A coarse-graining, ultrametric approach to resolve the phylogeny of prokaryotic strains with frequent homologous recombination

Tin Yau Pang (pang@hhu.de)

Research article

Keywords: phylogenetics, homologous recombination, ultrametric tree

Posted Date: December 10th, 2019

DOI: https://doi.org/10.21203/rs.2.18054/v1

Version of Record: A version of this preprint was published at BMC Evolutionary Biology on May 7th, 2020. See the published version at https://doi.org/10.1186/s12862-020-01616-5.

A coarse-graining, ultrametric approach to resolve the phylogeny of prokaryotic strains with frequent homologous recombination

Tin Yau Pang

Institute for Computer Science, Heinrich Heine University, Düsseldorf, 40225, Germany
Tel: +49-211-81-11651; Fax: +49-211-81-15767;

Email: pang@hhu.de

ABSTRACT

Background

A frequent event in the evolution of prokaryotic genomes is homologous recombination, where a foreign DNA stretch replaces a genomic region similar in sequence. Recombination can affect the relative position of two genomes in a phylogenetic reconstruction in two different ways: (i) one genome can recombine with a DNA stretch that is similar to the other genome, thereby reducing their pairwise sequence divergence; (ii) one genome can recombine with a DNA stretch from an outgroup genome, increasing the pairwise divergence. While several recombination-aware phylogenetic algorithms exist, many of these cannot account for both types of recombination; some algorithms can, but do so inefficiently. Moreover, many of them reconstruct the ancestral recombination graph (ARG) to help infer the genome tree, and require that a substantial portion of each genome has not been affected by recombination, a sometimes unrealistic assumption.

Results

Here, we propose a coarse-graining approach for phylogenetic reconstruction (CGP), which is recombination-aware but forgoes ARG reconstruction, applicable even if all genomic regions have experienced substantial amounts of recombination, and can be used on both nucleotide and amino acid sequences. CGP considers the local density of substitutions along pairwise genome alignments, fitting a model to the empirical distribution of substitution density to infer the pairwise coalescent time. Given all pairwise coalescent times, CGP reconstructs an ultrametric tree representing vertical inheritance. Based on simulations, we show that the proposed approach can reconstruct ultrametric trees with accurate topology, branch lengths, and root positioning. Applied to a set of *E. coli* strains, the reconstructed trees are most consistent with gene distributions when inferred from amino acid sequences, a data type that cannot be utilized by many alternative approaches.

Conclusions

The CGP algorithm is more accurate than alternative recombination-aware methods for ultrametric phylogenetic reconstructions.

KEYWORDS

phylogenetics, homologous recombination, ultrametric tree

BACKGROUND

The transfer of DNA stretches from one prokaryotic genome to another—also called horizontal gene transfer (HGT) or lateral gene transfer (LGT)—is a major driver of prokaryotic evolution [1]. It is caused by a variety of mechanisms, including transformation, transduction, conjugation, and gene transfer agents [2, 3]. Many prokaryotic genomes encode defense systems against foreign DNA, such as the restriction modification system [4]. A foreign DNA stretch that enters the prokaryotic cell and survives these host defenses may be incorporated into the host genome. If the incoming DNA stretch is highly similar to a stretch on the host genome, homologous recombination may occur, where the incoming DNA stretch homologously recombines with the host stretch and overwrites it [5, 6]. Alternatively, the incoming stretch may be inserted directly into the host genome through non-homologous recombination.

HGT allows the fast spread of genes in prokaryotic pangenomes, and facilitates rapid adaptation to environmental changes. A point in case is the spread of antibiotic resistance genes in pathogenic bacteria via HGT [7]. But recombination is also crucial for the long-term maintenance of prokaryotic populations, as it helps to repair DNA damaged by deleterious mutations, thereby avoiding the mutational meltdown of Muller's ratchet [8] in that sense, prokaryotic recombination may fulfill the same function as does sex in eukaryotes.

Computational modelling also suggests that recombination may help prokaryotes to purge selfish mobile genetic elements [9].

Recombination can severely affect phylogeny reconstructions. Its effects on genome divergence are complex. Depending on the circumstances, recombination can speed up the divergence of a genome pair or slow it down [6]; its effects may severely affect the accuracy of estimated branch lengths of phylogenetic tree. For example, (i) when a stretch of genome X is replaced by DNA from genome Y, some of the single nucleotide polymorphisms that previously differentiated X and Y will be erased, shortening the apparent evolutionary distance between the two genomes. Conversely, (ii) when X recombines with a DNA stretch of an outgroup genome (a genome that diverged before the split of the X and Y lineages), then it introduces further nucleotide polymorphisms into X, thereby increasing the apparent X-Y distance.

Multilocus sequence typing (MLST) aims to extract sequences of housekeeping genes from prokaryotic genomes, which can then be utilized to resolve evolutionary relationships [10]. However, MLST genes may also experience frequent recombination, and phylogenetic reconstruction without accounting for recombination can compromise the resulting trees [11]. In fact, the frequency with which recombination affects a gene can be of the same order of magnitude as the corresponding mutation rate [5]. Thus, if two lineages recombine with each other, application of conventional phylogenetic algorithms without accounting for recombination will generally lead to an underestimation of the age of the common ancestors [12]. When there are more than two strains, recombination affects the reliability of inference of relative divergence times between strains and may hence compromise both tree topology and branch length estimates.

There are several popular recombination-aware algorithms, including ClonalFrameML (and its predecessor ClonalFrame) [13, 14], the Bacter package in BEAST2

(which implements the ClonalOrigin model) [15, 16], and Gubbins [17]; there are also non-phylogenetic algorithms that detect recombination, such as BratNextGen and fastGEAR [18, 19]. These recombination-aware algorithms may reconstruct the ancestral recombination graph (ARG), which describes the history of transfer and homologous recombination of local genomic stretches across the genomes, to help infer the tree of phylogenetic inheritance of the genomes. While these algorithms can identify genomic stretches with high numbers of substitutions due to recombination with distant strains and thus account for type (ii) recombination effects, many do not take type (i) recombination effects into account; Bacter can account for type (i) effects, but is not computationally efficient for long genome sequences. Some of these algorithms rely on the assumption of low recombination rates, such that a substantial part of a genome remains clonal and has not been affected by recombination. This is unrealistic at least for some bacteria: *e.g.*, *E. coli* strains whose DNA sequences have diverged by more than 1.3% share very few stretches larger than a few kb that have not been affected by recombination in at least one of the two lineages [5]. Moreover, ARG may only reveal the latest recombination events on a genomic stretch, but its ability to recover the earlier events on the same stretch is limited, since each recombination erases the history of previous recombinations; this uncertainty on earlier recombination events may introduce error in branch length prediction.

In this paper, we propose a novel approach to phylogenetic reconstruction that neither assumes low recombination rates nor relies on ARG reconstruction. Our approach follows a coarse-graining model, which considers the local density of substitutions on a sequence alignment instead of site-specific substitutions [5]; the model describes how different parameters, such as mutation rate, recombination rate or coalescent time between a pair of genomes, affect the shape of their distribution of substitution density. It fits the empirical distribution of substitution density of genome pairs to the model, which allows the inference

of the matrix of coalescent time between the genome pairs and thereafter their ultrametric phylogenetic tree. In short, it forgoes the reconstruction of ARG and infers the branch lengths—coalescent times—from the relative abundance of genomic segments with different number of substitutions, and is also applicable to both nucleotide and amino acid sequences. The source code implementing this model is available at https://github.com/TinPang/coarse-graining-phylogenetics.

METHODS

Overview of the CGP algorithm that reconstructs ultrametric phylogenetic tree

The proposed coarse-graining phylogenetic (CGP) algorithm takes n aligned genome sequences as input (Figure 1a), which can be either nucleotide or amino acid sequences. For each pair of sequences, it divides them into L_{seg} equal-sized segments, each segment has l_s sites, and enumerates the single sites polymorphisms (SSPs)—sites with substitution—on each segment to obtain the distribution of local SSP density of the genome sequence pair (Figure 1b). This algorithm considers segments instead of nucleotide / amino-acid sites as the basic unit of a genome, because the local SSP density can be defined conveniently on segments; an SSP can be a single nucleotide polymorphism (SNP) on a nucleotide sequence, or a single amino acid polymorphism (SAP) on an amino acid sequence. It then fits the empirical SSP distribution of all pairs to a model to infer the matrix of coalescent time of the genome pairs (Figure 1d). While searching for the best fit model parameters and pairwise coalescent times, it constrains these $n(n\text{-}1)/2$ coalescent times (Figure 1e), so that the matrix can be bijectively mapped to a UPGMA tree that describes the phylogenetic inheritance of the n genome sequences (Figure 1f).

Model describing the evolution of single site polymorphism distributions of genome pairs in a Fisher-Wright population

The model behind CGP, which is used to fit the empirical distribution of local SSP (Figure 1d), is based on a Fisher-Wright haploid population with non-overlapping generations, constant population size, and homologous recombination [20, 21]. In this framework, a node in one generation inherits the genome of a random node in the previous generation, followed by mutation and homologous recombination. A genome sequence is divided into L_{seg} consecutive and non-overlapping segments, where every segment has length l_s (*i.e.*, consists of l_s sites). The rate of mutation is μ per site per generation. The rate for a site to be covered by a foreign DNA stretch attempting to recombine with the host genome is ρ per generation; the rate for a segment to be covered by a recombination-attempting DNA stretch is also approximately ρ, assuming that the segment is much shorter than the DNA stretch. Here, $\rho=\rho_{ini}L$, where ρ_{ini} is the probability for a recombination-attempting foreign DNA stretch to start at any given site, and L is the average length of the foreign DNA stretch. When a recombination attempt happens on a segment, it either succeeds, and the foreign DNA replaces the host DNA at the segment, or it fails. The success rate of an attempt is approximately $\exp(-\delta/\delta_{\mathrm{TE}})$, where δ is the divergence between the incoming DNA and the host DNA, and δ_{TE} is the transfer efficiency, a constant that governs the success rate [5]. The average site divergence in the population is denoted as θ, with $\theta=2\mu N_e$ and population size N_e.

CGP's model [5, 6] considers the evolution of a SSP distribution between a pair of genomes, X and Y. As the alignment of genome X and Y is divided into L_{seg} consecutive and non-overlapping segments with l_s sites, let $f(x|t)$ be the distribution of segment divergence, where $x=0, 1, \ldots, l_s$ represents the number of SSPs on a segment of the XY alignment, $t\geq 0$ is the (continuous) XY coalescent time, and $f(x|t)$ is normalized to unity by summing over x. To save computational resources, we assume an upper bound $l_s^{cutoff}\leq l_s$ to x. At $t=0$, the most

recent common ancestor (MRCA) of XY splits into two lineages; initially, the two have identical genomes, and thus $f(x|0)=\delta_{x,0}$ (where $\delta_{x,0}$ is the Kronecker delta, *i.e.*, $f(x|0)$ is non-zero only at $x=0$). At $t>0$, mutations and recombinations occur, and the evolution of $f(x|t)$ is described by the following equation:

$$\frac{df(x|t)}{dt} = 2l_s\mu \sum_{y=0}^{l_s^{cutoff}} (M(x|y) - I(x|y))f(y|t) + 2\rho \sum_{y=0}^{l_s^{cutoff}} (P(x|y,\theta,\delta_{TE},l_s) - I(x|y))f(y|t) \quad (1)$$

The first term of Eq (1) accounts for mutations on a segment—$M(x|y)$ models a mutation event, where a segment in the pair XY with y SSPs increases to $x=y+1$ SSPs during a mutation (*i.e.,* $M(x|y)=0$ for $x\neq y+1$); $I(x|y)$ is the identity matrix. For simplicity, we ignore back mutations.

The second term accounts for recombination—$P(x|y,\theta,\delta_{TE},l_s)$ models a recombination event (see Eq. (S4) of *Dixit et al.* [5] or Eq. (3) of *Dixit et al.* [6] for a detailed derivation). Since a segment can recombine with its counterpart on another genome, it assumes that each segment of a genome, along with its counterparts in different genomes of the population, have their own phylogeny, which is detached from the genomes' phylogeny, and the segment population structure is approximated by the coalescent model. For an attempted recombination between Y and an external donor D, we can use the coalescent model to calculate the probability distribution for the segment divergence δ between D and X, and then obtain x from $x=l_s\delta$. As mutation and recombination can equally occur on either X or Y, there is a factor 2 attached to both terms. See Supplementary Text for the exact form of $P(x|y,\theta,\delta_{TE},l_s)$. We solved Eq. (1) with boundary condition $f(x|0)=\delta_{x,0}$ to obtain the theoretical SSP distribution $f(x|t)$ at different coalescent times t.

We fit the theoretical distribution obtained with the CGP model to an empirical SSP distribution to infer the coalescent time of the pair of genomes. Let us consider an alignment

for a genome pair XY that is divided into L_{seg} segments, with empirical SSP distribution $g_{XY}(x)$ following the normalization condition:

$$\sum_{x=0}^{l_s^{cutoff}} g_{XY}(x) = L_{seg}$$

Let us denote the theoretical distribution as $f_{\mu,\rho,\theta,\delta TE}(x|t)$, which is normalized to unity. The probability to observe the empirical distribution $g_{XY}(x)$ given the theoretical distribution $f_{\mu,\rho,\theta,\delta TE}(x|t)$ is

$$\prod_x [f_{\mu,\rho,\theta,\delta_{TE}}(x|t)]^{g_{XY}(x)} \quad (2)$$

If we take the logarithm of this expression, it becomes the (negative) cross entropy between $g(x)$ and $f_{\mu,\rho,\theta,\delta TE}(x|t)$ [22, 23]. The higher their similarity, the higher is this negative cross entropy; it attains its maximum when $f_{\mu,\rho,\theta,\delta TE}(x|t)$ is equal to $g(x)$.

Suppose that we have n genomes (X_i, i=1,...,n), where their phylogeny is described by an ultrametric tree T; the $n(n\text{-}1)/2$ pairwise SSP distributions have evolved according to the model with parameters μ, ρ, θ, δ_{TE}. Let $t_T(X_a,X_b)$ be the coalescent time of X_a and X_b in the tree T. We use a score function, $S(X_1,X_2, \ldots,X_n|\mu, \rho, \theta, \delta_{TE},T)$, which is defined as the logarithm of the probability to observe the $n(n\text{-}1)/2$ empirical SSP distributions given the model and the tree, to quantify the model fit to the empirical SSP distributions. This score is the summation of the $n(n\text{-}1)/2$ negative mutual entropy terms:

$$S(X_1,\ldots X_n|\mu,\rho,\theta,\delta_{TE},T) = \sum_{all\ (X_a,X_b)\ pairs} log\{\prod_x [f_{\mu,\rho,\theta,\delta_{TE}}(x|t_T(X_a,X_b))]^{g_{X_aX_b}(x)}\} \quad (3)$$

Since the $n(n\text{-}1)/2$ SSP distributions are not completely independent of each other, Eq. (3) is not exactly a probability and so we call it a score. We developed an algorithm that samples the tree+model space and searches for the configuration with the maximum score using

Monte Carlo simulation with annealing and Metropolis acceptance (See Supplementary Text for details).

RESULTS

A coarse-graining approach to phylogenetic reconstruction

Figure 1 gives a brief illustration on how the proposed coarse-graining phylogenetics (CGP) algorithms fits the distribution of local single site polymorphisms (SSPs) density of the genome pairs to infer their phylogenetic tree, forgoing the reconstruction of ancestral recombination graph. In short, CGP is based on a mathematical model [5, 6] that quantitatively describes the evolution of genomic sequence divergence; this model is applicable to both nucleotide and amino acid sequences, and does not assume low recombination rate. Recombination can introduce DNA stretches characterized by a high density of substitutions, and the model considers substitution densities defined on the genomic segments. A nucleotide genome sequence alignment (or a corresponding concatenation of amino acid sequence alignments) is divided into a chain of consecutive, non-overlapping segments, each with l_s sites; for a pair of genomes, the single site polymorphisms (SSPs) on each segment are counted, resulting in an SSP distribution. CGP takes the SSP distribution of every pair of considered genomes as input. The coalescent time of two genomes can be inferred by fitting the CGP model to the empirical SSP distribution. The ultrametric tree describing the vertical component of inheritance among n genomes can be inferred from the coalescent times resulting from the fits to the $n(n\text{-}1)/2$ empirical SSP distributions, implemented by the score function of Eq. (3). We developed the CGP algorithm, which employs Monte Carlo simulation to sample the model+tree space, identifying the tree and parameters that result in the highest score.

Testing the CGP algorithm on simulated genomes

We performed Fisher-Wright simulations with recombination to generate genome sequences, allowing us to test different phylogenetic reconstruction algorithms. In the simulation, each recombination-attempting DNA stretch starts at a random site of a genome, with equal chance to be either a micro (geometric distribution, mean 100bp) or a macro stretch (geometric distribution, mean 1kb). We used three sets of parameters that correspond to prokaryotic populations with r/m=2, 40, 80 (r/m is the ratio between substitutions contributed by mutations and by recombinations; these three settings are denoted as low, intermediate, and high recombination level, respectively), and prepared the test groups, each with 4-10 genome sequences. For comparison, r/m values observed in nature range from 0.02 to 63.6 [11].

The MRCA of a group of random genomes in a simulated population has an average age close to the age of the population root node t_{root}. We would like to mimic the condition where a single lineage diverges from the rest of the population and forms its own subpopulation, so that the genomes in its subpopulation continue exchanging DNA among themselves and with the rest outside. Hence, when picking genomes in the population to form test groups, we constrained the age of the MRCA of the genomes in a test group, $t_{\text{test-group-root}}$, to be $t_{\text{test-group-root}} \ll t_{\text{root}}$ (see Supplementary Text for the details and Supplementary File S3 for the genome sequences in each test-group).

We applied CGP, as well as the previously published methods RaxML [24], ClonalFrameML [14], and Gubbins [17] to the sequences of each test group. The RAxML and Gubbins trees are midpoint-rooted. ClonalFrameML requires an initial tree as input and we used the RAxML tree. CGP uses segment size l_s=150 and l_s^{cutoff}=100. We compared each reconstructed tree with the true tree, measuring their unrooted symmetric distance (SD) [25], as well as their rooted and unrooted branch score distance (BSD) [26] to quantify the accuracy of the reconstructed phylogeny (see Supplementary File S1 for the these values); the

lower the unrooted SD / unrooted BSD / rooted BSD, the more accurate is the topology / branch lengths / root positioning, respectively. We normalized the branch lengths of each tree by its total branch length when calculating BSD.

CGP can predict the topology of a phylogeny of vertical inheritance as accurately as the other algorithms. Figure 2 shows the histograms of unrooted SD; ClonalFrameML is excluded as it uses the topology of RAxML trees. The distributions of SD of CGP are not significantly different from the distributions of RAxML and Gubbins. Two-sided Wilcoxon signed-rank tests (WSRT) at low, intermediate, and high recombination levels resulted in p=0.25, 0.69, 0.54 between CGP and RAxML, and p=0.25, 0.38, 0.92 between CGP and Gubbins.

Branch length predictions are more accurate with CGP than with alternative programs at higher levels of recombination. Figure 3 plots the distributions of the unrooted BSD of different algorithms and their z-scores; the unrooted BSD of trees reconstructed by different algorithms on the same test group sequences are pooled together to calculate their z-scores to help data visualization. The distribution of the unrooted BSD of CGP is significantly lower than RAxML and Gubbins ($p<10^{-10}$ at low, intermediate, and high recombination level compared to both RAxML and Gubbins). The unrooted BSD of CGP is significantly lower than ClonalFrameML except at low recombination levels (p=0.76, 2.2×10^{-7}, 10^{-7} at low, intermediate, and high recombination levels).

CGP can perform accurate root positioning. Figure 4 plots the distribution of the rooted BSD and their z-scores; the rooted BSD of trees reconstructed by different algorithms on the same test group sequences are pooled together to calculate their z-scores. The distribution of the rooted BSD of CGP is significantly lower than the other algorithms ($p<2\times10^{-17}$ at all recombination levels for CGP compared with the other three algorithms).

Testing the CGP algorithm on real *E. coli* genomes

We tested CGP, RAxML, ClonalFrameML, and Gubbins using *E. coli* and *Shigella* genome sequences (see Supplementary Table S1 for their names); we refer to them as *E. coli*, as these two species have intertwined phylogenies. We prepared test groups, each with 10 random strains, where each strain is represented by a nucleotide and an amino acid sequence made from its concatenated core genes (see Supplementary File S2 for the strains in each test group, and also the 1,636 orthologous gene families of core genes; see Supplementary File S3 for their sequences). We applied CGP, RAxML, ClonalFrameML (with the topology from RAxML trees), and Gubbins on the nucleotide sequences, and CGP and RAxML on amino acid sequences; the RAxML and Gubbins trees are midpoint-rooted. CGP uses segment length l_s=150 and l_s^{cutoff}=100 for nucleotide sequences, and l_s=50, l_s^{cutoff}=50 for amino acid sequences.

To assess the accuracy of the phylogenetic trees reconstructed by the different algorithms, we compared the reconstructed trees with the phylogenetic signal inferred from the distribution of orthologous gene families in different genomes. We applied the GLOOME algorithm [27], which considers the interior nodes of the tree as ancestral strains and reconstructs their gene distribution; it takes a tree and the presence-and-absence of genes across the extant strains as input, and performs a reconstruction of presence-and-absence of genes in the ancestral strains based on the GLOOME posterior likelihood (GPL). We used GPL as a score to quantify the accuracy of the tree fed into GLOOME; the more consistent the phylogenetic signal from the gene distributions with a given tree, the higher the GPL (see Supplementary File S2 for the GPL values of the reconstructed trees).

Figure 5 plots the distribution of the GPLs and the corresponding z-scores; the GPLs of trees of the same test groups reconstructed by different methods are pooled together to calculate the z-scores. Trees reconstructed from amino acid sequences have a higher GPL

than trees calculated from nucleotide sequences; moreover, CGP trees based on amino acids are more accurate than trees calculated using RAxML ($p<4\times10^{-14}$ when comparing CGP on amino acid sequences with any other algorithm; other recombination-aware algorithms are not applicable to amino acid sequences). Considering only trees reconstructed from nucleotide sequences, the CGP trees generally have higher GPL than RAxML, ClonalFrameML, and Gubbins trees ($p=5.2\times10^{-4}$, 0.049, 1.4×10^{-15}, respectively).

DISCUSSION

We introduced a coarse-graining phylogenetic (CGP) model, which infers a phylogenetic tree from the estimated pairwise coalescent times of genomes. We conducted extensive analyses to compare the accuracy of the CGP algorithm with other state-of-the-art algorithms to demonstrate its ability to reliably predict the topology, branch lengths, and root positioning of phylogenetic trees. The CGP model does not rely on the assumption of low recombination rates, which allows it to predict branch lengths accurately even if the vast majority of the considered genome segments have experienced recombination on the timescale covered by the phylogeny.

Analyses performed on the real *E. coli* genome sequences showed that trees reconstructed from core genome amino acid sequences are more accurate, *i.e.*, more consistent with the signal inferred from the distribution of genes in the extant genomes, than trees calculated from nucleotide sequences. Amino acid sequences of core genes tend to evolve more slowly than the corresponding DNA sequences, as these genes show dN/dS values <1 [28], and accordingly, the divergence of a pair of amino acid sequences is lower than that of their nucleotide counterparts (see Supplementary File S3 for nucleotide and amino acid sequence divergence between *E. coli* genome pairs). Thus, amino acid sequences

may be more "clonal" than nucleotide sequences and thus may provide more accurate phylogenetic signals.

The major source of error of the CGP algorithm comes from the mismatch between the genomic segments as basic unit of the algorithm and the genomic stretches affected by homologous recombination, as the algorithm does not try to match the boundary of the segments to the boundary of the actual recombination stretches. This mismatch gives rise to segments that lie on the boundary and cover multiple recombination stretches, which subsequently reduces the accuracy of the predictions of the algorithm. Hence, a possible direction for further development is to find out the criteria to fine tune the segment size l_s so as to minimize these boundary-overlapping segments; alternatively, we can improve the theoretical model so that the segments do not have to be equal-sized and we can match the segments to the recombination stretches.

The computational demand of the CGP algorithm is independent of sequence length, as CGP considers only SSP distributions that are represented by a vector of $1+l_s^{cutoff}$ elements in the computer code. Calculation of the CGP score (Eq. (3)) involves multiplication of $(1+l_s^{cutoff})\times(1+l_s^{cutoff})$ matrices; thus, the computational time scales as $O((l_s^{cutoff})^k)$, where $k\leq 3$ depends on the algorithm that carries out the matrix operations. When reconstructing a tree of n genomes, the score calculation involves the summation over $n(n-1)/2$ pairs, making it scale as $O(n^2)$. The segment size l_s affects the efficiency and accuracy of the algorithm. While a smaller l_s leads to lower accuracy, increasing l_s leads to higher computational demand; a large l_s combined with a small l_s^{cutoff} can also reduce the accuracy. Hence, one needs to set l_s and l_s^{cutoff} carefully to balance the need for speed and accuracy.

The current algorithm that implements the CGP model is very simple; it should be considered a proof of concept. While it makes use of Monte Carlo simulation to sample the tree+parameter space, a hill-climbing method may be more efficient. Other possible

346 improvements involve better local search moves in the ultrametric tree space; one might even
347 drop the stringent ultrametricity constraint, and replace it with a more flexible matrix-tree
348 mapping method that allows a more efficient search in the tree space. The mutation matrix in
349 the current model can be improved to include back mutations and a more complex mutation
350 model. We leave these possible improvements to future studies.
351

352 **DECLARATIONS**

353 **Ethics approval and consent to participate**

354 Not applicable

355 **Consent for publication**

356 Not applicable

357 **Availability of data and materials**

358 All data generated or analysed during this study are included in this published article, its
359 supplementary information files, and GitHub repository (https://github.com/TinPang/coarse-
360 graining-phylogenetics).

361 **Competing interests**

362 The authors declare that they have no competing interests

363 **Funding**

364 This work was supported by the German Research Foundation (CRC 680 and CRC 1310).

Authors' contributions

Not applicable

Acknowledgements

We would like to thank Martin Lercher and Arndt von Haeseler for helpful comments and advice.

SUPPLEMENTARY MATERIALS

1. Source code of the CGP algorithm: https://github.com/TinPang/coarse-graining-phylogenetics
2. Supplementary Text, Figures and Tables.
3. Supplementary File S1: Analyses of the simulated genomes: the symmetric distance (SD) and branch score distance (BSD) between the reconstructed trees and the true trees.
4. Supplementary File S2: Analyses of the real genomes: b-number of the *E. coli* core genes used to make the 'super-gene' sequences, strains in each test-group, and also the GLOOME posterior likelihood (GPL) of the reconstructed trees.
5. Supplementary File S3: Sequences and their phylogenetic trees: sequences of the simulated genomes in different test-groups, their true trees and also the phylogenetic trees reconstructed by different algorithms; sequences of the *E. coli* genomes and their trees reconstructed by different algorithms; genes to orthologous gene families map provided by ProteinORTHO.

388

389 **REFERENCES**

390 1. Pál C, Papp B, Lercher MJ. Adaptive evolution of bacterial metabolic networks by
391 horizontal gene transfer. Nat Genet. 2005;37:1372–5.

392 2. Ochman H, Lawrence JG, Groisman EA. Lateral gene transfer and the nature of bacterial
393 innovation. Nature. 2000;405:299–304.

394 3. Lang AS, Zhaxybayeva O, Beatty JT. Gene transfer agents: phage-like elements of genetic
395 exchange. Nat Rev Microbiol. 2012;10:472–82.

396 4. Wilson GG, Murray NE. Restriction and Modification Systems. Annu Rev Genet.
397 1991;25:585–627.

398 5. Dixit PD, Pang TY, Studier FW, Maslov S. Recombinant transfer in the basic genome of
399 Escherichia coli. Proc Natl Acad Sci U S A. 2015;112:9070–5.

400 6. Dixit PD, Pang TY, Maslov S. Recombination-Driven Genome Evolution and Stability of
401 Bacterial Species. Genetics. 2017;:genetics.300061.2017.

402 7. Huddleston JR. Horizontal gene transfer in the human gastrointestinal tract: potential
403 spread of antibiotic resistance genes. Infect Drug Resist. 2014;7:167–76.

404 8. Takeuchi N, Kaneko K, Koonin E. Horizontal Gene Transfer Can Rescue Prokaryotes
405 from Muller's Ratchet: Benefit of DNA from Dead Cells and Population Subdivision. G3
406 GenesGenomesGenetics. 2014;4:325–39.

407 9. Croucher NJ, Mostowy R, Wymant C, Turner P, Bentley SD, Fraser C. Horizontal DNA
408 Transfer Mechanisms of Bacteria as Weapons of Intragenomic Conflict. PLOS Biol.
409 2016;14:e1002394.

410 10. Spratt BG. Multilocus sequence typing: molecular typing of bacterial pathogens in an era
411 of rapid DNA sequencing and the internet. Curr Opin Microbiol. 1999;2:312–6.

412 11. Vos M, Didelot X. A comparison of homologous recombination rates in bacteria and
413 archaea. ISME J. 2009;3:199–208.

414 12. Schierup MH, Hein J. Consequences of recombination on traditional phylogenetic
415 analysis. Genetics. 2000;156:879–91.

416 13. Didelot X, Falush D. Inference of Bacterial Microevolution Using Multilocus Sequence
417 Data. Genetics. 2007;175:1251–66.

418 14. Didelot X, Wilson DJ. ClonalFrameML: Efficient Inference of Recombination in Whole
419 Bacterial Genomes. PLOS Comput Biol. 2015;11:e1004041.

420 15. Didelot X, Lawson D, Darling A, Falush D. Inference of Homologous Recombination in
421 Bacteria Using Whole-Genome Sequences. Genetics. 2010;186:1435–49.

16. Vaughan TG, Welch D, Drummond AJ, Biggs PJ, George T, French NP. Inferring Ancestral Recombination Graphs from Bacterial Genomic Data. Genetics. 2017;205:857–70.

17. Croucher NJ, Page AJ, Connor TR, Delaney AJ, Keane JA, Bentley SD, et al. Rapid phylogenetic analysis of large samples of recombinant bacterial whole genome sequences using Gubbins. Nucleic Acids Res. 2015;43:e15–e15.

18. Marttinen P, Hanage WP, Croucher NJ, Connor TR, Harris SR, Bentley SD, et al. Detection of recombination events in bacterial genomes from large population samples. Nucleic Acids Res. 2012;40:e6.

19. Mostowy R, Croucher NJ, Andam CP, Corander J, Hanage WP, Marttinen P. Efficient Inference of Recent and Ancestral Recombination within Bacterial Populations. Mol Biol Evol. 2017;34:1167–82.

20. Kingman JFC. Origins of the Coalescent: 1974-1982. Genetics. 2000;156:1461–3.

21. Fraser C, Hanage WP, Spratt BG. Recombination and the nature of bacterial speciation. Science. 2007;315:476–80.

22. Rubinstein RY, Kroese DP. The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning. Springer Science & Business Media; 2004.

23. Boer P-T de, Kroese DP, Mannor S, Rubinstein RY. A Tutorial on the Cross-Entropy Method. Ann Oper Res. 2005;134:19–67.

24. Stamatakis A. RAxML Version 8: A tool for Phylogenetic Analysis and Post-Analysis of Large Phylogenies. Bioinformatics. 2014;:btu033.

25. Robinson DF, Foulds LR. Comparison of phylogenetic trees. Math Biosci. 1981;53:131–47.

26. Kuhner MK, Felsenstein J. A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. Mol Biol Evol. 1994;11:459–68.

27. Cohen O, Ashkenazy H, Belinky F, Huchon D, Pupko T. GLOOME: gain loss mapping engine. Bioinformatics. 2010;26:2914–5.

28. Lapierre M, Blin C, Lambert A, Achaz G, Rocha EPC. The Impact of Selection, Gene Conversion, and Biased Sampling on the Assessment of Microbial Demography. Mol Biol Evol. 2016;33:1711–25.

FIGURES CAPTIONS

Figure 1. Illustration of the procedure of the proposed CGP algorithm. (a) The algorithm takes n aligned sequences as input, which can be nucleotide or amino acid sequences; substitutions on the sequences are represented by coloured markers. (b) Each of the $n(n\text{-}1)/2$ genome pairs is divided into equal-sized segments, and the pairwise substitutions on each segment is enumerated to obtain the distribution of local single site polymorphisms (SSPs) density (denoted as $g(x)$). (c) The algorithm aims to infer the distance matrix of the genome sequence pairs from the $n(n\text{-}1)/2$ SSP distributions. (d) In particular, the algorithm fits the empirical SSP distributions with a model; the input of this model involves a matrix of $n(n\text{-}1)/2$ coalescent times and other model parameters (mutation rate μ, recombination rate ρ, average population divergence θ and transfer efficiency δ_{TE}). (e) In the fitting process, the $n(n\text{-}1)/2$ coalescent times are constrained (matrix cells with the same colour have the same value), such that the matrix can be bijectively mapped to a UPGMA tree. (f) the algorithm explores the model parameter space and tree space to obtain the best fit ultrametric tree.

Figure 2. Histograms showing the distributions of the unrooted symmetric distance (SD) between true trees and trees reconstructed by CGP, RaxML, and Gubbins, from genome sequences derived from Fisher-Wright simulations with low, intermediate, and high recombination levels.

Figure 3. Boxplots showing the distributions of the unrooted branch score distance (BSD) and the distributions of the z-score of unrooted BSD between the true trees and trees reconstructed by CGP, RaxML, and Gubbins, from genome sequences from Fisher-Wright simulations with low, intermediate, and high recombination levels.

Figure 4. Boxplots showing the distributions of the rooted branch score distance (BSD) and the distributions of the z-score of rooted BSD between the true trees and trees reconstructed

by CGP, RaxML, and Gubbins, from genome sequences from Fisher-Wright simulations with low, intermediate, and high recombination levels.

Figure 5. Boxplots showing the distributions of the GLOOME posterior likelihood (GPL) and the distributions of the z-scores of GPL. The trees were reconstructed from observed *E. coli* genome sequences, applying CGP, RAxML, ClonalFrameML, and Gubbins to nucleotide sequences, and CGP and RAxML to amino acid sequences.

FIGURES

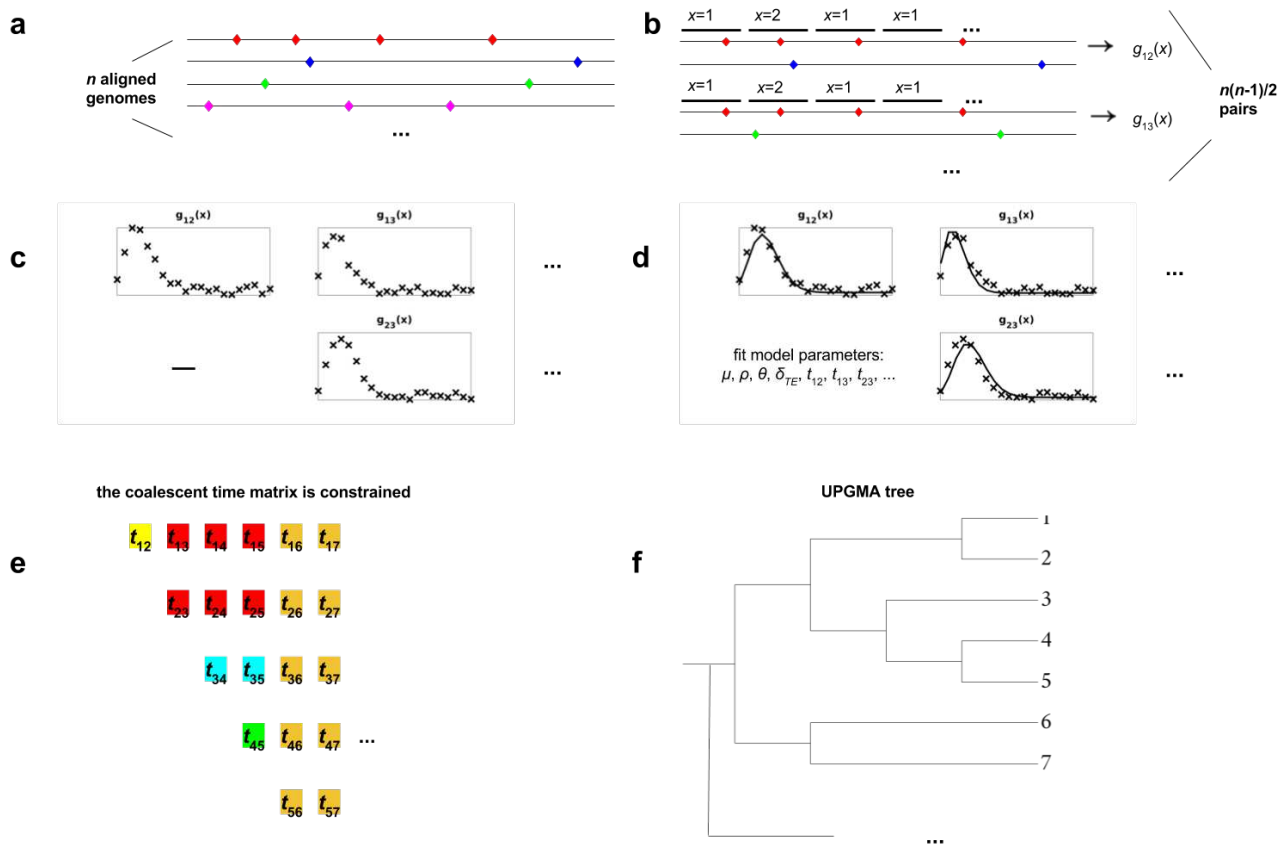
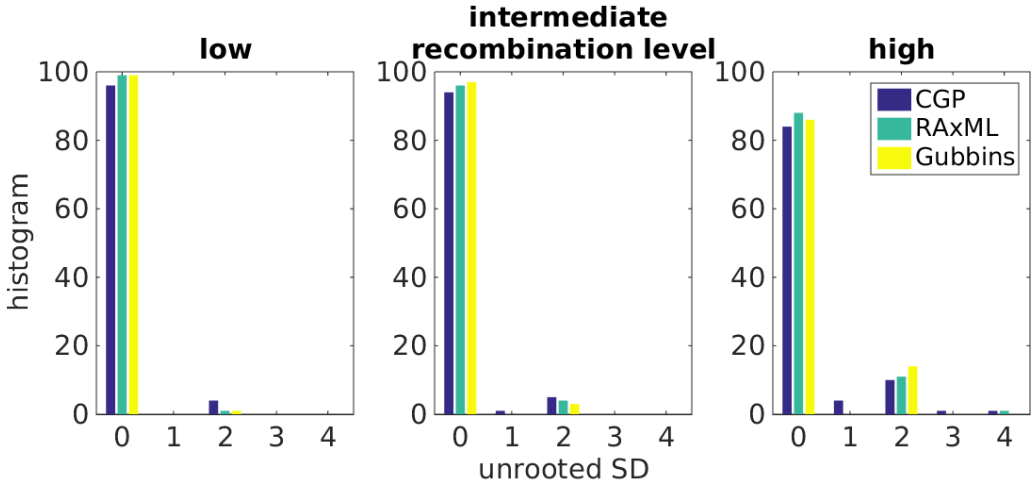


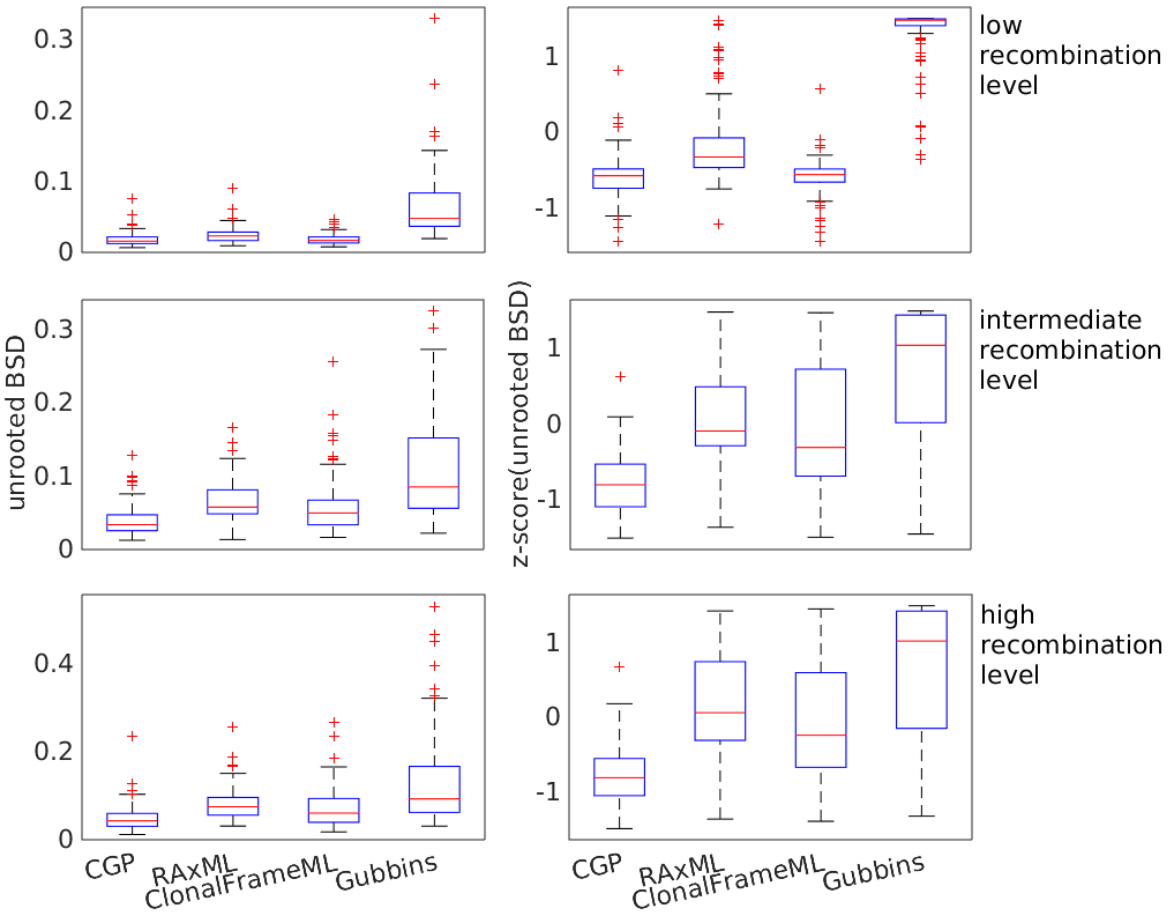
Figure 3.



492

493 **Figure 2.**

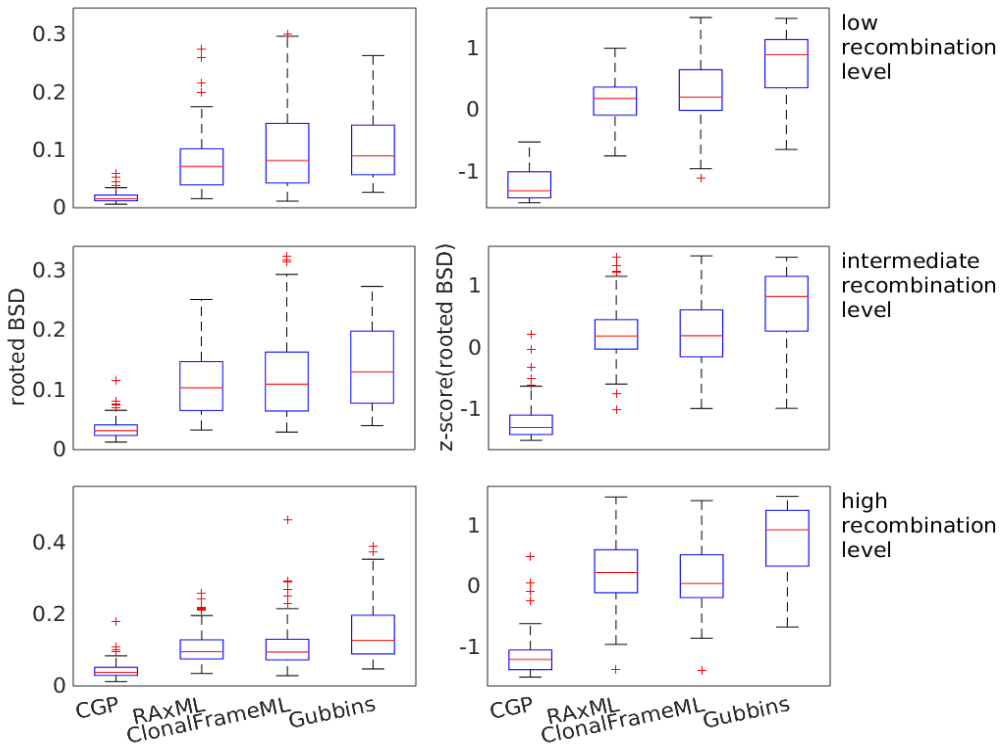
494



495

496 **Figure 3.**

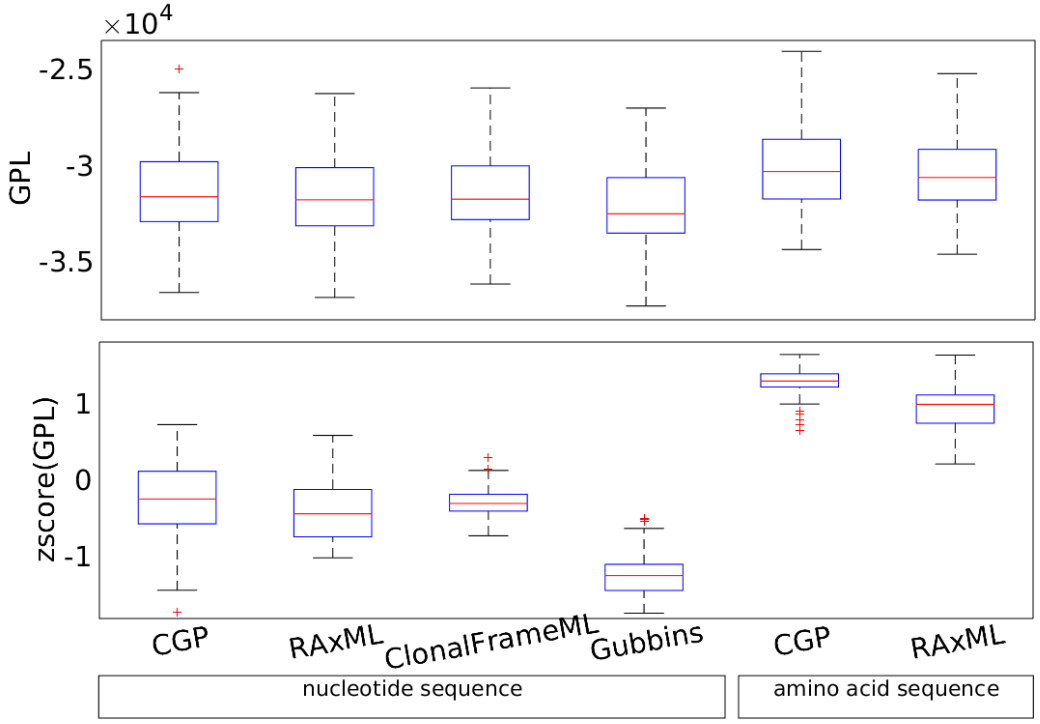
497



498

499 **Figure 4.**

500



501

502 **Figure 5.**

Figures

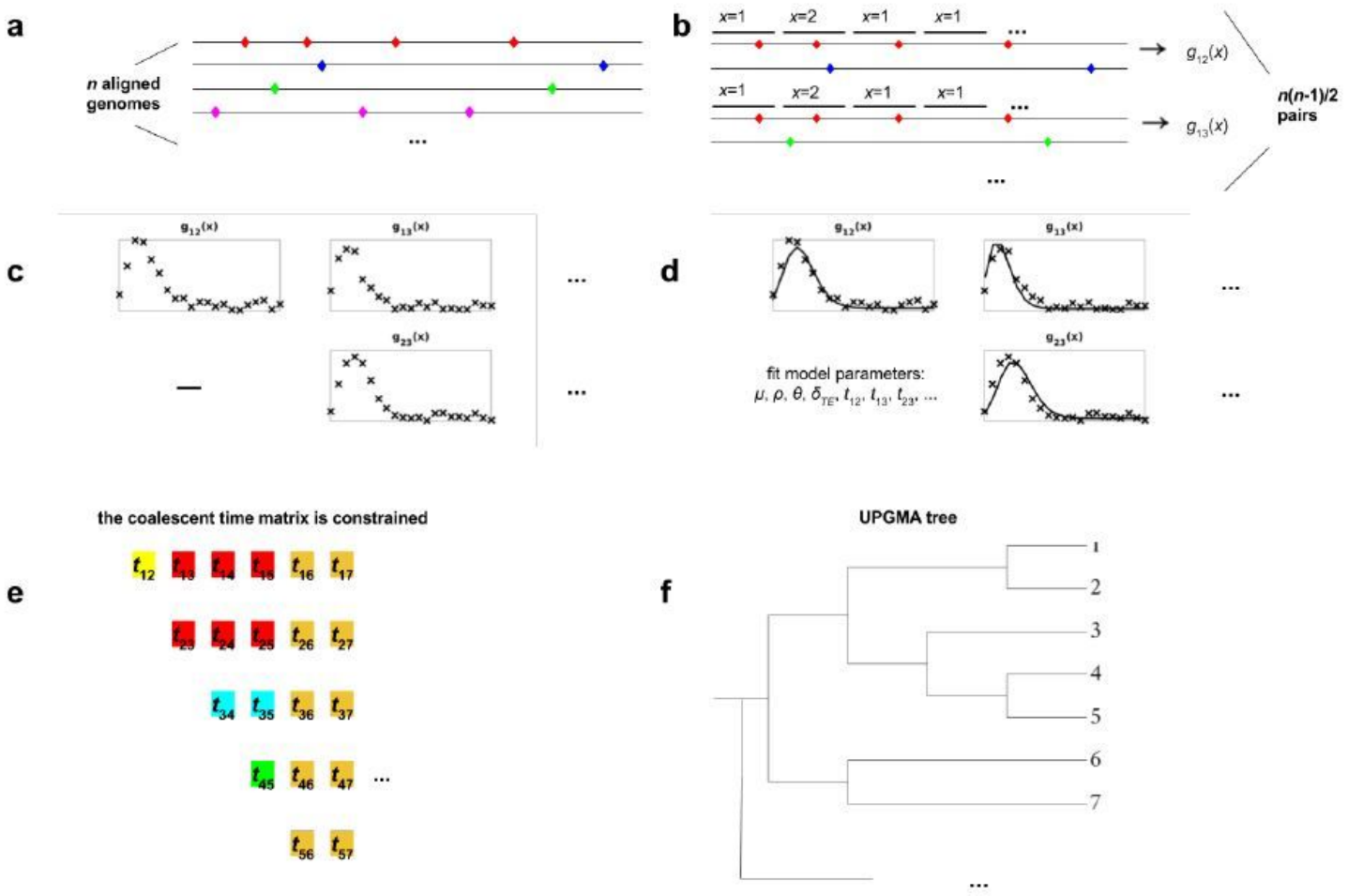


Figure 1

Illustration of the procedure of the proposed CGP algorithm. (a) The algorithm takes n aligned sequences as input, which can be nucleotide or amino acid sequences; substitutions on the sequences are represented by coloured markers. (b) Each of the n(n-1)/2 genome pairs is divided into equal-sized segments, and the pairwise substitutions on each segment is enumerated to obtain the distribution of local single site polymorphisms (SSPs) density (denoted as g(x)). (c) The algorithm aims to infer the distance matrix of the genome sequence pairs from the n(n-1)/2 SSP distributions. (d) In particular, the algorithm fits the empirical SSP distributions with a model; the input of this model involves a matrix of n(n-1)/2 coalescent times and other model parameters (mutation rate μ, recombination rate ρ, average population divergence θ and transfer efficiency δTE). (e) In the fitting process, the n(n-1)/2 coalescent times are constrained (matrix cells with the same colour have the same 467 value), such that the matrix can be bijectively mapped to a UPGMA tree. (f) the algorithm 468 explores the model parameter space and tree space to obtain the best fit ultrametric tree.

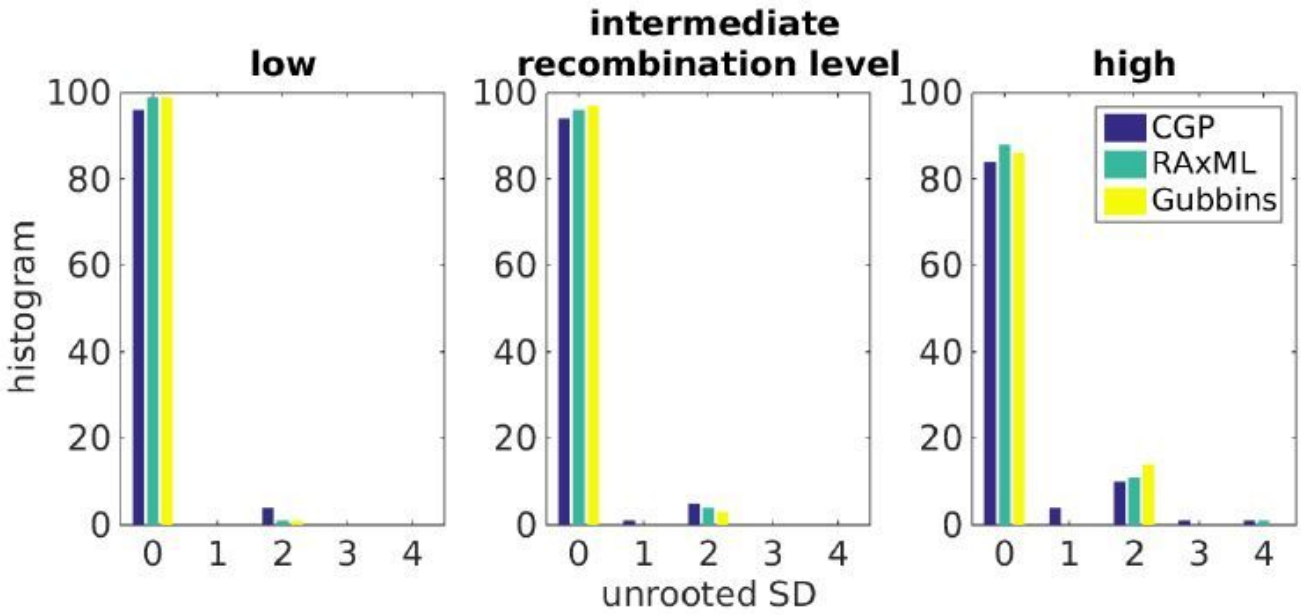


Figure 2

Histograms showing the distributions of the unrooted symmetric distance (SD) between true trees and trees reconstructed by CGP, RaxML, and Gubbins, from genome sequences derived from Fisher-Wright simulations with low, intermediate, and high recombination levels.

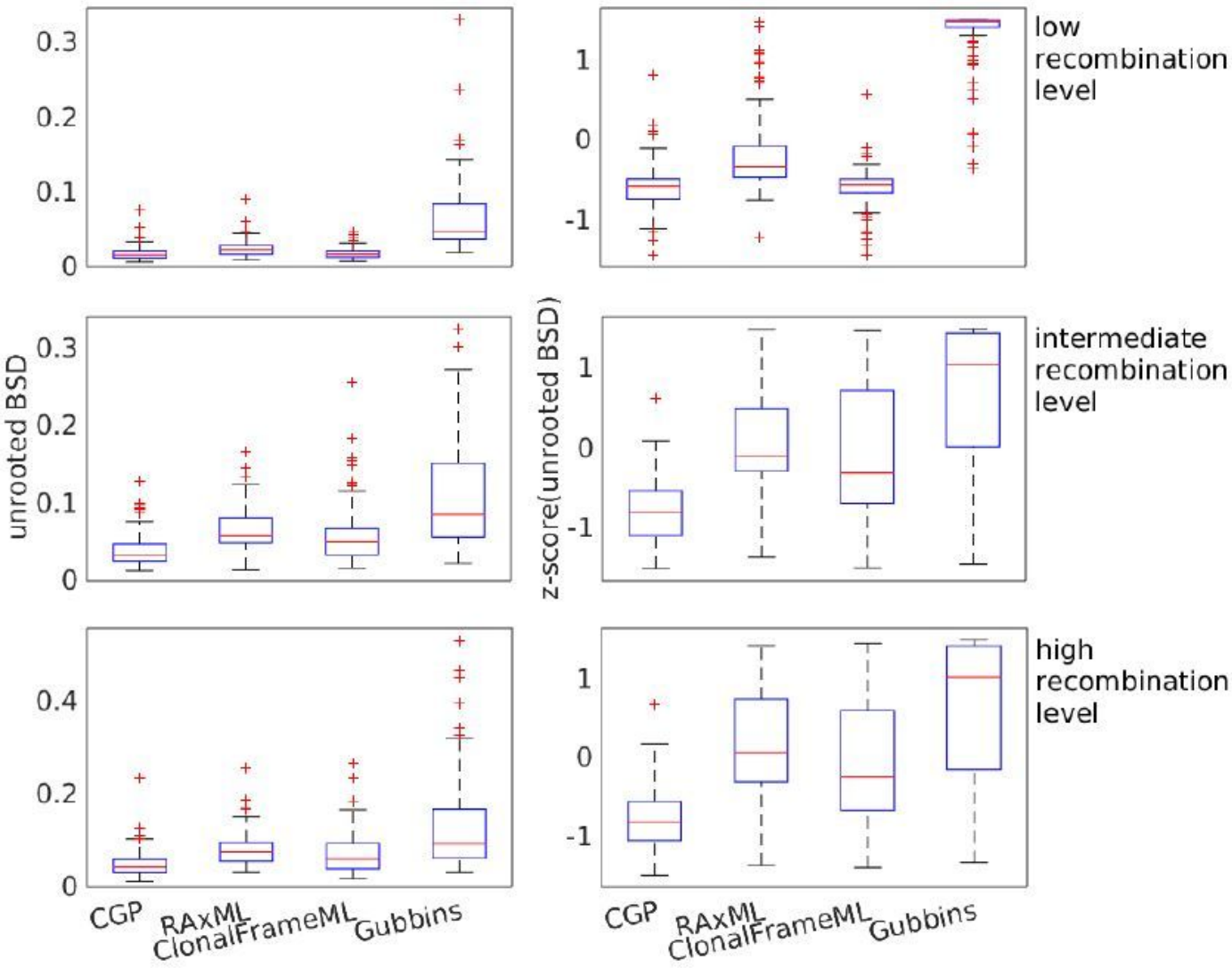


Figure 3

Boxplots showing the distributions of the unrooted branch score distance (BSD) and the distributions of the z-score of unrooted BSD between the true trees and trees reconstructed by CGP, RaxML, and Gubbins, from genome sequences from Fisher-Wright simulations with low, intermediate, and high recombination levels.

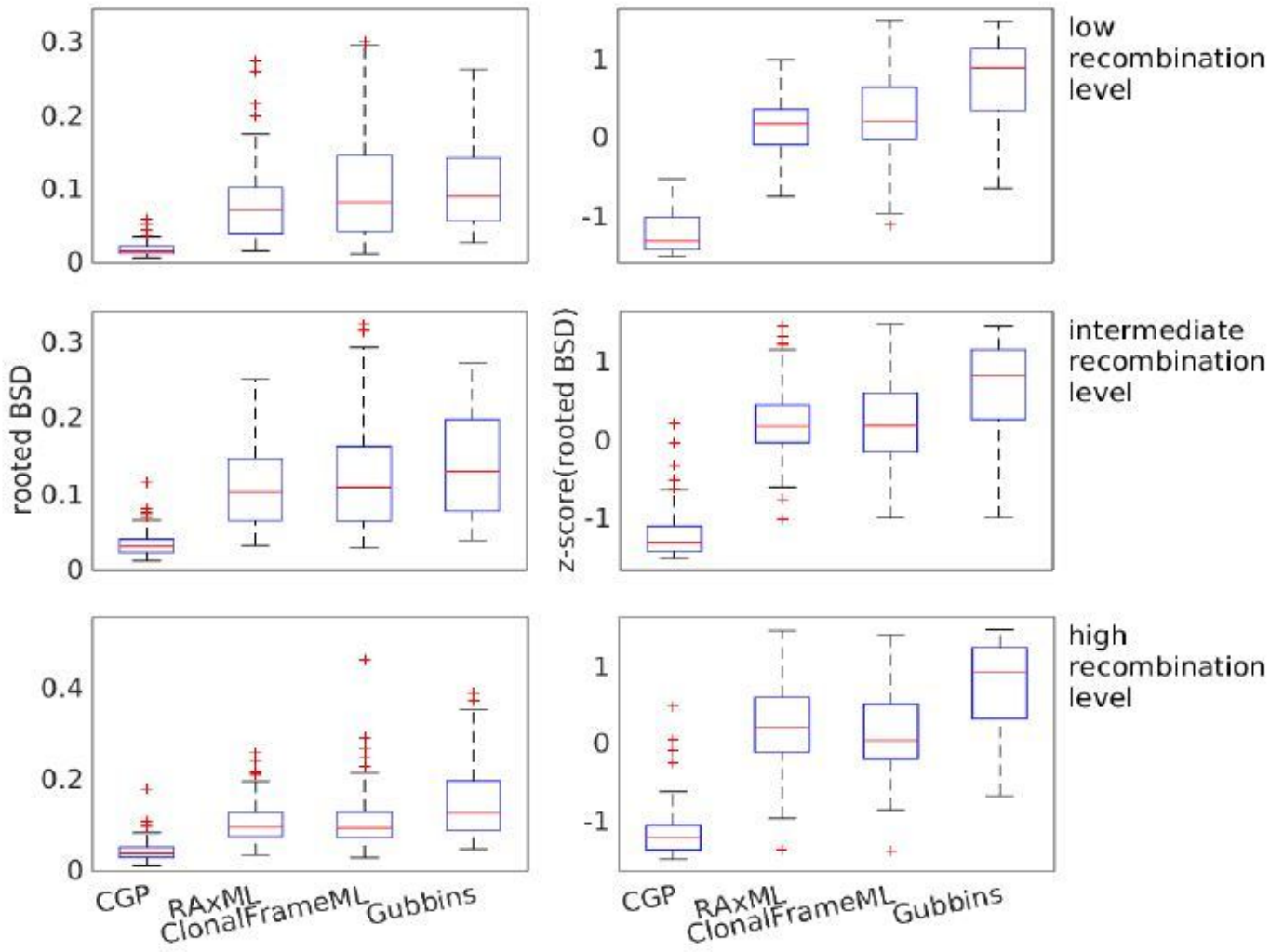


Figure 4

Boxplots showing the distributions of the rooted branch score distance (BSD) and the distributions of the z-score of rooted BSD between the true trees and trees reconstructed by CGP, RaxML, and Gubbins, from genome sequences from Fisher-Wright simulations with 480 low, intermediate, and high recombination levels.

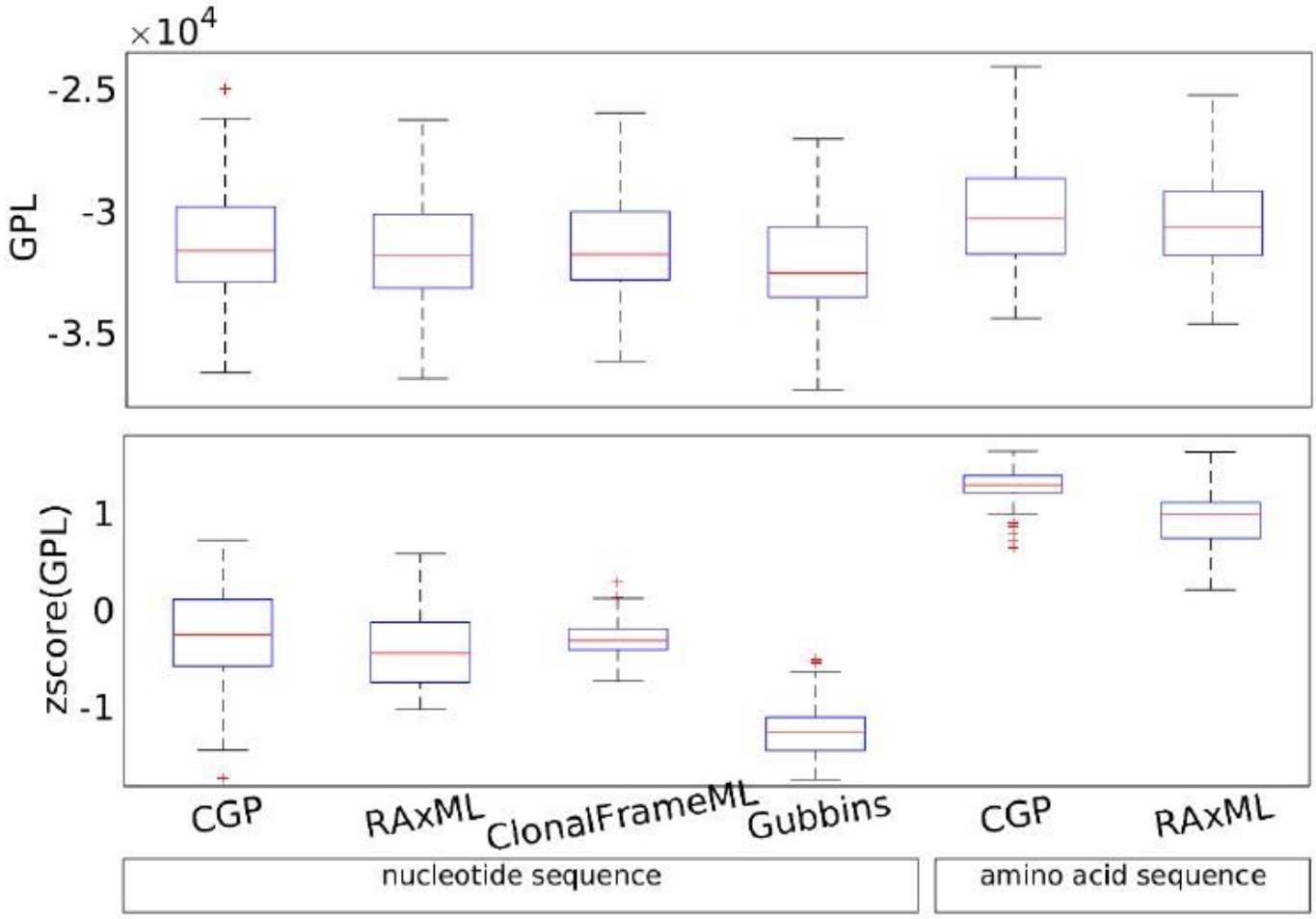


Figure 5

Boxplots showing the distributions of the GLOOME posterior likelihood (GPL) and the distributions of the z-scores of GPL. The trees were reconstructed from observed E. coli genome sequences, applying CGP, RAxML, ClonalFrameML, and Gubbins to nucleotide sequences, and CGP and RAxML to amino acid sequences.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementaryFileS3.tar.xz
- supplementarytext.pdf
- SupplementaryFileS2resultsrealgenomes.xlsx
- SupplementaryFileS1resultssimulatedgenomes.xlsx