JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 9, Number 6, 2002

Pp. 805–818

Bayesian Estimation of the Number of Inversions in the History of Two Chromosomes

THOMAS L. YORK,[1] RICHARD DURRETT,[1,2] and RASMUS NIELSEN[1]

ABSTRACT

We present a Bayesian approach to the problem of inferring the history of inversions separating homologous chromosomes from two different species. The method is based on Markov Chain Monte Carlo (MCMC) and takes full advantage of all the information from marker order. We apply the method both to simulated data and to two real data sets. For the simulated data, we show that the MCMC method provides accurate estimates of the true posterior distributions and in the analysis of the real data we show that the most likely number of inversions in some cases is considerably larger than estimates obtained based on the parsimony inferred number of inversions. Indeed, in the case of the *Drosophila repleta–D. melanogaster* comparison, the lower boundary of a 95% highest posterior density credible interval for the number of inversions is considerably larger than the most parsimonious number of inversions.

Key words: inversions, Bayesian estimation, MCMC, breakpoint graph.

1. INTRODUCTION

1.1. The biological problem

With the recent increase in the availability of genomic data, statistical methods for elucidating the evolution of genomes are becoming increasingly important. Such methods are relevant, not only in evolutionary studies, but also for comparative mapping. In general, genomes evolve by inversions, translocations, and chromosome fissions and fusions. For simplicity, in this article we will consider situations that involve only inversions. This occurs in (a) mitochondrial and chloroplast data, (b) sex chromosomes which do not undergo reciprocal translocations with autosomes, and (c) Drosophila species, where translocations and pericentric inversions are rare, so chromosome arms are preserved between species.

Our data consists of N markers with known order in two species. The problem of inferring the history of the two chromosomes is the problem of inferring the order and number of inversions. In most studies, this has been approached by estimating the "inversion distance," the smallest number of inversions which can produce the observed rearrangement. For example, in the data of Palmer and Herbon (1998), three inversions are necessary to transform the mitochondrial genome of cabbage (*Brassica oleracea*) into the mitochondrial genome of turnip (*Brassica campestris*). The problem of identifying the minimum number

[1]Department of Biological Statistics and Computational Biology, Cornell University, Ithaca, NY 14853.
[2]Department of Mathematics, Cornell University, Ithaca, NY 14853.

of inversions is known as "sorting by reversals" (SBR) for unsigned permutations. Sorting by reversals is NP-hard (Caprara, 1999), but an exact branch-and-bound method is available (Kececioglu and Sankoff, 1995).

There is no guarantee that the actual number of inversions that have occurred in the history of two chromosomes is equal to the minimal number. In many cases, the true number of inversions may, in fact, be much larger than the inversion distance. Several estimators of the true number of inversions have, therefore, been proposed. For example, Caprara and Lancia (2000) provided an estimator based on the number of break points, i.e., the number of adjacent pairs of markers in one genome that are not adjacent in the other. The inversion distance between two genomes is at least 1/2 the number of breakpoints but in most cases this bound is very crude. In this article, we will develop a Bayesian method for estimating the true number of inversions assuming an explicit biological model of evolution by inversions. The method is based on Markov Chain Monte Carlo (MCMC). The advantage of such a method is that it uses all the available information in the data and automatically provides measures of statistical uncertainty in terms of credible sets. We apply the method to two real data sets and show that, in one case, the most probable number of inversions is much larger than the minimum number of inversions.

1.2. The break point graph

The key to the study of the inversion distance between two genomes is the break point graph (Hannenhalli and Pevzner, 1995a). We first define the break point graph for the case where the orientations of the markers are known, so they are described by signed permutations. In this case, we can think of each marker as having two ends, a head and a tail. The break point graph of one signed permutation of N markers, p_a, relative to another, p_b, is a graph with $2N + 2$ vertices, one corresponding to each end of each of the N markers plus one for each end of the chromosome. The notation $(2, -3, 1, 4)$ will mean that marker 2 is leftmost and is oriented with its head to the *left* of its tail, and then next comes marker 3, with the its head to the *right* of the tail, etc. Now, for each marker k, label its head $2k - 1$ and its tail $2k$; we then may replace each signed number in the permutation by the pair of numbers representing the head and the tail, in the appropriate order: $k \to 2k-1 : 2k$ and $-k \to 2k : 2k-1$. We add 0 at the left end and $2N + 1 = 9$ at the right end to get

$$(2, -3, 1, 4) \to (0, 3 : 4, 6 : 5, 1 : 2, 7 : 8, 9).$$

The colon-separated pairs represent the two ends of one marker, and they will remain adjacent under any sequence of inversions; the comma-separated pairs represent marker ends which are adjacent in this permutation but which may be split up by inversions. For each comma-separated pair in p_a, (p_b), there is a *black* (*gray*) edge joining the corresponding vertices. As an example, Fig. 1 shows the break point graph of $p_a = (2, -3, 1, 4)$ relative to $p_b = (-1, -4, 2, 3)$. To help the reader check the construction of the graph, we note that

$$(-1, -4, 2, 3) \to (0, 2 : 1, 8 : 7, 3 : 4, 5 : 6, 9)$$

Note that each vertex has exactly one black edge and one gray edge incident to it. If we start at some vertex and follow an edge to another vertex, and then follow the other edge incident to that edge, etc., we will eventually return to the starting vertex. If there are vertices not on this *cycle* we can repeat the process until every vertex is on a cycle. This yields a *cycle decomposition* which in this (signed) case is unique.

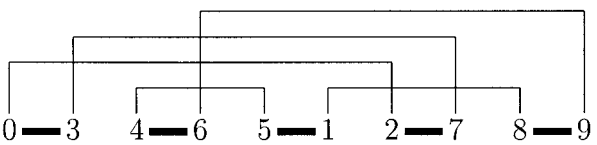

FIG. 1. The break point graph of $p_a = (2, -3, 1, 4)$ relative to $p_b = (-1, -4, 2, 3)$. The black edges are shown as thick lines. In this case, there are two cycles, 0-3-7-2-0 and 4-6-9-8-1-5-4.

Let the number of cycles in the cycle decomposition be $c(p_a, p_b)$. Performing an inversion, I, on p_a will cause two black edges to be broken and replaced by two others; the effect on the cycle decomposition will be to either split one of the cycles into two cycles, join two cycles together to form one cycle, or change one of the cycles so that it visits the same vertices but in a different order. The number of cycles will change by $\Delta c = c(Ip_a, p_b) - c(p_a, p_b) = +1, -1$ or 0. If $p_a = p_b$, the number of cycles is $c = N + 1$. Since an inversion increases the number of cycles by at most $+1$, it takes at least $N + 1 - c$ inversions to turn p_a into p_b. Although this is only a lower bound on the inversion distance, it is usually close to and often equal to the true value. For example, Bafna and Pevzner (1995) considered 11 comparisons of mitochondrial and chloroplast genomes and found that this lower bound gave the right answer in all cases.

In general, there are complications called *hurdles* that can prevent the number of cycles from being increased. The simplest example occurs when $p_a = (3, 2, 1)$ and $p_b = (1, 2, 3)$. In this case, the breakpoint graph has two cycles but no move will increase the number of cycles to 3. If $h(p_a, p_b)$ is the number of hurdles, then Hannenhalli and Pevzner (1995a) have shown that $n + 1 - c + h$ is a lower bound on the genomic distance. This is almost the answer in general. If the hurdles are arranged to form what they call a *fortress*, then one additional move is required. If we let $f(p_a, p_b) = 1$ when the breakpoint graph is a fortress and 0 otherwise, then

$$d(p_a, p_b) = n + 1 - c + h + f.$$

This result due to Hannenhalli and Pevzner (1995a) leads to a polynomial algorithm to compute the inversion distance between two signed permutations. For more details, see Chapter 10 of Pevzner (2000).

Most genomic data is in the form of unsigned permutations. That is, mapping techniques tell us the location of the markers on the chromosome but usually provide no information regarding the orientation of the marker on the chromosome. For example, consider the following comparative map between the human and cattle X chromosomes which comes from Band *et al.* (2000). Here, the second column gives the start of the gene when its exact location is known. The third column gives the cytological band to which it has been mapped. The symbols p and q refer to the two arms of the X chromosome with the numbers increasing as we move away from the centromere.

Gene	*Start (Kb)*	*Cyto*	*Cattle order*
ANT3		Xp22.32	1
AMELX	8,950	Xp22.31	2
SAT	18,652	Xp22.1	3
CYBB		Xp21.1	4
MAOA	38,289	Xp11.4	5
SYN1	42,783	Xp11.23	7
TIMP1	42,792	Xp11.23	8
SYP	44,288	Xp11.22	6
CITED1	64,082	Xq13.1	9
PLP1	97,418	Xq22	11
FACL4	103,471	Xq23	10
HPRT1	128,965	Xq26	14
TNFSF5	130,747	Xq26	13
SLC6A8	148,934	Xq28	12

In this case, if the data is grouped into blocks, then in most cases orientations can be assigned: $1, 2, 3, 4, 5 \to +1$, $6 \to 2?$, $7, 8 \to +3$, $9 \to 4?$, $11, 10 \to -5$, and $14, 13, 12 \to -6$. This leads to the following partially signed permutation $+1, +3, 2?, 4?, -5, -6$. There are only four possible ways to assign signs to the segments with unknown orientation. A little calculation shows that $+1, +3, -2, +4, -5, -6$ has the smallest distance: 4. In general one can reduce the computation of the distance from the unsigned case to the signed case, but in some situations the amount of work becomes prohibitive. For example, in the comparison of *Drosophila melanogaster* and *D. repleta* below, one would need to compute the distance for $2^{60} > 10^{18}$ assignments of signs.

2. A BAYESIAN APPROACH

2.1. Model assumptions

We model the process of chromosome rearrangement as follows:

- Rearrangement is assumed to be due entirely to inversions.
- The occurrence of inversions is a Poisson process with unknown mean λ; the probability of exactly L inversions having occurred is $P(L|\lambda) = e^{-\lambda}\lambda^L/L!,\ L = 1, 2, \ldots$.
- We assume a uniform prior distribution for λ, i.e., $P(\lambda) = 1/\lambda_{max}$ for $0 < \lambda \leq \lambda_{max}$.
- The number of markers which are present and in known order on both of the two chromosomes being compared is N. For each marker we may or may not know whether it has the same or opposite orientations on the two chromosomes. If we have (do not have) this information for all markers we represent the data, D, as a pair of signed (unsigned) permutations, Pa and Pb.
- We distinguish $N(N+1)/2$ possible inversions. We distinguish between inversions only if they produce distinct signed permutations of the markers. Thus, if we start with the sequence of markers (A, B, C), the inversion of any section containing B (but not A or C) yields $(A, -B, C)$. As far as the present model is concerned, all such inversions are the same; it doesn't matter precisely where the breaks in the chromosome are as long as the section between the breaks (i.e., the inverted section) includes B. The inversion of a section of chromosome which contains no markers is not counted as an inversion as all.
- Each of these $N(N+1)/2$ inversions occurs with equal probability.

2.2. MCMC method

There are in this model $(N(N+1)/2)^{L_X}$ equiprobable inversion sequences X of length L_X. Let Ω be the set of all possible inversion sequences, and let the probability measure associated with each $X \in \Omega$ be

$$P(X|\lambda) = (e^{-\lambda}\lambda^{L_X}/L_X!)(N(N+1)/2)^{-L_X}.$$

We are interested in approximating the posterior probability distributions for X and λ, i.e., approximating $P(X|D)$ and $P(\lambda|D)$, where D is the known marker order in the two sampled chromosomes. To do this, we establish a Markov chain with state space on $\Omega \times \mathbf{R}^+$ and stationary density $P(X, \lambda|D)$, $X \in \Omega$, $\lambda \in \mathbf{R}^+$. By sampling values of X and λ from this Markov chain at stationarity, we can approximate $P(X|D)$ and $P(\lambda|D)$. We may write the target distribution as $P(X, \lambda|D) = P(X, \lambda, D)/P(D) = P(D|X, \lambda)P(X|\lambda)P(\lambda)/P(D)$. Our update scheme (described in the next section) generates only inversion sequences (X) which transforms p_a into p_b. For such X, $P(D|X, \lambda) = 1$ and

$$P(X, \lambda|D) = (e^{-\lambda}\lambda^{L_X}/L_X!)(N(N+1)/2)^{-L_X}\frac{1}{\lambda_{max}}/P(D).$$

We update the Markov chain element (X, λ) using a Metropolis-Hastings scheme in which we alternate updating λ and X. A Gibbs step is used to update λ; i.e., $P(\lambda|X, D) \propto P(X|\lambda)P(\lambda) \propto e^{-\lambda}\lambda^{L_x}P(\lambda)$ is sampled from directly. Updating X is more involved. Think of X as an inversion path (Fig. 2) which comprises sequences of permutations, $p_0 = p_a, p_1, \ldots p_L = p_b$, and of inversions, $I_1, I_2, \ldots I_L$, with $p_i = I_i p_{i-1}, i = 1, 2, \ldots L$.

The proposed new path, Y, is constructed as follows:

1. Choose a section of X to replace. Choose, with probability $q_L(l, j)$, a length, l, $(0 \leq l \leq L)$, and a starting permutation, p_j, $(0 \leq j \leq L - l)$. The subpath from $p_\alpha = p_j$ to $p_\beta = p_{j+l}$ will be replaced in Y by a new one.
2. Generate a new subpath to replace this section. Using the breakpoint graph of p_α relative to p_β, choose an inversion, I_1', at random, but with $\Delta c = 1$ with high probability. Then, in the same way, choose I_2' using the break point graph of $I_1' p_\alpha$ relative to p_β, and so on, until $I_1' I_2' \ldots I_{l'}' p_\alpha = p_\beta$.

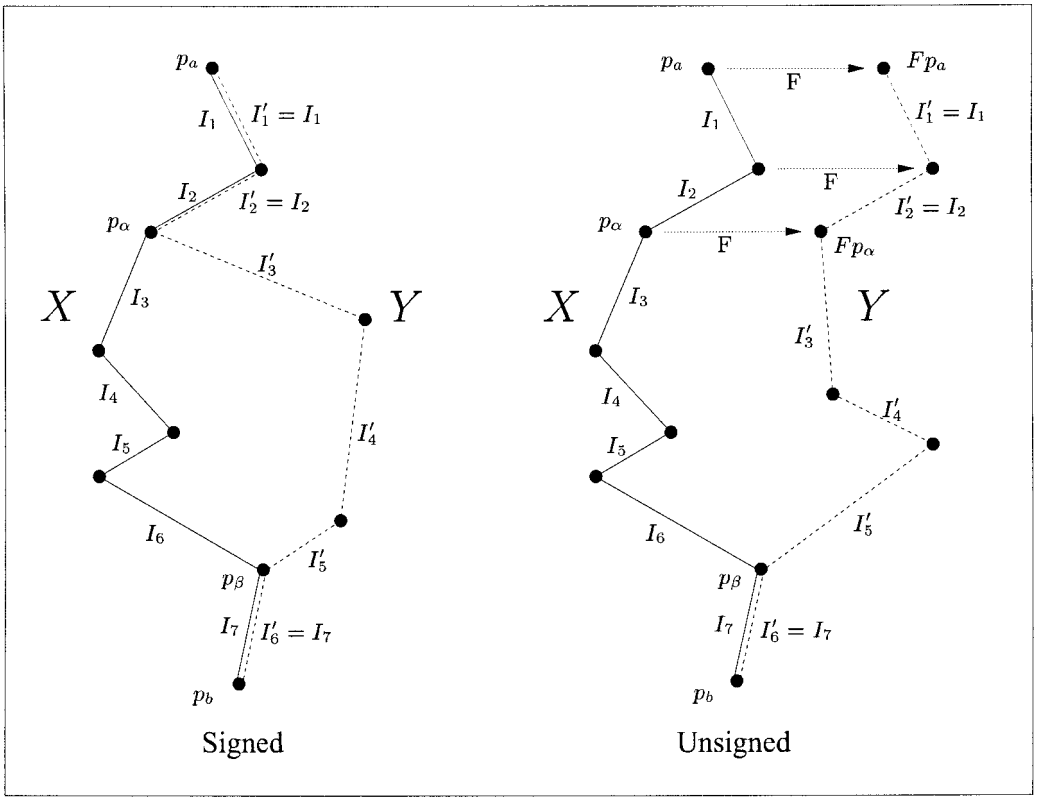

FIG. 2. The update scheme in the signed and unsigned cases. In each case, the dashed line is an inversion path, Y, proposed as an update for path X (solid line).

2.3. *Details of updating for signed permutations*

Choosing a section of X to replace. First the length, l, is chosen by sampling from a distribution $q(l)$, and then j is chosen uniformly at random from $0, 1, \ldots L_i - l$, so $q(l, j) = q(l)/(L + 1 - l)$. In practice, we use a $q(l)$ so that lengths small compared to N are roughly equally represented, while lengths large compared to N are strongly suppressed. The particular form we use is

$$q(l) \propto 1 - \tanh\left(\xi\left(\frac{l}{\alpha N} - 1\right)\right)$$

with typically $\xi = 8$ and $\alpha = 0.65$

Generating a new subpath. Let the permutations at the ends of the section to be replaced be $p_\alpha = p_j$ and $p_\beta = p_{j+l}$. We seek a sequence of inversions, $I'_1, I'_2, \ldots I'_{l'}$, and intermediate permutations $p'_0 = p_\alpha, p'_1, p'_2, \ldots p'_{l'} = p_\beta$ with $p'_i = I'_i p'_{i-1}$, $i = 1, 2, \ldots l'$. We use the break point graph of p'_{i-1} relative to p_β in deciding which inversion to choose as I'_{i+1}. In particular, we look at its cycle decomposition and identify which inversions would lead to a change in the number of cycles of $\Delta c = -1$, 0, or 1. The cycle decomposition of p_a relative to p_b has $N + 1$ cycles if p_a and p_b are identical and otherwise it has fewer. Thus, generating a sequence of inversions which turns p_a into p_b is a matter of bringing the number of cycles up to this value, and so a $\Delta c = +1$, (-1) step is a step toward (away from) this goal. By enumerating which inversions lead to $\Delta c = +1$, 0, or -1, and then choosing a $\Delta c = 0$, or $\Delta c = -1$ inversion only rarely, we make long paths unlikely compared to short ones. The idea is to simulate a random path from a distribution that approximates the posterior distribution. Under the assumption that shorter paths are more probable than longer paths, steps in which $\Delta c = +1$ should be chosen with higher probability than steps in which $\Delta c = 0$ or $\Delta c = -1$. Specifically, if N_{+1}, N_0, and N_{-1} are the numbers of inversions leading

to $\Delta c = +1$, 0, and -1, respectively, and are all nonzero, we let the relative probability of choosing $\Delta c = +1, 0, -1$ be $1, \varepsilon_1, \varepsilon_2$, and then, having chosen Δc, we choose among the corresponding inversions, giving each equal probability. So, for instance, if N_{+1}, N_0 and N_{-1} are all nonzero, the probability of choosing a particular $\Delta c = +1$ inversion would be $1/((1+\varepsilon_1+\varepsilon_2)N_{+1})$. However, if there are no $\Delta c = 0$ inversions, for instance, the probability of $\Delta c = 0$ must be zero, and we still let the relative probabilities of $\Delta c = 1$ and $\Delta c = -1$ be 1 and ε_2, so the probability of choosing a particular $\Delta c = +1$ inversion is $1/((1 + \varepsilon_2)N_{+1})$. When $N_{+1} = N_0 = 0$, the two permutations are equal. With probability $1 - \varepsilon_3$, we stop at this point and with probability ε_3 we keep going (with a $\Delta c = -1$ step since there is no other choice). The probability, q_{new}, of proposing a particular subpath from p_α to p_β with l' inversions will be the product of $l' + 1$ factors, one for each inversion and a factor of $1 - \varepsilon_3$ for actually stopping upon reaching p_β. Any nonnegative choices of ε_i are valid; however, the choice of ε_i may greatly influence rates of convergence. To determine appropriate values of ε_i, initial runs on sample data sets can be performed (see below).

The length of the proposed path Y is $L' = L + l' - l$, and the forward proposal probability is $q(Y|X) = q_L(l, j)q_{new}$. To calculate the acceptance probability, we also need $q(X|Y) = q_{L'}(l', j)q_{old}$. Here q_{old} is the probability of generating as a path from p_α to p_β precisely the subpath that occurred in X. This is done much as in the other direction: by getting cycle decompositions and identifying $\Delta c = +1, 0$, and -1 inversions, etc.

2.4. Details for unsigned permutations

The previous section describes an updating procedure for the signed case which relies on the fact that in that case it is not only easy to get c, but it is easy to identify which inversions lead to Δc of $+1, 0$, or -1. In the unsigned case, getting c is hard, and identifying $\Delta c = +1, 0, -1$ inversions would be even harder. Instead of trying to do this, we always work with signed permutations. However, when doing a calculation for the unsigned case, we let the starting permutation, p_a, wander over the set of 2^N *signed* permutations which all have the marker order specified in the data and differ from each other only in the orientations of the markers. In other words, we propose an update not only to the sequence of inversions but to the signs of the markers in p_a.

As in the signed case, we first choose a subpath to propose a replacement for; let it go from p_α to p_β. Let F be an operator which flips some markers. We perform these flips on $p_0 = p_a, p_1, \ldots, p_j = p_\alpha$, to get a new sequence of permutations $Fp_0, Fp_1, \ldots, Fp_j$, related by the same inversions, i.e., $I_i Fp_{i-1} = Fp_i$, $i = 1, \ldots, j$. Then we generate a path from Fp_α to p_β in the same way as for the signed case. Figure 2 illustrates the process. To flip a marker means to perform an inversion which affects only that one marker, and we know how to evaluate Δc for inversions. So we can easily evaluate the Δc from each of the N flips considered in isolation. Then each flip is performed with probability ϵ_4 for $\Delta c = -1$, $1/2$ for $\Delta c = 0$, or $\Delta c = 1$.

2.5. Details on convergence monitoring

We use the method of Gelman and Rubin (1992) to decide when the Markov chain has converged. This requires running some number, $m \geq 2$, of chains for the same data. Let $X_{i,j}$ be the i^{th} element of the j^{th} Markov chain, and $L_{i,j}$ its length. Define a between-chain variance $B = \frac{1}{m-1}\sum_j(\langle L\rangle_j - \langle L\rangle)^2$ and a within-chain variance $W = \frac{1}{m}\sum_j \frac{1}{n-1}\sum_i(L_{i,j} - \langle L\rangle_j)^2$, where $\langle L\rangle_j = \frac{1}{n}\sum_i L_{i,j}$ and $\langle L\rangle = \frac{1}{mn}\sum_{i,j} L_{ij}$. Convergence is indicated when $\sqrt{\hat{R}} = \sqrt{(n-1)/n + B/W}$ gets close to 1. We typically use between 5 and 10 chains and consider the burn-in phase to last until $\sqrt{\hat{R}} \leq 1.1$; at that point we start to accumulate L and λ histograms. Typically, we continue running for many times the length of the burn-in.

The idea of running several chains is that one can check that they have come to agree with each other before deciding that they have converged. For this purpose, it is desirable to start the different chains in widely dispersed states. In the present method, the initial states are constructed by generating a path from p_a to p_b in the same way as described in Section 2.4, except that some paths are generated using a low value of ϵ_1, and others using a large value, in order to start some chains with short paths and some with long paths.

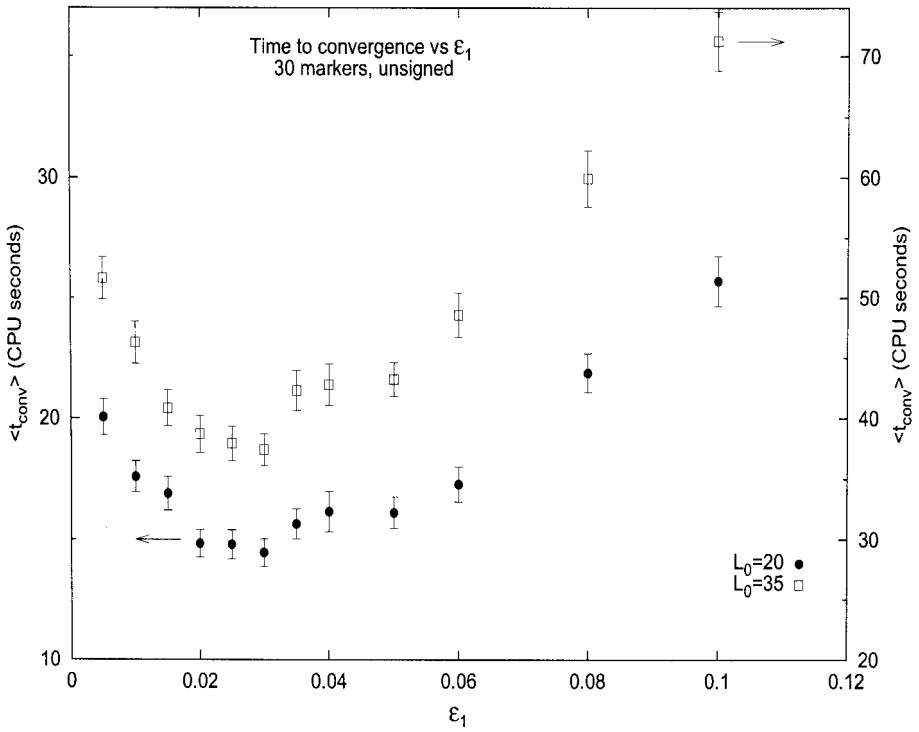

FIG. 3. Mean CPU time to convergence ($\sqrt{\hat{R}} < 1.1$) as a function of ε_1 for unsigned permutations of 30 markers. The number of random inversions performed to generate these data are $L_0 = 20$ and $L_0 = 35$.

2.6. Improving convergence

The update scheme described above has several parameters which may be tuned to improve convergence. In particular, α and ξ control the length of the section of path chosen to replace; ϵ_1, ϵ_2, and ϵ_3 control the generation of a new subpath transforming p_α into p_β (and in particular how strongly short paths are preferred); and (in the unsigned case) ϵ_4 controls the degree of preference for $\Delta c = 1$ marker flips. In order to choose reasonable values of these parameters, we performed several studies of convergence as a function of a parameter. In particular, for each of several values of a parameter, we found the number of Markov chain iterations and CPU time to convergence for each of a set of permutations representing simulated data. Each permutation was chosen by performing L_0 inversions with each being chosen uniformly at random from the $N(N+1)/2$ inversions. In this way, we arrived at the following values for the parameters: $\alpha = 0.65$, $\xi = 8$, $\epsilon_1 = 0.03$, $\epsilon_2 = \epsilon_1/2$, $\epsilon_3 = \epsilon_1^2$, and $\epsilon_4 = 0.025$. As an example, Fig. 3 shows the mean time to convergence as a function of ϵ_1 for unsigned permutations of 30 markers. The other parameters are as given above.

3. RESULTS FOR SIMULATED DATA

In this section, the markers are taken to be labeled such that p_b is the identity permutation $(1, 2, 3, \ldots, N)$. Then D is specified by giving the (signed or unsigned) permutation p_a. Let the number of paths of length L which sort D be $M(L, D)$. Since $P(X, \lambda|D)$ depends on X only through the length,

$$P(L, \lambda|D) = M(L, D)P(X, \lambda|D) \propto M(L, D)\frac{e^{\lambda}\lambda^{L}}{L!}(N(N+1)/2)^{-L}$$

and $P(L|D)$ and $P(\lambda|D)$ may be obtained by, respectively, integrating over λ and summing over L. For small N, it is feasible to get $M(L, D)$, for all L less than some maximum path length L_{max}, by directly counting the paths. For large L, $M(L, D) \approx (N(N+1)/2)^{L}/K(N)$, where $K(N)$ is the total number

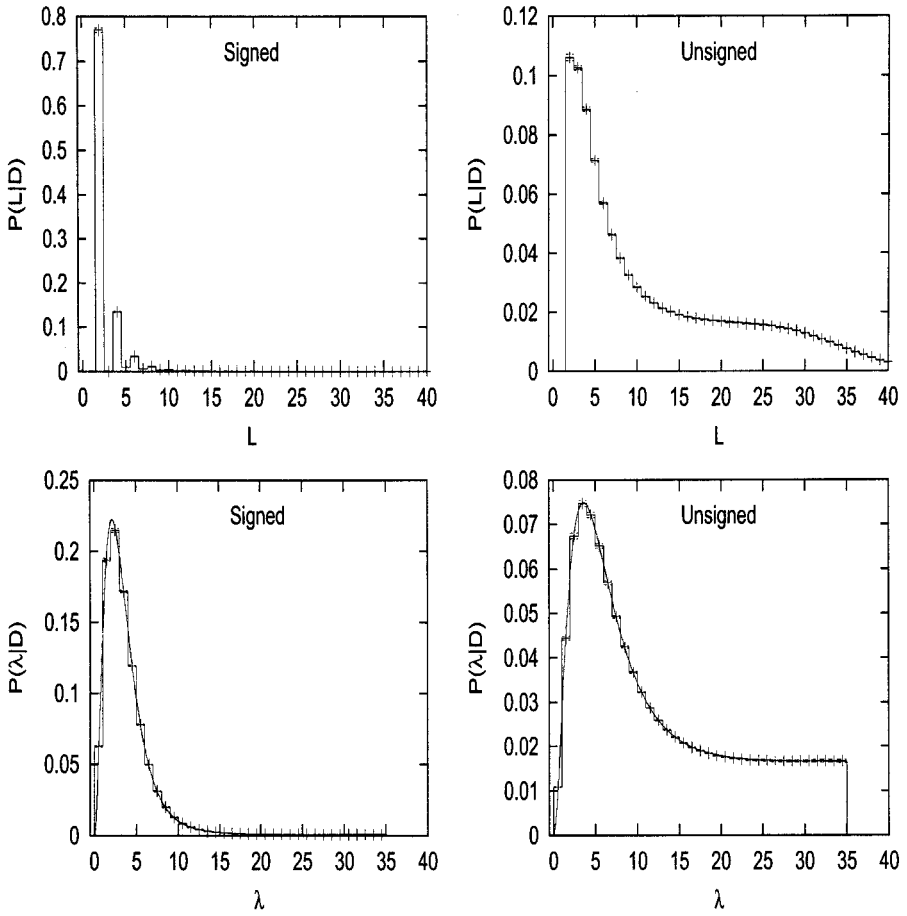

FIG. 4. Comparison of MCMC and exact results for $P(L|D)$ and $P(\lambda|D)$ for 7 signed markers, $D = (1, -5, -4, -3, -6, 2, 7)$, and $\lambda_{max} = 35$.

of permutations, and the numerator is the total number of paths of length L. For unsigned permutations $K(N) = N!$, and for signed permutations $K(N) = N!2^N$. In summing over L, we use this approximation for the $L > L_{max}$ terms. Figure 4 shows the result of this calculation compared with the result of our MCMC method. To obtain the results for the MCMC method, we ran 9 chains each consisting of 335,872 updates, while sampling values of λ and L in each iteration.

As Fig. 4 shows, the MCMC method provides a very close approximation to the true posterior distributions of L and λ. The close correspondence between the true and the estimated posterior distribution demonstrates that the MCMC method, in the current implementation, in fact does recover the true posterior distribution.

It is also worth noticing that the posterior distributions of L and λ have much larger variances when considering unsigned instead of signed markers. This tells us that much information regarding the inversion history is preserved in the signs of the markers. Whenever signs of the markers are available, estimators of the number of inversions should take this information into account.

4. APPLICATIONS TO REAL DATA

Human–cattle data

The first data set we analyze is the data set above with 14 unsigned markers on the X-chromosome of cattle and humans. In Fig. 5, we illustrate the convergence of $E[L|D]$. Eight replicate Markov chains

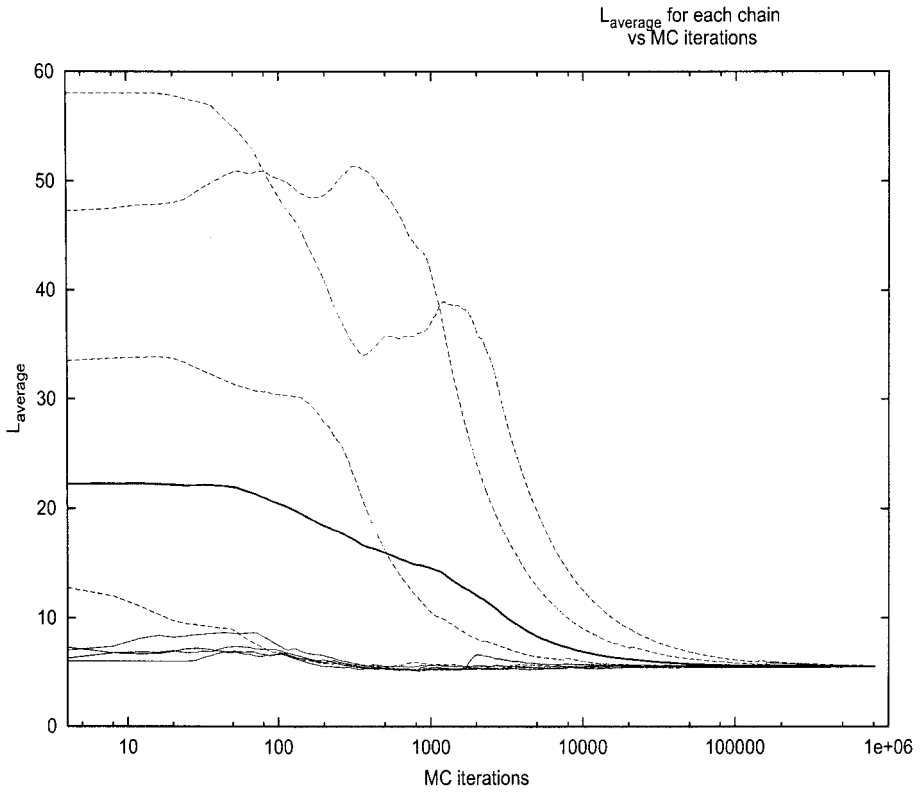

FIG. 5. Convergence of the ergodic average of the number inversions along the chain ($L_{average}$) for the data set containing 14 markers from human and cattle X chromosomes. The thick solid line shows the average among the 8 replicate chains.

were simulated. The initial inversion paths were simulated as one iteration of the chain with $\varepsilon_1 = 0.7$ (long initial inversion paths, dashed lines) or $\varepsilon_1 = 0.025$ (short initial inversions paths, solid lines). For each chain, 815,104 iterations were completed with $\varepsilon_1 = 0.03$ and $\varepsilon_4 = 0.025$ and a value of $\lambda_{max} = 80$ was used to ensure that the resulting posterior distributions were proper. The value of $\lambda_{max} = 80$ was chosen after initial trial runs to be sufficiently large to contain essentially all of the probability mass of the posterior distribution. None of the conclusions here are sensitive to the exact value of λ_{max} chosen. Notice that the ergodic averages of L along the chains converge to the same point for all chains: $E[L|D] \approx 5.49$. Also, $\sqrt{\hat{R}} < 1.1$ was achieved at 8192 iterations. The ergodic averages are converging relatively fast and simulation variance should be minimal in the estimation of L. Similar properties are true for λ (not shown). The run time for this data set was 254 seconds on an Athlon 1.2 GHz processor running Linux.

The posterior distribution for L and λ were obtained by sampling values of L and λ after each iteration of the Markov chain after the first 8,192 iterations (the burn-in time) of the chain (Fig. 6). The estimates of the posterior probability were combined by simply using the arithmetic averages of the posterior probability among runs. For this data set, the most probable value of L, i.e., the value of L that maximizes $P(L|D)$ is the parsimony inferred number of inversions (4). However, we also notice that it is quite likely that the true number of inversions is larger than 4. A 95% credible set for L, based on the highest posterior density (HPD) method is $(4 \leq L \leq 9)$. In fact, it is quite likely that the parsimony inferred number is not the true number of inversions. The expected number of inversions, given the data, is about twice as large as the parsimony inferred number.

The posterior density for the rate of inversions (λ) based on combining all eight runs is plotted in Fig. 6. The posterior density is obtained similarly to the posterior distribution of L by binning the data using a bin width of 0.25. Notice that $\mathrm{Var}(\lambda|D) > \mathrm{Var}(L|D)$ as expected. A 95% credible set for λ is given by $(1.05 \leq \lambda \leq 12.75)$ and $E(\lambda|D) = 6.49$.

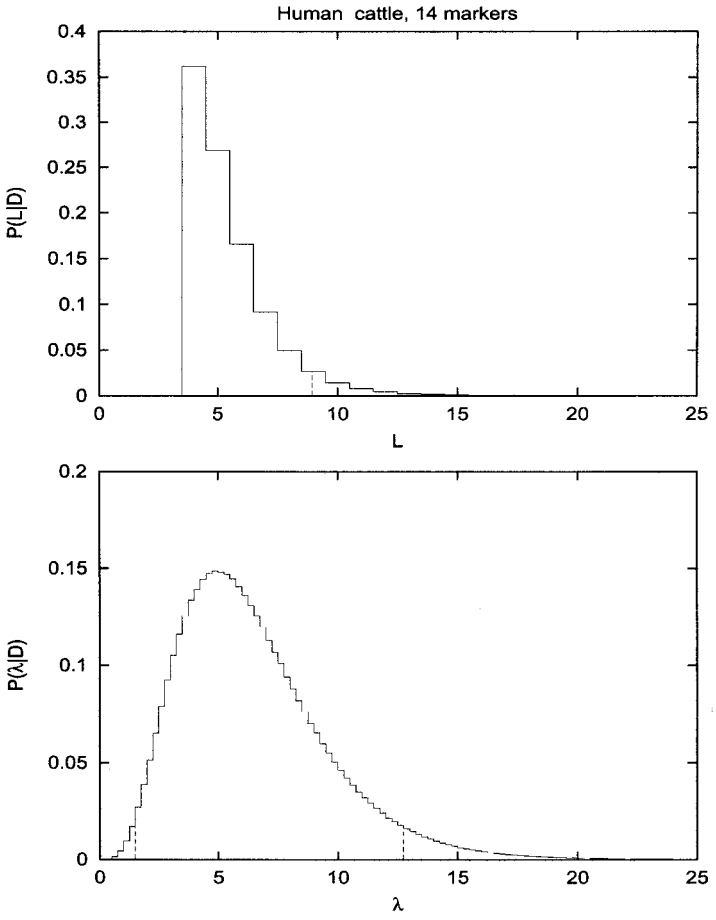

FIG. 6. The posterior distributionof L and λ for the data set of 14 markers from human and cattle X chromosomes. The vertical lines indicate 95% credible intervals.

D. melanogaster *and* D. repleta *data*

The second data set we analyze is a data set consisting of 79 unsigned markers from chromosome 3R of *Drosophila melanogaster* and chromosome 2 of *Drosophila repleta*, mapped by Ranz, Casals, and Ruiz (2001). The data is given in Table 1. The second column there gives the gene name, the fourth its physical location in *D. repleta*, the third its physical location in *D. melanogaster*, and the first its order in *D. melanogaster.*

Six Markov chains were simulated with results given in Fig. 7: three in which the initial inversion path was simulated using $\varepsilon_1 = 0.5$ (long initial inversion paths, dashed lines) and three simulated using $\varepsilon_1 = 0.025$ (short initial inversions paths, dotted lines) and with $\lambda_{max} = 200$. This data set contains considerably more inversions that the human–cattle data set, and convergence is consequently much slower. In this case, it took 1.7 million iterations before $\sqrt{\hat{R}} < 1.1$. A total of 43 million iterations were performed for each chain (Fig. 7). At this point, the ergodic average of L for each chain had essentially converged to the same point $E[L|D] \approx 92.61$. The run time for this data set was approximately 4 days on an Athlon 1.2 GHz processor running Linux. The posterior distribution of L is plotted in Fig. 8. Here, the maximum likelihood estimate $argmax_L P(L|D) = 87$ is slightly smaller than $E[L|D]$ because of the right-skew of the distribution. A 95% HPD credible set for L is given by $(71 \leq L \leq 118)$. The parsimony inferred number of inversions for this data set is 53 and the probability mass assigned to this number of inversions by the posterior distribution is essentially zero, in that in the 2.58×10^8 iterations this state was never visited. The parsimony number of inversions was found by running the chain under a very small fixed value of λ. There is a very high probability that the true number of inversions in the history of the two chromosomes

TABLE 1. RELATIVE ORDER OF MARKERS ON *D. Repleta* CHROMOSOME 2 AND *D. melanogaster* CHROMOSOME 3R

Order	*Name*	*D. mel.*	*D. repleta*	*Order*	*Name*	*D. mel.*	*D. repleta*
46	Hsrw	93D6	A1f	68	Rb97D	97D3	D3b
42	Cha	91B8	A1g	52	109B4	95B7	D3d
74	Acph-1	99C5	A1j	4	DS05926	83F2	D5a
45	Atpa	93A7	A2f	20	Hsp70A	87A7	D5b
54	Gdh	95C12	A3f	23	Hsp70B	87C1	D5c
47	145B9	94A1	A4a	11	dsx	84E1	E1b
73	Stg	99A5	B1b	1	lDsubFC4	82F	E1d
78	Gcn2	100C1	B1c	12	neur	85C3	E2c
71	wdn	98E3	B1i	62	57D6	96A5	E2f
75	RpL32	99D5	B2a	65	DS06282	97B1	E3d
24	Ry	87D11	B2c	18	DS04597	86C8	E4a
41	135D1	91B1	B2h	61	106F1	96A1	E4g
69	115B9	97E1	B3g	60	crb	95F3	E5a
76	Cec	99E5	B4a	32	Hsc70-4	88E8	E6k
25	37H1	87E6	B4c	33	Tm2	88E11	E6j
26	Act87E	87E9	B4d	64	E(spl)	96F10	F1a
50	152D12	95A7	C1a	35	Ubx	89D6	F1b
38	67H8	89F1	C2c	7	DS07700	84B3	F1c
27	94H1	87F12	C2d	6	Antp	84B1	F1d
2	DS08128	83E1	C2e	5	Pb	84A4	F1e
44	H	92E12	C3e	66	Amon	97C2	F1h
67	Tl	97D1	C4a	70	fkh	98D2	F2b
31	91F10	88D5	C4c	39	DS02256	90A1	F3a
77	tll	100B1	C4d	79	DS00911	100E1	F3h
30	put	88C9	C4e	22	Pp1-87B	87B15	F4b
59	120E2	95F2	C4g	51	nau	95B3	F4c
34	Act88F	88F7	C5e	13	MtnA	85E9	F4h
49	orb	94E9	C5f	28	ems	88A2	F5a
19	DS02168	86E1	C6a	55	Aats-glupro	95C13	F5d
56	Rox8	95D5	C6b	9	DS04025	84C7	F5e
57	69F5	95D10	C6c	10	aEst	84D8	F6a
58	Acp95EF	95E6	C6d	63	tld	96A22	F6f
72	Pkc98E	98F8	C6h	14	Syn	85F15	G2b
43	Dl	92A1	C7b	15	DS05661	86C1	G2d
8	DS08010	84C1	C7e	16	TfIIFb	86C4	G3c
53	11H7	95C7	D1e	40	DS06686	90E3	G4b
21	Gst	87B8	D1g	17	DS01290	86C6	G4c
3	Pak	83E5	D2a	37	Pxd	89E11	G4f
48	hh	94E1	D2d	36	abd-A	89E3	G4g
29	trx	88B4	D2h				

is considerably larger than the parsimony inferred number of inversions. The posterior distribution of λ is plotted in Fig. 8. A 95% HPD credible interval for λ is given by $(64.14 \leq \lambda \leq 125.00)$.

A different approach to the estimation of the number of inversions has been taken by Durrett (2002). He showed that when N markers are considered, if we add a 0 at the beginning and an $N+1$ at the end, then -2 + the number of conserved adjacencies (places where the two adjacent markers differ by 1) is an eigenfunction for the shuffling process with eigenvalue $(N-1)/(N+1)$. That is, a randomly chosen inversion reduces the average value of this statistic by a factor $(N-1)/(N+1)$. The number of conserved adjacencies in this data set is 11, while the initial number is 80, so setting

$$78 \cdot \left(\frac{78}{80}\right)^m = 9$$

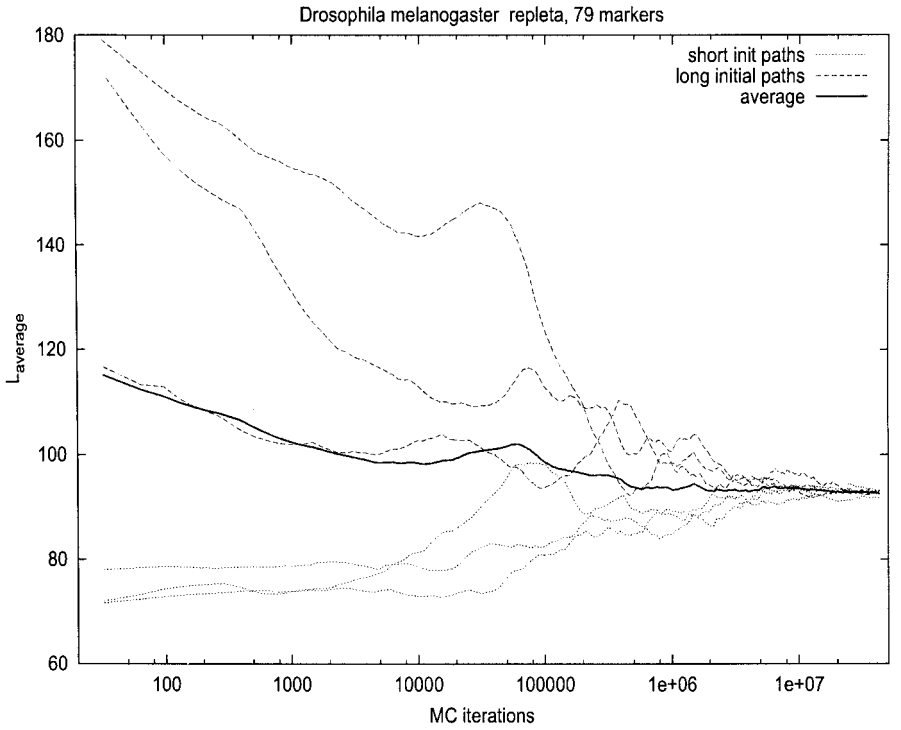

FIG. 7. Convergence of the ergodic average of the number inversions along the chain ($L_{average}$) for the data set containing 79 markers from *D. melanogaster* and *D. repleta.* The thick solid line shows the average among the 6 replicate chains. The vertical line indicates the end of the burn-in time.

and solving gives $m = \ln(9/78)/\ln(78/80) = 85.3$, which compares well with the estimates obtained above.

Even though we estimate that the order of the markers has been shuffled a large number of times, its distribution is not yet randomized. One way of seeing this is that $P(L|D)$ becomes very small for large values of L. Theoretical results of Durrett (2002) imply that when N is large, randomizing the marker order will take at least $(N/2)\ln N$ events. When $N = 79$, this is 173 inversions. One way of seeing that the distribution is not yet random is to observe that Spearman's rank correlation of the two marker orders is $\rho = 0.326$, which is significant at the $p = 0.001$ level. As Ranz, Casals, and Ruiz (2001) have already observed, this observation also casts doubt on the assumption that all inversions are equally likely. In 10,000 simulations of 40 randomly chosen inversions acting on 79 markers, the average rank correlation is only 0.0423, and only 4.3% of the runs had a rank correlation larger than 0.325.

5. DISCUSSION

Elucidating the evolutionary history of chromosomes is one of the major goals of computational genomics. We have here shown that a full probabilistic approach to the problem of inferring the number of inversions between two chromosomes is feasible, even for a data set with many inversions. We have seen that the probability that the true number of inversions is close to the minimum possible number of inversions is very small for the *Drosophila* data set. For such data sets, the parsimony estimates are not very meaningful and should not be applied.

In contrast to the parsimony approach, our Bayesian approach allows one to test hypotheses such as: Do all inversions occur at the same rate? Are inversion rates constant among lineages? The second question is motivated by the observation, see Graves (1996), that rodent chromosomes have undergone an unusually high number of genomic rearrangements per unit of evolutionary time.

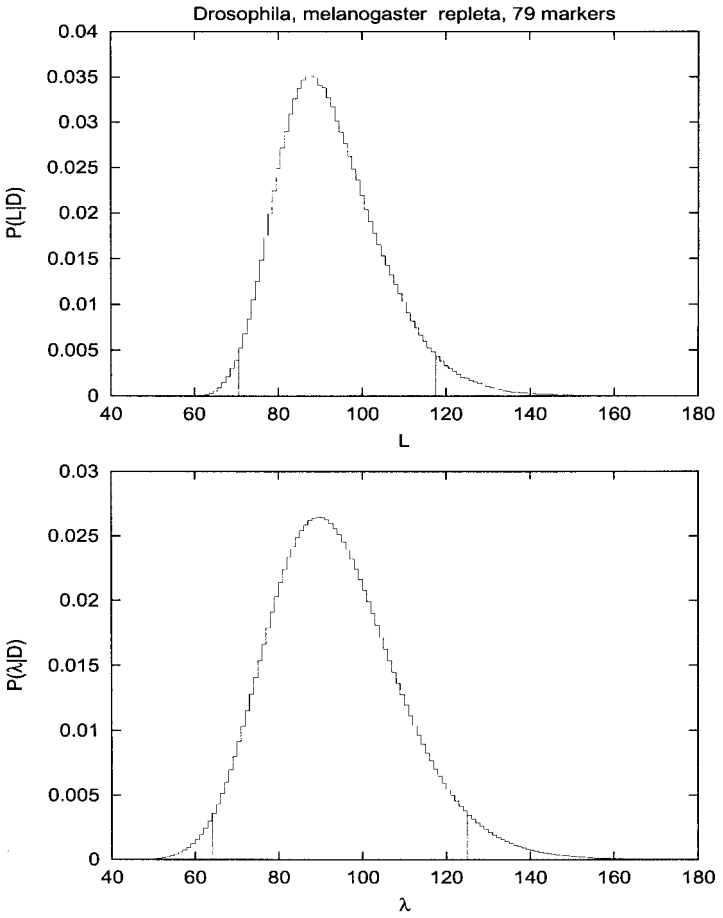

FIG. 8. The posterior distributionof L and λ for the data set of 79 markers from *D. melanogaster* and *D. repleta.* Dotted lines indicate 95% credible intervals. The vertical lines indicate 95% credible intervals.

Larget and Simon (2001) have considered a similar method applicable to data sets with fewer markers but with more than two species. They succeeded in developing a probabilistic method for estimating phylogenies for mitochondrial genomes based on taking genomic rearrangements into account. Extending MCMC methods similar to the ones used here to the problem of multiple species, will allow inferences of phylogenies using data sets with many markers (Larget, Simon, and Kadane, 2002). However, the heterogeneity of rates on different lineages casts doubt on the usefulness of the number of inversions as a molecular clock.

Beyond the phylogeny problem, another natural extension is to include translocations, transpositions, and gene duplications. Hannenhalli and Pevzner (1995b) have already solved the corresponding genomic distance problem. A Bayesian method which includes these evolutionary mechanisms would provide a general framework for the analysis of chromosomal evolution. However, in many comparisons (e.g., human vs. mouse) the method will now face the challenge of dealing with thousands of markers that have been subjected to hundreds of rearrangements.

ACKNOWLEDGMENTS

Durrett's research was partially supported by NSF grant DMS 9877066 and a supplement to C.F. Aquadro's NIH grant GM36431. Nielsen's research was supported by NSF grant DEB-0089487 and HFSP grant RGY0055/2001-M.

REFERENCES

Bafna, V., and Pevzner, P. 1995. Sorting by reversals: Genome rearrangements in plant organelles and the evolutionary history of *X* chromosome. *Mol. Biol. Evol.* 12, 239–246.

Band, M.R., Larson, J.H., Rebeiz, M., Green, C.A., Heyen, W., Donovan, J., Windish, R., Steining, C., Mahyuddin, P., Womack, J.E., and Lewin, H.A. 2000. An ordered comparative map of the cattle and human genomes. *Genome Res.* 10, 1359–1368.

Caprara, A. 1997. Sorting by reversals is difficult. *Proc. RECOMB 97*, 75–83.

Caprara, A. 1999. Sorting permutations by reversal and Eulerian cycle decompositions. *SIAM J. Discrete Math.* 12, 91–110.

Caprara, C., and Lancia, G. 2000. Experimental and statistical analysis of sorting by reversals. In *Comparative Genomics*, D. Sankoff and J.H. Nadeau, eds., Kluwer Academic Publishing, Dordrecht, The Netherlands.

Durrett, R. 2002. Shuffling chromosomes. Preprint.

Even, S., and Goldreich, O. 1981. The minimum-length generator sequence problem is NP-hard. *J. of Algorithms* 2, 311–313.

Gelman, A., and Rubin, D.B. 1992. Inference from iterative simulation using multiple sequences (with discussion). *Statist. Sci.* 7, 457–511.

Graves, J.A. 1996. Mammals that break the rules: Genetics of marsupials and monotremes. *Ann. Rev. Genet.* 30, 233–260.

Hannenhalli, S., and Pevzner, P.A. 1995a. Transforming cabbage into turnip (polynomial algorithm for sorting signed permutations by reversals). *Proc. 27th Ann. ACM Symposium on the Theory of Computing*, 178–189. Full version in the *J. ACM* 46, 1–27.

Hannenhalli, S., and Pevzner, P. 1995b. Transforming men into mice (polynomial algorithm for the genomic distance problem. *Proc. 36th Ann. IEEE Symposium on Foundations of Computer Science*, 581–592.

Kececioglu, J., and Sankoff, D. 1995. Exact and approximation algorithms for sorting by reversals, with applications to genome rearrangement. *Algorithmica* 13, 180–210.

Larget, B., Simon, D.L., and Kadane, J.B. 2002. Bayesian phylogenetic inference from animal mitochondrial genome arrangements. *J. Royal Stat. Sci.* To appear.

Palmer, J.D., and Herbon, L.A. 1988. Plant mitochondrial DNA evolves rapidly in structure but slowly in sequence. *J. Mol. Evol.* 28, 87–97.

Pevzner, P. 2000. *Computational Molecular Biology: An Algorithmic Approach*, MIT Press, Cambridge, MA.

Ranz, J.M., Casals, F., and Ruiz, A. 2001. How malleable is the eukaryotic genome? Extreme rate of chromosomal rearrangement in the Genus Drosophila. *Genome Res.* 11, 230–239.

Simon, D.L., and Larget. B. 2001. Phylogenetic inference from mitochondrial genome arrangement data. *Computational Science—ICCS 2001*, *Lecture Notes in Computer Science.*

Address correspondence to:
Rasmus Nielsen
Department of Biological Statistics and Computations Biology
439 Warren Hall
Cornell University
Ithaca, NY 14853

E-mail: rn28@cornell.edu