BAYESIAN AND MAXIMUM LIKELIHOOD ESTIMATION OF GENETIC MAPS

Thomas L. York[†], Richard T. Durrett[†‡], Steven Tanksley[¥] and Rasmus Nielsen[†]

[†]Department of Biological Statistics and Computational Biology, [‡]Department of Mathematics,

[¥]Department of Plant Breeding, Cornell University,

Ithaca, NY 14850, USA

Running head:

Estimation of Genetic Maps

Proofs to be sent to:

Rasmus Nielsen

Current address: Bioinformatics Center

University of Copenhagen

Universitetsparken 15, Building 10

2100 Kbh Ø, Denmark

Email: rasmus@binf.ku.dk

Phone: +45 35 32 12 79

Summary

There has recently been increased interest in the use of Markov Chain Monte Carlo (MCMC) based Bayesian methods for estimating genetic maps. The advantage of these methods is that they accurately can deal with missing data and genotyping errors. Here we present an extension of the previous methods that makes the Bayesian method applicable to large data sets. We present an extensive simulation study examining the statistical properties of the method and comparing it to the likelihood method implemented in MAPMAKER. We show that the Maximum A Posteriori (MAP) estimator of the genetic distances, corresponding to the maximum likelihood estimator, performs better than estimators based on the posterior expectation. We also show that while the performance is similar between MAPMAKER and the MCMC based method in the absence of genotyping errors, the MCMC based method has a distinct advantage in the presence of genotyping errors. A similar advantage of the Bayesian method was not observed for missing data. We also reanalyze a recently published set of data from the eggplant and show that the use of the MCMC based method leads to smaller estimates of genetic distances.

1. **Introduction**

Estimating the marker order and the distances between markers from controlled crosses is a classical problem in statistical genomics. There are many solutions to this problem. Most methods will proceed by first identifying linkage groups and then subsequently estimate marker order and distances between markers within each linkage group. Linkage groups are typically identified by examining the likelihood function for the recombination rate between pairs of markers. For example, the popular program MAPMAKER (Lander et al. 1987) assigns a pair of markers to the same linkage group if the LOD score in favor of linkage exceeds a certain threshold (the default value is 3.0). After markers have been assigned to linkage groups, marker order is usually estimated by finding the marker order which minimizes or maximizes some statistic, typically using a heuristic optimization. The two most common

statistics are the Sum of Adjacent Recombination fractions (SAR) and the LOD score. Keller (1999), George (1999), and Rosa (2002) considered Bayesian approaches for determining marker order and estimating adjacent recombination fractions. In these studies it has been argued that the Bayesian method has an advantage over previous methods in that it can directly incorporate uncertainty due to genotyping errors into the estimates of marker order and marker distances.

Here we will discuss an extension of the previous Bayesian methods that makes the method applicable to even very large data sets. We also demonstrate how the Bayesian approach may be used for inferences regarding linkage groups and to quantify uncertainty regarding marker order. Using simulations we compare our new method to the method implemented in the popular program MAPMAKER (Lander et al. 1987) and analyze the statistical properties of these methods in the presence of genotyping errors and missing data. The new approach is applied to a large data sets from the eggplant and we compare our results to the likelihood approach implemented in MAPMAKER (Lander et al. 1987). A computer program implementing the method is made publicly available.

2. **Theory and Methods**

The statistical method we will use is a Bayesian approach which combines the likelihood function with a prior distribution to form a posterior distribution. Inferences are then based on the posterior distribution. The priors we will use are uniform priors that assign equal probability mass to all possible observations. This implies that the obtained posterior distributions also can be interpreted directly as likelihood functions (or integrated likelihood functions). The method described here is essentially that of Rosa *et al.* (2002), but it is extended and improved in a few ways: 1) we extend the method to F2 data in addition to backcross data; 2) and we present some algorithmic improvements. As a result, our method can handle realistic data sets with hundreds of markers on multiple chromosomes. In the following, we will first briefly describe the likelihood function in the case of no errors and no

missing data and our choice of priors. In the subsequent sections we will then describe how the model incorporates errors and missing data and we will then provide details of the simulation algorithm.

(i) Model description

Our approach differs from previous approaches by directly modeling the presence of chromosomes. For dense sets of markers, this will alleviate the need for prior identification of linkage groups. Consider m markers in some arrangement λ on C chromosomes, of which C_{ne} are non-empty, i.e. have markers on them. There are then $m - C_{ne}$ adjacent pairs of markers and corresponding recombination fractions θ_j. For now, we assume that the set of genotypes, $\mathbf{G}$, is known perfectly at each marker for each of n individuals, and that for each marker pair we can find the number of recombinations between them, which, summed over individuals we call R. The likelihood is then:

$$p(\mathbf{G} \mid \lambda, \boldsymbol{\theta}) = 2^{-R_{\max} C_{ne}} \prod_{j=1}^{m-C_{ne}} \theta_j^{R_j} (1-\theta_j)^{(R_{\max} - R_j)}, \qquad (1)$$

where $R_{\max} = \kappa n$ is the maximum possible number of recombinations, and κ is 1 for backcross data and 2 for F2 data. This expression assumes that there is no interference among marker intervals. For the purpose of defining the likelihood function, this assumption is typically made (e.g. Lander et al. 1987; Rosa et al 2002). However, estimates of recombination fractions can still be converted to genetic distances using mapping functions that do not assume independence.

The prior distribution for each recombination fraction θ_i is taken to be $Uniform(0, ½)$. We let the prior probability of a marker and chromosome order be $p(\lambda) \propto 1/(2^{C_1+C_0} C_0!)$ where C_0 and C_1 are the numbers of chromosomes with exactly 0 and 1 markers on them, respectively. This prior arises from a process of translocations and inversions at stationarity (Durrett, Nielsen and York 2004). Note that

two marker orders differing only by a reordering of markers within a chromosome have the same prior probability. For the special case of $C = 1$ the prior distribution is discrete uniform. The joint posterior for marker order and recombination fractions, is then given by

$$p(\lambda, \boldsymbol{\theta} \mid \mathbf{G}) \propto p(\mathbf{G} \mid \lambda, \boldsymbol{\theta})\, 2^{m-C_{ne}}\, p(\lambda) \propto 2^{-(R_{\max}+1)C_{ne}} \prod_{j=1}^{m-C_{ne}} \theta_j^{R_j} (1-\theta_j)^{(R_{\max}-R_j)} p(\lambda) \qquad (2)$$

We are interested in estimating $p(\lambda, \boldsymbol{\theta} \mid \mathbf{G})$, and for this purpose we devise a based Markov Chain Monte Carlo (MCMC) algorithm. In brief, we define a Markov chain with state space on the set of possible values of λ and $\boldsymbol{\theta}$. We then simulate paths of this Markov chain using the Metropolis-Hastings algorithm and sample values of λ and $\boldsymbol{\theta}$ from the chain at stationarity. For more information regarding MCMC methods applied to marker orders, see for example Durrett, Nielsen and York (2004) and for a general introduction to MCMC, see for example Larget (2004). The Markov chain is simulated by repeatedly updating the values of $\boldsymbol{\theta}$ and λ. The *j*th component of $\boldsymbol{\theta}$, θ_j, is readily updated by drawing from $p(\theta_j \mid \lambda, \mathbf{G}) \propto \theta_j^{R_j} (1-\theta_j)^{(R_{\max}-R_j)}$, a beta distribution. Marker order is updated using a Metropolis-Hastings step with a proposal distribution (described below) which preferentially picks inversions and translocations which replace some recombination fractions with smaller ones.

(ii) Errors and Missing data

Let the observed genotypes (including errors and missing data) be $\mathbf{M}$, and let $\mathbf{G}$, the true genotypes, now be a parameter of the model, which also now includes parameter μ, the missing data rate, and π, the error rate. Following Rosa *et al.* (2002), updates to $\mathbf{G}$ are made by sampling from its full conditional distribution (i.e. using a Gibbs update):

$$p(\mathbf{G} \mid \lambda, \boldsymbol{\theta}, \mathbf{M}, \pi, \mu) \propto p(\mathbf{M} \mid \mathbf{G}, \pi, \mu) p(\mathbf{G} \mid \lambda, \boldsymbol{\theta}) \tag{3}$$

In the backcross case there are 3 observed genotypes, [homozygous (aa), heterozygous (aA), and missing data (ax)], and 2 possible true genotypes, aa and aA. The probability of missing data is μ, and the probability that a genotype not coded as missing is in error is π; i.e.:

$$p(m_{ij} = ax \mid g_{ij} = aa) = p(m_{ij} = ax \mid g_{ij} = aA) = \mu,$$

$$p(m_{ij} = aA \mid g_{ij} = aa) = p(m_{ij} = aa \mid g_{ij} = aA) = (1-\mu)\pi,$$

$$p(m_{ij} = aa \mid g_{ij} = aa) = p(m_{ij} = aA \mid g_{ij} = aA) = (1-\mu)(1-\pi),$$

where m_{ij} and g_{ij} are the observed and true genotypes, respectively, of individual i at locus j. It is convenient to update the g_{ij} one at a time using

$$p(g_{ij} \mid \mathbf{G}_{-ij}, \lambda, \boldsymbol{\theta}, \mathbf{M}, \pi, \mu) \propto p(m_{ij} \mid g_{ij}, \pi, \mu) p(g_{ij} \mid \mathbf{G}_{-ij}, \lambda, \boldsymbol{\theta}). \tag{4}$$

The second probability on the right depends only on the neighboring genotypes and recombination fractions; i.e., (dropping the *ij* subscript, and subscripting with L and R for left and right neighboring quantities):

$$p(g \mid g_L, \theta_L, g_R, \theta_R) \propto \theta_L^{\,r_L} (1-\theta_L)^{(1-r_L)} \theta_R^{\,r_R} (1-\theta_R)^{(1-r_R)}, \tag{5}$$

where r_L is 0 (1) if g and g_L are the same (different), and similarly for r_R. This is for a marker with neighboring markers on both sides, for the leftmost marker on a chromosome, for example, the factors involving θ_L and r_L become 1.

In the F2 case, an added complication arises when considering two heterozygous markers. Notating parental genotypes as AB/AB and ab/ab and the F1 genotype as AB/ab, the pair of heterozygous markers is either AB/ab (no recombinants) or Ab/aB (2 recombinants). This can be handled within the same framework by including linkage phase information in $\mathbf{G}$, i.e. $g_{ij} \in$ {a/a, a/A, A/a, A/A}; lumping together of A/a and a/A as heterozygous is a form of missing data.
There are then 6 observed genotypes, (aa, AA, aA, ax, Ax, and xx) and 4 possible true genotypes (a/a, a/A, A/a, A/A). In the absence of missing data we assume the probability of a genotype being miscoded is π, i.e. given a true genotype, one of the three possible observed genotypes is correct and occurs with probability $1-\pi$; we assume each of the two incorrect observed genotypes occurs with probability $\pi/2$. We assume the probabilities of missing data are μ_1 (data missing for one allele), and μ_2 (data missing for both alleles). When both errors and missing data are present we model their effects as follows (using the notation $p_{m,g} \equiv p(m \mid g, \pi, \mu_1, \mu_2)$):

$$p_{aa,a/a} = p_{AA,A/A} = p_{aA,a/A} = p_{aA,A/a} = (1-\mu_1-\mu_2)(1-\pi)$$

$$p_{aa,A/A} = p_{AA,a/a} = p_{aA,A/A} = p_{aA,a/a}$$
$$= p_{aa,a/A} = p_{aa,A/a} = p_{AA,a/A} = p_{AA,A/a} = (1-\mu_1-\mu_2)\frac{\pi}{2}$$

$$p_{ax,a/a} = p_{Ax,A/A} = \mu_1(1-\tfrac{3}{4}\pi)$$

$$p_{ax,A/A} = p_{Ax,a/a} = \tfrac{3}{4}\mu_1\pi$$

$$p_{ax,a/A} = p_{ax,A/a} = p_{Ax,a/A} = p_{Ax,A/a} = \frac{\mu_1}{2}$$

$$p_{xx,g} = \mu_2$$

Now $p(g \mid g_L, \theta_L, g_R, \theta_R) \propto \theta_L^{r_L}(1-\theta_L)^{(2-r_L)}\theta_R^{r_R}(1-\theta_R)^{(2-r_R)}$, where now r_L is 0 if g and g_ℓ are equal, 1 if one is homozygous the other heterozygous, and 2 if they are different but both homozygous (a/a and A/A) or both heterozygous (a/A and A/a).

The parameter π is updated using $p(\pi, \mu_1, \mu_2 \mid G, M, \lambda, \theta) \propto p(M \mid G, \pi, \mu_1, \mu_2)p(\pi, \mu_1, \mu_2)$, where $p(M \mid G, \pi, \mu_1, \mu_2) = \prod_{i,j} p(m_{ij} \mid g_{ij}, \pi, \mu_1, \mu_2)$; this is proportional to

$\mu_1^{n_1}\mu_2^{n_2}(1-\mu_1-\mu_2)^{n_3}\pi^{n_4}(1-\pi)^{n_5}(1-\frac{3}{4}\pi)^{n_6}$, where the exponents n_i depend on **G** and **M** and are easily found by counting how often each factor appears in the product. We use $p(\pi, \mu_1, \mu_2) = p(\pi)p(\mu_1, \mu_2)$ with a uniform prior for π. The error rate is updated by sampling from

($p(\pi \mid G, M) \propto \pi^{n_4}(1-\pi)^{n_5}(1-\frac{3}{4}\pi)^{n_6}$ using a Metropolis-Hastings step with proposal distribution $q(\pi) \propto \pi^{n_4}(1-\pi)^{n_5+\frac{3}{4}n_6}$. The fully conditional distribution for **G** (equation (3)), is independent of μ_1 and μ_2 when normalized, so it is not necessary to update μ_1 and μ_2 to estimate **G ,** λ, and **θ** properly.

(iii) Dynamic updates of G

For efficiency, rather than updating each g_{ij} one at a time, we update, for each individual, the genotypes of all the markers on a chromosome at once, using a dynamic programming approach akin to the Viterbi algorithm. Our algorithm differs in this respect from that of Rosa *et al.* (2002). Given marker order λ, let $\mathbf{g} = (g_1, g_2, ..., g_k)$ represent the true genotypes and $\mathbf{m} = (m_1, m_2, ..., m_k)$ the observed genotypes, with g_j the genotype at the j^{th} marker from the left, and θ_j the recombination fraction between markers j and $j+1$. To draw **g** from $p(\mathbf{g} \mid \mathbf{m}, \boldsymbol{\theta}, \pi, \mu)$ we first obtain the probability distribution of g_j conditional on observed genotypes and recombination fractions from the left end of the string of markers up to marker j, $p_L(g_j) \equiv p(g_j \mid \mathbf{m}_{\le j}, \boldsymbol{\theta}_{<j}, \pi, \mu)$. For the first marker at the leftmost end of markers,

$p_L(g_1) = p(g_1 \mid m_1, \pi, \mu) = p_{m_1, g_1}$. Because m_1 effects $p_L(g_2)$ only through $p_L(g_1)$, $p_L(g_2) = p(g_2 \mid p_L(g_1), m_2, \theta_1, \pi, \mu)$, and in general $p(g_j \mid \mathbf{m}_{\le j}, \boldsymbol{\theta}_{<j}, \pi, \mu) = p(g_j \mid p_L(g_{j-1}), m_j, \theta_{j-1}, \pi, \mu)$. Upon reaching the rightmost end, $p_L(g_k) = p(g_k \mid \mathbf{m}_{\le k}, \boldsymbol{\theta}_{<k}, \pi, \mu) = p(g_k \mid p_L(g_{k-1}), m_k, \theta_{k-1}, \pi, \mu)$ is conditional on *all* m_i and θ_i. We then draw g_k from $p_L(g_k)$ and work back toward the left, next drawing g_{k-1} from $p_L(g_{k-1} \mid g_k, \theta_{k-1})$, etc. The advantage of this method is that the correlation in the unknown missing data (or errors) among markers for the same individual can be taking into account when proposing updates.

(iv) Proposal distribution for λ

In order to analyze data sets with many markers it is important to propose new marker orders in an efficient way. It is helpful to break the problem of finding a good way of proposing new marker orders into two parts: choosing a basic rearrangement operation, (for example swapping the positions of two markers), and choosing a proposal distribution specifying the probability of each such rearrangement (which two markers to swap). The basic rearrangement operations we use are inversions and, in the multiple chromosome case, translocations. Rosa *et al.* (2002) use inversions (which they call "rotation of random length segments") and swapping the positions of two markers. By inversion is meant reversing the order of some sequence of markers on a chromosome, e.g. $abcdefg \rightarrow abfedcg$. Translocation means cutting two chromosomes and then joining the pieces together at the cut ends so as to get two new chromosomes, each containing material from both the original ones. Both of these operations leave the total number of chromosomes unchanged. The number of chromosomes, C, that the program will work with is supplied by the user. If this number is larger than the actual number of chromosomes in the genome, this is not a problem as translocations can redistribute the markers so that some chromosomes have no markers. For *m* markers on a chromosome there are $m(m-1)/2$ distinct inversions and Rosa *et al.* (2002) propose each with equal probability (and similarly for marker swapping). In order to be more

efficient for genomes with many markers, we use a non-uniform proposal distribution. The proposal distribution makes use of a table of estimated recombination fractions for each pair of markers, $\hat{\boldsymbol{\theta}}$, which is calculated just once, before starting the Markov chain. We use for $\hat{\theta}_{ij}$ the maximum likelihood estimate considering only the data for markers *i* and *j* and assuming no coding errors. We expect that for the correct marker order it will usually be true of adjacent markers a and b that b is among the closest few markers to a (as measured by $\hat{\theta}$) and vice versa. We use this idea to choose the first end of a section to propose inverting. Specifically we define $\rho_{\alpha\beta}$ to be the closeness rank of marker β relative to marker α, such that $\rho_{\alpha\alpha} = 0$, and if marker β is the closest distinct marker to α then $\rho_{\alpha\beta} = 1$, etc. We define $R_{ab} = (\rho_{ab} + \rho_{ba})/2$ and define R_{avg} to be the average of R_{ab} over all adjacent marker pairs. The break at the first end of the proposed inversion is chosen to lie between adjacent markers a and b with relative probability $f(R_{ab})$. The end of an inversion can also lie between a chromosome end and an adjacent marker; this is proposed with a relative probability of $f(R_{avg})$ for each such pair. The function f should be increasing in order to preferentially propose breaking apart markers which are not likely to belong together. [Specifically, we use $f(x) = \min(1.6^{x-m/C}, 1)$, where *C* is the number of chromosomes. The idea is that approximately *m/C* markers will be on the same chromosome as marker a and their rankings will be informative, but the rest of the markers are on different chromosomes and are all equally poor candidates to be adjacent.] Having chosen the first end between a and b, the other end is preferentially chosen between c and d so as to have smaller recombination fractions at the newly created adjacencies compared to the adjacencies which are lost in the inversion, i.e. inversions for which $\Delta = \hat{\theta}_{ac} + \hat{\theta}_{bd} - \hat{\theta}_{ab} - \hat{\theta}_{cd}$ is small are proposed with higher probability. [In particular, the probability is $e^{-20\Delta}$.] We define $\hat{\theta}_{avg}$ to be the average of $\hat{\theta}_{\alpha\beta}$ over all adjacent marker pairs and use $\hat{\theta} = \hat{\theta}_{avg}$ for adjacencies between chromosome ends and markers when calculating Δ. When a new marker order is

proposed, new recombination fractions are needed for the newly adjacent marker pairs; these are drawn from

$$p(\theta_j \mid \lambda, \mathbf{G}) = \theta_j^{R_j}(1-\theta_j)^{(R_{\max}-R_j)} / \int_0^{\frac{1}{2}} \theta^{R_j}(1-\theta)^{(R_{\max}-R_j)} d\theta \,. \quad (6)$$

With this proposal distribution for the recombination fractions, the acceptance probability of an inversion $..ab....cd.. \rightarrow ..ac....bd..$ is

$$P_a(\lambda \rightarrow \lambda') = \min\left(1, \frac{q(\lambda' \rightarrow \lambda)\int_0^{\frac{1}{2}} \theta^{R_{ab}}(1-\theta)^{(R_{\max}-R_{ab})} d\theta \int_0^{\frac{1}{2}} \theta^{R_{cd}}(1-\theta)^{(R_{\max}-R_{cd})} d\theta}{q(\lambda \rightarrow \lambda')\int_0^{\frac{1}{2}} \theta^{R_{ac}}(1-\theta)^{(R_{\max}-R_{ac})} d\theta \int_0^{\frac{1}{2}} \theta^{R_{bd}}(1-\theta)^{(R_{\max}-R_{bd})} d\theta}\right); \quad (7)$$

where $q(\lambda \rightarrow \lambda')$ is the probability of proposing order λ' from order λ. All factors depending on recombination fractions cancel out.

(v) Estimation

Marker orders and distances sampled from the Markov chain can be used for inferences. For example, the maximum *a posteriori* probability (MAP) estimate of marker order is given by the marker order that appears most often in the chain (for chains that have been running long enough). Because of the use of a uniform prior, this estimate is also the integrated maximum likelihood (ML) estimate. The use of the word 'integrated' means here that there may have been nuisance parameters, such as error rates, which have been integrated out using the MCMC procedure. Likewise, the map distance or recombination fraction between a particular pair of markers, given a particular marker order, can be estimated either by using the posterior expectation (estimated as the average value of the recombination fraction along the chain) or as the MAP value, which in our case again equals the integrated ML estimate. One of the

issues we will explore here is whether the posterior expectation or the MAP estimate is a better estimator of the recombination fraction and map distances.

We monitor convergence by running $N > 1$ chains and looking for agreement among them. In particular we use the Gelman and Rubin (1992) statistic applied to the map distance and error rate. For each state in a chain a total map distance L is calculated by summing the map distances between each pair of adjacent markers. For each chain the mean and variance of L are kept track of, and a between-chain variance, B (the variance of the N means), and a within-chain variance W (the mean of the N variances) are defined. Convergence is indicated when B becomes small compared to W. We typically define the burn-in to end when B/W becomes less than 0.1.

A program for performing this analysis is available from http://www.binf.ku.dk/users/rasmus/webpage/ras.html.

3. **Results**

(i) Simulated data

In this section we describe results based on simulated data that will help illuminate the statistical properties of the Bayesian methods. We have analyzed sets of simulated data using both our Bayesian method and the widely used mapping program MAPMAKER (Lander et al. 1987). All the results in this section are for data simulated from genomes with 8 markers on 1 chromosome, with the markers evenly spaced with separation d and assuming no interference. We describe results for BC and F2 crosses, and for various marker spacings (d), error rates π, and numbers of individuals n. Each data point shown in this section is based on 200 data sets or more. For these data sets our criterion for end of burn-in, $B/W < 0.1$, was reached at typically 250 updates, requiring 1 CPU second on a 2.8 GHz processor. The small number of markers was chosen so that Mapmaker's "compare" function, which searches exhaustively for the best marker order, would run in a reasonably short time. Mapmaker has other ways

to find the marker order suitable for larger numbers of markers but these tend to require interaction with an intelligent user, and are therefore less appropriate for the automated analysis of many data sets.

(ii) Estimation of genetic distances

The first question we will address is how well the Bayesian method estimates genetic distances and whether use of the posterior expectation or the MAP [corresponding to the ML estimate] provides the best point estimator of genetic distance. Data was simulated assuming no interference for 8 markers, for Backcrosses (BC) with $n = 50$, BC with $n = 100$, and F2 crosses (F2) with $n = 50$. Let $L = 7d$ be the distance from leftmost to rightmost marker. We estimate L considering only states with marker order $\hat{\lambda}$ (the MAP estimate of marker order). Using either the posterior expectation or the MAP method to estimate θ_j, the corresponding Haldane's distances are summed: $\hat{L} = \sum_j d_H(\hat{\theta}_j)$, where

$d_H(\theta) = -\ln(1-2\theta)/2$. As seen in Fig. 1, using the MAP estimator of θ gives a considerably better estimate than using the posterior expectation. The estimate based on the posterior expectation is very biased because of the long tail of the likelihood function. In addition to the bias, we also evaluate the root mean square error (RMSE). The RMSE is equal to the square root of the variance plus the bias squared, and is, therefore, a measure of the performance of the method that considers both variance and bias. The MAP method also has a much lower RMSE than the posterior expectation, showing that the bias in the posterior expectation is not compensated by a similarly reduced variance.

When considering the superior MAP estimator, we also notice that the F2 method has higher bias than the BC for samples sizes of $n = 50$. The reason is presumably that the marker state in F2 data is not known for double heterozygotes. However, when considering the RMSE, the F2 with a sample size of $n = 50$ performs intermediate between the BC with $n = 50$ and $n = 100$. The well-known good performance by the F2 cross is a consequence of the fact that F2 data has twice as many informative meioses as BC data.

(iii) Estimation of marker order

We use two different measures to determine how well a method performs in terms of estimation of marker order. First, we use the proportion of times the method estimates the correct marker order. Second, we use the average distance between the true and the inferred marker order. Distances are defined in terms of number breakpoints, i.e. the number of times in the inferred marker order, marker j from the true marker order is not followed by marker $j + 1$ from the true marker order, $j = 1, 2, \ldots, 7$. The estimate of the marker order chosen is the marker order that appears most often in the simulation of the Markov chain, i.e. the MAP estimate which is also identical to the (integrated) ML estimate.

As seen in Fig. 2, the estimate from Mapmaker and our MAP estimate have essentially identical properties in terms of identification of correct marker order, in the absence of errors and missing data. Despite the differences in implementation, this is not surprising since both methods are maximum likelihood methods under the same model. It also confirms that the MCMC method accurately is able to reproduce results obtained using exhaustive searches in Mapmaker.

We also notice that the estimate for 50 F2 crosses is almost as good as the estimate from 100 Backcrosses for genetic distance up to about 5-10 cM. For small genetic distances, the chance of a double heterozygote resulting from two recombination events is small and an F2 cross provides essentially twice as much information as a Backcross. As the probability that a double heterozygote results from two recombination events increases, the advantage of the F2 method diminishes.

(iv) Correcting for genotyping errors

To illustrate the effect of genotyping errors we simulated data with varying error rates. The MCMC mehod can take errors into account without *a priori* knowing the error rate (see Theory and Methods). We can then compare the performance of the method with and without correction for errors (Figure 3).

Without error correction the estimate of L is biased towards larger values because genotyping errors appear are interpreted as recombinations; this is equally true of our Bayesian method and Mapmaker. For the Bayesian method with error correction the bias in L is small for all error rates, although there is some negative bias for the smallest error rates. Error correction improves the RMSE in L for larger error rates while giving equally good estimates at small error rates. The difference in RMSE is large for error rates of 0.005 or larger. Mapmaker has a form of error correction which assumes a prior error rate, π^*, supplied by the user. Compared with the Bayesian method with error correction, Mapmaker with $\pi^* = 0.01$ gives estimates of L which are equally good at small error rates but somewhat worse at higher rates due to greater bias.

Errors in the data (miscoding of genotypes) degrade inferences regarding marker order (Figure 4), with Mapmaker and our Bayesian method performing similarly when error correction is not used. In the Bayesian method, error correction improves these inferences when errors are present, an effect which is quite small for BC data but larger for F2 data. Furthermore, there is no penalty for using error correction when analyzing error-free data. Mapmaker's inferences regarding marker order show little benefit from using error correction, and there is a penalty for using error correction when analyzing error-free data.

The estimate of the genotyping error rate, based on the posterior expectation, is shown in Figure 5. Notice that the estimate of the error rate is approximately unbiased.

(v) Missing data

We generated simulated data with $\mu_1 = 0.03$ and $\mu_2 = 0.07$; these values are close to the rates of missing data seen in the eggplant data discussed below. We analyzed these data sets with both Mapmaker and our Bayesian method. The results are very similar both for estimation of L and estimation of marker order, as may be seen in Figure 6 and 7. The RMSE is similar for the two methods

showing that the Bayesian method does not have the same advantage when correcting for missing data as it has in the correction of genotyping errors.

(vi) MarkerOrder Results For Eggplant Data

The method described here is computationally capable of handling large data sets of hundreds of markers and individuals. To illustrate this, we re-analyzed the data by Doganlar *et al.* (2002). They published a genetic map of eggplant based on a data set consisting of genotypes at 233 markers for 58 F2 individuals obtained using MAPMAKER (Lander et al. 1987).

Although the present method can explicitly model the presence of chromosomes, it is computationally simpler first to identify linkage and then to analyze each of these linkage groups separately. Therefore, we first generated MCMC output for the full data set and analyzed it for linkage groups, and then did a separate run for each linkage group to determine the marker order within each linkage group. We define linkage groups such that markers which belong to the same linkage group should almost always be found on a single chromosome, rather than being spread over 2 or more. For a set S of markers define its linkage fraction, $f(S)$, as the fraction of the MCMC output states for which all markers in S lie on the same chromosome. For threshold linkage fraction f_l we define the set of linkage groups by dividing the markers into sets S_i (with every marker belonging to exactly one of the S_i), in such a way that $f(S_i) > f_l$ for all i, and taking the union of any two distinct sets S_i and S_j gives a set with linkage fraction less than f_l. Setting f_l = 0.9 we found a set of 13 linkage groups. These correspond exactly to the chromosomes found by Doganlar *et al.* (2002) with the exception of their chromosome 9, which appears in our analysis as two linkage groups that we refer to as 9a and 9b.

Table 1 shows the linkage fractions of our 13 linkage groups and their pair-wise unions. Linkage fractions less than 0.1 are not shown. Our 13 groups all have linkage fraction $\geq$ 0.96. The union of groups 9a and 9b has a linkage fraction of only 0.37, and there is another pairing – groups 9b and 10 –

which is actually more strongly linked. Mapmaker defines linkage groups by looking at each pair of markers, finding the maximum likelihood recombination fraction, $\hat{\theta}$, and the likelihood ratio $L(\hat{\theta})/L(1/2)$. Two markers are considered linked if $\log_{10}(L(\hat{\theta})/L(1/2)) > LOD_l$; furthermore, linkage is transitive, i.e. if A and B are linked and B and C are linked then A and C are linked. By increasing LOD_l the number of linkage groups can be increased, but it was not possible to get our set of 13 linkage groups with the Mapmaker analysis by adjusting this threshold. With LOD_l in the range 2.8 to 3.5 we find the 12 groups reported by Doganlar *et al.* (2002) and with LOD_l = 3.6 we find 13 groups, but with chromosome 3 split into two instead of chromosome 9. For defining linkage groups the Bayesian method has the advantage of taking marker order into account, unlike Mapmaker. However, in this case it seems very likely that the identification of the chromosomes, in particular chromosome 9, in Doganlar *et al.* (2002) is correct since the markers on chromosome 9 are all found on the same chromosome in tomato.

(vii) Comparison of maps for each linkage group.

In our analysis we first lumped together any pair of markers with identical sets of genotypes; if $m_{ij} = m_{ij'}$ for all i, then markers j and j' are lumped together. This left 229 markers with distinct genotypes. For each of the 13 linkage groups found as described above we did a MCMC run with 4 replicate chains. By considering the Gelman-Rubin statistic applied to L and π, requiring $B/W < 0.1$ for both, we chose 1000 updates as a conservative burn-in length, for all linkage groups except number 12 for which a burn-in of 3000 updates was needed. Run lengths of 50 times burn-in were used. Linkage group 8, for example, with 16 markers, was run for 50,000 updates for each chain, taking 600 CPU seconds on a 2.8 GHz processor. In several cases adjacent markers are separated by an estimated map distance of zero; in each of these cases reversing the order of the markers gives an order with almost the same posterior probability, so we have lumped such pairs of markers (and in 1 case three markers) together. Having

done this the marker order from our Bayesian analysis agrees perfectly with the results of a Mapmaker analysis we performed, as well as with the map of Doganlar *et al.* (2002). Map distances also agree well, with the Bayesian error corrected estimates being consistently slightly smaller. Summing all the estimated kosambi distances between neighboring markers gives 1441 cM (Bayesian) and 1508 cM (Mapmaker).

(viii) Estimation of error rate.

The posterior density for the error rate for the eggplant data set is shown in Figure 8. The error rate seems to be relatively low, most likely less than 0.01. Therefore, it is not surprising that the map distances agree well between the map of Doganlar *et al.* (2002) and the results of the MCMC analysis.

4. Discussion

The results presented here show that the new MCMC approach has a distinct advantage over previous methods, in estimation of genetic distances and marker order in the presence of genotyping errors. However, genetic distances are best estimated using the MAP or maximum likelihood estimator and can be biased if estimated using the posterior expectation based on a uniform prior. Part of the problem appears to be that in small data sets, there is so little information regarding the genetic distance for each set of markers that the information introduced by the prior tends to dominate the information in provided by the likelihood function. It is possible that for very large data sets the posterior expectation performs considerably better as a point estimator.

The commonly used program, MAPMAKER (Lander et al. 1987) performs as well as the MCMC based method when the genotyping error rate is low. However, when the genotyping error rate is reasonably large, the new MCMC method performs considerably better. The new Bayesian method is applicable to real data sets, as illustrated by the application to the eggplant data, and should be used instead of more traditional methods when the genotyping error rate may be larger than 0.005.

5. Acknowledgments.

This work was supported by NSF/NIH Grant DMS/NIGMS – 0201037.

References

Doganalar S, Frary A, Daunay M-C, Lester RN, Tanksley SD. (2002) A Comparative Genetic Linkage Map of Eggplant (*Solanum melongena*) and Its Implications for Genome Evolution in the Solanaceae. Genetics 161: 1697-1711.

Durrett, R., R. Nielsen and T. F. York 2004. Bayesian estimation of genomic distance. Genetics. 166: 621-629.

Gelman, A and Rubin, DB (1992) Inference from iterative simulation using multiple sequences, Statistical Science, **7**, 457-511.

George AW, Mengersen KL, Davis, GP. (1999) A Bayesian approach to ordering gene markers. Biometrics 55: 419-429.

Keller, AE. (1999) Estimation of genetic map distances, detection of genotyping errors, and imputation of missing genotypes via Gibbs sampling, M.S. Thesis, Cornell University.

Lander ES, Green P, Abrahamson J, Barlow A, Daly M., Lincoln S, Newburg L (1987). MAPMAKER: an interactive computer package for constructing primary genetic linkage maps of experimental and natural populations. Genomics 1: 174-181

Larget, B. 2004. Introduction to Markov Chain Monte Carlo methods in molecular evolution, In R. Nielsen (ed.), Statistical methods in molecular evolution, Springer Verlag, New York.

Rosa JMG, Yandell BS, Gianola, D. (2002) A Bayesian approach for constructing genetic maps when markers are miscoded. Genet. Sel. Evol. 34: 353-369.

Figure Legends

Fig. 1. The Bias (**a**) and root mean square (rms) error (**b**) for 3 types of data, BC $n = 50$, BC $n = 100$, and F2 $n = 50$, for the maximum *a posteriori* probability (MAP) estimate and the estimate based on the posterior expectation. It is assumed that there are no errors and no missing data.

Fig. 2. The probability of estimating the correct marker order (**a**) and the expected number of breakpoints between the true and the estimated marker order(**b**) for 3 types of data, BC $n = 50$, BC $n = 100$, and F2 $n = 50$, using estimates based on the present MCMC approach and using Mapmaker. It is assumed that there are no errors and no missing data.

Fig. 3. The Bias (**a**) and root mean square (rms) error (**b**) for 2 types of data BC $n = 100$, and F2 $n = 50$, for the maximum *a posteriori* probability (MAP) estimate. The data is simulated under varying error rates (π).

Fig. 4. The probability of estimating the correct marker order (**a**) and the expected number of breakpoints between the true and the estimated marker order(**b**) for 2 types of data, BC $n = 100$, and F2 $n = 50$, using estimates based on the present MCMC approach and using Mapmaker. The data is simulated under varying error rates (π).

Fig 5. The mean estimate of the error rate for the simulated BC and F2 data simulated under varying error rates (π).

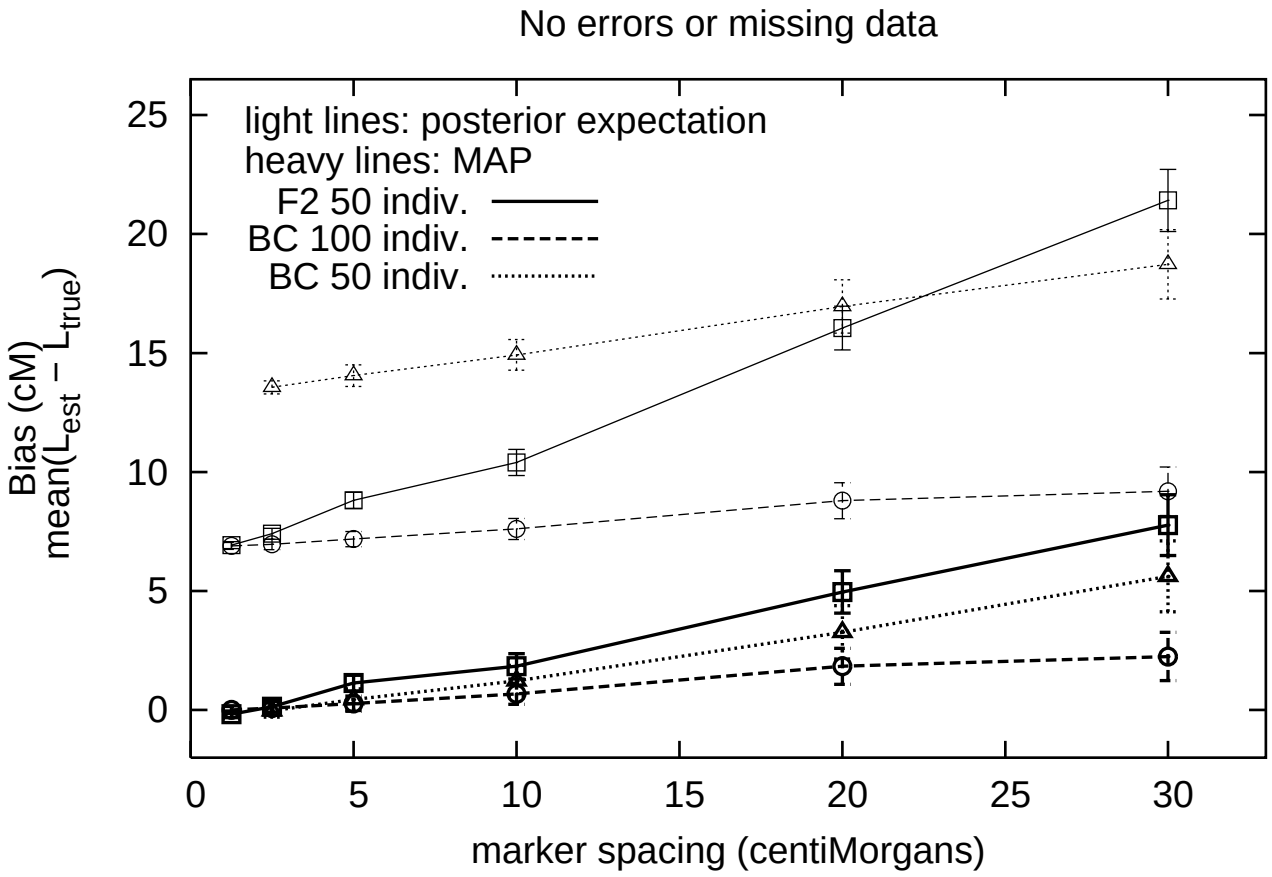
Fig 6. Estimation of map distances for the simulated BC and F2 data with missing data for various marker spacings.

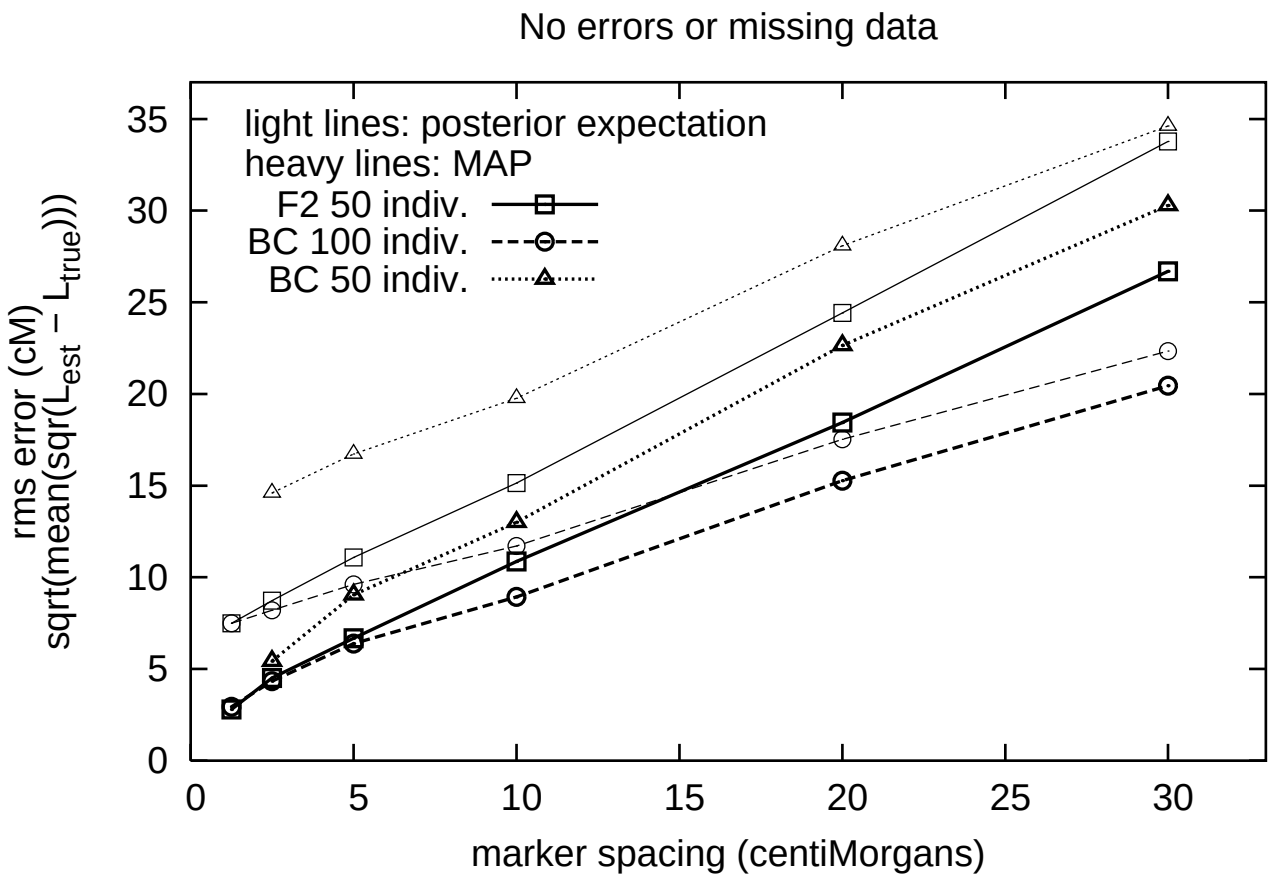
Fig. 7. Estimation of marker order for the simulated BC and F2 data with missing data for various marker spacings.

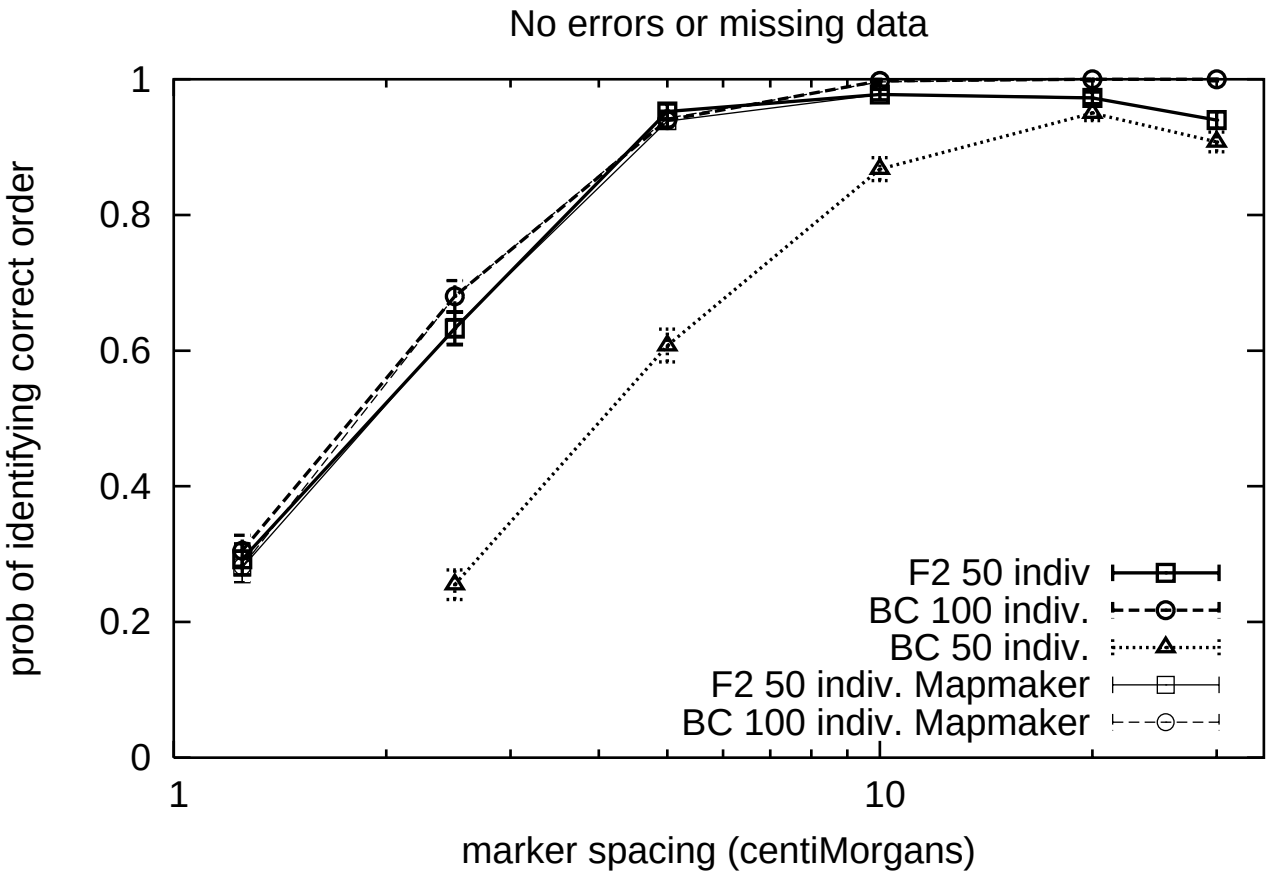
Fig 8. The posterior density of the error rate for chromosome 10 of the eggplant data.

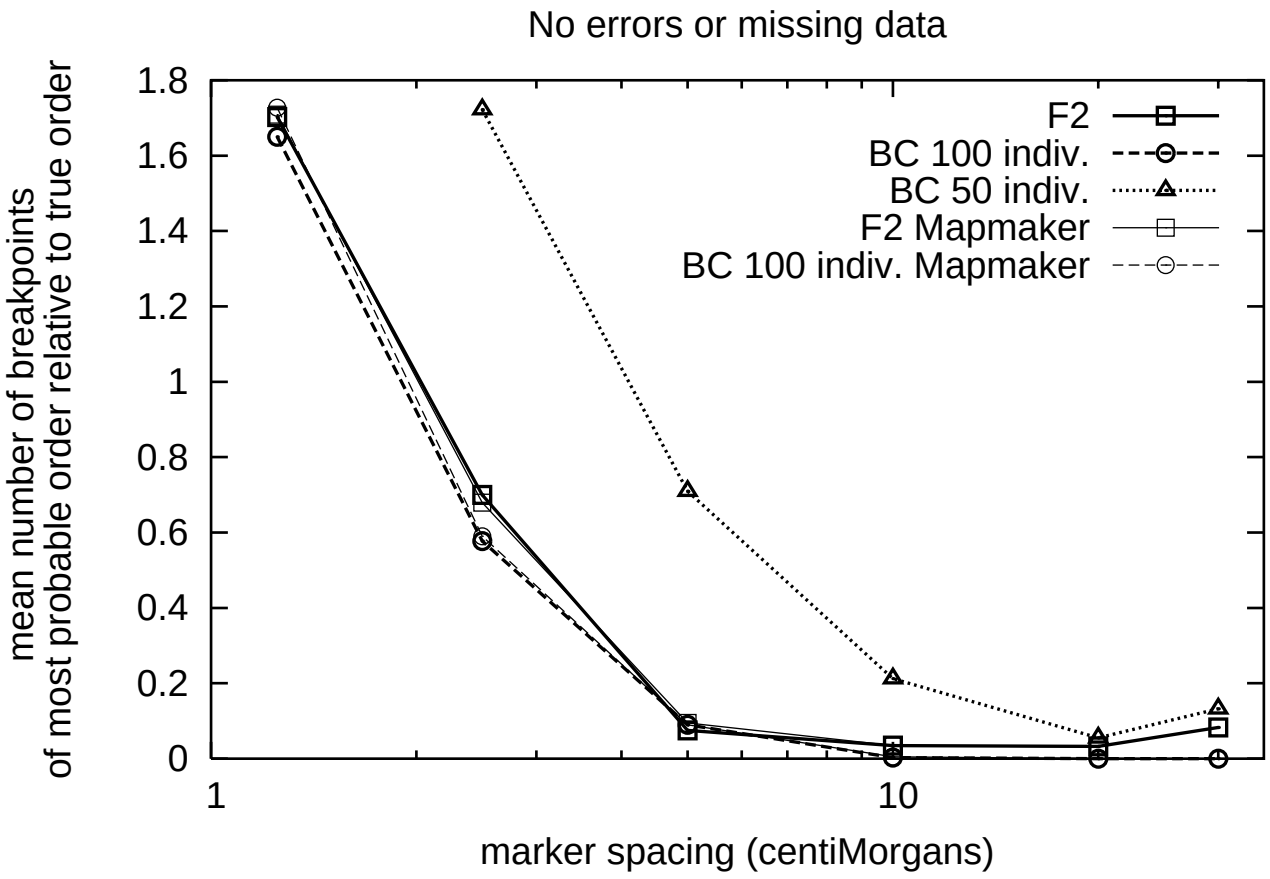
Table of linkage fractions.													
	1	2	3	4	5	6	7	8	9a	9b	10	11	12
1	1.00												
2		1											
3		0.10	0.97										
4				1									
5					1.00								
6						0.99							
7					0.21		1.00						
8								1.00					
9a									1				
9b									0.37	1			
10									0.17	0.38	1		
11												0.96	
12													1.00

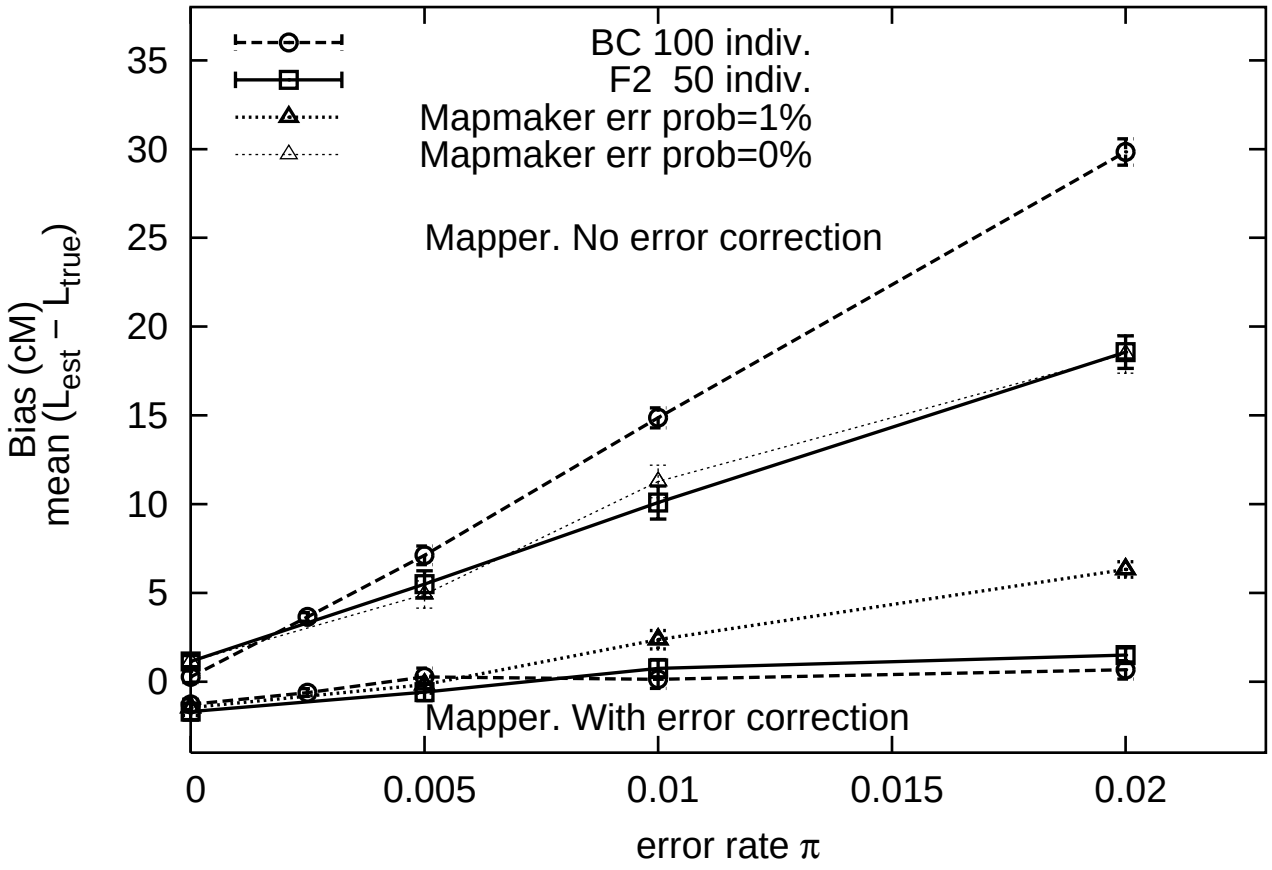
Table 1. Linkage fractions for the 13 identified linkage groups of the eggplant data.

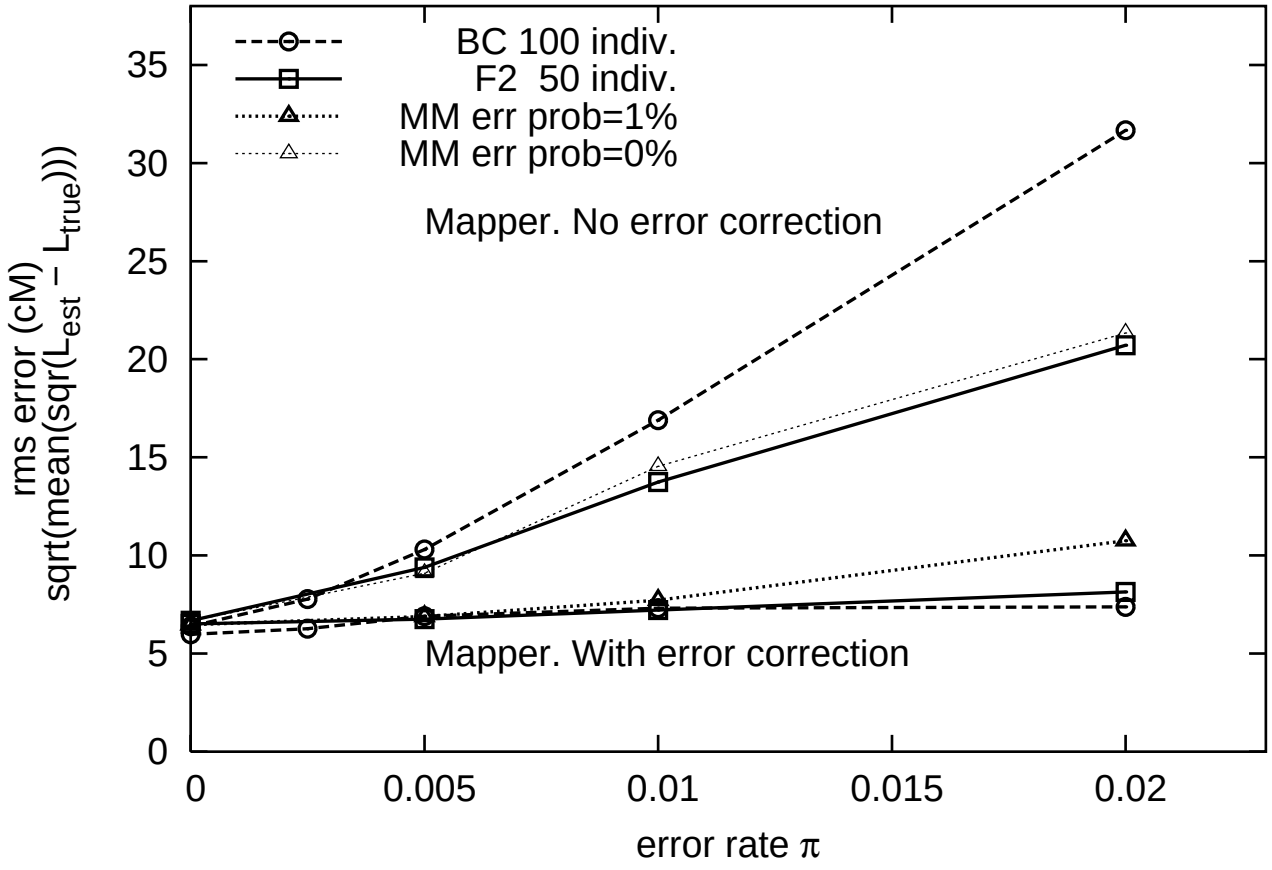

No errors or missing data
light lines: posterior expectation
heavy lines: MAP
F2 50 indiv.
BC 100 indiv.
BC 50 indiv.
Bias (cM)
mean(L_est – L_true)
marker spacing (centiMorgans)
0
5
10
15
20
25
30

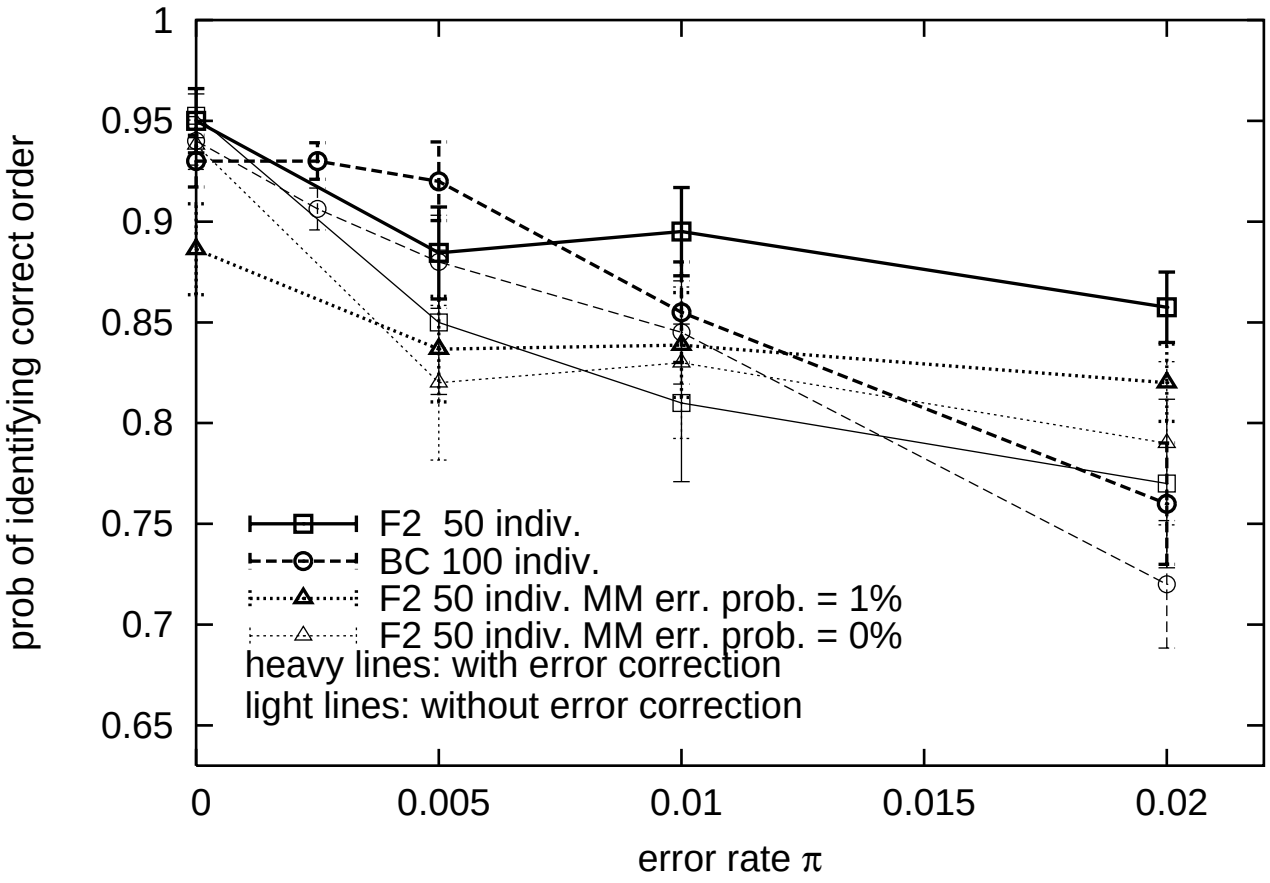

No errors or missing data
light lines: posterior expectation
heavy lines: MAP
F2 50 indiv.
BC 100 indiv.
BC 50 indiv.
rms error (cM)
sqrt(mean(sqr(L_est – L_true)))
marker spacing (centiMorgans)
0
5
10
15
20
25
30
35

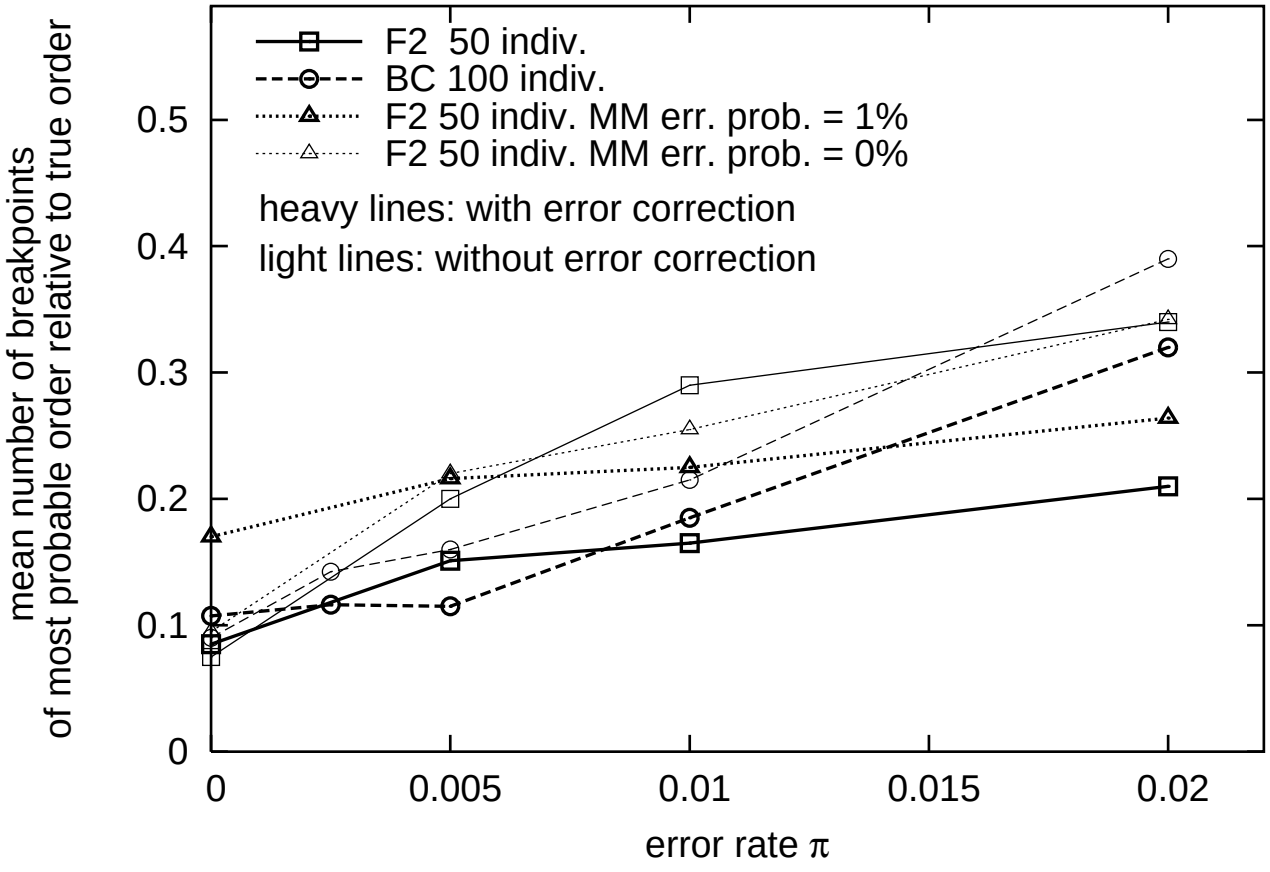

No errors or missing data
prob of identifying correct order
1
0.8
0.6
0.4
0.2
0
1
10
marker spacing (centiMorgans)
F2 50 indiv
BC 100 indiv.
BC 50 indiv.
F2 50 indiv. Mapmaker
BC 100 indiv. Mapmaker

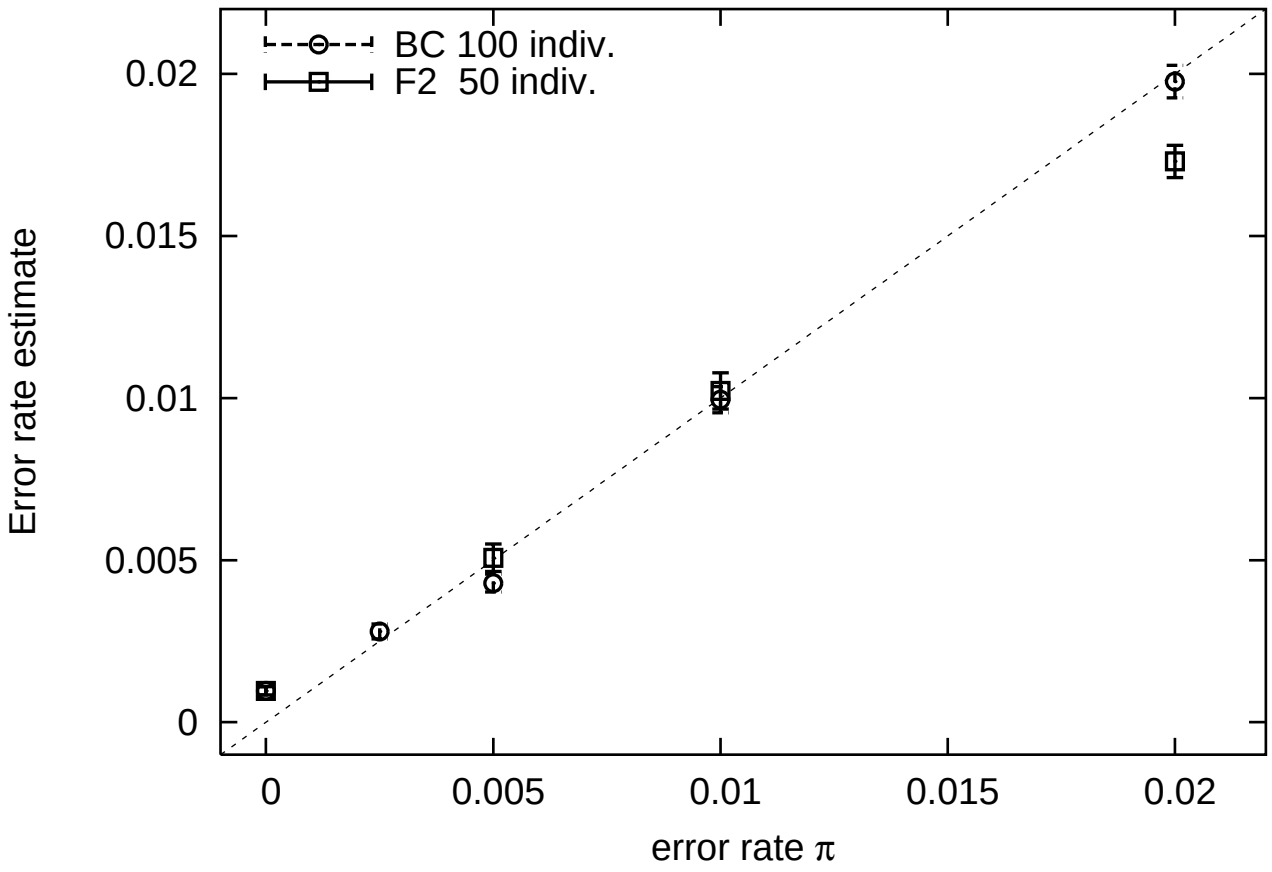

No errors or missing data
mean number of breakpoints
of most probable order relative to true order
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
1
10
marker spacing (centiMorgans)
F2
BC 100 indiv.
BC 50 indiv.
F2 Mapmaker
BC 100 indiv. Mapmaker

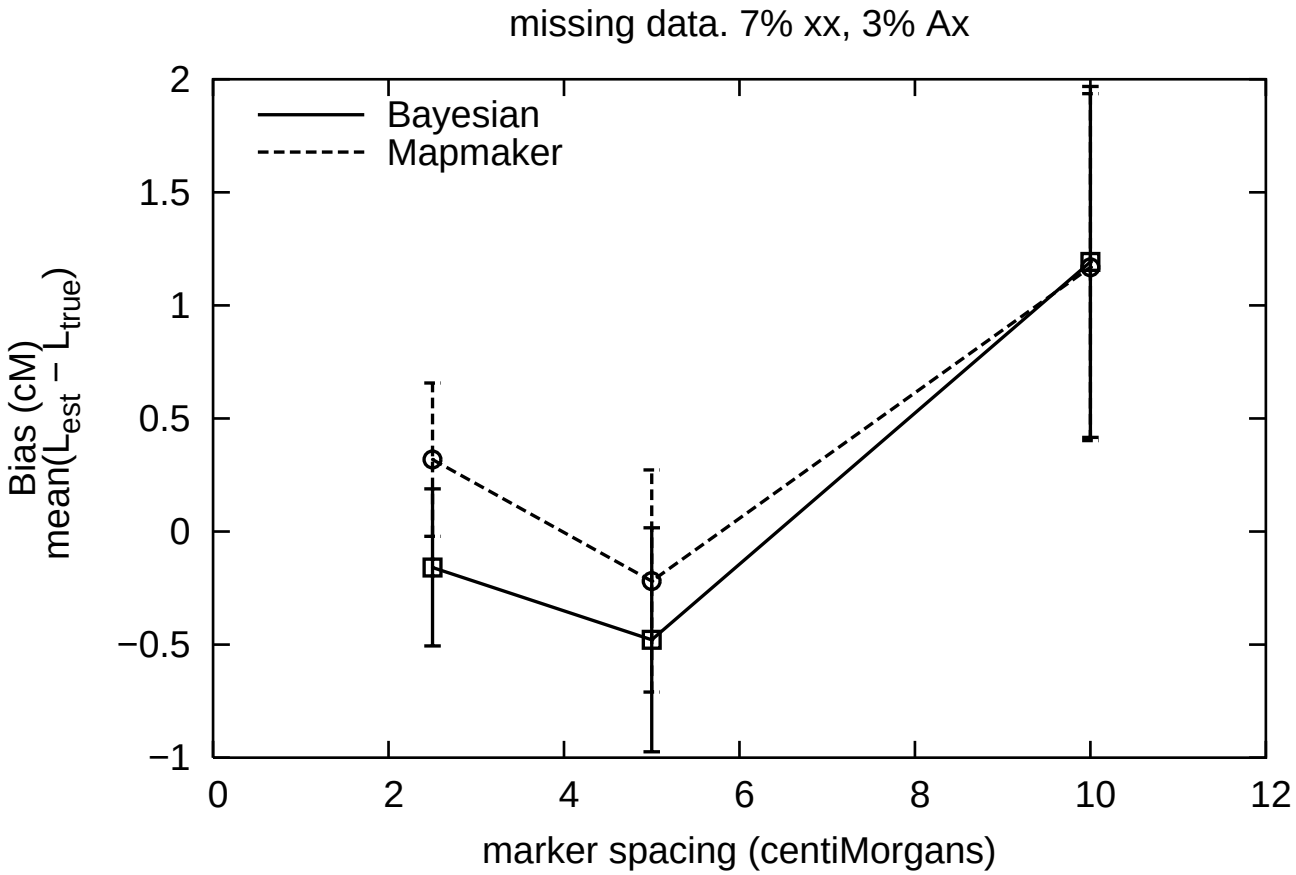

BC 100 indiv.
F2 50 indiv.
Mapmaker err prob=1%
Mapmaker err prob=0%
Mapper. No error correction
Mapper. With error correction
Bias (cM)
mean (L_est – L_true)
error rate π
0
5
10
15
20
25
30
35
0.005
0.01
0.015
0.02

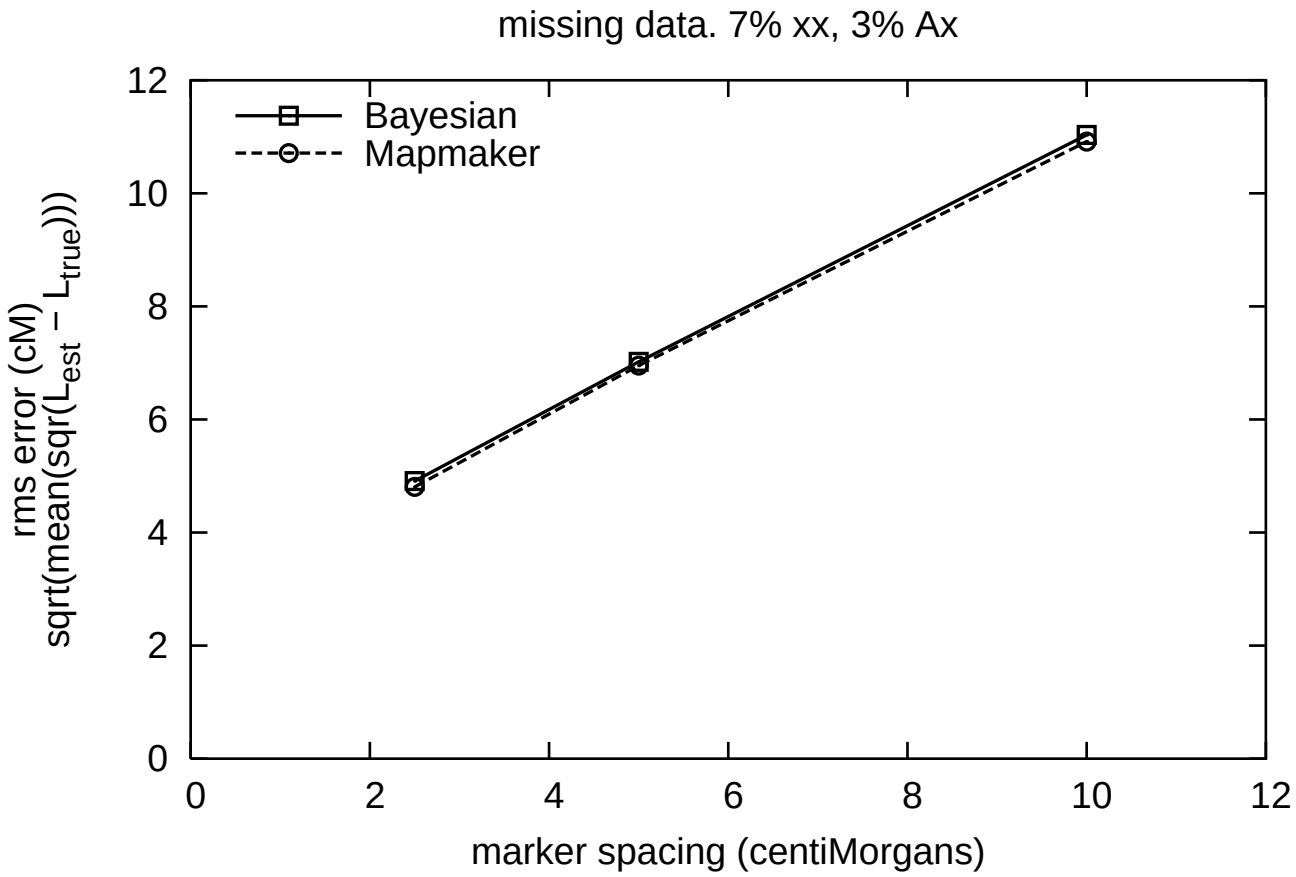

BC 100 indiv.
F2 50 indiv.
MM err prob=1%
MM err prob=0%
Mapper. No error correction
Mapper. With error correction
rms error (cM)
sqrt(mean(sqr(L_est – L_true)))
error rate π
0
5
10
15
20
25
30
35
0.005
0.01
0.015
0.02

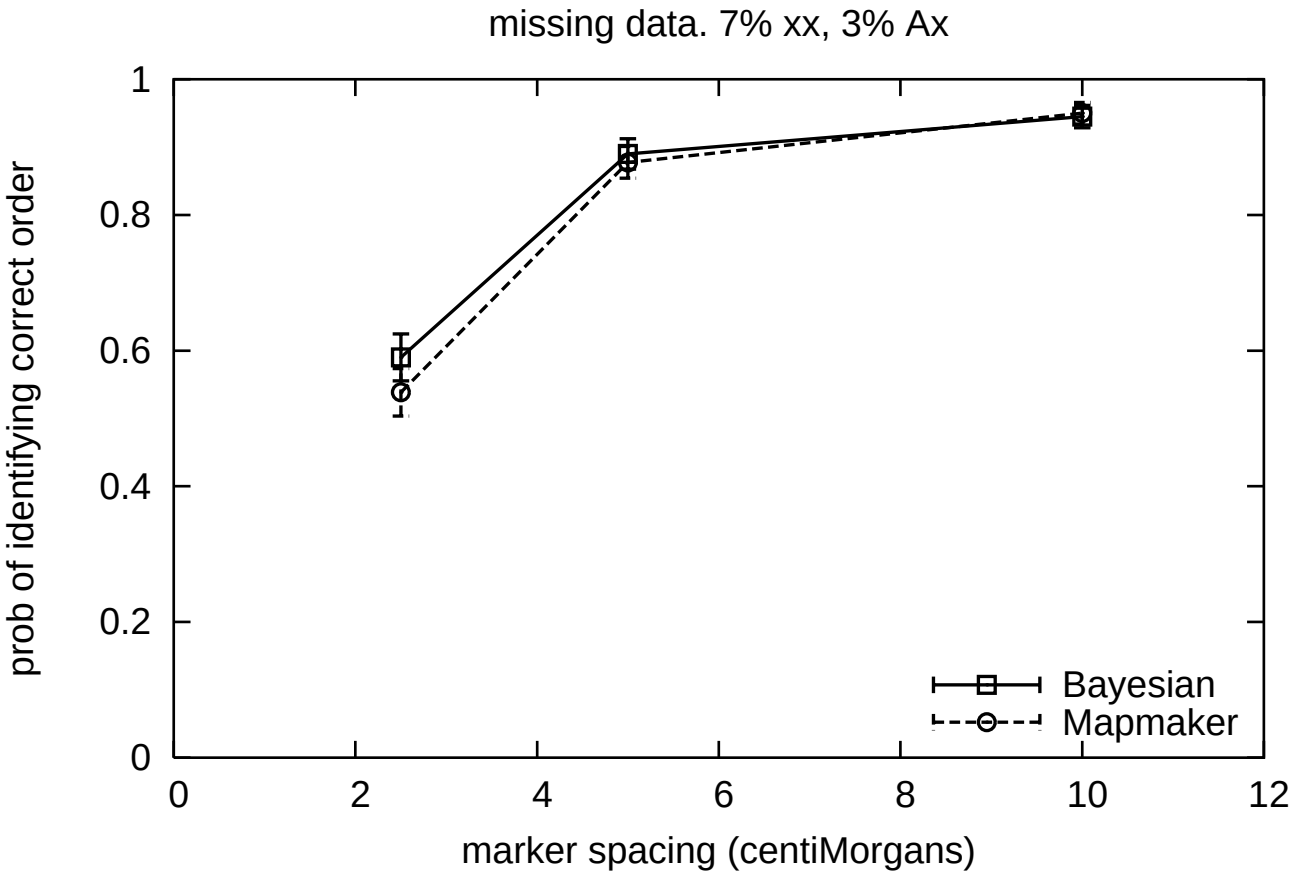

prob of identifying correct order
1
0.95
0.9
0.85
0.8
0.75
0.7
0.65
0
0.005
0.01
0.015
0.02
error rate π
F2 50 indiv.
BC 100 indiv.
F2 50 indiv. MM err. prob. = 1%
F2 50 indiv. MM err. prob. = 0%
heavy lines: with error correction
light lines: without error correction

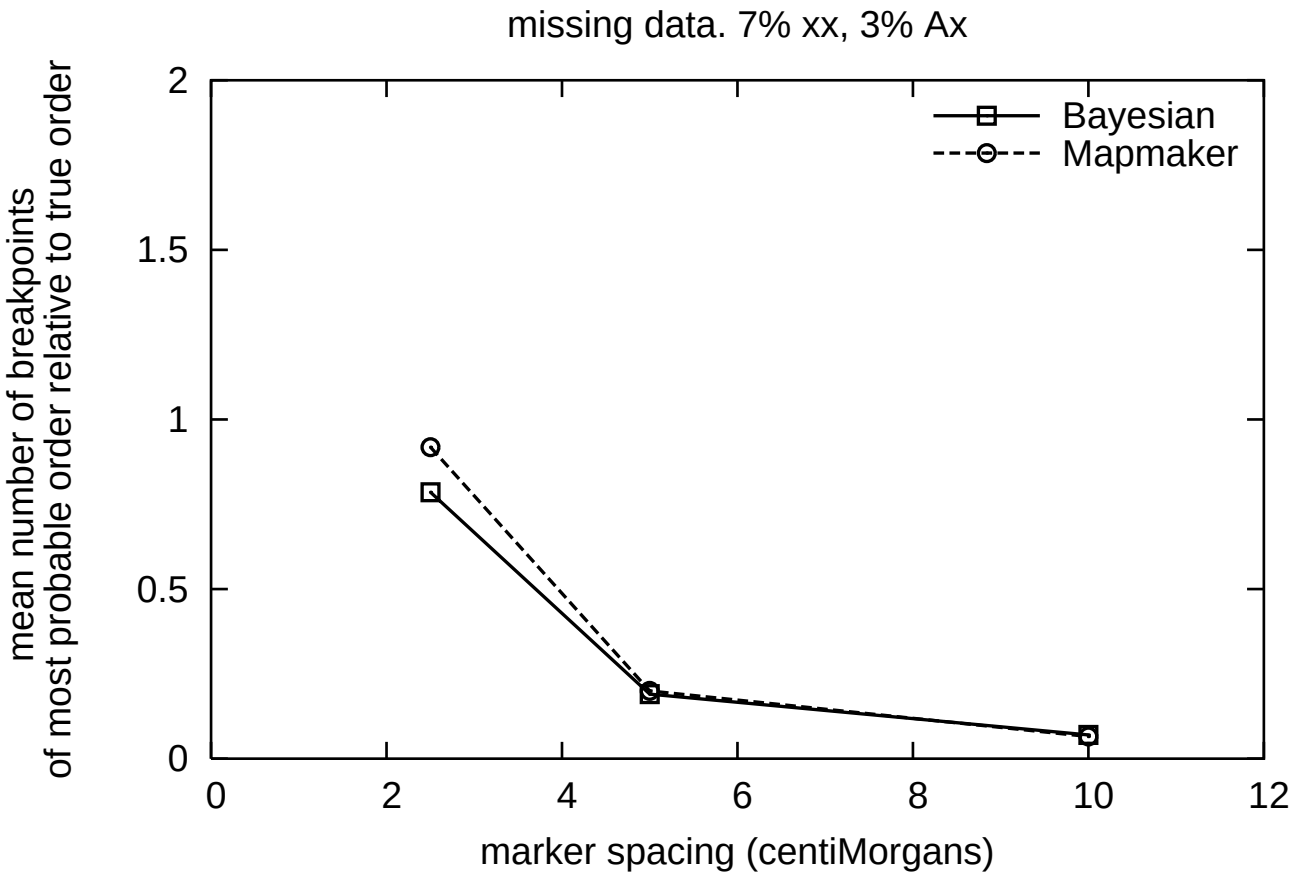

mean number of breakpoints
of most probable order relative to true order
0.5
0.4
0.3
0.2
0.1
0
0
0.005
0.01
0.015
0.02
error rate π
F2 50 indiv.
BC 100 indiv.
F2 50 indiv. MM err. prob. = 1%
F2 50 indiv. MM err. prob. = 0%
heavy lines: with error correction
light lines: without error correction


BC 100 indiv.
F2 50 indiv.
Error rate estimate
error rate π
0
0.005
0.01
0.015
0.02


missing data. 7% xx, 3% Ax
Bayesian
Mapmaker
Bias (cM)
mean(L_est – L_true)
2
1.5
1
0.5
0
−0.5
−1
0
2
4
6
8
10
12
marker spacing (centiMorgans)


missing data. 7% xx, 3% Ax
Bayesian
Mapmaker
rms error (cM)
sqrt(mean(sqr(L_est – L_true)))
12
10
8
6
4
2
0
0
2
4
6
8
10
12
marker spacing (centiMorgans)


missing data. 7% xx, 3% Ax
prob of identifying correct order
marker spacing (centiMorgans)
Bayesian
Mapmaker
0
0.2
0.4
0.6
0.8
1
0
2
4
6
8
10
12


missing data. 7% xx, 3% Ax
mean number of breakpoints
of most probable order relative to true order
marker spacing (centiMorgans)
Bayesian
Mapmaker
0
0.5
1
1.5
2
0
2
4
6
8
10
12