study we simulate whole genome sequences under mammalian models trees of 5, 15, 25, and 50 taxa (five trials for each setting). Running Evolver is considerably challenging as even specified by the program authors in the manual. Our data simulation and analysis together took months of computation to complete. For each simulated dataset we constructed trees given by gene concatenation, tree consensus, Huson's GBDP methods and several variants of it, and our LASTZ based method. We then we evaluate their error with the Robinson-Foulds distance that measures the number of false positives and false negatives. We also study our LASTZ based method on real data. In summary our results show that whole genome sequence methods give lower error than gene based methods on the simulated data, and that our variants of GBDP give lower error than the original method. In the rest of the paper we describe the Evolver simulation model and the phylogeny reconstruction methods that we study. After that we present in detail our results on simulated and real data followed by discussion and future avenues.

## 2 Methods

### 2.1 Evolver evolutionary model

Evolver [15] is a suite of programs designed to simulate the evolution of whole genome sequences. It does so by simulating the long-term averaged effects of mutations and selections over a species rather than using an explicit model of allele frequencies or gene flow. Evolver embodies two core components as part of its simulation, namely inter- and intra-chromosome modules. The inter-chromosome module simulates events involving two chromosomes, including chromosome fission, fusion, and segmental moves and copies. The intra-chromosome module simulates events that occur within a single chromosome, such as insertions, substitutions, deletions, inversions, transpositions, and duplications.

There are two types of evolutionary events are used in its simulations: mutations and constraint changes. Mutations are events that modify the primary sequence. Some examples of mutations are insertions and deletions. If a mutation affects a base than it is said to be accepted with a probability called accept probability. Evolver proposes mutations at rates specified by parameters in the model and are used in both inter- and intra-chomosome evolutions.

Constraint change events modify annotations and leave the primary sequence unchanged. Some examples of constraint change events are exon gains and losses such as create a new first or last UTR exon. Evolver's gene model is based on four types of constrained elements (CEs): CDS (protein-coding sequence), UTR (untranslated exonic sequence), NXE (non-exonic CE in a gene), NGE (non-gene element). Since there is no model of RNA genes the base pairing is not modeled. Genes are defined by the range of positions defined in annotations file. Evolver explicitly models protein evolution by including constraints such as the gene must begin with a start codon and must contain exactly one stop codon as last one in the coding sequence.

Mobile elements (MEs) are modeled as nucleotide sequences that are inserted at a randomly chosen point in the chromosome and are used in the intra-module. Their evolution is given by a birth-death model (see page 21 of the manual for full details).

Evolver executes in *evolutionary cycles* by first invoking the inter-chromosome simulator once of the entire genome followed by the intra-chromosome simulator once for each chromosome. The output of a cycle is used as an input of the next one.

Evolver requires four main input components:

- An ancestral genome sequence
- An annotation file describing the ancestral genome in terms of gene, non-gene conserved elements, CpG islands and tandem arrays
- A library of mobile elements and retroposed pseudo-gene sequences
- A parameter file that specifies a model of evolution that includes rates for each type of events used in simulation.

The Evolver output consists of several files:

- Alignments of the evolved genomes to the common ancestor as well as to each other
- Annotations file of the evolved genomes
- Statistics of the evolution process, for example number of accepted and rejected substitutions and genome characteristics

**Effect of model tree branch lengths on Evolver** Evolver uses ticks as a unit of time to evolve along a particular branch. It calculates the number of *evolutionary cycles* per tick at the base level, gene level, or chromosome level. For example if a branch length is 0.06 and we specify a tick length of 0.02 then Evolver evolves the genome at length 0.02, then 0.04, and then 0.06. At each step of the evolution Evolver will produce the genome sequences and annotation files (all of which take considerable space). If the branch length is greater than the step length then only one evolutionary cycle is performed.

An evolutionary cycle does not necessarily mean only one evolutionary event. Several events may take place in one cycle such as substitutions, insertions, and deletions. There are event rates in the parameter file that specify the number of events per tick. For example from the manual: "the rate of UTR loss is specified as UTRs lost per exon per tick. Thus if the genome has 105 UTRs and the rate of UTR loss is 10-4 the average number of UTRs lost in the genome will be 10 per tick".

The default step value in Evolver is 0.002 while our model tree average branch lengths are higher. Below in Subsection 3.2 we describe how we select the step length.

### 2.2 Phylogeny reconstruction methods

We first describe the supermatrix and supertree methods both of which are widely used for whole genome phylogeny reconstruction. We then describe the Genome BLAST Distance Phylogeny method of Huson *et. al.* [3] and our method for estimating distance matrices from whole genome data on which we then run the neighbor joining method to produce a tree.

**Gene concatenation (supermatrix)** In this method we align individual genes separately and then concatenate them to form one large alignment on which we infer a tree. The location of the individual genes are given in the output files given by Evolver. Since genes may be duplicated we use the first copy we encounter while parsing the genome. We use the MAFFT program to multiply align each gene [24] and RAxML [25] for tree inference both with default parameters.

**Tree consensus (supertree)** In the supertree approach we infer trees on invidually aligned genes separately. Here also the exact location of individual genes are available to us, we use the first duplicated copy, and we use MAFFT [24] and RAxML[25] for alignment and tree reconstruction respectively. We then determine the majority consensus tree with the PHYLIP consense tool[26]. This is the tree given by bipartitions that occur in the majority of the trees [27].

**Genome blast distance phylogeny (GBDP)** The GBDP method uses blastn [12, 28] to determine high scoring segment pairs (HSPs) of min length 50 between a pair of genomes from which an evolutionary distance is estimated. We describe the full GBDP algorithm below in Algorithm 1. Our description follows the steps described in the original study [3].

We implement GBDP with Python programs, the blastn standalon program, and the neighbor program in PHYLIP [26]. We study two variants of this method as well:

- **GBDP-JC**: Instead of the distances given by the above formula here we choose to utilize the Jukes-Cantor evolutionary distance. Instead of selecting long high scoring segments we replace steps 3 and 4 we a different one. We concatenate all high scoring aligned pairs to form one large alignment and determine the Jukes-Cantor evolutionary distance

$$d(G_i, G_j) = -\frac{3}{4} log(1 - \frac{4}{3} p_{ij}), \text{ where } p_{ij} = \frac{total\_mismatches}{total\_length - gaps}$$

- **GBDP-TOPK**: Instead of considering all high scoring segments to determine the maximum set we select only the top $k$ longest ones as given by blastn. This in step 2 we would set $k$ to a specific value. This considerably reduces error as we show below.

**Algorithm 1** Genome blast distance phylogeny (GBDP)

**Input:** Input genome sequences given in a list $G$ where the length of $G$ is $n$
**Output:** Phylogeny $T$ on $G$
**Procedure:**

**for** $i = 0$ to $n - 1$ **do**
  **for** $j = i + 1$ to $n - 1$ **do**
    1. Determing high scoring segments between genomes $G_i$ and $G_j$ using blastn software (with word size set to 50)
    2. The output of blastn are $k$ high scoring pairwise alignments $HSP_0, ..., HSP_{k-1}$ of $G_i$ and $G_j$.
    3. Assume without loss of generality that the high scoring pairs are ranking by decreasing length. We greedly select the maximum set $S$ of non-overlapping pairs: we start with $HSP_0$ and add $HSP_m$ (for $0 < m < k$) if it doesn't overlap with existing HSPs in $S$.
    4. Let $G_i^{match}$ and $G_j^{match}$ be the number of nucleotides in maximal set of HSPs $S$ that we calculated above. We define

$$d_{match}(G_i, G_j) = -log(\frac{|G_i^{match}|+|G_j^{match}|}{2min(|G_i|,|G_j|}$$

    Since blastn is asymmetric the final evolutionary distance is given by the average

$$d(G_i, G_j) = \frac{d_{match}(G_i,G_j)+d_{match}(G_j,G_i)}{2}$$

  **end for**
**end for**
Output the neighbor joining tree on $d$

**LASTZ-based genome distance phylogeny (LGDP)** Instead of high scoring segment pairs given by blastn as in the above method we consider aligned blocks given by the LASTZ program [13] to determine evolutionary distances. For a given pair of genomes $X$ and $Y$ we determine aligned blocks with LASTZ from which we compute Jukes-Cantor [14] evolutionary distances. We explain our method in full in Algorithm 2.

**Algorithm 2** LASTZ-based genome distance phylogeny (LGDP)

**Input:** Input genome sequences given in a list $G$ where the length of $G$ is $n$
**Output:** Phylogeny $T$ on $G$
**Procedure:**

**for** $i = 0$ to $n - 1$ **do**
  **for** $j = i + 1$ to $n - 1$ **do**
    1. Align genomes $G_i$ and $G_j$ with LASTZ
    2. The output of LASTZ are aligned blocks of $G_i$ and $G_j$.
    3. Concatenate all aligned blocks into one large alignment and determine the Jukes Cantor evolutionary distance

$$d(G_i, G_j) = -\frac{3}{4}log(1 - \frac{4}{3}p_{ij}), \text{ where } p_{ij} = \frac{total\_mismatches}{total\_length-gaps}$$

  **end for**
**end for**
Output the neighbor joining tree on $d$

## 3 Results

We describe here our experimental setup with Evolver including challenges encountered in running it. We then present our experimental results followed by discussion.

### 3.1 Model trees

We performed a BLAST search against the manually annotated Swiss-Prot database [29] with the human haemoglobin gene (HBA_HUMAN) as the query sequence. We then obtained top 5, 15, 25, and 50 ranked sequences with repeat species omitted and computed the neighbor joining tree via ClustalW [30] also available at the Swiss-Prot server. This gave us four model trees from real data on which we then simulate whole genome sequences starting from human chromosome 20 as the root sequence. In Figure 1 we show our 50 taxa model tree.

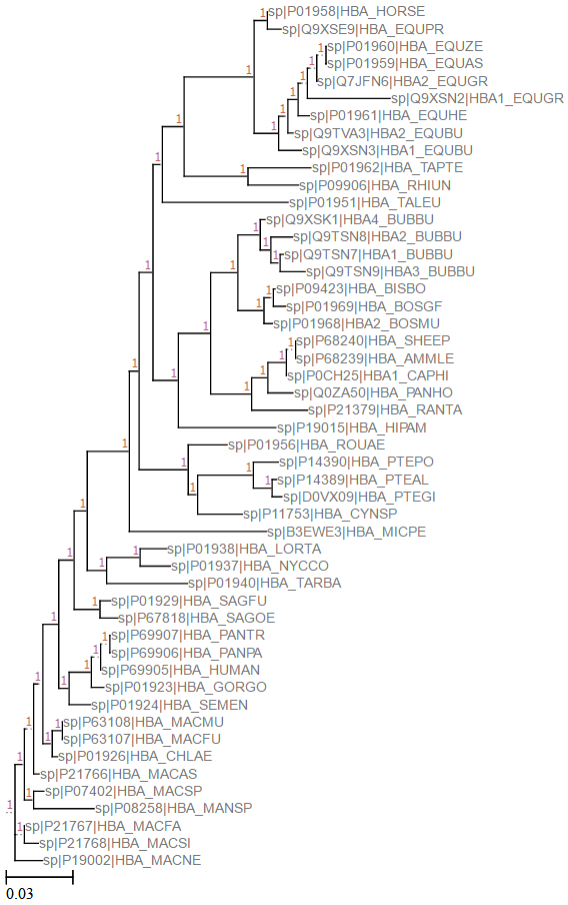


Fig. 1: Our 50 taxa haemoglobin neighbor joining gene tree that we use as a model tree

### 3.2 Evolver simulation

In total we simulate five sets of whole genome sequences for each model tree. We started the simulation using a subset of the annotated human genome GRCh38/hg38 [31, 32]. We extracted the complete chromosome sequence for chromosome 20 along with the annotations from the UCSC genome browser, golden path download site [31–33], specifically knownGene, mgcGenes, cpgIslandsExt, ensGene and knownGeneOld5. We used the evolverSimControl [17] suite to process the data into an Evolver infile dataset using same parameters as described in the Alignathon study [17]. The simulation process shuffles the sequences, genes, and chromosomes of the genome along a given phylogenetic tree as described earlier. We used the repetitive element library from the human genome already given in the Evolver package [17, 31, 32].

As described earlier Evolver uses ticks as a unit of time to evolve along a particular branch. It calculates the number of evolutionary cycles per tick at the base level, gene level, or chromosome level. The default step length in Evolver is 0.001 while the average branch lengths of our 5, 15, 25, and 50 taxon model trees are 0.002, 0.017, 0.011, 0.01 respectively. As expected this led to a large output by Evolver of genome sequences and annotations since it simulates data for every 0.001 portion of the step along a given branch. Furthermore Evolver may timeout early and not simulate genome sequences for all taxa possibly due to the birth-death process governing mobile element evolution (see page 21 of the manual). We increased the step length incrementally by 0.001 while simulating on the 50 taxon model tree and stopped at 0.006. There the data output was manageable and Evolver didn't time out before simulating all taxa - smaller step lengths and larger trees led to more frequent timeouts.

The simulation process is highly computationally challenging as even stated by the authors of the program in the program manual. In fact we encountered several computational challenges in producing the simulated data, particularly on larger 25 and 50 taxon trees that haven't been studied before with this software. For example Evolver takes days to finish generating data for a 25 taxon tree on a typical dual core CPU. We ran it on a multiprocessor machine with a total of 32 cores and 128GB RAM so we could generate multiple simulated datasets and produce data for 50 taxon trees as well. In total the data for our simulation study occupied about at least terrabyte of disk space and took weeks to produce (and additional weeks to analyze due its scale). Another challenge in running Evolver on large trees is that the evolutionary process may terminate earlier on the model tree thus not producing sequences on all leaves. This may be due to its birth death model that governs the mobile element evolution (see page 21 of Evolver manual).

We make our data freely available at `https://web.njit.edu/~usman/genomephylogeny`. This includes the five model trees and 25 simulated whole genome sequences along with annotation files outputted by Evolver.

### 3.3 Phylogeny reconstruction

In order to construct gene concatenation and tree consensus phylogenies we need location of genes within the simulated genomes. Evolver produces annotated files that contain such information for each simulated genome. We thus extract simulated genes from each sequence from which we construct gene concatenation and tree consensus phylogenies as described earlier. Due to duplication there may be multiple copies of the same gene in which case we select the first one encountered while parsing the genome. For the two genome methods we provide the genome sequences directly as input.

### 3.4 Robinson-Foulds distances on simulated data

**Comparison of whole genome sequence and gene based methods** For each model tree we simulated five sets of whole genome sequences. On each set we constructed trees with the four methods described above and determined the Robinson-Foulds distance [34] with the PHYLIP program treedist. This is just the number of false positives plus false negatives. Since both our model and estimated trees are binary our false positives are equal to false negatives thus making the Robinsons-Fould distance twice the number of false positives.

In Figure 2 we make several observations. First see that the error of all four methods increases as the tree size becomes larger suggesting the problem becomes difficult as we add taxa. One way to overcome this problem is to use divide-and-conquer methods for large-scale phylogeny reconstruction such as disk covering methods [35–37]. Disk-covering methods divide the problem into overlapping subsets with a distance [38] or tree-based decomposition [35], construct trees on each subset, and combine the overlapping subtrees with the strict consensus merger [37]. This approach is likely to work here since we see that smaller taxon trees are easier to reconstruct.

Second we see that the gene and supertree consensus methods give higher error than the LASTZ based whole genome approach. This may not be too surprising since the model produces evolutionary events in non-coding regions that gene-based methods would miss. At the same time there are many genes in the root genome that undergo evolution, yet their signal isn't strong enough to determine all branches correctly.

Thirdly, of the two whole genome sequence methods we see that the LASTZ-based genome distance phylogeny (LGDP) has lower error than the genome BLAST distance phylogeny (GBDP). This may be because we are using the Jukes-Cantor distances from the alignments to determine an evolutionary distance matrix whereas GBDP relies on number of bases covered in the BLAST alignments. It's also possible that the quality of the high scoring segments may

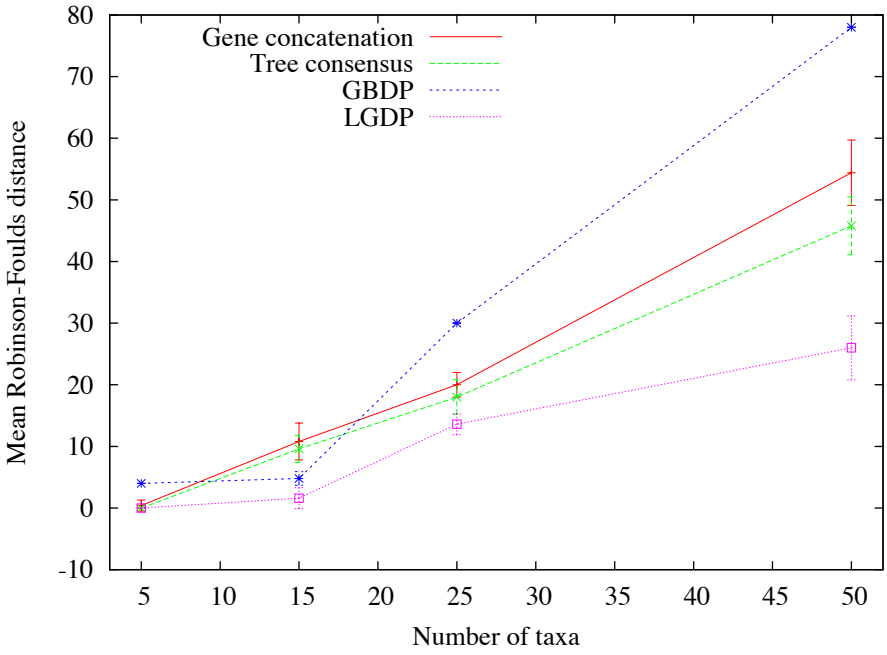


Fig. 2: Shown here are the average Robinson-Foulds distances of the four methods on genome sequences evolved from our four model trees. Each point is the average across five simulated datasets per model tree and has error bars to indicate the standard deviation.

affect the distance matrix calculation and so we test the effect of considering the top $k$ high scoring segments from which to compute the maximal set. We describe these two variants earlier in our Section on Methods and compare them to GBDP and LGDP in the next subsection.

**Comparison of variants of GBDP and LGDP** Due to time constraints we are able to perform these experiments only on the first trial of our simulated data for 5, 15, and 25 taxon model trees. In Figure 3 we show the error of the original GBDP, the two variants GBDP-JC and GBDP-TOPK (for k=1000, 10K, 20K, and 50K), and LGDP. Recall that GBDP-JC is the original method except the Jukes-Cantor distance matrices are used and in GBDP-TOPK we use the top $k$ largest high scoring segments instead of all to calculate the maximum non-overlapping set.

We make several interesting observations. First we see that Jukes-Cantor distances on high scoring alignments given by blastn (the GBDP-JC method) gives lower error than the original GBDP. Thus it appears that using a more exact evolutionary distance is better than counting number of bases in maximal non-overlapping alignments, which is the original method.

However, the results with GBDP-TOPK show this may not necessarily the case. When we consider just the top 1000 longest high scoring pairs to determine the maximal non-overlapping set we see that the resulting tree has the lowest error of the GBDP variants. In fact GBDP-TOP1K is comparable to LGDP and better at 25 taxa. As we increase the threshold $k$ from 1K to 50K the error increases and approaches the original GBDP. Remember that in this variant we use the original distance formula proposed in GBDP. The top $k$ scoring pairs are not only long but also tend to be of better quality in terms of alignment score and significance. Thus we see that the better quality high scoring pairs give lower error.

### 3.5 Comparison of LGDP and gene concatenation on real data

This section is particularly challenging for us since model species trees are generally constructed with gene concatenation methods. For example the Tree of Life mammalian phylogeny located at `http://tolweb.org/Eutheria/`

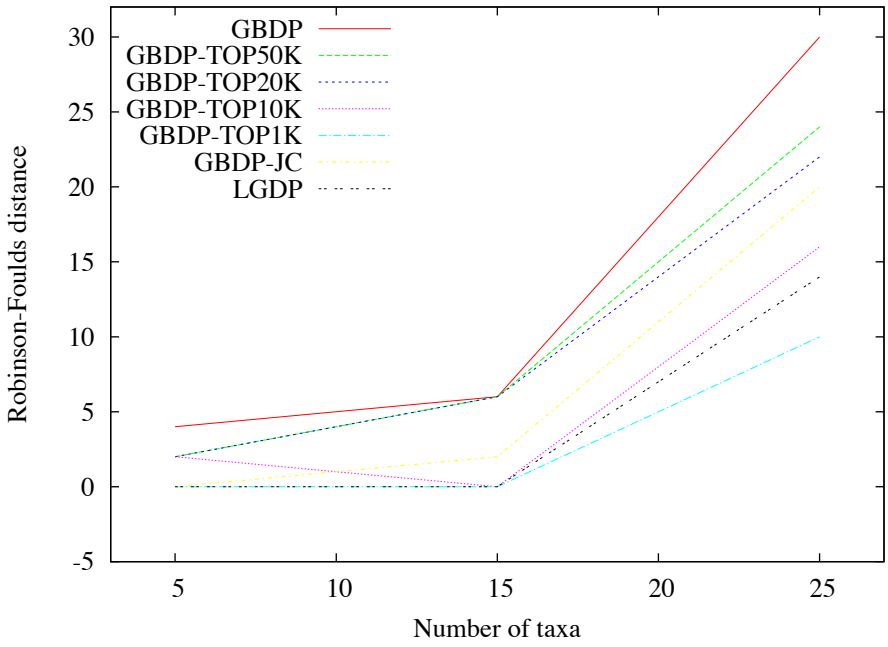


Fig. 3: Shown here are the Robinson-Foulds distances of variants of GBDP on the first set of evolved genome sequences from our three model trees. We see that as $k$ increases, in other words as we consider more top high scoring pairs given by blastn, the error of GBDP-TOPK approaches GBDP.

15997 (and also on Wikipedia at https://en.wikipedia.org/wiki/Evolution_of_mammals) were given by a maximum likelihood gene concatenated tree containing thousands of genes [39]. Still we want to evaluate our method on real data and see the similarities and differences to the gene concatenated model species. For this purpose we consider the mammalian phylogeny at The Broad Institute as a reference tree [40]. Note that this phylogeny is also gene based as explained in pages S18 and S19 of the Supplementary Section of the paper [40]. It was constructed from orthologous gene trees from family gene trees in the Ensemble Compara v57 database. After filtering the authors obtained a tree based on 17,709 genes in total.

In order to also have our gene concatenated tree on this data we sought eight genes: gadd45, cd8, cd4, il2, il10, m33, cytochrome b, and haemoglobin from the SWISSPROT curated database [29]. These genes are spread across essential functionality but we were able to obtain them only for 24 of the taxa in the mammalian tree. Thus we consider the model tree reduced to 24 taxa as the reference one. We aligned each gene with MAFFT with default parameters, concatenated the alignments, and then determined the RAxML tree with default parameters.

We obtained whole genome sequences for each of the 24 taxa from the NCBI genome database [28]. From there we construct our LASTZ-based genome tree. In Table 1 we see that the gene concatenated tree even with eight genes has a lower Robinson-Foulds distance. Both trees including the reference are binary, thus false positives and false negatives are just half the Robinson-Foulds distance.

Table 1: 29 mammals tree comparison

| Approach | Robinson-Foulds distance |
|---|---|
| Gene concatenated tree from eight genes | 18 |
| LASTZ-genome based tree | 28 |

While the LASTZ-based whole genome tree has higher error we see in Figure 4 it reconstructs several subtrees correctly. For example we see the subtrees given by human and chimp, rat and mouse, cat and dog, armadillo and elephant, cow and dolphin and alpaca all to be correct. We also see larger correct subtrees. It is possible that the differences between the LASTZ-based tree and the reference are due to evolutionary changes in non-coding regions that our method accounts for.

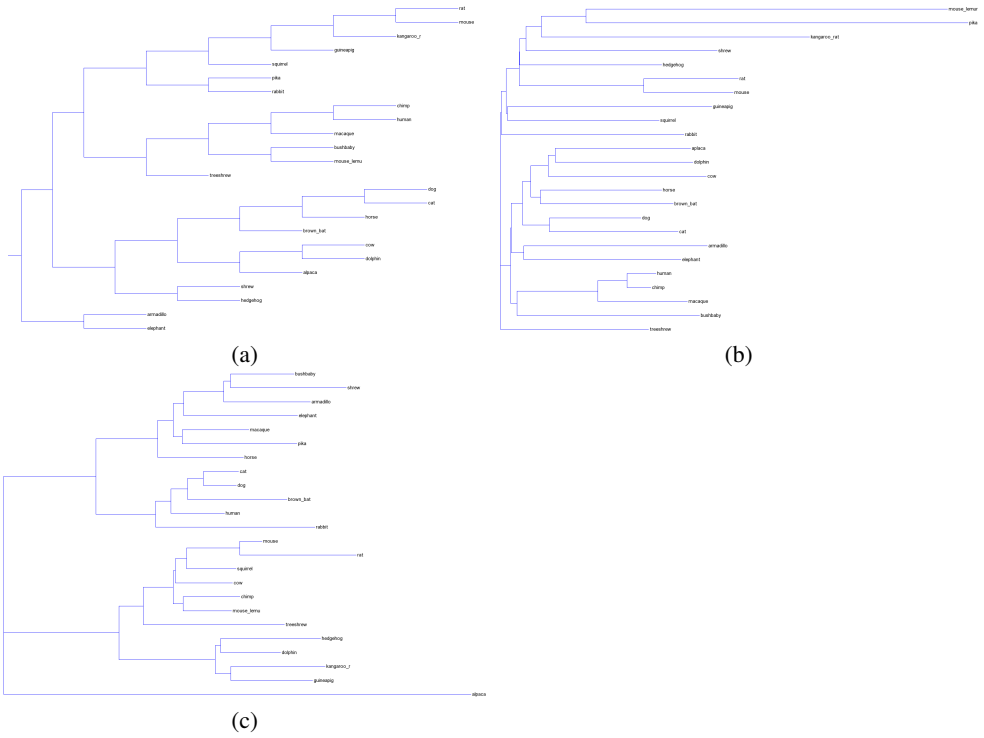


Fig. 4: In (a) we have the model reference tree on 24 mammals [40], in (b) the LASTZ-based genome distance phylogeny from whole genome sequences, and in (c) the gene concatenated tree from eight genes.

## 4 Discussion and Future work

Our research here opens the avenue for several lines of future work. First, the Evolver step length corresponds to evolutionary rates - low values means more evolution and high values mean less - and we studied only one setting in this paper. We wish to study the effect of lower and higher step lengths moving forward so we can see how higher and lower evolutionary rates affect the phylogenetic reconstruction compared to our current setting. We also want to evolve bacterial genomes - as opposed to just a human chromosome as done here - so that we can evaluate microbial whole genome phylogenetic reconstruction accuracy.

Second, divide and conquer methods like disk covering methods [37, 38, 35] can be highly effective here leading to accurate phylogenies from whole genome data for large groups of species. Third, our results on the simulated data provide insight into the whole genome methods that we have not applied to real data yet. Moving forward we want to evaluate the GBDP phylogeny (and its variants) on the real dataset. In fact the insight that using just the top $k$ high scoring alignments gives better trees can be extended to the LGDP method as well. As of now we use all the alignment blocks outputted by LASTZ in that method but based on what we see in GBDP-TOPK we expect it to perform better if we consider the top 1000 ones.

Another line of future work is to extend the Evolver evolutionary model to incorporate horizontal gene transfer which has been observed in genome evolution [41, 42]. This would allow the evaluation of phylogenetic network methods [43–45] as well as new network methods from large-scale whole genome sequence data.

We attempted to use an experimentally evolved bacterial phylogeny [46] as a reference tree as part of our real data study. However, the simulated data includes only short reads of evolved genomes and not completely assembled ones. This brings up an interesting question of determining evolutionary distances between genomes using short read data only. Some recent methods have been proposed [47, 48] but their accuracy on simulated data is unknown.

## 5 Conclusion

We conduct for the first-time a large-scale simulation study to evaluate the accuracy of phylogenies from simulated whole genome sequences under the Evolver evolutionary model. We show that whole genome sequence methods give trees with lower error than gene concatenation and tree consensus methods and that simple variants of existing whole genome sequence phylogeny reconstruction methods can give lower error.

## 6 Acknowledgements

We thank the NJIT Academic and Research Computing Systems Group (ARCS) for their support in running Evolver on the university's symmetric multiprocessor computing node.

## References

1. J. Eisen. Phylogeny of bacterial and archaeal genomes using conserved genes: supertrees and supermatrices. *PLOSone*, 8:e6251, 2013.
2. C. Abergel C. Notredame. Using multiple alignment methods to assess the quality of genomic data analysis. *Bioinformatics and Genomes*, pages 30–50, 2003.
3. Stefan R Henz, Daniel H Huson, Alexander F Auch, Kay Nieselt-Struwe, and Stephan C Schuster. Whole-genome prokaryotic phylogeny. *Bioinformatics*, 21(10):2329–2335, 2005.
4. Jan P Meier-Kolthoff and Markus Goeker. Victor: Genome-based phylogeny and classification of prokaryotic viruses. *bioRxiv*, page 107862, 2017.
5. Jeffrey T Foster, Stephen M Beckstrom-Sternberg, Talima Pearson, James S Beckstrom-Sternberg, Patrick SG Chain, Francisco F Roberto, Jonathan Hnath, Tom Brettin, and Paul Keim. Whole-genome-based phylogeny and divergence of the genus brucella. *Journal of bacteriology*, 191(8):2864–2870, 2009.
6. Soichirou Satoh, Mamoru Mimuro, and Ayumi Tanaka. Construction of a phylogenetic tree of photosynthetic prokaryotes based on average similarities of whole genome sequences. *PloS one*, 8(7):e70290, 2013.
7. Liang Liu, Zhenxiang Xi, Shaoyuan Wu, Charles C Davis, and Scott V Edwards. Estimating phylogenetic trees from genome-scale data. *Annals of the New York Academy of Sciences*, 1360(1):36–53, 2015.
8. Md Shamsuzzoha Bayzid and Tandy Warnow. Naive binning improves phylogenomic analyses. *Bioinformatics*, 29(18):2277–2284, 2013.
9. Erich D Jarvis, Siavash Mirarab, Andre J Aberer, Bo Li, Peter Houde, Cai Li, Simon YW Ho, Brant C Faircloth, Benoit Nabholz, Jason T Howard, et al. Whole-genome analyses resolve early branches in the tree of life of modern birds. *Science*, 346(6215):1320–1331, 2014.
10. Sudhindra R Gadagkar, Michael S Rosenberg, and Sudhir Kumar. Inferring species phylogenies from multiple genes: concatenated sequence tree versus consensus gene tree. *Journal of Experimental Zoology Part B: Molecular and Developmental Evolution*, 304(1):64–74, 2005.
11. Joseph Felsenstein. Evolutionary trees from dna sequences: a maximum likelihood approach. *Journal of molecular evolution*, 17(6):368–376, 1981.
12. Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990.
13. R.S. Harris. Improved pairwise alignment of genomic dna. *Ph.D. Thesis, The Pennsylvania State University.*, 2007.
14. Thomas H Jukes, Charles R Cantor, et al. Evolution of protein molecules. *Mammalian protein metabolism*, 3(21):132, 1969.
15. S. Batzoglou R. Edgar, G. Asimenos and A. Sidow. Evolver, 2009.

16. Dent Earl, Keith Bradnam, John St John, Aaron Darling, Dawei Lin, Joseph Fass, Hung On Ken Yu, Vince Buffalo, Daniel R Zerbino, Mark Diekhans, et al. Assemblathon 1: a competitive assessment of de novo short read assembly methods. *Genome research*, 21(12):2224–2241, 2011.
17. D. Earl et al. Alignathon: a competitive assessment of whole-genome alignment methods. *Genome Res.*, 24:2077–2089, 2014.
18. Jaebum Kim, Denis M Larkin, Qingle Cai, Yongfen Zhang, Ri-Li Ge, Loretta Auvil, Boris Capitanu, Guojie Zhang, Harris A Lewin, Jian Ma, et al. Reference-assisted chromosome assembly. *Proceedings of the National Academy of Sciences*, 110(5):1785–1790, 2013.
19. Krisztian Buza, Bartek Wilczynski, and Norbert Dojer. Record: reference-assisted genome assembly for closely related genomes. *International journal of genomics*, 2015, 2015.
20. Benedict Paten, Dent Earl, Ngan Nguyen, Mark Diekhans, Daniel Zerbino, and David Haussler. Cactus: Algorithms for genome multiple sequence alignment. *Genome research*, 21(9):1512–1528, 2011.
21. Alejandro Hernandez Wences and Michael C Schatz. Metassembler: merging and optimizing de novo genome assemblies. *Genome biology*, 16(1):207, 2015.
22. Virag Sharma, Anas Elghafari, and Michael Hiller. Coding exon-structure aware realigner (cesar) utilizes genome alignments for accurate comparative gene annotation. *Nucleic acids research*, page gkw210, 2016.
23. Michael Hiller, Bruce T Schaar, and Gill Bejerano. Hundreds of conserved non-coding genomic regions are independently lost in mammals. *Nucleic acids research*, 40(22):11463–11476, 2012.
24. K Kazutaka et al.l. Mafft: a novel method for rapid multiple sequence alignment based on fast fourier transform. *Nuc. Acid Res.*, 30(14):3059–3066, 2002.
25. A. Stamatakis. Raxml version 8: A tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 2014.
26. J. Felsenstein. Phylip (phylogeny inference package) version 3.5c. *Department of Genetics, University of Washington, Seattle.*, 1993.
27. David M Hillis, Craig Moritz, Barbara K Mable, and Dan Graur. *Molecular systematics*, volume 23. Sinauer Associates Sunderland, MA, 1996.
28. Eric W Sayers, Tanya Barrett, Dennis A Benson, Evan Bolton, Stephen H Bryant, Kathi Canese, Vyacheslav Chetvernin, Deanna M Church, Michael DiCuccio, Scott Federhen, et al. Database resources of the national center for biotechnology information. *Nucleic acids research*, 39(suppl 1):D38–D51, 2011.
29. Emmanuel Boutet, Damien Lieberherr, Michael Tognolli, Michel Schneider, and Amos Bairoch. Uniprotkb/swiss-prot: the manually annotated section of the uniprot knowledgebase. *Plant bioinformatics: methods and protocols*, pages 89–112, 2007.
30. Julie D Thompson, Desmond G Higgins, and Toby J Gibson. Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic acids research*, 22(22):4673–4680, 1994.
31. W.J. Kent et al. The human genome browser at ucsc. *Genome Res.*, 12(6):996–1006, 2002.
32. K.R. Rosenbloom et al. The ucsc genome browser database: 2015 update. *Nucleic Acids Res.*, 43 (Database issue):D670–81, 2015.
33. K.R. Rosenbloom et al. Ecode data in the ucsc genome browser: year 5 update. *Nucleic Acids Res.*, 41(Database issue):D56–63, 2013.
34. D.F. Robinson and L.R Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53:131–147, 1981.
35. Usman W Roshan, Tandy Warnow, Bernard ME Moret, and Tiffani L Williams. Rec-i-dcm3: a fast algorithmic technique for reconstructing phylogenetic trees. In *Computational Systems Bioinformatics Conference, 2004. CSB 2004. Proceedings. 2004 IEEE*, pages 98–109. IEEE, 2004.
36. Luay Nakhleh, Usman Roshan, Katherine St John, Jerry Sun, and Tandy Warnow. Designing fast converging phylogenetic methods. *Bioinformatics*, 17(suppl 1):S190–S198, 2001.
37. Daniel H Huson, Scott M Nettles, and Tandy J Warnow. Disk-covering, a fast-converging method for phylogenetic tree reconstruction. *Journal of Computational Biology*, 6(3-4):369–386, 1999.
38. Daniel H Huson, Lisa Vawter, and Tandy J Warnow. Solving large scale phylogenetic problems using dcm2. In *ISMB*, volume 99, page 1, 1999.
39. William J Murphy, Eduardo Eizirik, Stephen J O'brien, Ole Madsen, Mark Scally, Christophe J Douady, Emma Teeling, Oliver A Ryder, Michael J Stanhope, Wilfried W de Jong, et al. Resolution of the early placental mammal radiation using bayesian phylogenetics. *Science*, 294(5550):2348–2351, 2001.
40. K Lindblad-Toh et al. A high-resolution map of human evolutionary constraint using 29 mammals. *Nature*, 478:476–482, 2011.
41. Ravi Jain, Maria C Rivera, and James A Lake. Horizontal gene transfer among genomes: the complexity hypothesis. *Proceedings of the National Academy of Sciences*, 96(7):3801–3806, 1999.
42. E Bapteste, E Susko, J Leigh, D MacLeod, RL Charlebois, and WF Doolittle. Do orthologous gene phylogenies really support tree-thinking? *BMC Evolutionary Biology*, 5(1):33, 2005.

43. Daniel H Huson, Regula Rupp, and Celine Scornavacca. *Phylogenetic networks: concepts, algorithms and applications*. Cambridge University Press, 2010.
44. Bernard ME Moret, Luay Nakhleh, Tandy Warnow, C Randal Linder, Anna Tholse, Anneke Padolina, Jerry Sun, and Ruth Timme. Phylogenetic networks: modeling, reconstructibility, and accuracy. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 1(1):13–23, 2004.
45. Luay Nakhleh. Evolutionary phylogenetic networks: models and issues. In *Problem solving handbook in computational biology and bioinformatics*, pages 125–158. Springer, 2010.
46. Johanne Ahrenfeldt, Carina Skaarup, Henrik Hasman, Anders Gorm Pedersen, Frank Møller Aarestrup, and Ole Lund. Bacterial whole genome-based phylogeny: construction of a new benchmarking dataset and assessment of some existing methods. *BMC Genomics*, 18(1):19, 2017.
47. Frederic Bertels, Olin K Silander, Mikhail Pachkov, Paul B Rainey, and Erik van Nimwegen. Automated reconstruction of whole-genome phylogenies from short-sequence reads. *Molecular biology and evolution*, 31(5):1077–1088, 2014.
48. Huan Fan, Anthony R Ives, Yann Surget-Groba, and Charles H Cannon. An assembly and alignment-free method of phylogeny reconstruction from next-generation sequencing data. *BMC genomics*, 16(1):522, 2015.