

# The Bacterial Sequential Markov Coalescent

Nicola De Maio<sup>\*,†,1</sup> and Daniel J. Wilson<sup>\*,†,‡</sup>

<sup>\*</sup>Institute for Emerging Infections, Oxford Martin School, <sup>†</sup>Nuffield Department of Medicine, and <sup>‡</sup>Wellcome Trust Centre for Human Genetics, University of Oxford, Oxford, OX1 3PA, United Kingdom

ORCID ID: 0000-0002-1776-8564 (N.D.M.)

**ABSTRACT** Bacteria can exchange and acquire new genetic material from other organisms directly and via the environment. This process, known as bacterial recombination, has a strong impact on the evolution of bacteria, for example, leading to the spread of antibiotic resistance across clades and species, and to the avoidance of clonal interference. Recombination hinders phylogenetic and transmission inference because it creates patterns of substitutions (homoplasies) inconsistent with the hypothesis of a single evolutionary tree. Bacterial recombination is typically modeled as statistically akin to gene conversion in eukaryotes, *i.e.*, using the coalescent with gene conversion (CGC). However, this model can be very computationally demanding as it needs to account for the correlations of evolutionary histories of even distant loci. So, with the increasing popularity of whole genome sequencing, the need has emerged for a faster approach to model and simulate bacterial genome evolution. We present a new model that approximates the coalescent with gene conversion: the bacterial sequential Markov coalescent (BSMC). Our approach is based on a similar idea to the sequential Markov coalescent (SMC)—an approximation of the coalescent with crossover recombination. However, bacterial recombination poses hurdles to a sequential Markov approximation, as it leads to strong correlations and linkage disequilibrium across very distant sites in the genome. Our BSMC overcomes these difficulties, and shows a considerable reduction in computational demand compared to the exact CGC, and very similar patterns in simulated data. We implemented our BSMC model within new simulation software FastSimBac. In addition to the decreased computational demand compared to previous bacterial genome evolution simulators, FastSimBac provides more general options for evolutionary scenarios, allowing population structure with migration, speciation, population size changes, and recombination hotspots. FastSimBac is available from <https://bitbucket.org/nicofmay/fastsimbac>, and is distributed as open source under the terms of the GNU General Public License. Lastly, we use the BSMC within an Approximate Bayesian Computation (ABC) inference scheme, and suggest that parameters simulated under the exact CGC can correctly be recovered, further showcasing the accuracy of the BSMC. With this ABC we infer recombination rate, mutation rate, and recombination tract length of *Bacillus cereus* from a whole genome alignment.

**KEYWORDS** bacterial evolution; recombination; coalescent; simulations; ABC

**B**ACTERIAL whole-genome sequencing has rapidly replaced multilocus sequence typing for population analyses of bacterial pathogens thanks to its fast and cost-effective provision of higher resolution genetic information (Didelot

*et al.* 2012; Wilson 2012). Methods using genomic data to infer epidemiological, phylogeographic, phylodynamic, and evolutionary patterns are often hampered by recombination (*e.g.*, Schierup and Hein 2000; Posada and Crandall 2002), and the bacterial setting is no exception (Hedge and Wilson 2014). Recombination causes different sites in the genome to have different inheritance histories. For these reasons, in recent years many methods have been proposed to measure, identify, and account for bacterial recombination (*e.g.*, Didelot and Falush 2007; Marttinen *et al.* 2008; Tang *et al.* 2009; Didelot *et al.* 2010; Marttinen *et al.* 2012; Croucher *et al.* 2014; Didelot and Wilson 2015). Among these, simulators of bacterial evolution (*e.g.*, Didelot *et al.* 2009b; Mostowy *et al.* 2014; Brown *et al.* 2015) have been used for parameter inference and hypothesis testing (Fearnhead

Copyright © 2017 Maio and Wilson

doi: <https://doi.org/10.1534/genetics.116.198796>

Manuscript received December 1, 2016; accepted for publication February 14, 2017; published Early Online March 2, 2017.

Available freely online through the author-supported open access option.

This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

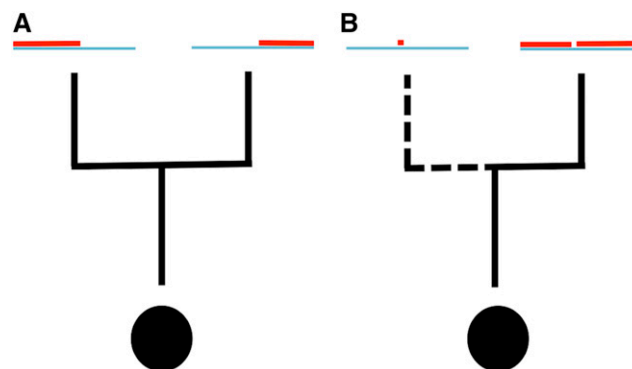
Supplemental material is available online at [www.genetics.org/lookup/suppl/doi:10.1534/genetics.116.198796/-/DC1](http://www.genetics.org/lookup/suppl/doi:10.1534/genetics.116.198796/-/DC1).

<sup>1</sup>Corresponding author: John Radcliffe Hospital, Headley Way, Oxford, OX3 9DU, United Kingdom. E-mail: [nicola.de.maio.85@gmail.com](mailto:nicola.de.maio.85@gmail.com) and [nicola.demaio@ndm.ox.ac.uk](mailto:nicola.demaio@ndm.ox.ac.uk)

*et al.* 2005; Fraser *et al.* 2005; Wilson *et al.* 2009; Ansari and Didelot 2014), and for benchmarking (e.g., Falush *et al.* 2006; Didelot and Falush 2007; Turner *et al.* 2007; Buckee *et al.* 2008; Marttinen *et al.* 2012; Hedge and Wilson 2014).

Simulating bacterial evolution poses specific difficulties as the process of bacterial recombination is very different to that of other organisms. Eukaryotic recombination is predominantly modeled as a cross-over process, with recombination events breaking a chromosome into two parts with different ancestries (Figure 1). While it is possible to simulate eukaryotic evolution with recombination forward in time (Peng and Kimmel 2005; Carvajal-Rodríguez 2008; Hernandez 2008; Arenas 2013), coalescent-based (Kingman 1982) backward in time models (Hudson 1983; Griffiths and Marjoram 1997; Wiuf and Hein 1999) are usually more computationally efficient (e.g., Hudson 2002; Arenas and Posada 2007, 2010; Ewing and Hermisson 2010; Excoffier and Foll 2011). Yet, the coalescent with recombination itself may not be sufficiently fast when large genomic segments are considered (McVean and Cardin 2005). One of the reasons is that the structure describing the evolutionary history of all positions (the ancestral recombination graph, ARG) grows subexponentially with genome size and recombination rate (Wiuf and Hein 1999). For this reason, a faster approximation to the coalescent with recombination, the sequential Markov coalescent [SMC, see McVean and Cardin (2005), Marjoram and Wall (2006)] was proposed. Similar to the exact sequential form of the coalescent with recombination (Wiuf and Hein 1999), the SMC starts by considering one evolutionary tree at the left (*i.e.*, 5') end of the sequence, and generates new trees affected by recombination as it moves toward the right (3') end. However, the SMC does not generate an ARG, but rather a sequence of local trees. The SMC makes the simplifying assumption that, if the local tree for the considered position is known, then all local trees to its left can be ignored when considering trees to its right. In fact, crossover recombination makes evolutionary histories less correlated as physical distance increases. This model has been extended to incorporate complex population history (Chen *et al.* 2009), and to have improved accuracy (Wang *et al.* 2014) and computational efficiency (Staab *et al.* 2015).

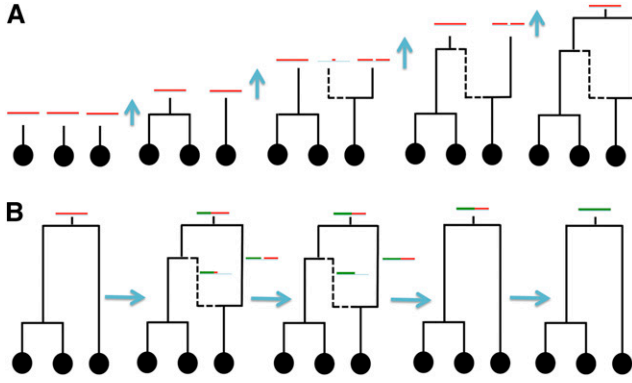
Bacterial recombination is different from eukaryotic recombination (Smith *et al.* 1993, 2000), and is generally modeled like gene conversion: a bacterial recombination event imports only a small fragment of DNA from a donor genome, while most of the genome is inherited clonally (Figure 1). This results in sites very distantly located in the genome remaining very tightly linked genetically. In fact, a single genealogy, known as the clonal frame (Milkman and Bridges 1990), represents the evolutionary history of all nonrecombining sites, no matter how physically far they are from each other. So, methods for eukaryotic recombination cannot be applied to bacteria at genomic scales. While bacterial evolution can be simulated forward in time, backward in time coalescent methods are usually more efficient, and are generally based on the coalescent with gene conversion [CGC,



**Figure 1** Graphical representation of eukaryotic and bacterial recombination models. Black circles represent sampled sequences, black lines are ancestral lineages (dashed if they represent bacterial recombination lineages). Blue segments represent the genome sequence, and red segments represent the portion of the genome that is ancestral to the particular lineage. (A) Crossover event: the entire genome to the left of the crossover site is inherited from one parent; the entire genome to the right is inherited from the other parent. (B) Gene conversion, or bacterial recombination: most of the genome is inherited from a single parent lineage, except a short segment.

see Wiuf and Hein (2000), and Figure 2A]. Recently, efficient methods implementing the CGC have been developed for simulating bacterial evolution (Didelot *et al.* 2009b; Brown *et al.* 2015). However, these approaches struggle to simulate whole genomes at high recombination rates (e.g., requiring up to hours for one bacterial genome alignment with  $\rho > 0.01$ , see Brown *et al.* 2015, and *Results*). Like the coalescent with crossover recombination, the CGC also generates large ARGs, and this contributes to the computational demand.

Here, we present a new approximation to the CGC (Figure 2B), inspired by the SMC, to efficiently and accurately model bacterial recombination. We model the clonal frame, and simulate the coalescent and recombination processes from one end of the genome to the other, conditioning on the clonal frame throughout. However, we ignore recombination events that occurred at distant, previously considered, positions. This approach differs from other approximations to the CGC (e.g., Didelot *et al.* 2010; Ansari and Didelot 2014), as we can simulate entire genomes while allowing recombining lineages with overlapping ancestral material to coalesce with one another, and allowing recombination events to split the ancestral material of recombinant lineages. In fact, frequent recombination events can break down ancestral material intervals further and further, reducing them far below the expected length of an individual recombination interval. Ignoring these complexities leads to biases when considering elevated recombination rates (Didelot *et al.* 2010), and, by accounting for them, we aim to produce a model more faithful to the CGC. We call this model the bacterial sequential Markov coalescent (BSMC), which we implement within new simulation software called FastSimBac. FastSimBac is faster than previous methods (between one to two orders



**Figure 2** Graphical representation of the bacterial coalescent (CGC) and BSMC models. Black circles represent sampled genomes, black lines are ancestral lineages (continuous if they belong to the clonal frame, dashed otherwise). Red segments represent, for each extant lineage, the portion of the genome that is ancestral to any sampled descendent of that lineage. Time is considered backward from bottom to top, and mergers of lineages represent coalescent events. (A) Example of simulation under the CGC; recombination and coalescent events are simulated backward in time starting with one lineage per sample at the present. (B) Example of BSMC simulation: first a clonal frame is simulated; then the process moves left to right across the genome (which for simplicity is linear), and left portions of the genome are gradually forgotten (represented in green). The BSMC stops at each recombination start and end position; recombination events are forgotten at their end, but the clonal frame is never forgotten.

of magnitude for typical genome sizes and recombination rates). Also, by building on top of popular simulators *ms* (Hudson 2002) and *MaCS* (Chen *et al.* 2009), our software can simulate general evolutionary scenarios, allowing migration, speciation, demographic changes, recombination hotspots, and between-species recombination. We show that the BSMC can accurately approximate the exact CGC, and can be used to infer recombination parameters using Approximate Bayesian Computation (ABC). We demonstrate its utility by inferring *Bacillus cereus* recombination and mutation parameters from a whole genome alignment.

## Materials and Methods

### BSMC algorithm

We assume that a given set of parameters is specified *a priori*:  $\lambda$  is the mean length of a recombining segment,  $G$  is the total genome length, and  $\rho$  is the recombination rate.  $\lambda$  and  $G$  are measured in base pairs, while  $\rho = 2N_e r$  is the per-individual, per-generation, and per-base pair gene conversion initiation rate  $r$  scaled by twice the effective population size  $N_e$ . Our BSMC algorithm crosses the genome from left to right, and discards most previous local trees, but always keeps track of, and conditions on, the clonal frame. The current local ARG  $A(x_{\text{cur}})$  keeps track of all, and only the lineages with non-empty ancestral material to the right of  $x_{\text{cur}}$ . All lineages in  $A(x_{\text{cur}})$  are possible targets of new recombination events and coalescent events. Recombination events and coalescent

events are not allowed on forgotten lineages [not in  $A(x_{\text{cur}})$ ]. To determine which lineage is in  $A(x_{\text{cur}})$  and which is not, we record and update for each lineage  $l$  its ancestral material to the right of  $x_{\text{cur}}$ :  $a_l(x_{\text{cur}})$ . One aim of the algorithm is to generate the sequence of local trees along the genome. For a given position  $x_{\text{cur}}$ , the local (or marginal) tree  $T(x_{\text{cur}})$  is the genealogy describing the inheritance history of site  $x_{\text{cur}}$ .  $T(x_{\text{cur}})$  can be obtained from  $A(x_{\text{cur}})$  by removing all branches that are not ancestral at  $x_{\text{cur}}$ . A graphical example of the algorithm is given in Supplemental Material, Figure S1 in File S1. More specifically, the BSMC algorithm proceeds as follows:

1. **Initialization:**  $x_{\text{cur}} = 0$  (current position, maximum is 1), and  $T_{\text{cf}}$  (the clonal frame) is simulated under the coalescent without recombination. The initial local ARG  $A(x_{\text{cur}})$ , and local tree  $T(x_{\text{cur}})$ , are set to  $T(0) = A(0) = T_{\text{cf}}$ . The ancestral material of every lineage  $l$  in  $A(0)$  is set to  $a_l(0) = [x_{\text{cur}}, 1] = [0, 1]$ , the whole genome. The list of recombination end points  $E$  (the right ends of recombination segments) is initialized as empty:  $E = ()$ .
2. **Position of new event:** The distance until the next potential recombination initiation (that occurs at position  $x_{\text{new}}$ ) is drawn according to an exponential distribution  $(x_{\text{new}} - x_{\text{cur}}) \sim \text{Exp}[(\rho G/2)\bar{A}(x_{\text{cur}})]$ , where  $\bar{A}(x_{\text{cur}})$  is the sum of all branch lengths in  $A(x_{\text{cur}})$ , expressed in units of  $2N_e$  generations. If  $x_{\text{new}} > E_0$ , where  $E_0$  is the first (and smallest) element of the list  $E$  of recombination end points (if  $E$  is empty then  $E_0 = \infty$ ), then the recombination initiation at  $x_{\text{new}}$  is cancelled, and the next considered position is set to  $x_{\text{new}} = E_0$ ;  $E_0$  is then removed from  $E$ , and the next event becomes a recombination termination, so go to step 4. Otherwise, if  $x_{\text{new}} \geq 1$  terminate the algorithm, and if  $x_{\text{new}} < 1$  the next event is a new recombination initiation at  $x_{\text{new}}$ , so go to step 3.
3. **New recombination event:** sample a lineage  $l$  randomly from  $A(x_{\text{cur}})$  proportionally to branch length. Then, sample a time  $t$  uniformly along the time spanned by  $l$ . The new recombination occurs at time  $t$  on branch  $l$ , and a new lineage  $l'$  is created, with its most recent end joining  $l$  at time  $t$ . A new coalescent time and coalescing lineage is sampled for  $l'$  conditional on  $A(x_{\text{cur}})$  [under the algorithm of Wiuf and Hein (1999)]. The right end of the recombining interval  $x_{\text{end}}$  is sampled from the distribution  $(x_{\text{end}} - x_{\text{new}}) \sim \text{Geom}(\lambda)/G$ , where  $\text{Geom}(\lambda)$  is the geometric distribution with mean  $\lambda$ . If  $x_{\text{end}} < 1$ , it is added to  $E$  while keeping  $E$  sorted in increasing order. The new local ARG is defined as  $A(x_{\text{new}}) = A(x_{\text{cur}}) \cup l'$ , and ancestral material of all lineages in  $A(x_{\text{new}})$  is updated (ancestral material to the left of  $x_{\text{new}}$  is deleted). Any lineage with no ancestral material to the right of  $x_{\text{new}}$  is removed from  $A(x_{\text{new}})$ . The new local tree  $T(x_{\text{new}})$  is defined from  $A(x_{\text{new}})$  and is printed to file. The current position is updated:  $x_{\text{cur}} = x_{\text{new}}$ . Return to step 2.
4. **Terminate a recombination event:** the new local ARG is initialized as  $A(x_{\text{new}}) = A(x_{\text{cur}})$ . The ancestral material of

all lineages in  $A(x_{\text{new}})$  is updated (ancestral material to the left of  $x_{\text{new}}$  is deleted). Any lineage with no ancestral material to the right side of  $x_{\text{new}}$  is removed from  $A(x_{\text{new}})$ . The new local tree  $T(x_{\text{new}})$  is defined from  $A(x_{\text{new}})$  and is printed to file. The current position is updated:  $x_{\text{cur}} = x_{\text{new}}$ . Return to step 2.

A large part of the complexity of the algorithm is attributable to the process of updating the ancestral material of lineages after a new recombination event is added to the local ARG. This step is described more in detail in [File S1](#). Our algorithm and model differ from the approximation of the CGC used by Didelot *et al.* (2010) and Ansari and Didelot (2014); in fact, we allow recombinant lineages to be affected by further recombinations, and to coalesce with each other if their ancestral materials overlap. To increase realism, we use the first positions simulated by the algorithm (generally  $10\lambda$  bases) as burn-in, that is, they are simulated but not considered part of the genome. While we simulate a linear genome, bacterial genomes are typically circular, so we assume that an arbitrary start position has been chosen. The version of the algorithm above conveys the basics of the model of within-population recombination; in our simulation software FastSimBac we have included many additional event types described in [File S1](#): mutation, migration, speciation, demographic change, recombination hotspots, and between-species recombination.

### Performance testing

We simulated bacterial genome evolution under the coalescent with gene conversion using SimBac (Brown *et al.* 2015). We always simulated 50 contemporaneous samples. We performed simulations under four different recombination intensities:  $\rho = 2N_e r = 0.001, 0.002, 0.005, 0.01$ , with  $\rho$  the population-scaled per-generation per-base pair recombination initiation rate. We used four genome sizes:  $G = 1, 2, 5$ , and 10 Mbp, and mean recombination tract length  $\lambda = 500$ . These values encompass a range of biologically relevant scenarios for bacteria (Vos and Didelot 2009; Didelot and Maiden 2010). We simulated 10 replicates for each combination of parameters, and, for each replicate, the simulated collection of local trees, and the clonal frame, were stored. Sequence data were generated from local trees using SeqGen (Rambaut and Grassly 1997) under an HKY85 model (Hasegawa *et al.* 1985) with transition/transversion rate ratio  $\kappa = 3$ . Some of the parameter combinations were too computationally demanding for SimBac: ( $\rho = 0.005$ ,  $G = 10$  Mbp), ( $\rho = 0.01$ ,  $G = 5$  Mbp), ( $\rho = 0.01$ ,  $G = 10$  Mbp). Every time we could run SimBac, we used its clonal frame as a fixed input for our software FastSimBac. The clonal frame is a major source of variation in sequence patterns between simulations (Ansari and Didelot 2014), so fixing the clonal frame, we reduce the variance in the difference of summary statistics between the two methods. Both the BSMC and the CGC assume that the clonal frame is generated by a standard coalescent process, so fixing it does not

introduce biases, and gives better resolution to spot differences between models. For any scenario in which we could not run SimBac, the clonal frame was generated within FastSimBac. We used local trees from FastSimBac to generate genome alignments with SeqGen as before.

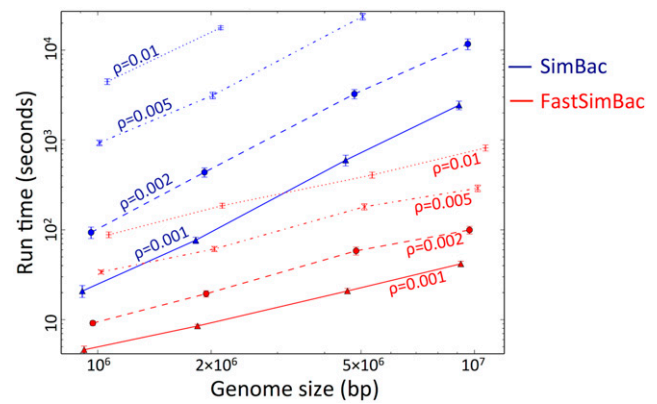
### ABC inference

We performed ABC with the local-linear regression approach (Beaumont *et al.* 2002) as implemented in the R package abc (Csilléry *et al.* 2012). We implemented and tested the performance of an ABC scheme based on the BSMC, using FastSimBac simulations to infer parameters from datasets themselves simulated under the CGC with SimBac. We used a uniform prior distribution over  $[0, 0.005]$  for the recombination rate  $\rho$ , and over  $[10, 1000]$  for the mean length  $\lambda$  of recombining intervals. The same priors were used for both simulating datasets and performing inference. The aim of the ABC analyses was to infer  $\rho$  and  $\lambda$ . For simplicity, the clonal frame simulated in SimBac was assumed to be known, as was the mutation rate  $\theta = 0.005$ . The clonal frame is an important confounding factor in real data analysis, which can be hard to estimate correctly (Hedge and Wilson 2014). However, including clonal frame inference in our ABC would make it too computationally demanding; also, fixing the clonal frame in this context allows us to focus on differences between the BSMC and the CGC. We simulated 1 Mbp alignments with 20 samples. For each true data set from SimBac, we simulated 10,000 datasets under the BSMC in FastSimBac. Only 1% of the simulations in FastSimBac were retained for parameter inference [the 1% with summary statistics most closest to true data, see Beaumont *et al.* (2002)]. We used two summary statistics: G4 (the proportion of incompatible sites) between neighboring SNPs, and G4 between SNPs at least 20 kbp away. Specifically, for the first summary statistic we counted the number of SNPs inconsistent with the first SNP occurring to their right; for the second summary statistic, for each SNP we selected the first SNP to its right at least 20 kbp away. We chose these summary statistics because G4 [and linkage disequilibrium (LD)] at short distances ( $h \ll \lambda$ ) is informative of the recombination rate  $\rho$  (the expected number of recombination events initiating or terminating within a short interval  $h$  is approximately proportional to  $2h\rho$ ). On the other hand, G4 at long distances ( $h \gg \lambda$ ) is informative of the product  $\rho\lambda$  (the expected number of recombination events affecting any of two distant bases is approximately proportional to  $2\rho\lambda$ ). The approximately linear relationship between G4 and number of recombinations might not hold for extreme values of the parameter space, in which case this simple two-summary statistics ABC could have problems inferring  $\rho$  and  $\lambda$  (see Figure 2 in [File S1](#)).

We also used the ABC-MCMC inference scheme (Marjoram *et al.* 2003) on a real *B. cereus* genome alignment (Didelot *et al.* 2010; Ansari and Didelot 2014). We used uniform prior distributions on  $[0.0, 0.25]$  for  $\rho$ , on  $[1, 10000]$  for  $\lambda$ , and on  $[0.01, 0.2]$  for  $\theta$  (the per-base pair per-individual, and



per-generation mutation rate scaled by  $2N_e$ ).  $\rho$ ,  $\lambda$  and  $\theta$  are also the three parameters that we inferred. We simulated whole genome alignments of 13 samples and 5,240,935 bp, as for the real dataset. For this analysis we used more summary statistics than in the ABC above (seven instead of two), so to allow estimation of the mutation rate  $\theta$ , to address potential limitations of the two previously considered summary statistics (see Figure S2 in File S1), and to address the potential impact of biological complexities on individual summary statistics. The seven summary statistics used are: number of polymorphic sites (observed value 629,942); G4 for consecutive SNPs (observed value 0.167) and for SNPs at least 2 kbp away (observed value 0.297); mean LD (measured as  $r^2 = [(p_{AB} - p_A p_B)^2 / p_A(1 - p_A)p_B(1 - p_B)]$  where  $p_A$  is the frequency of allele A in the first SNP,  $p_B$  the frequency of B in the second SNP, and  $p_{AB}$  the frequency of the AB haplotype) for consecutive SNPs (observed value 0.396) and for SNPs at least 2 kbp away (observed value 0.274); and mean number of haplotypes (considering a certain number of SNPs at the time) for pairs of consecutive SNPs (observed value 3.003) and for groups of four SNPs made of two pairs of consecutive SNPs, the two pairs being at a distance of at least 2 kbp. The number of SNPs, G4 and  $r^2$  were also used as summary statistics by Ansari and Didelot (2014). We can simulate entire genomes (instead of SNP pairs as Ansari and Didelot (2014)) and so include summary statistics for groups of  $>2$  SNPs. Due to the considerable computational demand, we fixed the clonal frame to that estimated and used by Didelot *et al.* (2010) and Ansari and Didelot (2014). However, recombination can cause errors in the estimate of the clonal frame, in particular of branch lengths (Hedge and Wilson 2014). In fact, with increasing recombination, and, in the absence of population structure, all genetic distances between samples are expected to converge to a common value. The consequent branch length errors can potentially bias inference, so we attempt to correct branch length errors within our ABC approach (see File S1). Lastly, to improve the realism of our model, we account for invariable sites. In fact,  $\sim 1$  out of every 6 bp (after removing sites with limited coverage) in the alignment are polymorphic, and a large proportion of the genome is expected to be coding; so, in principle, one would expect many homoplasies to occur simply due to multiple substitutions at the same site, and not necessarily requiring recombination events. Using back of the envelop calculations (see File S1) we estimated that around half (48.44%) of the genome is invariant and that the transition-transversion ratio is 5.21. We used these estimates as fixed values within an HKY (Hasegawa *et al.* 1985) substitution model with invariant sites, instead of the basic JC model (Jukes and Cantor 1969) implemented in our basic inference and in Ansari and Didelot (2014). This model, together with the local trees simulated by FastSimBac, was used in SeqGen to simulate the alignment from which summary statistics were extracted at each step of the ABC-MCMC. Each run consisted of 10,000 ABC-MCMC steps (of which



**Figure 3** Comparison of computational demand between the bacterial sequential Markov coalescent (BSMC) and the coalescent with gene conversion (CGC). The BSMC implemented in FastSimBac is faster than the CGC implemented in SimBac. On the vertical axis is the time required to generate local trees per replicate (in seconds on a logarithmic scale). On the horizontal axis is the genome size (in base pair on a logarithmic scale). Red lines refer to FastSimBac, blue lines to SimBac. Each point is the mean over 10 replicates, and bars represent SEs of the mean. SimBac was not run for highest recombination rates and genome sizes due to time limitations.

1000 were used as burn-in), and required between 2 weeks to 1 month with one processor.

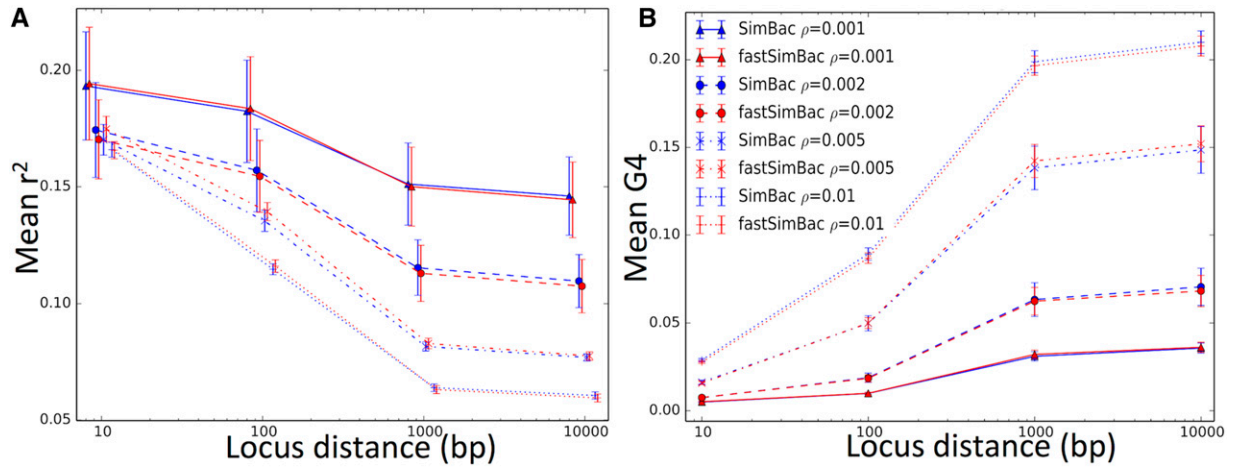
#### Data availability

FastSimBac is distributed as open source under the terms of the GNU General Public License, is available from <https://bitbucket.org/nicofmay/fastsimbac>, and is distributed as open source under the terms of the GNU General Public License.

## Results and Discussion

### Computational efficiency of BSMC

FastSimBac substantially reduces computational demand for simulating bacterial genome evolution. Compared to SimBac (the most efficient software currently available), FastSimBac improves speed by  $\sim 1$  order of magnitude for low recombination rate ( $\rho = 0.001$ ) and genome size ( $10^6$  bp), and up to two orders of magnitude for elevated recombination rate ( $\rho = 0.01$ ) and genome size ( $10^7$  bp) (Figure 3). Further, FastSimBac allows simulation of scenarios with both high recombination rate and genome size, which are currently out of reach of other methods due to excessive requirements for time and RAM. The performance of FastSimBac relative to the CGC improves as we increase either genome size or recombination rate (Figure 3). The running time required appears linear with genome size for FastSimBac, while not for SimBac. Another benefit of FastSimBac is that, by avoiding the generation of a global ARG, it has small RAM usage, allowing to run multiple simulations in parallel. The computational demand of FastSimBac also appears approximately linear in the number of samples (Figure S3 in File S1).



**Figure 4** Comparison of LD and site incompatibility between the BSMC and the CGC. The BSMC generates patterns of LD (measured as  $r^2$ ) and pairwise genetic incompatibility between sites (G4) very similar to the CGC. On the horizontal axis is the base pair distance between SNPs at which LD and G4 are measured.  $r^2$  is calculated as  $[(p_{AB} - p_A p_B)^2 / p_A(1 - p_A)p_B(1 - p_B)]$ , and G4 (the four-gamete test) is one if a SNP pair is incompatible and zero otherwise. For each distance  $d$ , and for any SNP  $x$ , LD and G4 are calculated between  $x$  and the first SNP at least  $d$  base pair to the right of  $x$ . Red lines refer to FastSimBac, blue lines to SimBac, and different point and line styles refer to different recombination rates (see legend). Genome length is 1 Mbp. Each point is the mean over 20 replicates, and bars are SEM. (A) Genome-wide mean LD. (B) Genome-wide mean G4.

### Accuracy of the BSMC

Next, we compared the simulated patterns of genetic variation and local tree features between the exact CGC simulated under SimBac, and the BSMC simulated with FastSimBac. LD (measured as  $r^2$ ), as expected, decreases considerably with increasing recombination rate (Figure 4), while the opposite holds for pairwise genetic incompatibility between sites (the four-gamete test, G4). There is substantial variation across different replicates in mean LD, probably because each replicate has a distinct clonal frame, and the clonal frame influences site patterns across the whole genome. LD and G4 at 1 kbp scales are already very close to those at longer distances, suggesting that a distance of  $2\lambda$  is sufficient to reach nearly as much LD as any arbitrary distance. Most importantly, values simulated under the BSMC mimic closely those simulated under the CGC, suggesting that, even at high recombination rates and short distances, the BSMC is a very accurate approximation (Figure 4). Similar results are also observed at different genome sizes (Figure S4 in File S1).

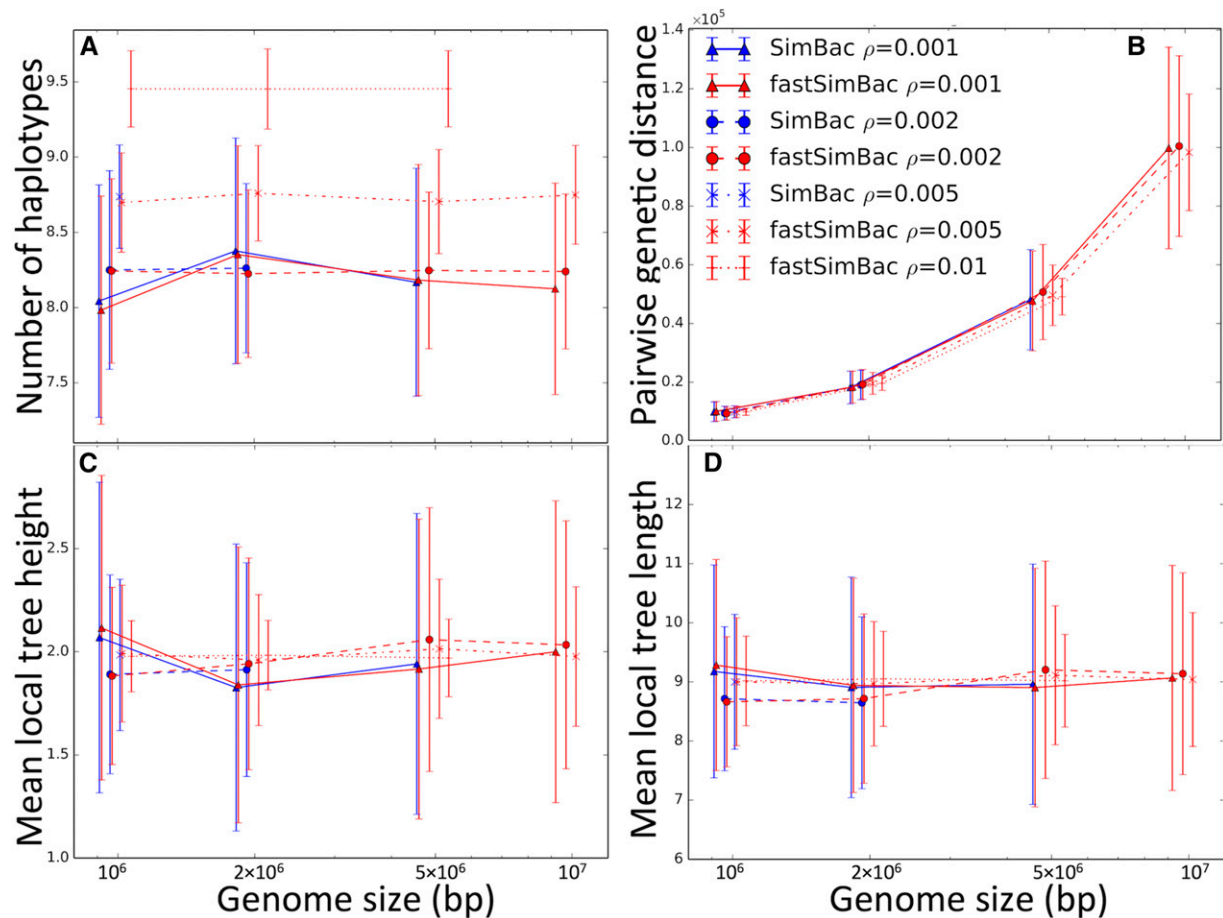
As expected, the number of haplotypes in nonoverlapping windows of 10 SNPs increases with recombination rate (Figure 5A), and, again, the BSMC very closely mimics the CGC. The genomic variation in number of haplotypes (Figure S5A in File S1) is very slightly underestimated, probably because long-range correlations in local trees (after conditioning on the clonal frame) are ignored in the BSMC, while present in the CGC. The mean pairwise genetic distances between samples are unaffected by recombination and by the model used for simulations (Figure 5B), but recombination does affect their variance (Figure S5B in File S1) because it tends to break down the relatedness of samples. Again, both patterns in the CGC are very closely approximated by the BSMC. Mean local tree height (Figure 5C) and mean local tree size (total

sum of the branch lengths, Figure 5D), are highly variable depending on the simulated clonal frame, and are not strongly affected by the simulation parameters, nor by our BSMC approximation.

### BSMC-based ABC inference

We investigated the accuracy and applicability of the BSMC approximation by performing ABC inference. First, we reconstructed parameters simulated under the exact CGC. We use two summary statistics based on G4, the pairwise genetic incompatibility between sites (see *Materials and Methods*). Although we simulated datasets under the exact CGC, and performed ABC simulations under a different model (the BSMC), inference was accurate. The 95% posterior confidence intervals for the population-scaled recombination rate  $\rho$ , and the mean length of recombining intervals  $\lambda$  contain the simulated values in both our replicates (Figure 6 and Figure S6 in File S1). This suggests that the BSMC can be used for accurately inferring bacterial evolutionary parameters. However, the elevated computational demand of this ABC approach keeps us from performing a more thorough simulation study. Also, here we assume that the exact clonal frame is known, and focus on differences between the BSMC and the CGC; this is not usually true for real datasets, where clonal frame imprecision could likely lead to higher inference error.

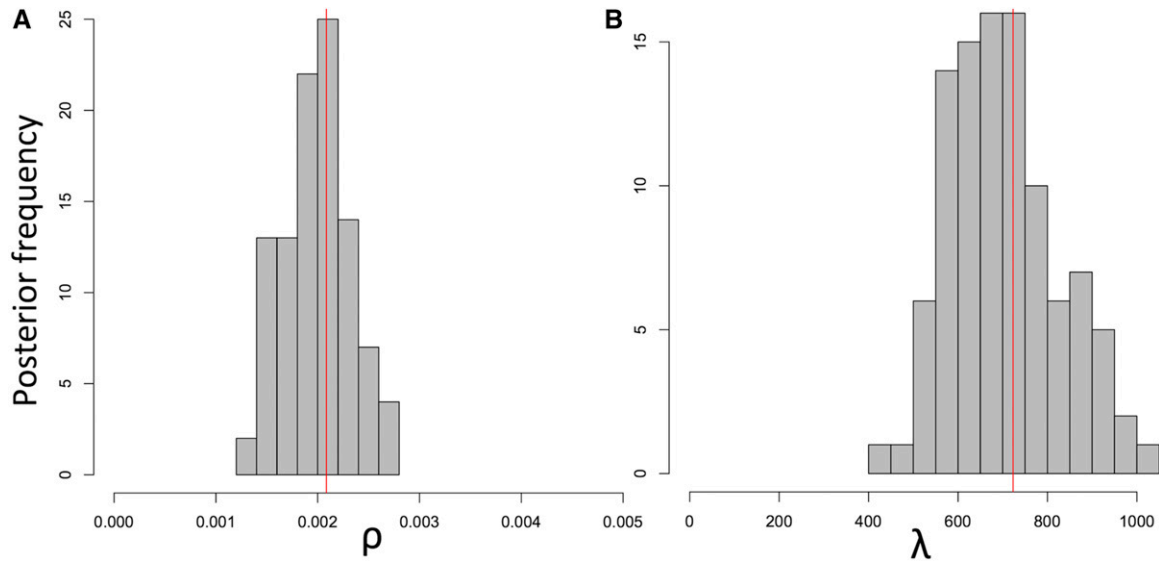
As an additional example of the applicability of the BSMC and of FastSimBac, we used ABC-MCMC (Marjoram *et al.* 2003) to infer  $\rho$ ,  $\lambda$ , and the scaled mutation rate  $\theta$  for the *B. cereus* bacterial group. Bacteria of the *B. cereus* group mostly live in the soil, feeding on dead organic matter, but they can occasionally infect humans and cause a range of diseases, from food poisoning to deadly anthrax (Arnesen *et al.* 2008). Disagreement has been found between *B. cereus*



**Figure 5** Comparison of simulated patterns between the BSMC and the CGC. Bacterial evolution simulated under the BSMC generates very similar patterns to the exact CGC. (A) Genome-wide mean number of simulated haplotypes over nonoverlapping sliding windows of 10 SNPs; (B) Mean pairwise genetic distance between samples; (C) Mean local tree height; (D) Mean local tree size (sum of all branch lengths). On the horizontal axis is genome size in base pair and on logarithmic scale. Red lines refer to FastSimBac, blue lines to SimBac, and different line and dot styles indicate different recombination rates (see legend). Each point is the mean over 50 replicates, and bars are SDs. SimBac and FastSimBac were not run for the highest recombination rates and genome sizes due to time and memory limitations.

species designation and MLST clade structure and population history, probably due to the contribution of plasmids and genetic recombination to the bacterial phenotype (Priest *et al.* 2004; Sorokin *et al.* 2006; Didelot *et al.* 2009a; Zwick *et al.* 2012). Furthermore, analyses of MLST data have shown discordant results regarding the prevalence of recombination relative to mutation in *B. cereus*. Estimates range from  $\rho/\theta \approx 0.05$  (Hanage *et al.* 2006), to  $\rho/\theta \approx 0.2$  (Didelot *et al.* 2009a), to  $\rho/\theta \approx 0.3$  (Didelot and Falush 2007), up to  $\rho/\theta \approx 2$  (Pérez-Losada *et al.* 2006), leading to a state of uncertainty regarding the contribution of recombination to *B. cereus* evolution. Improving our understanding of recombination in *B. cereus* would help us recognize the effect of homologous recombination on epidemiological inference and species delimitation (Didelot and Maiden 2010), and predict the acquisition and spread of infectivity and resistance factors (Perron *et al.* 2012). Genome-wide data from multiple strains provide a greater opportunity to study recombination. Here, we consider the genome alignment described in Didelot *et al.* (2010) and Ansari and Didelot (2014) comprising

13 genomes from the *B. cereus* group. Didelot *et al.* (2010) performed MCMC inference on this dataset using an approximate coalescent model with bacterial recombination (the ClonalOrigin model) that did not allow recombinant lineages to be affected by further recombination, nor recombinant lineages to coalesce with one another. They inferred a mean recombination tract length of  $\lambda = 171$  bp with interquartile range [168, 175], and  $\rho/\theta = 0.21$  with interquartile range [0.20, 0.23]. Ansari and Didelot (2014) used a model similar to ClonalOrigin within an ABC-MCMC approach, and accounted for the propensity for lineages to recombine more with closely related lineages than with distantly related ones. They inferred  $\rho = 0.077$  with confidence interval  $CI_\rho = [0.036, 0.127]$ ,  $\lambda = 152$  bp with  $CI_\lambda = [74, 279]$ , and  $\theta = 0.0528$  with  $CI_\theta = [0.0437, 0.0640]$ . The ClonalOrigin model employed by these methods approximates the coalescent with gene conversion, but less closely than the BSMC. In fact, the ClonalOrigin model leads to overestimation of  $\rho$  under recombination and mutation rates relevant to this scenario (Didelot *et al.* 2010). Our BSMC-based ABC-MCMC approach instead allows



**Figure 6** Accurate inference of recombination parameters with the BSMC-based ABC. Recombination parameters simulated under the exact CGC (red vertical lines) were reconstructed using simulations under the BSMC within an ABC inference scheme. Inference from another independent ABC run is shown in Figure S6 and File S1. (A) Posterior distribution of  $\rho$ . (B) Posterior distribution of  $\lambda$ .

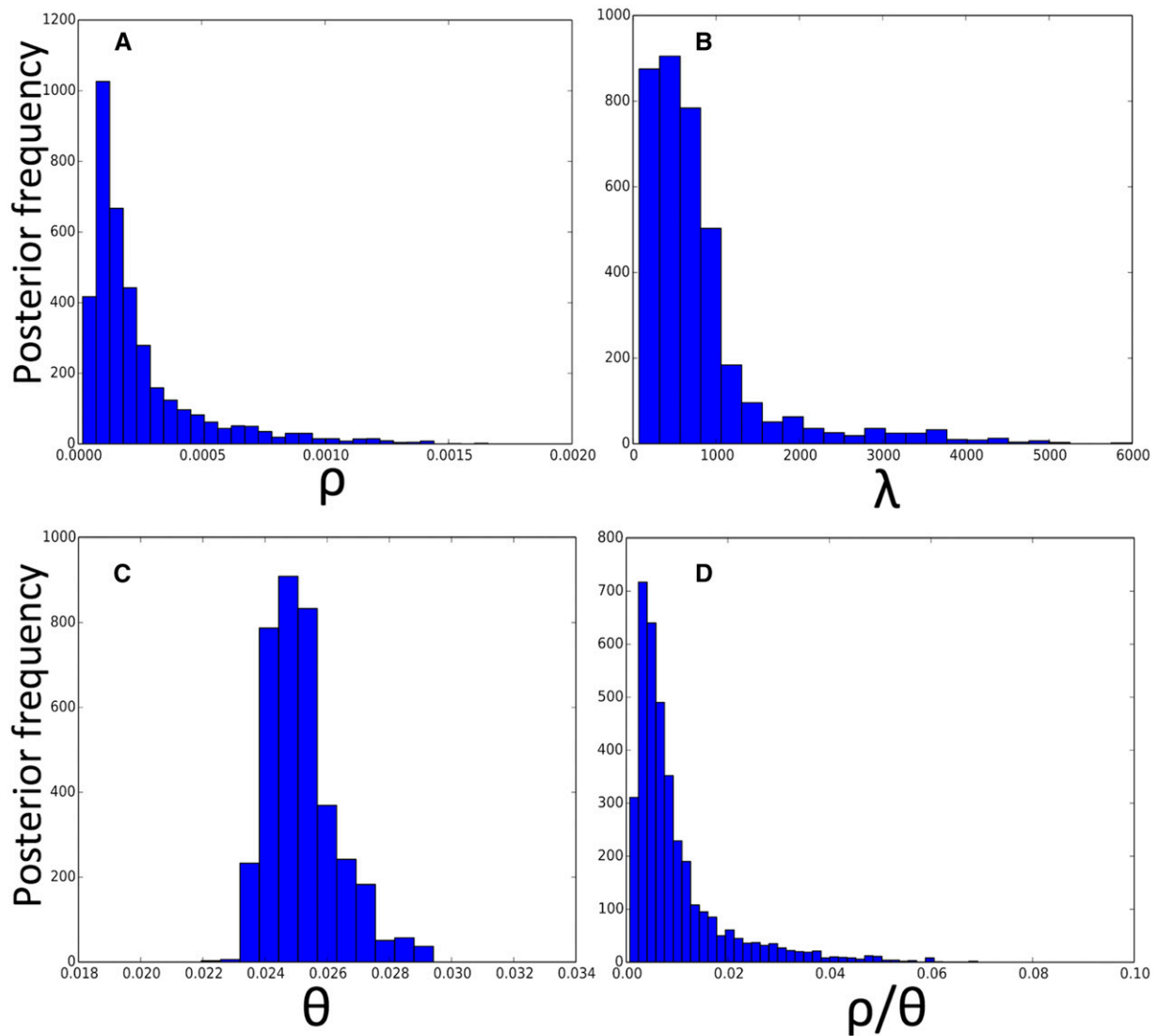
recombination events to split the ancestral material of recombinant lineages. Furthermore, in contrast to both these previous analyses, we account for differences in transition and transversion rates, for invariant sites, and for biases in branch length estimation (see *Materials and Methods* and File S1).

With our BSMC-based approach, we infer higher mean recombination tract length  $\lambda$  (median 592 bp and 95% confidence interval [119, 3406], Figure 7) than previous estimates [171 and 152 bp from Didelot *et al.* (2010) and Ansari and Didelot (2014) respectively]; Our estimate is closer to values inferred from genome-wide likelihood-based analyses in *Clostridium difficile* (Didelot and Wilson 2015). We also inferred a considerably lower contribution of recombination relative to mutation ( $\rho/\theta$ , median 0.0065 and 95 % confidence interval [0.001, 0.038], Figure 7) than previous genome-wide studies [0.21 and  $\approx 1.46$  from Didelot *et al.* (2010) and Ansari and Didelot (2014), respectively]. This suggests that recombination contributes much less to *B. cereus* evolution than previously thought, and that these bacteria are considerably clonal, although, due to variation in recombination rates between clades, our results do not necessarily apply to all species in the *B. cereus* group (Sorokin *et al.* 2006). These results were confirmed by an additional independent run of the analysis (Figure S7 in File S1), and can be explained by the fact that we account for invariant sites, for transition/transversion bias, and for multiple substitutions at the same position (using a finite sites model). In fact, invariant sites and a high transition to transversion rate ratio usually cause more homoplasies than expected under a homogeneous substitution rate. This happens because an uneven distribution of substitutions along the genome (attributable to invariant sites) increases the probability that two

substitutions hit the same base, potentially causing homoplasies. A transition/transversion bias increases the probability that bases hit by multiple substitutions become homoplasies. These homoplasies, if unaccounted for in the model, can be interpreted as the effect of short recombinant fragments, downwardly biasing estimates of  $\lambda$ , and upwardly biasing estimates of  $\rho/\theta$ . Our approach naturally accounts for homoplasies due to multiple substitutions at the same site by modeling sequence evolution along local trees with a continuous-time DNA substitution model including invariant sites and transition/transversion bias. Supporting our interpretation, when we ran our method without accounting for invariant sites we estimated lower  $\lambda$  and higher  $\rho/\theta$  (Figure S8 in File S1). Another potential factor is that our BSMC allows interactions between recombination events, breaking recombinant segments into smaller pieces as expected under the CGC; this process, if unaccounted for, could lead to a downward bias in the estimation of  $\lambda$ .

We can measure the total impact of recombination on genome evolution as  $\rho^*\lambda/\theta$ , for which we infer a posterior median of 3.7 (95 % confidence interval [2.9, 5.9]); this is considerably smaller than previous estimates [ $\approx 35.9$  and  $\approx 221.7$  for Didelot *et al.* (2010) and Ansari and Didelot (2014) respectively]. Our  $\rho^*\lambda/\theta$  confidence interval is smaller than for other parameters because  $\rho$  and  $\lambda$  are strongly inversely correlated in our posterior (Figure S9 in File S1). This makes  $\rho^*\lambda$  easier to estimate than the two parameters separately. This problem of identifiability of  $\lambda$  and  $\rho$  has been previously observed in bacteria (Ansari and Didelot 2014), and is also seen in eukaryotes (Padhukasahasram *et al.* 2004, 2006), although analysis of bacterial data are simpler due to the lack of crossover recombination. We found no correlation between other pairs of parameters (Figure S9, A–C, in File S1).





**Figure 7** Posterior distributions of parameters for genome-wide evolution of *B. cereus*. We inferred BSMC parameters using an ABC-MCMC inference scheme. (A) Posterior distribution of  $\rho$ . (B) Posterior distribution of  $\lambda$ . (C) Posterior distribution of  $\theta$ . (D) Posterior distribution of  $\rho/\theta$ .

While our ABC-MCMC seems to capture well the complexity of real data for five out of seven summary statistics, for two of them (G4 at large distances and  $r^2$  at short distances) there are discrepancies (Figure S9, D–J, Figure S7, E–K, in [File S1](#)). This suggests the existence of further neglected biological complexities, for example larger rate of recombination between closely related lineages (see Ansari and Didelot 2014), variable recombination rate between *B. cereus* clades (Sorokin *et al.* 2006), nonhomologous recombination (Didelot and Maiden 2010), population structure (such as due to niche adaptation Sorokin *et al.* 2006), recombination with other bacterial groups, variable selective pressure and mutation rate, and alignment errors. Error in clonal frame estimation, despite our efforts to correct branch lengths (see [File S1](#)), could also play a role in these discrepancies and reduce accuracy.

In conclusion, the BSMC offers not only a very computationally convenient approximation to the CGC, but also an

accurate one. Our implementation of the BSMC model in the simulation software FastSimBac allows faster simulations of bacterial genome evolution (and therefore parameter inference with ABC), under a broader range of parameter values. FastSimBac allows specification of the clonal frame upon which simulations can be conditioned, which may grant simulations a closer fit to particular datasets when the clonal frame is readily estimable. By virtue of building on top of the popular simulators ms (Hudson 2002) and MaCS (Chen *et al.* 2009), our software includes options for many evolutionary scenarios that have been included in previous eukaryotic coalescent simulators (Hudson 2002; Chen *et al.* 2009), but which have remained unavailable for simulating bacterial genomes, such as population structure and migration, speciation, changes in population size, and recombination hot-spots. Applications of our model and software are not restricted to simulations, but also include inference of

recombination rates and other parameters of bacterial evolution. Our analysis of recombination in the *B. cereus* group showcases the applicability of our method for inference from genome-wide alignments. However, our ABC method is very computationally demanding, and so it would be challenging to apply it to scenarios with particularly high recombination rates, or large sample sizes. In the future, we intend to use the BSMC within a likelihood framework for accurate and efficient inference of the clonal frame and recombination parameters simultaneously. We believe that the BSMC and FastSimBac will prove very useful for both benchmarking and for statistical inference based on bacterial genome sequence data.

## Acknowledgments

We are grateful to the creators of MaCS, on which the FastSimBac code is partly based. We also thank Xavier Didelot for sharing the *B. cereus* dataset. We thank the two anonymous reviewers and the editor for valuable comments. N.D.M. was supported by a James Martin Research Fellowship of the Oxford Martin School. D.J.W. is a Sir Henry Dale Fellow, jointly funded by the Wellcome Trust and the Royal Society (grant 101237/Z/13/Z).

## Literature Cited

- Ansari, M. A., and X. Didelot, 2014 Inference of the properties of the recombination process from whole bacterial genomes. *Genetics* 196: 253–265.
- Arenas, M., 2013 Computer programs and methodologies for the simulation of dna sequence data with recombination. *Front. Genet.* 4: 9.
- Arenas, M., and D. Posada, 2007 Recodon: coalescent simulation of coding dna sequences with recombination, migration and demography. *BMC Bioinformatics* 8: 458.
- Arenas, M., and D. Posada, 2010 Coalescent simulation of intracodon recombination. *Genetics* 184: 429–437.
- Arnesen, L. P. S., A. Fagerlund, and P. E. Granum, 2008 From soil to gut: *Bacillus cereus* and its food poisoning toxins. *FEMS Microbiol. Rev.* 32: 579–606.
- Beaumont, M. A., W. Zhang, and D. J. Balding, 2002 Approximate Bayesian computation in population genetics. *Genetics* 162: 2025–2035.
- Brown, T., X. Didelot, D. J. Wilson, and N. De Maio, 2015 Simbac: simulation of whole bacterial genomes with homologous recombination. *Microb. Genom.* 2. doi: 10.1099/mgen.0.000044.
- Buckee, C. O., K. A. Jolley, M. Recker, B. Penman, P. Kriz *et al.*, 2008 Role of selection in the emergence of lineages and the evolution of virulence in *Neisseria meningitidis*. *Proc. Natl. Acad. Sci. USA* 105: 15082–15087.
- Carvajal-Rodríguez, A., 2008 GENOMEPOP: a program to simulate genomes in populations. *BMC Bioinformatics* 9: 223.
- Chen, G. K., P. Marjoram, and J. D. Wall, 2009 Fast and flexible simulation of dna sequence data. *Genome Res.* 19: 136–142.
- Croucher, N. J., A. J. Page, T. R. Connor, A. J. Delaney, J. A. Keane *et al.*, 2015 Rapid phylogenetic analysis of large samples of recombinant bacterial whole genome sequences using gubbins. *Nucleic Acids Res.* 43: e15.
- Csilléry, K., O. François, and M. G. Blum, 2012 abc: an R package for approximate Bayesian computation (ABC). *Methods Ecol. Evol.* 3: 475–479.
- Didelot, X., and D. Falush, 2007 Inference of bacterial microevolution using multilocus sequence data. *Genetics* 175: 1251–1266.
- Didelot, X., and M. C. Maiden, 2010 Impact of recombination on bacterial evolution. *Trends Microbiol.* 18: 315–322.
- Didelot, X., and D. J. Wilson, 2015 Clonalframeml: efficient inference of recombination in whole bacterial genomes. *PLOS Comput. Biol.* 11: e1004041.
- Didelot, X., M. Barker, D. Falush, and F. G. Priest, 2009a Evolution of pathogenicity in the *Bacillus cereus* group. *Syst. Appl. Microbiol.* 32: 81–90.
- Didelot, X., D. Lawson, and D. Falush, 2009b Simmlst: simulation of multi-locus sequence typing data under a neutral model. *Bioinformatics* 25: 1442–1444.
- Didelot, X., D. Lawson, A. Darling, and D. Falush, 2010 Inference of homologous recombination in bacteria using whole-genome sequences. *Genetics* 186: 1435–1449.
- Didelot, X., D. W. Eyre, M. Cule, C. Ip, M. A. Ansari *et al.*, 2012 Microevolutionary analysis of *Clostridium difficile* genomes to investigate transmission. *Genome Biol.* 13: R118.
- Ewing, G., and J. Hermisson, 2010 MSMS: a coalescent simulation program including recombination, demographic structure and selection at a single locus. *Bioinformatics* 26: 2064–2065.
- Excoffier, L., and M. Foll, 2011 Fastsimcoal: a continuous-time coalescent simulator of genomic diversity under arbitrarily complex evolutionary scenarios. *Bioinformatics* 27: 1332–1334.
- Falush, D., M. Torpdahl, X. Didelot, D. F. Conrad, D. J. Wilson *et al.*, 2006 Mismatch induced speciation in *Salmonella*: model and data. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* 361: 2045–2053.
- Fearnhead, P., N. G. Smith, M. Barrigas, A. Fox, and N. French, 2005 Analysis of recombination in *Campylobacter jejuni* from mlst population data. *J. Mol. Evol.* 61: 333–340.
- Fraser, C., W. P. Hanage, and B. G. Spratt, 2005 Neutral micro-epidemic evolution of bacterial pathogens. *Proc. Natl. Acad. Sci. USA* 102: 1968–1973.
- Griffiths, R. C., and P. Marjoram, 1997 An ancestral recombination graph. *Inst. Math. Appl.* 87: 257–270.
- Hanage, W. P., C. Fraser, and B. G. Spratt, 2006 The impact of homologous recombination on the generation of diversity in bacteria. *J. Theor. Biol.* 239: 210–219.
- Hasegawa, M., H. Kishino, and T.-a. Yano, 1985 Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *J. Mol. Evol.* 22: 160–174.
- Hedge, J., and D. J. Wilson, 2014 Bacterial phylogenetic reconstruction from whole genomes is robust to recombination but demographic inference is not. *MBio* 5: e02158.
- Hernandez, R. D., 2008 A flexible forward simulator for populations subject to selection and demography. *Bioinformatics* 24: 2786–2787.
- Hudson, R. R., 1983 Properties of a neutral allele model with intragenic recombination. *Theor. Popul. Biol.* 23: 183–201.
- Hudson, R. R., 2002 Generating samples under a Wright-Fisher neutral model of genetic variation. *Bioinformatics* 18: 337–338.
- Jukes, T. H., and C. R. Cantor, 1969 Evolution of protein molecules, pp. 21–132 in *Mammalian protein metabolism*, edited by H. N. Munro. Academic Press, New York.
- Kingman, J. F. C., 1982 The coalescent. *Stoch. Proc. Appl.* 13: 235–248.
- Marjoram, P., and J. D. Wall, 2006 Fast coalescent simulation. *BMC Genet.* 7: 16.
- Marjoram, P., J. Molitor, V. Plagnol, and S. Tavaré, 2003 Markov chain Monte Carlo without likelihoods. *Proc. Natl. Acad. Sci. USA* 100: 15324–15328.
- Martinen, P., A. Baldwin, W. P. Hanage, C. Dowson, E. Mahenthiralingam *et al.*, 2008 Bayesian modeling of recombination events in bacterial populations. *BMC Bioinformatics* 9: 1.

- Marttinen, P., W. P. Hanage, N. J. Croucher, T. R. Connor, S. R. Harris *et al.*, 2012 Detection of recombination events in bacterial genomes from large population samples. *Nucleic Acids Res.* 40: e6.
- McVean, G. A., and N. J. Cardin, 2005 Approximating the coalescent with recombination. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* 360: 1387–1393.
- Milkman, R., and M. M. Bridges, 1990 Molecular evolution of the *Escherichia coli* chromosome. iii. Clonal frames. *Genetics* 126: 505–517.
- Mostowy, R., N. J. Croucher, W. P. Hanage, S. R. Harris, S. Bentley *et al.*, 2014 Heterogeneity in the frequency and characteristics of homologous recombination in pneumococcal evolution. *PLoS Genet.* 10: e1004300.
- Padhukasahasram, B., P. Marjoram, and M. Nordborg, 2004 Estimating the rate of gene conversion on human chromosome 21. *Am. J. Hum. Genet.* 75: 386–397.
- Padhukasahasram, B., J. D. Wall, P. Marjoram, and M. Nordborg, 2006 Estimating recombination rates from single-nucleotide polymorphisms using summary statistics. *Genetics* 174: 1517–1528.
- Peng, B., and M. Kimmel, 2005 simuPOP: a forward-time population genetics simulation environment. *Bioinformatics* 21: 3686–3687.
- Pérez-Losada, M., E. B. Browne, A. Madsen, T. Wirth, R. P. Viscidi *et al.*, 2006 Population genetics of microbial pathogens estimated from multilocus sequence typing (mlst) data. *Infect. Genet. Evol.* 6: 97–112.
- Perron, G. G., A. E. Lee, Y. Wang, W. E. Huang, and T. G. Barraclough, 2012 Bacterial recombination promotes the evolution of multi-drug-resistance in functionally diverse populations. *Proc. Biol. Sci.* 279: 1477–1484.
- Posada, D., and K. A. Crandall, 2002 The effect of recombination on the accuracy of phylogeny estimation. *J. Mol. Evol.* 54: 396–402.
- Priest, F. G., M. Barker, L. W. Baillie, E. C. Holmes, and M. C. Maiden, 2004 Population structure and evolution of the *Bacillus cereus* group. *J. Bacteriol.* 186: 7959–7970.
- Rambaut, A., and N. C. Grassly, 1997 Seq-Gen: an application for the monte carlo simulation of DNA sequence evolution along phylogenetic trees. *Comput. Appl. Biosci.* 13: 235–238.
- Schierup, M. H., and J. Hein, 2000 Consequences of recombination on traditional phylogenetic analysis. *Genetics* 156: 879–891.
- Smith, J. M., N. H. Smith, M. O'Rourke, and B. G. Spratt, 1993 How clonal are bacteria? *Proc. Natl. Acad. Sci. USA* 90: 4384–4388.
- Smith, J. M., E. J. Feil, and N. H. Smith, 2000 Population structure and evolutionary dynamics of pathogenic bacteria. *BioEssays* 22: 1115–1122.
- Sorokin, A., B. Candelon, K. Guilloux, N. Galleron, N. Wackerow-Kouzova *et al.*, 2006 Multiple-locus sequence typing analysis of *Bacillus cereus* and *Bacillus thuringiensis* reveals separate clustering and a distinct population structure of psychrotrophic strains. *Appl. Environ. Microbiol.* 72: 1569–1578.
- Staab, P. R., S. Zhu, D. Metzler, and G. Lunter, 2015 scrm: efficiently simulating long sequences using the approximated coalescent with recombination. *Bioinformatics* 31: 1680–1682.
- Tang, J., W. P. Hanage, C. Fraser, and J. Corander, 2009 Identifying currents in the gene pool for bacterial populations using an integrative approach. *PLOS Comput. Biol.* 5: e1000455.
- Turner, K. M., W. P. Hanage, C. Fraser, T. R. Connor, and B. G. Spratt, 2007 Assessing the reliability of eburst using simulated populations with known ancestry. *BMC Microbiol.* 7: 30.
- Vos, M., and X. Didelot, 2009 A comparison of homologous recombination rates in bacteria and archaea. *ISME J.* 3: 199–208.
- Wang, Y., Y. Zhou, L. Li, X. Chen, Y. Liu *et al.*, 2014 A new method for modeling coalescent processes with recombination. *BMC Bioinformatics* 15: 273.
- Wilson, D. J., 2012 Insights from genomics into bacterial pathogen populations. *PLoS Pathog.* 8: e1002874.
- Wilson, D. J., E. Gabriel, A. J. Leatherbarrow, J. Cheesbrough, S. Gee *et al.*, 2009 Rapid evolution and the importance of recombination to the gastroenteric pathogen *Campylobacter jejuni*. *Mol. Biol. Evol.* 26: 385–397.
- Wiuf, C., and J. Hein, 1999 Recombination as a point process along sequences. *Theor. Popul. Biol.* 55: 248–259.
- Wiuf, C., and J. Hein, 2000 The coalescent with gene conversion. *Genetics* 155: 451–462.
- Zwick, M. E., S. J. Joseph, X. Didelot, P. E. Chen, K. A. Bishop-Lilly *et al.*, 2012 Genomic characterization of the *Bacillus cereus* sensu lato species: backdrop to the evolution of *Bacillus anthracis*. *Genome Res.* 22: 1512–1524.

Communicating editor: L. M. Wahl