Genetic Epidemiology 25: 25–35 (2003)

Linkage Analysis With Sequential Imputation

Zachary Skrivanek, Shili Lin*, and Mark Irwin

Department of Statistics, Ohio State University, Columbus, Ohio

Multilocus calculations, using all available information on all pedigree members, are important for linkage analysis. Exact calculation methods in linkage analysis are limited in either the number of loci or the number of pedigree members they can handle. In this article, we propose a Monte Carlo method for linkage analysis based on sequential imputation. Unlike exact methods, sequential imputation can handle large pedigrees with a moderate number of loci in its current implementation. This Monte Carlo method is an application of importance sampling, in which we sequentially impute ordered genotypes locus by locus, and then impute inheritance vectors conditioned on these genotypes. The resulting inheritance vectors, together with the importance sampling weights, are used to derive a consistent estimator of any linkage statistic of interest. The linkage statistic can be parametric or nonparametric; we focus on nonparametric linkage statistics. We demonstrate that accurate estimates can be achieved within a reasonable computing time. A simulation study illustrates the potential gain in power using our method for multilocus linkage analysis with large pedigrees. We simulated data at six markers under three models. We analyzed them using both sequential imputation and GENEHUNTER. GENEHUNTER had to drop between 38–54% of pedigree members, whereas our method was able to use all pedigree members. The power gains of using all pedigree members were substantial under 2 of the 3 models. We implemented sequential imputation for multilocus linkage analysis in a user-friendly software package called SIMPLE. *Genet Epidemiol* 25:25–35, 2003.

Key words: IBD; Monte Carlo; NPL statistics; pedigrees; power study

Grant Sponsor: National Institute on Alcohol Abuse and Alcoholism, NIH; Grant number: U10AA08403; Grant Sponsor: NSF; Grant number; DMS-9971770; Grant Sponsor: NIH; Grant number: GM31575.
*Correspondence to: Shili Lin, Department of Statistics, Ohio State University, 404 Cockins Hall, 1958 Neil Ave., Columbus, OH 43210-1247. E-mail: shili@stat.ohio-state.edu
Received for publication 30 August 2002; Revision accepted 29 January 2003
Published online in Wiley InterScience (www.interscience.wiley.com).
DOI: 10.1002/gepi.10249

INTRODUCTION

Linkage analysis extracts inheritance information from pedigree data to evaluate the cosegregation of marker and trait alleles. Thus it is important to utilize available information on multiple markers and all pedigree members. Unfortunately, algorithms for exact analysis are computationally limited in either the number of markers or the number of pedigree members they can handle. Peeling and hidden Markov model (HMM) approaches are two such exact methods that are most frequently used.

Peeling [Elston and Stewart, 1971; Cannings et al., 1978] is a computational algorithm that recursively aggregates inheritance information from pedigree members. The algorithm scales linearly with the number of pedigree members, but exponentially with the number of loci. Genotype elimination [Lange and Goradia, 1987; O'Connell and Weeks, 1999] and set-recoding [O'Connell and Weeks, 1995] were proposed to reduce the computational requirements, so that data from more loci can be processed jointly. Despite these improvements, peeling is still limited in the number of loci that it can handle.

HMM methods model the underlying inheritance pattern as an inhomogeneous Markov chain, with each entry of the transition matrix as a function of the recombination fraction between adjacent loci [Lander and Green, 1987]. In addition to the assumptions of Hardy-Weinberg equilibrium and linkage equilibrium, the key to the algorithm is the assumption of no genetic interference. In contrast to peeling, the HMM method scales linearly with the number of loci, but exponentially with the number of pedigree members. Many improvements have been made to

reduce computational requirements, so that more pedigree members can be analyzed. Properties of the transition matrix [Kruglyak et al., 1995] and symmetries in founder phases [Kruglyak et al., 1996] were exploited to reduce the amount of calculations. Fast Fourier transformations [Kruglyak and Lander, 1998] further speed up calculations. Using observed genotypes to reduce the inheritance space [Markianos et al., 2001a] and to form equivalence classes [Markianos et al., 2001b] allows for potentially more pedigree members. Idury and Elston [1997] described a ''divide and conquer'' algorithm that speeds up some of the calculations and allows for sex-specific recombination without any computational penalty. This ''divide and conquer'' method was incorporated into the software package Merlin [Abecassis et al., 2002], which also uses an approximation method to expand the size of the pedigree it can handle in some cases. Other algorithmic improvements, such as efficient tree traversal, were made to the HMM algorithm and incorporated into Allegro [Gudbjartsson et al., 2000]. However, even with these improvements, the HMM formulation inevitably scales exponentially with the number of pedigree members.

Monte Carlo methods were proposed to overcome these computational limitations. Two major approaches of Monte Carlo methods to linkage analysis are Markov chain Monte Carlo (MCMC) and sequential imputation. MCMC algorithms can be designed such that they scale linearly in both the number of loci and the number of pedigree members [Thompson, 2000]. Thus, MCMC is an extremely powerful estimation method that can practically deal with any number of loci and pedigree of arbitrary size and complexity [Luo et al., 2001]. However, due to strong dependencies among realizations of the Markov chain, convergence can be slow [Thompson, 2000].

Sequential imputation is another Monte Carlo method that has been successfully applied to a variety of areas [Bergman, 2001; Blake et al., 2001]. Irwin et al. [1994] illustrated how to use sequential imputation in linkage analysis to calculate the likelihood (and hence LOD scores), utilizing the peeling algorithm for a single locus, which results in an algorithm that also scales linearly in both the number of loci and the number of pedigree members in terms of computational operations. For pedigrees that are not very complex (e.g., single-locus peelable), sequential imputation can be more efficient computationally than MCMC methods in many circumstances. However, it should be noted that sequential imputation is not meant to be a replacement for MCMC, as it cannot handle very complex pedigrees, such as the 1,544-member Hutterite pedigree successfully dealt with using MCMC methods [Luo et al., 2001]. Furthermore, as it is currently implemented, the memory requirement can be prohibitive for a large number of markers, but this restriction can be lifted, as discussed later.

This article extends the method of sequential imputation to nonparametric linkage analysis. This is an important step forward in making sequential imputation a viable alternative for linkage analysis, as nonparametric linkage analysis is frequently more suited for analyzing complex traits whose underlying genetic model is unknown or unclear.

The idea is to simulate inheritance vectors conditioned on phase-known multilocus genotypes that were imputed sequentially. Then the inheritance vectors can be used to estimate any linkage statistic of the form [Whittemore and Halpern, 1994b; Kruglyak et al., 1996]

$$E[\mathrm{S}(\phi, x_d)|\mathbf{x}_m] = \sum_{\phi} \mathrm{S}(\phi, x_d) P(\phi|\mathbf{x}_m), \qquad (1)$$

where $\mathbf{x}_m$ is the observed marker data, x_d is the observed disease phenotypes, and ϕ is the unobserved inheritance vector. The inheritance vector [Lander and Green, 1987], $\phi=(p_1, m_1, \ldots, p_n, m_n)$, is a binary representation of the inheritance information at a location in the genome for each of the n nonfounders. The i^{th} nonfounder is assigned 2 bits, p_i and m_i, corresponding to the genetic information inherited from the father and mother. Each bit is either 1 or 0, depending on whether the allele was inherited from the grandmother or grandfather, respectively. The inheritance distribution, $\mathrm{P}(\phi \mid \mathbf{x}_m)$, is the distribution of the inheritance vectors conditioned on the observed marker data.

The scoring function, $\mathrm{S}(\phi, x_d)$, for inheritance vector ϕ and observed disease phenotypes x_d, measures the amount of identical by descent (IBD) sharing. An example of a scoring function for sib pairs is to assign a score of $\frac{1}{2}$, $\frac{1}{4}$, or 0 to a sib pair that shares 2, 1, or 0 alleles IBD, respectively. Suppose two sibs have the following inheritance vector, (1,0,1,0), which implies that they both inherited the grandmaternal allele from their father and the grandpaternal allele from their mother. Therefore they share two alleles IBD, and would get a score of $\frac{1}{2}$ with this scoring function.

The class of linkage statistics represented in (1) encompasses a wide range of nonparametric IBD statistics, including S_{pairs} and S_{all} [Whittemore and Halpern, 1994a], the most popular allele-sharing statistics for nonparametric analysis. We note that, if we add genetic parameters for the disease model to the score function, the statistic in the form (1) becomes a parametric statistic. In fact, the familiar LOD score is included in this class [Kruglyak et al., 1996].

METHODS

Our approach is to estimate the linkage statistic in formula (1) instead of calculating it exactly. We decompose the information that we have on the m markers into $\mathbf{x}_m$ $\{x_1,\ldots,x_m\}$. We denote the unobserved ordered genotypes (genotypes with parental source information) at the m markers, $\{y_1,\ldots,y_m\}$, as $\mathbf{y}$. After obtaining the starting point of the sequential imputation (step 1), we sequentially sample the ordered genotypes and calculate the appropriate importance sampling weight (steps 2 and 3). Then we sample the inheritance vector ϕ at a particular location, given the sampled ordered genotypes at the m markers (step 4). Finally, we calculate the score using ϕ (step 5). These steps are summarized as follows:

Step 1. Calculate $\mathrm{P}(x_1)$ and sample y_1 from $\mathrm{P}(y_1 \mid x_1)$.

Step 2. For t=2,…, m, carry out the following steps:

(a) Calculate $\mathrm{P}(x_t \mid x_1, y_1,\ldots, x_{t-1}, y_{t-1})$.

(b) Sample y_t from $\mathrm{P}(y_t \mid x_1, y_1,\ldots, x_{t-1}, y_{t-1}, x_t)$.

Step 3. Form $w(\mathbf{y}) = \mathrm{P}(x_1)\,\Pi_{t=2}^{m}\ P(x_t|x_1, y_1, \ldots, x_t{-}1, y_t{-}1)$.

Step 4. Sample ϕ at a location of interest according to $\mathrm{P}(\phi \mid \mathbf{y})$, where $\mathbf{y}$ are the ordered genotypes sampled in steps 1–3. Note that $\mathrm{P}(\phi \mid \mathbf{y})$=$\mathrm{P}(\phi \mid \mathbf{y}, \mathbf{x}_m)$.

Step 5. Calculate the score $\mathrm{S}(\phi, x_d)$.

Steps 1–5 are carried out N times to form $w(\mathbf{y}_1),\ldots, w(\mathbf{y}_\mathrm{N})$ and $\mathrm{S}(\phi_1, x_d),\ldots, \mathrm{S}(\phi_\mathrm{N}, x_d)$. The probability calculations and the sampling in steps 1–3 are done by means of single-locus peeling, and sampling using reverse peeling [Ploughman and Boehnke, 1989; Ott, 1989].

The sampling of the inheritance vector in step 4 involves a series of Bernoulli trials. Since each of the bits that make up the inheritance vector is conditionally independent of each other, given the ordered genotypes, $\mathbf{y}$, we can sample these bits separately.

Irwin et al. [1994] showed that the sampling distribution of the ordered genotypes, $\mathrm{P}^*(\mathbf{y} \mid \mathbf{x}_m)$, satisfies:

$$\mathrm{P}^*(\mathbf{y}|\mathbf{x}_m) = \frac{\mathrm{P}(\mathbf{y}|\mathbf{x}_m)\mathrm{P}(\mathbf{x}_m)}{w(\mathbf{y})}.$$

From this equality it follows:

$$\begin{aligned} &E_{\phi,\mathbf{y}}[\mathrm{S}(\phi, x_d)w(\mathbf{y})|\mathbf{x}_m] \\ &= \sum_{\phi} \mathrm{S}(\phi, x_d) \sum_{\mathbf{y}} \mathrm{P}(\phi|\mathbf{y})\mathrm{P}^*(\mathbf{y}|\mathbf{x}_m)w(\mathbf{y}) \\ &= \mathrm{P}(\mathbf{x}_m) \sum_{\phi} \mathrm{S}(\phi,\ x_d) \sum_{\mathbf{y}} \mathrm{P}(\phi|\mathbf{y})\mathrm{P}(\mathbf{y}|\mathbf{x}_m) \\ &= \mathrm{P}(\mathbf{x}_m)\mathrm{E}_{\phi}[\mathrm{S}(\phi, x_d)|\mathbf{x}_m]. \end{aligned}$$

This result, and the fact that the average of the weights is an unbiased estimator of $\mathrm{P}(\mathbf{x}_m)$ [Irwin et al., 1994], gives a consistent estimator for the linkage statistic in (1):

$$\widehat{\mathrm{E}}[S(\phi, x_d)|\mathbf{x}_m] = \sum_{j=1}^{\mathrm{N}} \mathrm{S}(\phi_j,\ x_d)\,\frac{w(\mathbf{y}_j)}{w(+)},$$

where $w(+)$=$\sum_{j=1}^{\mathrm{N}} w(\mathbf{y}_j)$. Therefore, the estimate is a weighted average of the scores.

In step 5, to calculate the score $\mathrm{S}(\phi, x_d)$, we first assign each of the founders two unique allele labels. We pass these founder allele labels down the pedigree, using the sampled inheritance vector. We then measure the number of founder allele labels in common among the affecteds via the IBD scoring function.

THE NULL DISTRIBUTION

The IBD statistic measures the amount of IBD sharing. If the amount of sharing among the affecteds is significantly more than what would be expected under the null hypothesis of no linkage, then there is evidence of linkage. Therefore, it is necessary to measure the mean and variance of the scores under the null hypothesis of no linkage. To estimate the null mean and variance, we simply pass the founder allele labels through the pedigree with 50% probability that a particular allele label will be passed on to an offspring, and calculate the score. We repeat this process many times to get a sample of scores from the null distribution. The mean and variance of this sample give unbiased estimates of the null mean and variance of the scoring function. Although one could do this exactly [Kruglyak et al., 1996], we find this to be inefficient for the

size of pedigrees we are considering. Therefore, we implemented the above Monte Carlo version of the GENEHUNTER procedure [Kruglyak et al., 1996]. We then standardize $\widehat{\mathrm{E}}[\mathrm{S}(\phi, x_d)|\mathbf{x}_m]$ by the estimated null mean and null standard deviation to form the standardized statistic. Furthermore, the sampled scores under the null distribution are used to estimate the exact *P*-value. We note that this leads to conservative estimates of the standardized statistic and *P*-value, as pointed out by Kruglyak et al. [1996].

TO REWEIGHT OR NOT?

In the methods described above, we sampled the inheritance vectors (step 4) at every location of interest (usually along the entire chromosome in which the markers reside), and then estimated the statistic using the sampled inheritance vectors. Alternatively, we could sample inheritance vectors at only a few locations of the chromosome and estimate the linkage statistics at neighboring locations by reweighting, another importance sampling idea exploited by Irwin et al. [1994]. For instance, suppose that inheritance vectors were sampled at position d_0. We can estimate the statistic at a nearby location, say d_1, by

$$\sum_{j=1}^{\mathrm{N}} \mathrm{S}(\phi_j, x_d) \frac{\mathrm{P}_{d_1}(\phi_j|\mathbf{y}_j)}{\mathrm{P}_{d_0}(\phi_j|\mathbf{y}_j)} \frac{w(\mathbf{y}_j)}{w(+)}. \qquad (2)$$

This reweighted statistic is a consistent estimator of the linkage statistic at d_1.

We found that reweighting does not perform well in estimating IBD statistics, however. This is most likely due to the fact that the distribution of the inheritance vectors under d_1 is too far away from the distribution under d_0, resulting in large variability in (2). The computational savings in doing reweighting instead of sampling at a particular location for estimating the likelihood, as proposed by Irwin et al. [1994], can be substantial, since the alternative would involve peeling. On the other hand, there was no such clear advantage in using reweighting in this application of sequential imputation, as sampling inheritance vectors does not pose much computational burden at all. Hence, reweighting is not adopted here.

THE SOFTWARE PACKAGE

We implemented sequential imputation for linkage analysis in a software package called SIMPLE (Sequential Imputation for MultiPoint Linkage Estimation). The nonparametric IBD statistics currently available in SIMPLE include the score functions S_{all} and S_{pairs} [Whittemore and Halpern, 1994a; Kruglyak et al., 1996]. Furthermore, SIMPLE can calculate LOD scores. SIMPLE takes input files with the same format as those used in GENEHUNTER, enabling the user to easily switch to SIMPLE if the pedigree is too large to be handled by GENEHUNTER in its entirety. The software is freely available from our web site.

COMPUTATIONAL REQUIREMENTS

Producing the weights and ordered genotypes (steps 1–3) takes the majority of the computing time. To complete a single iteration, we need to peel each marker locus and then do reverse peeling [Ploughman and Boehnke, 1989; Ott, 1989] to sample the ordered genotypes. Thus, the complexity and memory requirements are the same as those required to do m single-locus peels. The key difference in computational cost between this algorithm and a standard peeling algorithm for linkage analysis, such as that implemented in LINKAGE [Lathrop et al., 1984], is that we are only doing a single-locus peel at a time, so the calculations are linear in the number of markers. Efficiencies in peeling algorithms can be applied to the peeling step here to improve the overall efficiency. Currently, some genotype elimination has been implemented in SIMPLE to achieve such efficiencies. As in peeling, this stage is sensitive to missing data.

In step 4 of the algorithm, we sample the inheritance vector at a location of interest, conditioned on the ordered genotypes sampled. For one iteration, this involves simulating the two inheritance bits for each of the nonfounders, resulting in the calculations being linear in the number of pedigree members. The computational time required for calculating the score (step 5 of the algorithm) depends on its complexity. In particular, the current algorithm for calculating S_{all} is computationally limited in the number of affecteds it can handle; see Markianos et al. [2001a] for a detailed discussion. Missing data have no effect on either of these last two steps, since they are conditioned on complete ordered genotypes.

The memory requirement is most influenced by the number of loci analyzed. This is due to storing the joint inheritance vector probabilities across all loci to speed up calculations, leading to the storage being exponential in the number of loci analyzed. In steps 1–3, we store the inheritance

vector probabilities only for the markers, whereas in steps 4 and 5, we store them for the markers plus a location of interest. These probabilities are stored for all locations where the statistics are to be estimated.

We now present a summary of results for time and memory requirements for analyzing a small, medium, and large pedigree, respectively. We chose the first three pedigrees (pedigrees 1, 2, and 3) that were presented in a simulated data set from Genetics Analysis Workshop (GAW) 12. The small, medium, and large pedigrees have 52, 86, and 100 members, respectively. They have 15, 17, and 34 members with missing data. Eight markers, with 6–8 alleles each and an average heterozygosity of 0.77, were analyzed. We ran SIMPLE for 1,000 iterations and estimated S_{pairs}. GENEHUNTER was not capable of analyzing any of these pedigrees without seriously reducing the number of pedigree members. GENEHUNTER would have had to drop 24 (46%), 50 (58%), and 58 (58%) members in the small, medium, and large pedigrees, respectively, to be able to analyze them. We used version 2.1.3 of GENEHUNTER here and throughout this paper.

We conducted the study on a Sun Blade 100 with an Ultrasparc IIe 500-mHz processor. This study can be used as a rough guideline to the time and memory requirements for using SIMPLE. The results are shown in Table I. In Table I, we show the time and memory requirements to process all eight markers for 1,000 iterations in steps 1–3. Since the number of points where linkage statistics are estimated depends on the user, we report the time and memory requirements per point in steps 4 and 5. Because the computational time grows linearly with the number of iterations, an estimate of the time for analyzing these pedigrees with 2,000 iterations would be approximately twice the reported times, for example. On the other hand, the memory is not affected by the number of iterations. For steps 4 and 5, the computational time and memory grow linearly with the number of points to be analyzed. For example, to estimate the time and memory to analyze these eight markers with 5 points between each pair of adjacent markers (43 points in total), multiply the reported time and memory by 43.

TABLE I. Time and Memory Requirements for 1,000 Iterations[a]

	Steps 1–3		Steps 4 and 5	
Pedigree size	Time (hr:min)	Memory (MB)	Time (sec)	Memory (MB)
Small	1:37	4.3	0.57	0.42
Medium	1:41	4.1	1.33	0.42
Large	3:47	7.5	1.61	0.42

[a]We report time and memory requirements to complete 1,000 iterations of steps 1–3 and steps 4 and 5 of the algorithm (including the calculation of the estimate) for eight markers in each of three pedigrees of sizes small (52 members), medium (86 members), and large (100 members). Results are reported per disease location for steps 4 and 5. Note that time units are different for steps 1–3 and steps 4 and 5. MB, megabytes.

The time and memory requirements to produce the weights and ordered genotypes (steps 1–3) for the small and medium pedigrees were similar. Though the medium pedigree was substantially larger than the small pedigree, they both had a comparable amount of missing data. This would explain why they took similar amounts of time and memory to be analyzed. On the other hand, the large pedigree had twice as much missing data, and therefore took more than twice as long and almost twice as much memory as the other two pedigrees analyzed. The memory requirements to sample the inheritance vectors (step 4), calculate the scores (step 5), and form the weighted estimates were the same for all three pedigrees. This is expected, since the number of loci (eight markers and 1 point) analyzed was the same for all three pedigrees. On the other hand, the time increased as the size of the pedigree increased, since the number of inheritance vectors to be sampled increased accordingly.

ACCURACY OF ESTIMATES

We did a number of validation studies of SIMPLE using GENEHUNTER to verify that the scores were estimated accurately within reasonable computing time. The scores were always close to the scores produced by GENEHUNTER. Of course, the accuracy is a function of the number of iterations. To get a rough estimate of the necessary sample size to reach a certain desired accuracy, one may run SIMPLE for a small number of iterations (say, 100) to estimate the sampling variability (automatically calculated in SIMPLE), which we note is unlikely to be very accurate. From this estimate, one can estimate the necessary number of iterations.

To illustrate the accuracy of SIMPLE, we analyzed pedigree 76 of the Collaborative Studies on the Genetics of Alcoholism (COGA) data set from GAW 11. We removed three members so that GENEHUNTER could analyze it. The pedigree is

shown in Figure 1. Note that it has a marriage loop. There are 14 members in the (reduced) pedigree, with four founders. Eight markers are used from chromosome one: D1S1613, D1S550, D1S532, D1S1588, D1S1631, D1S1675, D1S534, and D1S1595. They have 9–12 alleles, with an average heterozygosity of 0.75. The markers are spaced 11.2, 8.4, 18.1, 12.5, 11.9, 9.0, and 9.8 cM apart. Two founders (14%) are missing all of their marker data. In addition, 7 other members (50%) are missing data for D1S1631, 2 members (14%) are missing data for D1S534, and 3 members (21%) are missing data for other markers.

The linkage statistics S_{pairs} and S_{all} were estimated at five locations between each adjacent pair of markers, using both GENEHUNTER and SIMPLE with 5,000 iterations. As can be seen from the plots in Figure 2, the estimated standardized scores produced by SIMPLE were close to the exact scores produced by GENEHUNTER. For the S_{pairs} estimates, the median error was 0.021, while the maximum error was 0.112, which we note is almost double the second largest error. For

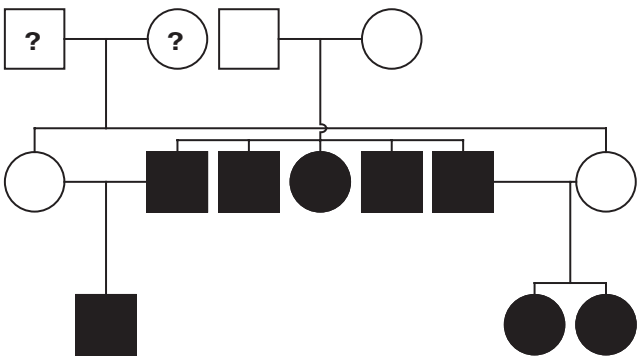

Fig. 1. Pedigree used in validation study. Individuals with question mark have no marker data or information on disease phenotypes.

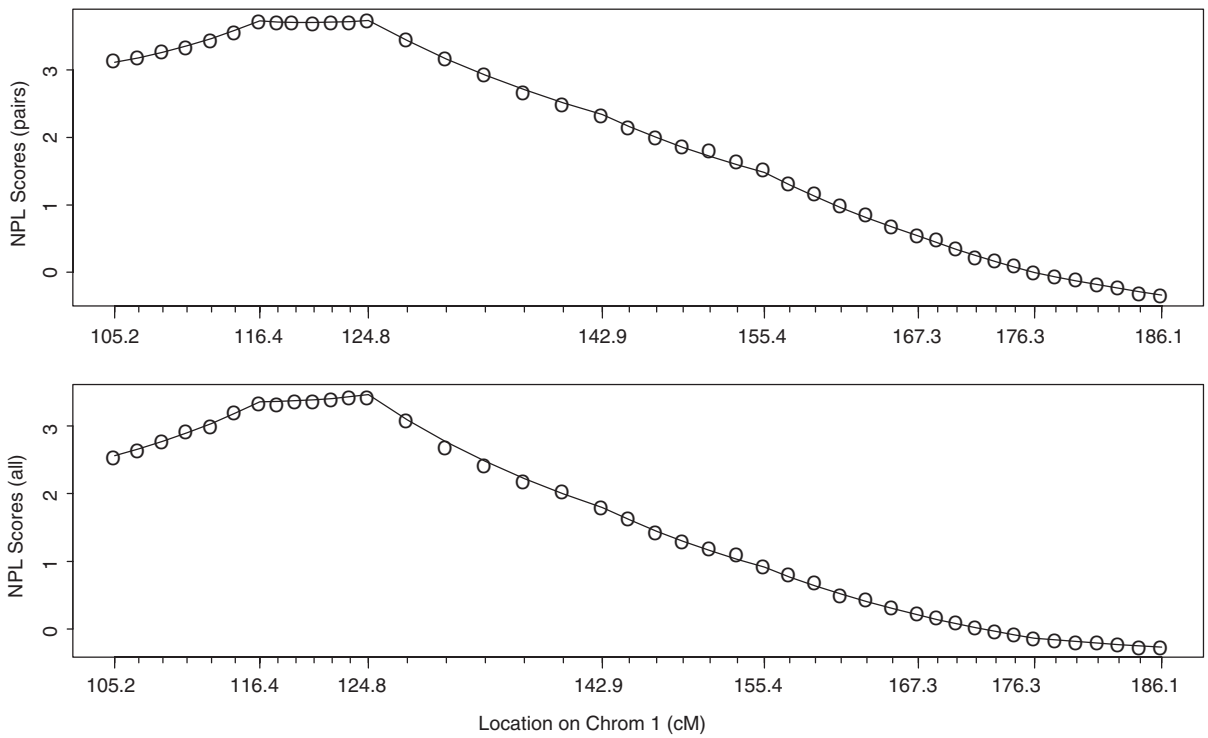

Fig. 2. Standardized scores from validation study. Scores produced by GENEHUNTER are given by line, and scores produced by SIMPLE are plotted with circles. S_{pairs} are plotted at top, and S_{all} are plotted at bottom. Markers are indicated by extended tick marks, and locations (in cM) are indicated on x-axis at bottom. Chrom, Chromosome.

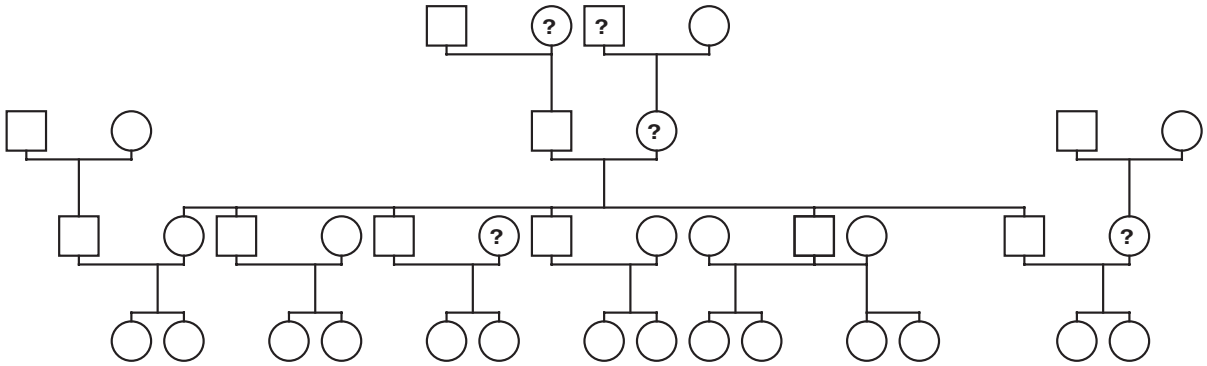

Fig. 3. Pedigree structure for power study. Individuals with question mark have no marker or disease data.

the S_{all} estimates, the median and maximum errors were 0.018 and 0.090, respectively, slightly smaller than their S_{pairs} counterparts.

POWER STUDY

To illustrate the potential benefit to multipoint linkage analysis by processing all pedigree members of a large pedigree, we performed a simulation study. We used the S_{pairs} statistic to analyze the full pedigree shown in Figure 3 with SIMPLE and then with GENEHUNTER, which needed to discard some members of the pedigree. The pedigree had 37 members, 11 of whom were founders, and 5 members had missing marker and disease data. The ascertainment criterion was that at least one sib in each of the seven sibships in the last generation had to be affected.

We used six markers with equally frequent alleles for each marker. The markers were spaced 15 cM apart. We simulated the marker and disease data under three disease models. In all three cases, disease data were simulated at a locus in the middle of the marker map at 37.5 cM. In model I, the penetrances for genotypes aa, Aa, and AA were 0, 0.9, and 0.95, with a disease allele frequency of P(A)=0.1. In model II, the penetrances were 0.05, 0.4, and 0.6, with a disease allele frequency of 0.05. In model III, the penetrances were 0.05, 0.5, and 0.7, with a disease allele frequency of 0.3.

Five hundred pedigrees were simulated under all three models. GENEHUNTER had to drop between 14 (38%) to 20 (54%) members in order to process the pedigrees, among them, the number of affecteds ranging from 0–13, with an average of 5.3. To estimate power, we calculated the proportion of pedigrees that had maximum scores exceeding given thresholds. Four thresholds levels were used: 2.33, 3.09, 3.72, and 4.27, as suggested by Kruglyak et al. [1996]. These thresholds correspond to asymptotic significance levels of 0.01, 0.001, 0.0001, and 0.00001, respectively. The results are summarized in Table II.

TABLE II. Power Estimates for a Single Pedigree[a]

	Model I		Model II		Model III	
Level	SIMPLE	GH	SIMPLE	GH	SIMPLE	GH
0.01	44	40	38	26	21	19
0.001	26	24	23	12	11	7
0.0001	15	10	15	5	5	3
0.00001	8	3	10	3	2	1

[a]Power was defined as percentage of pedigree with scores exceeding given threshold. Thresholds used for asymptotic significance levels of 0.01, 0.001, 0.0001, and 0.00001 were 2.33, 3.09, 3.72, and 4.27, respectively. GH, GENEHUNTER.

From the initially simulated pedigrees, we resampled, with replacement, 500 data sets of size k, with k ranging from 2–50 pedigrees for each of the three models. We estimated power by the proportion of data sets with standardized scores that exceeded the threshold values. The results for the three models, using threshold 3.09, are shown in Figure 4. We plotted the proportions for both SIMPLE and GENEHUNTER as points, and included a curve that was calculated by a spline smoother [Hastie and Tibshirani, 1990]. We see that under all three models, SIMPLE yields higher power than GENEHUNTER.

For models I and II, we calculated the minimal sample sizes needed, based on the spline smooth curve (only one of the curves is shown in Figure 4; the remaining plots are available from our web site), to reach 50%, 65%, and 80% power for each of the threshold levels 2.33, 3.09, 3.72, and 4.27. The results are summarized in Table III. Since the power was much weaker for model III, we

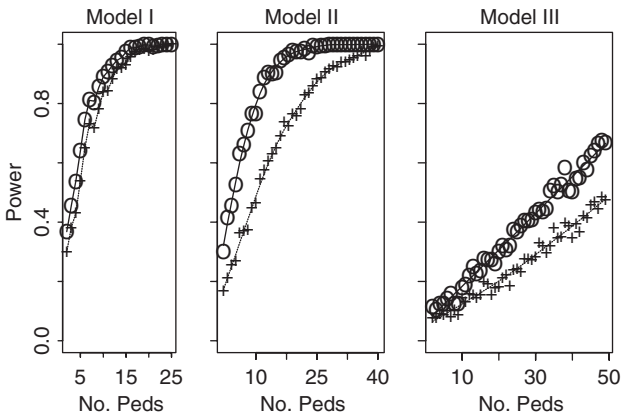

Fig. 4. Power curves for SIMPLE (solid line and ○) and GENEHUNTER (dashed line and +), based on a threshold of 3.09 for all three genetic models. Peds, Pedigrees.

TABLE III. Sample Size Estimates for Models I and II[a]

		Model I		Model II	
Power	Level	SIMPLE	GH	SIMPLE	GH
50%	0.01	2	2	2	5
50%	0.001	4	5	5	11
50%	0.0001	6	7	8	17
50%	0.00001	8	10	11	24
65%	0.01	3	4	4	8
65%	0.001	6	6	7	15
65%	0.0001	8	10	11	23
65%	0.00001	11	13	13	29
80%	0.01	5	6	6	12
80%	0.001	7	10	11	21
80%	0.0001	11	14	14	29
80%	0.00001	14	17	18	37

[a]For nominal significance levels of 0.01, 0.001, 0.0001, and 0.00001, we report minimal sample size necessary (based on a spline fit) to achieve 50%, 65%, and 80% power. Corresponding thresholds are 2.33, 3.09, 3.72, and 4.27, respectively.

reported the results for powers 40%, 50%, and 65% at thresholds 2.33 and 3.09 for this model. The results are summarized in Table IV. For model I, SIMPLE performed slightly better than GENEHUNTER. However, for model II, SIMPLE only requires approximately half as many pedigrees as GENEHUNTER for the powers considered. In model III, GENEHUNTER needs approximately 50% more pedigrees than SIMPLE to achieve the same power. In all three models, the reduction in number of pedigrees necessary to achieve the given power using SIMPLE grows as the desired power increases and as the threshold becomes more stringent.

TABLE IV. Sample Size Estimates for Model III[a]

Power	Level	SIMPLE	GH
40%	0.01	11	18
40%	0.001	28	42
50%	0.01	17	26
50%	0.001	36	*
65%	0.01	26	36
65%	0.001	48	*

[a]For nominal significance levels of 0.01 and 0.001, we report minimal sample size necessary (based on a spline fit) to achieve 40%, 50%, and 65% power. Corresponding thresholds are 2.33 and 3.09, respectively. Asterisks indicate that required sample size is greater than 50.

TABLE V. Type I Error Rates[a]

	Empirical	
Nominal level	SIMPLE	GH
0.01	0.008	0.005
0.001	0.0005	0.003
0.0001	0.0	0.002
0.00001	0.0	0.0

[a]For nominal levels of 0.01, 0.001, 0.0001, and 0.00001, we report estimated type I error rates for a sample of 15 pedigrees. Corresponding thresholds are 2.33, 3.09, 3.72, and 4.27, respectively.

TYPE I ERROR

We studied the type I error rates for a data set of 15 pedigrees, which was chosen to reflect a realistic situation. To estimate type I error, we simulated marker genotypes for 10,000 pedigrees, using the same pedigree structure and missing data pattern used in the power study (Fig. 3), fixing the last generation as all affected. From these 10,000 simulated pedigrees, we resampled 2,000 data sets of size 15 pedigrees with replacement. We then calculated the proportion of data sets with standardized scores exceeding each of four thresholds, to estimate the type I error rates. The results for both SIMPLE and GENEHUNTER are shown in Table V. GENEHUNTER dropped 17 (46%) members in each of the pedigrees simulated. The estimated type I error rates were close to the nominal significance levels.

A LARGER EXAMPLE

The purpose of this example is twofold. First, we would like to demonstrate more fully the capability of SIMPLE with a larger pedigree and a larger number of markers. Second, we would also like to study the potential effect of dense marker maps on the standard errors of the estimates. To

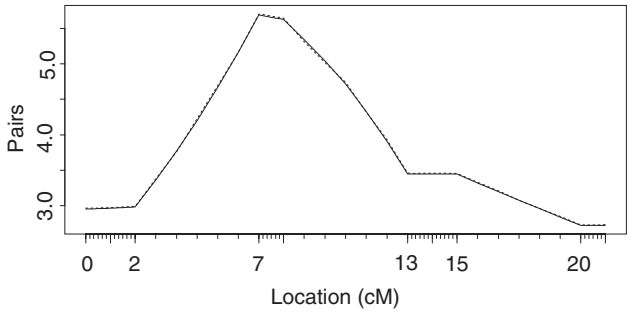

Fig. 5. Standardized scores from larger example. Scores produced by 5K run are given by solid line, and scores produced by 50K run are given by dashed line. Markers are indicated by extended tick marks, and locations (in cM) are indicated on x-axis.

fulfill this purpose, we selected the largest GAW 12 pedigree (100 members) in our computational requirements study. Because the marker map provided by GAW 12 was coarser than what we desired, we simulated our own data at 10 markers, with intermarker distances of 1, 1, 5, 1, 5, 1, 1, 5 and 1 cM. The genetic model was taken to be model I as in the power study, and the disease locus was assumed to be right in the middle between markers 5 and 6. Data on 34 of the 100 individuals were assumed to be missing, consistent with those privided by GAW 12.

Results for S_{pairs} with run lengths of 5,000 (5K) and 50,000 (50K) iterations are plotted in Figure 5. The agreement between the two curves is excellent, with a maximum difference of 0.028, which is less than 1% of either of the estimated scores. The standard errors of the estimates are all very small, with maximums of 0.100 and 0.098 for the 5K and 50K runs, respectively. Although the standard errors for many of the estimates for the 50K run are smaller, the differences are usually very small. These results indicate that SIMPLE can indeed effectively handle a moderate number of markers and large pedigrees with a significant amount of missing data. Furthermore, the method of sequential imputation does not seem to be overly affected by closely spaced markers, as the standard errors for the 50K run are not significantly smaller than those for the 5K run. We caution, though, that these observations should be viewed as tentative, as they are based on only a single example.

DISCUSSION

Linkage analysis is an important tool in localizing disease loci. When analyzing complex traits in humans, it is desirable to process many loci and use all informative members in a given pedigree. We present sequential impuation, a Monte Carlo method, to perform nonparametric multipoint linkage analysis for large pedigrees. This method can handle either more loci or larger pedigrees than the conventional exact calculation methods: peeling and HMM.

One advantage of this method over the HMM is that it can process larger pedigrees, which can lead to an increase in power. We demonstrated the potential gain in power in our simulation study using S_{pairs} and three genetic models, although the magnitude of power gains varied from model to model. Substantial power gains are observed under models II and III, while the gains under model I are minimal. The different levels of power gains in the three models are due to differences in the amount of IBD information carried by the affected individuals dropped.

We would expect the gains in power to be even greater with S_{all} due to the nature of the statistic. Unlike S_{pairs}, S_{all} gives increasing scores to the larger number of affected pedigree members sharing an allele IBD. Since GENEHUNTER often discards affected members, we would expect this to adversely affect the power to a greater degree with S_{all} than with S_{pairs}. One drawback of using S_{all}, however, is the computational intensity of its current implementation. Markianos et al. [2001b] addressed this issue and proposed a method to reduce the computational burden.

Another advantage of sequential imputation over the HMM method is that it can incorporate genetic interference in its calculations. Currently, in addition to Haldane's no interference model, SIMPLE can calculate linkage statistics using the chi-square model [Foss et al., 1993; Zhao et al., 1995], a recombination model that is suitable for modeling crossover interference in humans [Lin and Speed, 1996].

Because sequential imputation and MCMC are currently the two major approaches of Monte Carlo approximation methods in linkage analysis, it is important for these methods to be compared with each other to assess their relative merits in terms of their accuracy and efficiency for different types of problems. However, MCMC methods have not been compared among themselves to assess their relative performances, nor has the reliability of the programs implementing these methods been compared thoroughly except for a recent small-scale comparative study of IBD probabilities produced by two popular MCMC software packages (Dr. Ellen Wijsman, personal communication). Due to the lack of knowledge on the reliability of these programs at the moment, it is premature to carry out comparisons between sequential imputation and MCMC using available software. Furthermore, in order to carry out such a comparison in a fair and complete manner, a great many issues need to be considered before the design and execution of such a study can take place, which is certainly not within the scope of this paper. Among many others, selection of measures for comparison is a key issue that one needs to address in order to compare the performances of two simulation-based methods. We plan to study these issues and carry out a comparative study in a future contribution.

In the algorithm described in this paper, we decomposed the data into the information that we had at the m loci, and sequentially imputed the ordered-genotypes locus by locus. We note that other decompositions are possible. For instance, one could decompose the data into sets of loci. This would involve a multilocus peel per iteration, which obviously increases the computational cost. The advantage is that it should decrease the Monte Carlo variability and hence require less iterations to reach the same accuracy. Furthermore, the order of the sequential imputation does not have to be the physical order of the loci. In fact, the simulation variability should decrease by processing the more informative loci first. SIMPLE, by default, uses the number of alleles as a measure of informativeness, and sorts the loci accordingly. The user may override this default and provide his/her own process order.

As the current implementation of sequential imputation is based on single-locus peels, the memory required for the peeling calculations is linear in the number of loci examined. As part of the peeling calculations, inheritance vector probabilities for all loci are needed. As the number of calculations performed during peeling that call for these inheritance vector probabilities will be much larger than the number of these probabilities, calculating them as required during peeling will lead to many repeated calculations. To avoid these repetitions and to speed up the program, the current implementation of SIMPLE stores all inheritance vector probabilities to be used during peeling. This approach to increasing speed leads to memory requirements that are exponential in the number of loci. Currently most users should be able to process 10–13 markers, depending on the amount of memory their computer system has. Thus for problems with more markers than a user's computer can handle, the least informative loci should be dropped from the analysis. In a future release of SIMPLE, we plan to include a runtime option not to store the table of inheritance vector probabilities, to allow more loci to be processed. This modification should allow over 30 loci to be processed on most systems.

In addition to computing NPL and LOD scores, the current version of SIMPLE has other capabilities. It can perform haplotype analysis to find highly probable haplotype configurations. SIMPLE can also be instructed to output the estimated IBD distributions for all pairs of related individuals. This IBD-sharing information can be used for many other purposes. Among them is the potential to fit variance component models for quantitative traits on large pedigrees, which will be implemented in a future release of our software. Finally, we also plan to implement the allele-sharing model of Kong and Cox [1997] to further increase the capabilities of SIMPLE.

ELECTRONIC-DATABASE INFORMATION

The URL for the software package and the supplementary material:

http://www.stat.ohio-state.edu/~statgen/SOFTWARE/SIMPLE

ACKNOWLEDGMENTS

We thank Dr. Daniel Schaid and the reviewers for their constructive comments and helpful suggestions. This work was supported in part by NSF grant DMS-9971770 (to S.L.). The Collaborative Study on the Genetics of Alcoholism (COGA) (H. Begleiter, SUNY HSCB Principal Investigator,

T. Reich, Washington University, Co-Principal Investigator) includes nine different centers where data collection, analysis, and/or storage take place. The nine sites and Principal Investigators and Co-Investigators are: Indiana Univeristy (T.-K. Li, J. Nurnberger, Jr., P.M. Conneally, and H. Edenberg); Univeristy of Iowa (R. Crowe and S. Kuperman); University of California at San Diego (M. Schuckit); University of Connecticut (V. Hesselbrock); State University of New York, Health Sciences Center at Brooklyn (B. Porjesz and H. Begleiter); Washington University in St. Louis (T. Reich, C.R. Coninger, J. Rice, and A. Goate); Howard University (R. Taylor); Rutgers University (J. Tischfield); and the Southwest Foundation (L. Almasy). This national collaborative study is supported by NIH grant U10AA08403 from the National Institute on Alcohol Abuse and Alcoholism. GAW12 was supported by NIH grant GM31575.

REFERENCES

Abecassis G, Cherny S, Cookson W, Cardon L. 2002. Merlin—rapid analysis of dense genetic maps using sparse gene flow trees. Nat Genet 30:97–101.

Bergman N. 2001. Posterior Cramer-Rao bounds for sequential estimation. In: Doucet A, de Freitas N, Gordon N, editors. Sequential Monte Carlo methods in practice. New York: Springer Verlag. p 321–38.

Blake A, Isard M, MacCormick J. 2001. Statistical models of visual shape and motion. In: Doucet A, de Freitas N, Gordon N, editors. Sequential Monte Carlo methods in practice. New York: Springer Verlag. p 339–57.

Cannings C, Thompson E, Skolnick M. 1978. Probability functions on complex pedigrees. Adv Appl Prob 10:26–61.

Elston R, Stewart J. 1971. A general model for the genetic analysis of pedigree data. Nat Genetics 21:523–42.

Foss E, Lande R, Stahl F, Steinberg C. 1993. Chiasma interference as a function of genetic distance. Genetics 133:631–91.

Gudbjartsson D, Jonasson K, Frigge M, Kong A. 2000. Allegro, a new computer program for multipoint linkage analysis. Nat Genet 25:12–3.

Hastie T, Tibshirani R. 1990. Generalized additive models. London: Chapman and Hall.

Idury R, Elston R. 1997. A faster and more general hidden Markov model algorithm for multipoint likelihood calculations. Nat Genet 47:197–202.

Irwin M, Cox N, Kong A. 1994. Sequential imputation for multilocus linkage analysis. Proc Natl Acad Sci USA 91:1684–88.

Kong A, Cox N. 1997. LOD scores and accurate linkage tests. Am J Hum Genet 61:1179–88.

Kruglyak L, Lander E. 1998. Faster multipoint linkage analysis using Fourier transforms. J Comput Biol 5:1–7.

Kruglyak L, Daly M, Lander E. 1995. Rapid multipoint linkage analysis of recessive traits in nuclear families, including homozygosity mapping. Am J Hum Genet 56:519–27.

Kruglyak L, Daly M, Reeve-Daly M, Lander E. 1996. Parametric and nonparametric linkage analysis: a unified multipoint approach. Am J Hum Genet 58:1347–63.

Lander E, Green P. 1987. Construction of multilocus genetic linkage maps in humans. Proc Natl Acad Sci USA 84:2363–7.

Lange K, Goradia T. 1987. An algorithm for automatic genotype elimniation. Am J Hum Genet 40:250–6.

Lathrop G, Lalouel J, Julier C, Ott J. 1984. Strategies for multilocus linkage in humans. Proc Natl Acad Sci USA 81:3443–6.

Lin S, Speed T. 1996. Incorporating crossover interference into pedigree analysis using the χ^2 model. Hum Hered 46:315–22.

Luo Y, Lin S, Irwin M. 2001. Two-locus modeling of asthma in a Hutterite pedigree via Markov chain Monte Carlo. Genet Epidemiol [suppl] 21:24–9.

Markianos K, Daly M, Kruglyak L. 2001a. Efficient multipoint linkage analysis through reduction of inheritance space. Am J Hum Genet 68:963–77.

Markianos K, Katz A, Kruglyak L. 2001b. A new computational approach for rapid multipoint linkage analysis of qualitative and quantitative traits in large, complex pedigrees, and its implementation in GENEHUNTER. Am J Hum Genet 69:228.

O'Connell J, Weeks D. 1995. The VITESSE algorithm for rapid exact multilocus linkage analysis via genotype set-recoding and fuzzy inheritance. Nat Genet 11:402–8.

O'Connell J, Weeks D. 1999. An optimal algorithm for automatic genotype elimination. Am J Hum Genet 65:1733–40.

Ott J. 1989. Computer-simulation methods in human linkage analysis. Proc Natl Acad Sci USA 86:4175–8.

Ploughman L, Boehnke M. 1989. Estimating the power of a proposed linkage study for a complex trait. Am J Hum Genet 44:543–51.

Thompson E. 2000. Statistical inferences from genetic data on pedigrees. Volume 6. NSF-CBMS Regional Conference Series in Probability and Statistics, Beachwood, OH:IMS.

Whittemore A, Halpern J. 1994a. A class of tests for linkage using affected pedigree members. Biometrics 50:118–27.

Whittemore A, Halpern J. 1994b. Probability of gene identity by descent: computation and applications. Biometrics 50:118–27.

Zhao H, Speed T, McPeek M. 1995. Statistical analysis of crossover interference using the chi-square model. Genetics 139:1031–44.