The Metropolized Partial Importance Sampling MCMC mixes slowly on minimum reversal rearrangement paths

István Miklós, Bence Mélykúti, and Krister Swenson

***Abstract*—Markov chain Monte Carlo has been the standard technique for inferring the posterior distribution of genome rearrangement scenarios under a Bayesian approach. We present here a negative result on the rate of convergence of the generally used Markov chains. We prove that the relaxation time of the Markov chains walking on the optimal reversal sorting scenarios might grow exponentially with the size of the signed permutations, namely, with the number of syntheny blocks.**

***Index Terms*—Stochastic programming (G.1.6.k), Markov processes (G.3.e), Analysis of Algorithms and Problem Complexity (F.2.m) Biology and genetics (J.3.a)**

I. Introduction

THE fact that the gene orders of genomes evolve by inversions was discovered earlier [40] than the DNA double-helix itself [43]. Although the computational problem was clearly stated already in 1941, the first study of the computational complexity of sorting by inversions was published only in the '90s [22]. The first polynomial running time algorithm was given by Hannenhalli and Pevzner [19], which has been subsequently improved [21]. The best algorithm today takes sub-quadratic time to find an optimal sorting path [41], and a linear running time algorithm exists that calculates the minimum number of inversions needed to transform one genome into another (without giving a sorting path) [3].

Unfortunately, the problem does not scale well with the number of genomes: the inversion median problem – namely, finding an intermediate genome that minimizes the sum of distances from three input genomes – is known to be NP-complete [12]. Several heuristic approaches have been published on finding the optimal inversion median of three genomes, and some of them are based on considering all optimal sorting paths. Siepel introduced an algorithm for finding all sorting reversals [38]. Braga *et al.* [11] gave an algorithm to find all optimal sorting scenarios, however, the running time of this algorithm might grow exponentially with the length of the input genome [7]. Counting all optimal sorting paths and the problem of sampling from the uniform distribution of them in polynomial time are still unsolved.

Markov chain Monte Carlo methods (MCMC) [29], [20] for genome rearrangement have been introduced a few years ago, which try to explore the posterior distribution of rearrangement paths instead of highlighting a single optimal one. They define different models where genomes can evolve by reversals [23], [42], [24], reversals and translocations [14], or reversals, transpositions and inverted transpositions [30], [33]. The general theory of MCMC states that the Markov chain will be in the prescribed distribution after an infinite number of random steps. A Markov chain has to approximate its target distribution in a reasonable time, in other words, it has to mix quickly to be applicable in practice.

We conjectured that the mixing of MCMC methods on genome rearrangement might be slow, since for a related problem we had already had a negative result: we had showed that the sampling protocol of Ajana *et al.* [1] generates a distribution of minimum reversal sorting paths that might be very far from the uniform distribution [28].

We present a negative result in this paper: if we restrict the state space of a special type of MCMC that is used for genome rearrangement problems to the uniform distribution of minimum reversal sorting paths, the resulting Markov chain mixes slowly. Although it does not prove, it gives rise to our conjecture that the same Markov chain might mix slowly on larger spaces containing suboptimal solutions.

II. The graph of desire and reality

The genome rearrangement problem calls for a transformation of one genome into another using a set of possible mutations. Genomes are typically described as signed permutations: numbers represent the different genes and the signs represent the reading directions of genes. It is easy to show that the signed permutations with the usual composition of permutations form a group, and the mutations act on them (group action). Therefore, transforming a genome g_1 into g_2 is equivalent to sorting $g_2^{-1}g_1$ into the identity permutation, $+1,\ +2,\ \ldots, +n$ (writing products from left to right, and hence assuming that mutations act from the right).

A signed permutation can be represented as a graph of desire and reality (see, for example, [4]). In this representation, the signed permutation is transformed into a double-length non-signed permutation replacing $+i$ by $2i-1,\ 2i$ and replacing $-i$ by $2i,\ 2i-1$. This unsigned permutation is framed by 0 and $2n+1$, where n is the length of the signed permutation. Vertices of the graph of desire and reality are the numbers of the unsigned permutation together with 0 and $2n+1$. Starting with 0, every other pair of vertices are connected in the unsigned permutation with a black line, and they are called *reality edges*, since they show the reality, i.e. what the neighbor of 0 is, etc. Also starting with 0, every node $2i$ and $2i+1$ are connected with a grey arc above the row of vertices, and these grey arcs are called *desire edges*, since they show which nodes

should be neighbors to get the identity permutation. Since each vertex of the graph of desire and reality has a degree of 2, the graph decomposes into cycles. We can distinguish *oriented* and *unoriented* cycles. A cycle is oriented iff there are two reality edges with different directions on a traversing of the cycle, otherwise it is unoriented. By definition, intersecting cycles form *components*, which partition the graph. A component is oriented if it contains at least one oriented cycle, otherwise it is unoriented. The Hannenhalli-Pevzner theorem says that the minimum number of reversals necessary to sort a permutation σ that contains only oriented components is $n + 1 - c(\sigma)$, where n is the length of the signed permutation and $c(\sigma)$ is the number of cycles. Since a reversal can increase the number of cycles at most by 1 (see for example [38]), the theorem claims that if a permutation contains only oriented components, there is always a reversal that increases the number of cycles by 1 and does not create an unoriented component.

III. MCMC and Partial Importance Sampling

A discrete time Markov chain over a state space I is a random walk over the state space I and can be given by a non-negative $I \times I$ matrix P for which

$$\sum_j p_{i,j} = 1 \tag{1}$$

for all i. $p_{i,j}$ describes what the probability is that the random walk jumps into state j in the next step if the actual state is i. Under some mild conditions, such a random walk globally converges to a distribution. Roughly speaking, after a large number of steps, the actual state will be a random state following a given distribution, regardless of the starting state of the chain.

The Metropolis-Hastings algorithm is a general algorithm to create a Markov chain that converges to a prescribed distribution π. It needs a primary Markov chain that is irreducible and aperiodic on the state space I. Irreducibility means that there is a non-zero probability to reach any state j from any state i after a finite number of steps. Aperiodicity means that the greatest common divisor of the lengths of the possible cycles of the Markov chain with non-zero probability is 1. Moreover, it is also necessary that for any $x, y \in I$, $p_{x,y} > 0$ implies $p_{y,x} > 0$. The Metropolis-Hastings algorithm transforms this chain to another chain in the following two steps:

- (proposal) Draw a random y from the primary chain's transition distribution $T(\cdot|x_t)$, where x_t is the state in which the chain is after step t.
- (acceptance) Draw a random u from $U[0, 1]$. Let $x_{t+1} = y$ if
$$u \leq \frac{\pi(y)p_{y,x_t}}{\pi(x_t)p_{x_t,y}} \tag{2}$$
and let $x_{t+1} = x_t$ otherwise.

The resulting Markov chain (x_t) will be reversible, irreducible and aperiodic, and hence, it will converge to π since the detailed balance holds [29], [20].

Sometimes each point in the state space I can be represented as a vector, and the primary Markov chain modifies x_t by changing a subset (or window) of its coordinates, w. Let w' denote the newly drawn coordinates of y proposed from x_t. It is easy to show that the acceptance ratio in Eqn. (2) can be replaced by

$$\frac{\pi(y)T(x_t, w'|y)}{\pi(x_t)T(y, w|x_t)} \tag{3}$$

where $T(a, w|b)$ tells the probability of proposing a from b by choosing and modifying the coordinates w, without changing the equilibrium distribution, π, even if y can be proposed by altering a larger set of coordinates of x_t [26]. When the newly drawn coordinates of y do not depend on the respective coordinates of x_t, the algorithm is called Metropolized Partial Importance Sampling.

In the case of genome rearrangements, the state space of MCMC is the set of allowed transition paths between two genomes. Such a state space can be considered as being comprised of $(n + 2 - c(\sigma))$-tuples of genomes (g_1, intermediate genomes connecting g_1 to g_2, and g_2). A Metropolized Partial Importance Sampler cuts out a subpath from the current path, which is framed by genomes g_k and g_ℓ and draws a new subpath transforming g_k into g_ℓ. This subpath is drawn from a distribution that does not depend on the cut out subpath. In published implementations [14], [23], [24], [30], [33], [42], the new subpath is drawn step by step, drawing a new intermediate genome by considering the list of mutations that act on the current intermediate genome. If the allowed transition paths are the minimum reversal sorting paths, then the next intermediate genome is drawn by applying a random, uniformly distributed sorting reversal on the current intermediate genome. In the next section, we prove that this kind of MCMC mixes slowly in the worst case.

IV. ParIS mixes slowly on minimum reversal paths

A. Speed of Convergence of Markov chain Monte Carlo algorithms

The Markov chain Monte Carlo methods provide an algorithm that constructs a Markov chain for any input data. Below D denotes the data, and the convergence is measured as a function of the size of D. We would like to measure the convergence of a Markov chain on state space I_D with what is called the maximal variation distance from the equilibrium distribution after step k starting in an arbitrary position i_D. We define

$$\tau_{i_D}(\epsilon) := \min\{k_0 | \forall k \geq k_0, d_v(\delta_{i_D}^T P_D^k, \pi_D) \leq \epsilon\} \tag{4}$$

where δ_{i_D} is the vector whose coordinates correspond to the possible states of the state space I_D and which contains 1 for the coordinate representing state i_D, and contains 0s for all remaining coordinates, P_D is the transition probability matrix of the Markov chain, π_D is the equilibrium distribution, and $d_v(\cdot, \cdot)$ is the variational distance defined as

$$d_v(\pi_1, \pi_2) = \frac{1}{2} \sum_{i \in I} |\pi_1(i) - \pi_2(i)| \tag{5}$$

We say that a Markov chain converges quickly if

$$\max_{i_D \in I_D} \tau_{i_D}(\epsilon) \tag{6}$$

is a polynomial function of both $\log(1/\epsilon)$ and $|D|$, and the Markov chain converges slowly if there exists an ϵ such that for all $l \in \mathbb{N}$,

$$\max_{i_D \in I_D} \tau_{i_D}(\epsilon) = \Omega(|D|^l) \tag{7}$$

Aldous [2] showed that

$$\max_{i_D \in I_D} \tau_{i_D}(\epsilon) \geq \frac{\rho_D}{2(1-\rho_D)} \log\left(\frac{1}{2\epsilon}\right) \tag{8}$$

where ρ_D is the second largest eigenvalue modulus, that is $\max\{\lambda_{2,D}, |\lambda_{r,D}|\}$, where $\lambda_{2,D}$ is the second largest eigenvalue of the transition matrix P_D, and $\lambda_{r,D}$ is the smallest eigenvalue of P_D (if the Markov chain is reversible, all eigenvalues are real numbers). Consequently, if the second largest eigenvalue converges to 1 exponentially with the size of the data, then the MCMC converges slowly.

The Cheeger's inequality gives a lower bound on the second largest eigenvalue. We define the ergodic flow of a set $S_D \subseteq I_D$ as

$$F(S_D) := \sum_{x \in S_D, y \in I_D \setminus S_D} P_D(y|x)\pi_D(x) \tag{9}$$

and the conductance of a Markov chains

$$\Phi_D := \inf\left\{ \frac{F(S_D)}{\pi_D(S_D)} \middle| S_D \subset I_D, \quad 0 < \pi_D(S_D) \leq \frac{1}{2} \right\} \tag{10}$$

It can be shown [27] that

$$1 - 2\Phi_D \leq \lambda_{2,D} \tag{11}$$

It follows that the convergence of a Markov chain is necessarily slow if there are sets S_D, for which $F(S_D)/\pi_D(S_D)$ converges to 0 exponentially with $|D|$. A heuristic explanation is that the small ergodic flow between S_D and its complement means a bottleneck, and a Markov chain having a bottleneck cannot be quickly mixing. Below we construct a series of data with such S_Ds, hence prove that the proposed MCMC mixes slowly in at least one case.

B. The example

For each $n \in \mathbb{N}$ we construct a $13n-2$-long signed permutation. Fig. 1. shows the general structure of the permutation from our example. Its graph of desire and reality can be split into two parts. The first part is a single component that consists of $4n-2$ rainbow motifs, each chained to the next, with a six-long cycle chained to the end. The second part contains n repeats of ten-long cycles being equivalent to the -1, -2, -3, -4 permutation. Such permutation exists for every n. The general permutation of the first part is shown in Fig. 2, the second part contains the numbers in the identical order, one positive sign is followed by four negative signs, namely, the second part of the permutation is

$$8n-1,\ -(8n),\ -(8n+1),\ -(8n+2),\ -(8n+3),\ 8n+4, \ldots$$

It is easy to show that the first part of the permutation needs $4n$ reversals to get sorted, and it has exactly two optimal sorting paths by reversals. Moreover, these two sorting paths have only the start and end genome in common, all the intermediate genomes of the two sorting paths are different. Each ten-long cycle in the second part of the permutation needs 4 reversals to get sorted, and each of them has 26 optimal sorting paths. Of these 26, $4! = 24$ paths reverse single numbers one by one, and they form a four-dimensional hypercube, i.e. they have 14 common intermediate genomes in addition to the start and end genomes. The remaining two sorting paths reverse the first or last three numbers of such ten-long cycles alternately, twice each. The Hannenhalli-Pevzner theorem says that all sorting paths of a permutation are combinations of the sorting paths over its components, therefore there are

$$|I_D| = 2 \times 26^n \times \frac{(8n)!}{(4n)!(4!)^n} \tag{12}$$

sorting paths of the nth member of the series. This set of paths can be partitioned into two, equal size parts based on which path they use for sorting the first component. Let S_D be one of these sets. We are going to show that $F(S_D)/\pi_D(S_D)$ converges to 0 exponentially fast with n, and hence, exponentially fast with $|D| = 13n-2$.

The first observation is that

$$\frac{F(S_D)}{\pi_D(S_D)} = \frac{1}{|S_D|} \sum_{x \in S_D, y \in I_D \setminus S_D} P_D(y|x) \tag{13}$$

since π_D is the uniform distribution. We proceed by cutting S_D into three parts such that the first two parts are 'negligibly' small, and the third contains an ergodic flow towards the complement of S_D that is too small. Let $S_{D,1}$ be the subset of S_D which contains the paths in which there are less than $\left(7+\frac{9}{11}\right)n$ intermediate genomes between the first and last sorting reversals of the first component. Since each sorting path contains $8n$ reversals and $4n$ reversals sort the first component, there exists a $c_1 > 1$ for which

$$\frac{|S_{D,1}|}{|S_D|} = O\left(\frac{1}{c_1^n}\right) \tag{14}$$

since the reversals sorting the first component can be positioned without constraints in $\binom{8n}{4n}$ ways into each complete sorting path, and in less than $\binom{(7+\frac{9}{11})n}{4n}$ ways if all these mutations must be put in a window that is less than $\left(7+\frac{9}{11}\right)n$ long, and the number of possible windows in an $8n$-long series of reversals is $O(n^2)$.

The remaining set $S_D \setminus S_{D,1}$ contains sorting paths in which the complete sorting of at least $\frac{9}{11}n$ ten-long cycles are between the first and the last sorting reversals of the large component. Let $S_{D,2}$ be the subset of $S_D \setminus S_{D,1}$ that contains paths in which at most $\frac{3}{4}n$ ten-long cycles are sorted with single number reversals between the first and the last sorting reversals of the large component. It is obvious that there exists a $c_2 > 1$ for which

$$\frac{|S_{D,2}|}{|S_D|} = O\left(\frac{1}{c_2^n}\right) \tag{15}$$

since the number of ten-long cycles that are sorted with single number reversals are binomially distributed with mean $\frac{24}{26}k$ for $k \geq \frac{9}{11}n$. Hence $\frac{3}{4}n < \frac{24}{26}k$, and we can apply the Chernoff inequality.

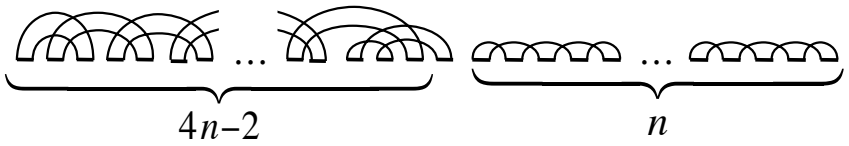

Fig. 1. The general structure of the graph of desire and reality of the signed permutation that we generated. See main text for details.

$$6n-2, 6n-3, 1, 6n-4, 6n-1, 6n-5, 2, \ldots k, 6n-2k-2, 6n+k-2, 6n-2k-3, k+1, \ldots$$
$$2n-2, 2n+2, 8n-4, 2n+1, 2n-1, -(8n-2), 2n, 8n-3, 8n-1$$

Fig. 2. The general form of the signed permutation for the first component on Fig. 1

Let $S_{D,3}$ be $S_D \setminus (S_{D,1} \cup S_{D,2})$. We have

$$\frac{F(S_D)}{\pi_D(S_D)} = \frac{1}{|S_D|} \quad \left(\sum_{x\in S_{D,1}, y\in I_D\setminus S_D} P_D(y|x) + \sum_{x\in S_{D,2}, y\in I_D\setminus S_D} P_D(y|x) + \sum_{x\in S_{D,3}, y\in I_D\setminus S_D} P_D(y|x)\right) \quad (16)$$

$|S_{D,1}|$ and $|S_{D,2}|$ are upper bounds for the first and the second sum, hence

$$\frac{F(S_D)}{\pi_D(S_D)} = O\left(\frac{1}{\min\{c_1, c_2\}^n}\right) + \frac{1}{|S_D|} \sum_{x\in S_{D,3}, y\in I_D\setminus S_D} P_D(y|x) \quad (17)$$

Recall that

$$P_D(y|x) = \sum_w T_D(y, w|x) \min\left\{1, \frac{\pi_D(y) T_D(x, w'|y)}{\pi_D(x) T_D(y, w|x)}\right\} = \sum_w \min\left\{T_D(y, w|x), T_D(x, w'|y)\right\} \quad (18)$$

and hence, $P_D(y|x)$ can be bounded by

$$P_D(y|x) \le \sum_w T_D(x, w'|y) \quad (19)$$

Let $c = \min\{c_1, c_2\}$, and we have

$$\frac{F(S_D)}{\pi_D(S_D)} \le O\left(\frac{1}{c^n}\right) + \frac{1}{|S_D|} \sum_w \sum_{\substack{x\in S_{D,3},\\ y\in I_D\setminus S_D}} T_D(x, w'|y) \quad (20)$$

where the first sum runs only on windows w that contain at least the first and the last reversal sorting the large component. The inner sum sums for all y the probability that such a subpath is proposed in the w' window that transforms y into in the $S_{D,3}$ set. For a particular y, there is a $c_3 > 1$ such that the probability of the transformation towards any $x \in S_{D,3}$, namely, $\sum_{x\in S_{D,3}} T_D(x, w'|y)$ is $O\left(\frac{1}{c_3^n}\right)$. This is because at least $\frac{3}{4}n$ ten-long cycles should be sorted by single number reversals for a successful transition. However, in the proposal distribution the number of ten-long cycles that are sorted by single number reversals is binomially distributed with mean $\frac{2}{3}k$ for k smaller than n and we can again apply the Chernoff bound. The number of ys in the subset $I_D\setminus S_D$ is exactly $|S_D|$, the number of possible windows is only $O(n^2)$, hence for some $1 < c_3^* < c_3$

$$\frac{F(S_D)}{\pi_D(S_D)} = O\left(\left(\frac{1}{\min\{c_1, c_2, c_3^*\}}\right)^n\right) \quad (21)$$

□

V. Discussion and Conclusion

In this paper we showed that the Metropolized Partial Importance Sampler might mix slowly on the set of minimum reversal paths. The cause of slow mixing are the big gaps in the optimal sorting paths, like the gaps between the two optimal sorting paths of the large component in our example. Due to these big gaps, large portions of the actual sorting path should be replaced in the proposal to get an irreducible chain. The large changes cause small acceptance ratios, and eventually slow mixing. One might argue that the Metropolized Partial Importance Sampling could be improved on the above mentioned example if it resampled mutations only on one component (whose mutations might not be consecutive on the current path). However, big gaps are common in genome rearrangements paths, for example, it can be shown that hurdle-cutting and hurdle merging [19] sorting paths are disjoint except for the start and the end genome. Both the hurdle-cutting and the hurdle-merging paths might be numerous, and we conjecture that the Metropolized Partial Importance Sampler might mix slowly even on sorting two hurdles.

Our result does not prove but suggests that the similar MCMC methods on the posterior distribution of all sorting paths [14], [23], [24], [30], [33], [42] might also mix slowly. Indeed, the key point in our proof is that the back-proposal probability is vanishingly small for the majority of the set of paths S_D, and we saw similar behavior in the case of the posterior distribution of rearrangement paths. The BADGER software [39], [24] has a pre-burn-in phase in which the proposal and backproposal probabilities are omitted from the Metropolis-Hastings ratio, and this makes the likelihood improve significantly. If that pre-burn-in phase is switched off, the burn-in phase remains at low likelihood values and no convergence is obtained. Indeed, our experiments [13] showed that without this pre-burn-in phase, the Markov chain does not converge on Yersinia phylogenies. Therefore we had to use the BADGER software instead of our software, which does not apply this pre-burn-in trick [33].

However, this proof does not imply in any sense that no fast mixing Markov chain exists for sampling from the uniform distribution of minimum reversal sorting paths or posterior distributions of genome rearrangement paths under a Bayesian framework. Indeed, there are at least two possible ways to improve the mixing of Markov chains: with novel proposals that might destroy bottlenecks and with parallel chains that exchange information. We show one example for each.

- Let a reversal be described as a double cut-and-join (DCJ) mutation [8]. The DCJ representation of a reversal tells which adjacencies are changed in the signed permutation. Let sorting paths be described by their series of reversals in DCJ representation. For example, the sorting path: $+3,+4,-1,-2 \rightarrow +1,-4,-3,-2 \rightarrow +1,+2,+3,+4$ is represented by $(0,b3|b1,e2)\ (e1,e4|b2,5)$. This means that before the first reversal, the beginning of gene 3 was at the beginning of the permutation (represented as 0), the beginning of gene 1 was in adjacency with the end of gene 2, and the first reversal swapped the positions $b3$ and $b1$. Similarly, the second reversal breaks the adjacencies between $e1$ and $e4$ and between $b2$ and the end of the permutation by swapping $e4$ and $b2$. Note that $(a,b|c,d)$ means the same reversal as $(d,c|b,a)$, but differs from, for example, $(b,a|c,d)$.
  Let the vertices of a graph be the minimum reversal paths of a signed permutation. Let two points of this graph be connected iff at most four, not necessarily consecutive reversals can be removed from each of their DCJ representations such that the remaining patterns will be the same (note that the remaining representations of DCJ mutations might not represent valid DCJ operations). Our conjecture is that the graph will always be connected if the signed permutation contains only oriented components. Above this conjecture, it is an open question if such fixed number of removals holds for all signed permutations, and if so, the so-obtained Markov chain (namely, remove a fixed number of not necessarily consecutive reversals and put back reversals not necessarily to the same place) can be transformed into a quickly mixing MCMC. The hope that such a Markov chain might be quickly mixing is due to the fact that in such Markov chain there is a polynomial lower bound for the backproposal probabilities (and hence for the acceptance ratio) while the diameter of the Markov chain will grow also polynomially with the problem size.
- For an n long, signed permutation that can be sorted in k steps, we create a Markov chain whose states are $k+1$-tuples. The first coordinate of any element in the state space contains the signed permutation, and the lth coordinate contains a transformation path from the given signed permutation to an other signed permutation that can be sorted in $l-1$ steps.
  We define a Markov chain on this set that changes two consecutive coordinates, the lth and $l+1$st in the following way. The new lth coordinate is the shortened path in the old $l+1$st coordinate, and the new $l+1$st coordinate is a random extension of the old lth coordinate. Applying the appropriate Metropolis-Hastings ratio, the Markov chain will converge to the uniform distribution. We could prove that this Markov chain on its own generally will not converge quickly to the equilibrium distribution [31], however, the mixing is quick on Yersinia data if the following inside-swapping step is also added to the transition kernel of the Markov chain. The inside-swapping step swaps two consecutive commuting reversals on one of the sorting paths. To do such a step, we first choose a random i between 2 and $k+1$, then we count all the neighboring reversals in the sorting path in the ith coordinate that can be swapped. We select a random pair, calculate how many commuting reversal neighbors there are after swapping them, and calculate the corresponding Metropolis-Hastings ratio with which we accept the change. We compared this Markov chain with the Importance Sampling method of Ajana *et al.* [1], and showed that this latter method explores only a negligible part of the possible sorting reversals. Since the Partial Importance Sampling method applies the same Importance Sampling transition kernel, this again suggests that the slow convergence of the Markov chain we described in this manuscript might be a general problem in case of real data, not only for the example we gave.

We also would like to highlight that a commonly used method, Parallel Tempering [16], also known as $(MC)^3$ [37] will not work. Indeed, we showed that an MCMC might mix slowly even if the target distribution is the uniform one, and the uniform distribution cannot be further heated.

ACKNOWLEDGMENT

This work was supported by the BBSRC grant BB/C509566/1. IM was also supported by a Bolyai postdoctoral fellowship and an OTKA grant F61730. BM acknowledges financial support from the EPSRC through the Life Sciences Interface Doctoral Training Centre, University of Oxford, UK.

REFERENCES

[1] Ajana, Y., Lefebvre, J.-F., Tillier, E.R.M., El-Mabrouk, N.: Exploring the Set of All Minimal Sequences of Reversals - An Application to Test the Replication-Directed Reversal Hypothesis. *Lecture Notes in Computer Science*, vol. 2452, pp. 300–315, 2002.

[2] Aldous, D.J.: Some inequalities for reversible Markov chains. *Journal of the London Mathematical Society*, vol. 2 num. 25, pp. 564–576, 1982.

[3] Bader, D.A., Moret, B.M.E., Yan, M.: A linear-time algorithm for computing inversion distance between signed permutations with an experimental study. *J. Comp. Biol.*, vol. 8, num. 5, pp 483–491, 2001.

[4] Bader, M., Ohlebusch, E.: Sorting by weighted reversals, transpositions and inverted transpositions. *Proceedings of RECOMB2006, Lecture Notes in Bioinformatics*, vol. 3909, pp. 563–577, 2006.

[5] Bafna, V., Pevzner, A.: Sorting by transpositions. *SIAM J. Disc. Math.*, vol. 11, num 2, pp 224–240, 1998.

[6] Bergeron, A.: A very elementary presentation of the Hannenhalli-Pevzner theory. *Proceedings of CPM2001*, pp. 106–117, 2001.

[7] Bergeron, A., Chauve, C., Hartman, T., St-Onge, K.: On the properties of sequences of reversals that sort a signed permutation. *In: Proceedings of JOBIM2002* pp. 99–107, 2002.

[8] Bergeron, A., Mixtacki, J., Stoye, J.: A unifying view of genome rearrangements . *Proceedings of WABI2006*, pp 163–173, 2006.

[9] Berman, P., Hannenhalli, S., Karpinski, M.: 1.375-Approximation Algorithm for Sorting by Reversals. *Proceedings of ESA2002*, pp. 200–210, 2002.

[10] Blanchette, M., Kunisawa, T., Sankoff, D: Parametric genome rearrangement. *Gene*, vol. 172, pp. GC11–GC17, 1996.

[11] Braga, A., Sagot, M.F., Scornavacca, C., Tannier, E.: The solution space of sorting by reversals. *Lecture Notes in Bioinformatics*, vol. 4463, pp. 293–304, 2007.

[12] Caprara, A.: Formulations and hardness of multiple sorting by reversals. *Proc. 3rd Annual International Conference on Research in Computational Molecular Biology*, pp. 84-94, 1999.

[13] Darling, A., Miklós, I., Ragan, M.: Dynamics of genome rearrangement in bacterial populations. *PLoS Genetics*, vol. 4, num. 7., e1000128.

[14] Durrett, R., Nielsen, R., York, T.L.: Bayesian estimation of genomic distance. *Genetics*, vol. 166, pp. 621–629, 2004.

[15] Eriksen, N.: (1+ε)-approximation of sorting by reversals and transpositions. *Proceedings of WABI2001, LNCS*, vol. 2149, pp. 227–237, 2001.
[16] Geyer, C.J.: Markov chain Monte Carlo maximum likelihood. In: Keramigas, E., Editor, 1991. Computing Science and Statistics: The 23rd Symposium on the Inference, Interface Foundation, Fairfax, pp. 156163.
[17] Gu, Q-P., Peng, S., Sudborough, H.I.: A 2-Approximation Algorithm for Genome Rearrangements by Reversals and Transpositions. *Theor. Comp. Sci.*, vol. 210, no. 2, pp. 327–339, 1999.
[18] Hannenhalli, S.: Polynomial algorithm for computing translocation distance between genomes. *Proceedings of CPM1996*, pp.168–185, 1996.
[19] Hannenhalli, S., Pevzner, P.A.: Transforming Cabbage into Turnip: Polynomial Algorithm for Sorting Signed Permutations by Reversals. *Journal of ACM*, vol. 46, no. 1, pp. 1–27, 1999.
[20] Hastings, W.K.: Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.
[21] Kaplan, H., Shamir, R., Tarjan, R.: A faster and simpler algorithm for sorting signed permutations by reversals. *SIAM J. Comput.*, vol. 29, no. 3, pp. 880–892, 1999.
[22] Kececioglu, J.D., Sankoff, D.: Exact and Approximation Algorithms for Sorting by Reversals, with Application to Genome Rearrangement. *Algorithmica*, vol. 13, pp. 180–210, 1995.
[23] Larget, B., Simon, D.L., Kadane, B.J.: Bayesian phylogenetic inference from animal mitochondrial genome arrangements. *J. Roy. Stat. Soc. B.*, vol. 64, no. 4, pp. 681–695, 2002.
[24] Larget B, Simon DL, Kadane JB, Sweet D.: A Bayesian analysis of metazoan mitochondrial genome arrangements. *Mol. Biol. Evol.*, vol. 22, no. 3, pp. 486–495, 2005.
[25] Liu, J.S.: Monte Carlo strategies in scientific computing. Springer Series in Statistics, New-York. (2001)
[26] Lunter, G.A., Miklós, I., Drummond, A.J., Jensen, J.L., Hein, J.J.: Bayesian Coestimation of Phylogeny and Sequence Alignment. *BMC Bioinformatics*, vol. 6, article number 83, 2005
[27] Lawler, G.F., Sokal, A.D.: Bounds on the L^2 spectrum for Markov chains and Markov processes: A generalization of Cheegers inequality. *Transactions of the American Mathematical Society*, vol. 309, no. 2, pp. 557–580, 1988.
[28] Mélykúti, B.: The Mixing Rate of Markov Chain Monte Carlo Methods and some Applications of MCMC Simulation in Bioinformatics. *MSc thesis*, http://ramet.elte.hu/~miklosi/MSc/Melykuti_thesis.pdf
[29] Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equations of state calculations by fast computing machines. *J. Chem. Phys.*, vol. 21, no. 6, pp. 1087–1091, 1953.
[30] Miklós, I.: MCMC Genome Rearrangement. *Bioinformatics*, vol. 19, pp. ii130–ii137, 2003.
[31] Miklós, I., Darling, A.: An efficient approach to sampling optimal inversion histories with parallel Markov-chain Monte Carlo. *manuscript in preparation.*
[32] Miklós, I., Hein, J.: Genome rearrangement in mitochondria and its computational biology. *Proceedings of the 2nd RECOMB Satellite Workshop on Computational Genomics, Lecture Notes in Bioinformatics*, vol. 3388, pp. 85–96, 2005.
[33] Miklós, I., Ittzés, P., Hein, J.: ParIS genome rearrangement server. *Bioinformatics*, vol. 21, no. 6, pp. 817-820, 2005.
[34] Nadau, J.H., Taylor, B.A.: Lengths of chromosome segments conserved since divergence of man and mouse. *PNAS*, vol. 81, pp. 814–818, 1984.
[35] von Neumann, J.: Various techniques used in connection with random digits. *National Bureau of Standards Applied Mathematics Series*, vol. 12, pp. 36–38, 1951.
[36] Palmer, J.D., Herbon, L.A.: Plant mitochondrial DNA evolves rapidly in structure, but slowly in sequence. *J. Mol. Evol.*, vol. 28, pp. 87–97, 1988.
[37] Fredrik Ronquist and John P. Huelsenbeck: MrBayes 3: Bayesian phylogenetic inference under mixed models *Bioinformatics*, vol. 19, pp. 1572–1574, 2003.
[38] Siepel, A.: An algorithm to find all sorting reversals. *Proceedings of RECOMB2002*, pp. 281–290, 2002.
[39] Simon, D. and B. Larget. 2004. Bayesian Analysis to Describe Genomic Evolution by Rearrangement (BADGER), version 1.01 beta. Department of Mathematics and Computer Science, Duquesne University
[40] Sturtevant, A.H., Novitski, E.: The homologies of chromosome elements in the genus Drosophila. *Genetics*, vol. 26, pp. 517–541, 1941.
[41] Tannier, E., Sagot, M.-F.: Sorting by reversals in subquadratic time. *Proccedings of the 15th CPM, Lecture Notes in Computer Science*, pp. 1–13, 2004.
[42] York, T.L., Durrett, R., Nielsen, R.: Bayesian estimation of inversions in the history of two chromosomes. *J. Comp. Biol.*, vol. 9, pp. 808–818, 2002.
[43] Watson, J.D., Crick, F.H.C.: Molecular Structure of Nucleic Acids: A Structure for Deoxyribose Nucleic Acid, *Nature*, vol. 171, pp. 737–738, 1953.

István Miklós got his PhD in Theoretical Biology and Ecology at the Eötvös Loránd University, Budapest in 2002. He was a postdoc in the Bioinformatics group, Department of Statistics at the University of Oxford from 2002 until 2004 and at the Eötvös Loránd University from 2004 until 2006. He got a young researcher position at the Rényi Institute in 2006, and he is a visitor in the Bioinformatics group in Oxford in the 2007/2008 academic year.

Bence Mélykúti received his MSc degree in mathematics from the Eötvös Loránd University, Budapest in 2006. He is now a doctoral candidate at the Life Sciences Interface Doctoral Training Centre, University of Oxford, UK as a member of Keble College.

Krister Swenson received his Bachelors in Computer Science at University of New Hampshire and is currently a Doctoral Student at the School of Computer and Communication Sciences, Ecole Polytechnique Fédérale de Lausanne, Switzerland.