Efficient Inference of Haplotypes From Genotypes on a Large Animal Pedigree

Eyal Baruch, Joel Ira Weller,1 Miri Cohen-Zinder, Micha Ron and Eyal Seroussi

Institute of Animal Sciences, ARO, The Volcani Center, Bet Dagan 50250, Israel

Manuscript received June 23, 2005
Accepted for publication November 19, 2005

ABSTRACT

We present a simple algorithm for reconstruction of haplotypes from a sample of multilocus genotypes. The algorithm is aimed specifically for analysis of very large pedigrees for small chromosomal segments, where recombination frequency within the chromosomal segment can be assumed to be zero. The algorithm was tested both on simulated pedigrees of 155 individuals in a family structure of three generations and on real data of 1149 animals from the Israeli Holstein dairy cattle population, including 406 bulls with genotypes, but no females with genotypes. The rate of haplotype resolution for the simulated data was >91% with a standard deviation of 2%. With 20% missing data, the rate of haplotype resolution was 67.5% with a standard deviation of 1.5%. In both cases all recovered haplotypes were correct. In the real data, allele origin was resolved for 22% of the heterozygous genotypes, even though 70% of the genotypes were missing. Haplotypes were resolved for 36% of the males. Computing time was insignificant for both data sets. Despite the intricacy of large-scale real pedigree genotypes, the proposed algorithm provides a practical rule-based solution for resolving haplotypes for small chromosomal segments in commercial animal populations.

T he genotyping data of diploid organisms obtained by current laboratory techniques provide unordered allele pairs for each marker. Reconstruction of haplotypes from this data is a crucial step in many applications. Haplotypes of tightly linked markers provide insight into old and rare recombination events, and thus are more informative than single markers. In particular, haplotypes are essential for linkage disequilibrium (LD)-based gene mapping. Numerous studies have shown that individual loci affecting economic traits (QTL) can be detected via linkage to genetic markers (reviewed by WELLER 2001). However, using genetic linkage, the location of a QTL in animal populations generally cannot be resolved to <10 cM, and an interval of this magnitude will still contain ~80 genes. Using LD mapping, MEUWISSEN and GODDARD (2000) proposed a method to narrow the confidence interval for a QTL to a few centimorgan units. Haplotype resolution is also important for commercial animal breeding. MACKINNON and GEORGES (1998) proposed that if a QTL of economic importance is localized to a relatively small chromosomal segment, frequency of the favorable allele can be increased by selection on haplotypes, even though the actual QTL has not been identified.

Approaches to reconstruction of haplotypes by genotype inference can be divided into statistical methods and nonstatistical methods. The first approach applies computational or statistical inference to find the most likely haplotype configuration consistent with the observed genotypic data. Several recent studies have proposed modifications for this method, e.g., CLARK’S (1990) parsimony method, the expectation maximization algorithm (EXCOFFIER and SLATKIN 1995), the partition ligation variant (QIN et al. 2002), Phase (STEPHENS et al. 2001), Haplotyper (QIN et al. 2002), and the phylogenetic approach (GUSFIELD 2001; ESKIN et al. 2003). However, these algorithms generally require large data sets (on the order of tens of thousands of individuals) from populations where a biological “bottleneck” has previously occurred, as presumably happened in humans (Daly et al. 2001; HELMUTH 2001; GABRIEL et al. 2002).

In the nonstatistical approach, every possible trio of two parents and an offspring is examined. The haplotypes are resolved by forward inference of Mendelian rules from the parental genotypes to the offspring genotype and backward inference from offspring genotype to the parents. As the pedigree size increases, the number of computations for most rule-based algorithms increases exponentially. Thus, analysis of pedigrees consisting of hundreds of individuals would require several hours or even days of computations.

Many studies tried to deal with the time-consuming computations by applying specific biological assumptions, such as absence of recombination within the haplotype. O’CONNELL (2000) with ZAPLO described a genotype elimination algorithm, which was intended for single nucleotide polymorphism (SNP) markers under the assumption of zero recombination. TAPADAR et al. (1999) proposed an evolution-based method with
an optimality criterion of a minimum number of recombinations over all the possible haplotype configurations of pedigree members. The difficulty arises when there is a need to handle missing genotypes, as is usually the case in actual data sets. QIAN and BECKMANN (2002) developed a six-rule-based algorithm that exhaustively searches all possible minimum recombinant haplotype configurations and completes missing genotypes. To overcome the poor performance of this algorithm in large pedigrees, LI and JIANG (2003) proposed a polynomial time-exact algorithm for haplotype reconstruction without recombinants. This algorithm first identifies all the necessary constraints on the basis of Mendelian laws and the zero recombination assumption, but can be applied only to data that have no missing genotypes. Another formulation of the same algorithm is the branch-and-bound strategy (LI and JIANG 2004) that utilizes a partial order relationship and some other special relationships among variables to decide the branching order. This algorithm can be applied to pedigree with missing data, but the running time will (linearly) increase with the rate of missing data. Moreover, when multiple solutions exist, which is the usual case, especially in real pedigree data, the best haplotype configuration is selected on the basis of a maximum-likelihood approach. Hence it may lead to incorrectly resolved haplotype configuration.

The aim of this study was to develop an efficient rule-based haplotype reconstruction method suitable for pedigrees of several hundred individuals, assuming no recombination and allowing for missing data. The method was applied to both simulated and actual data from the Israeli dairy cattle population and compared to alternative algorithms.

METHODS

Definitions: Following ERONEN et al. (2004), we assumed a set (map for a chromosomal segment) \( M \) of \( l \) markers \( 1, \ldots, l \) and denote the set of alleles of marker \( i \) by \( A_i \), where \( A_i = \{ a_{i1}, a_{i2}, a_{i3}, \ldots, a_{ik} \} \). The number of different alleles of marker \( i \) in the population is denoted by \( k \). For SNP, we will assume \( k = 2 \). The single-locus genotype \( G(i) \) over \( M \) is an unordered allele pair at locus \( i \), which corresponds to the pair of homologous chromosomes, \( G(i) \in \{ \{a_{ip}, a_{iq} \} | a_{ip}, a_{iq} \in A_i \} \). Let \( G(s, t) \) denote the allelic sequence from the \( s \)th to the \( t \)th marker on both chromosomes. \( G(s, t) = \{ H_1(s, t), H_2(s, t) \} \), where \( H_1(s, t) \in \prod_{i=s}^{t} A_i \). For any \( i \), a vector of (known) alleles on one of the homologous chromosomes and \( H_2(s, t) \) is its counterpart on the other chromosome. We will denote \( H(i, \hat{i}) \) by \( H(i) \).

The \( i \)th genotype will be denoted “resolved” if two haplotypes, \( H_1(i) \) and \( H_2(i) \), are determined, such that \( G(i) = \{ H_1(i), H_2(i) \} \), and the \( H_1(i) \) and \( H_2(i) \) haplotype sources (paternal or maternal) are also determined. \( H_1(i), H_2(i) \), and \( G(i) \) are termed “consistent” when all markers in \( G(1, \ldots, l) \) are resolved and \( \{ H_1(i), H_2(i) \} \) for \( i = 1, \ldots, l \) is a possible haplotype configuration for genotypes \( G(1, l) \). Thus a haplotype configuration fully describes the haplotypes in a member of the pedigree and the origin of each allele included in the haplotypes.

The \( i \)th marker has a “conflict” if \( G(i) \) must contain \( \{ H_1(i), H_2(i) \} \), based on Mendelian rules, but does not. The conflicts were divided into three categories. Conflicts due to no common allele between the parent and the putative progeny that were both originally genotyped in the raw data are denoted type 1. Conflicts due to no common allele between the parent and putative progeny, but either the parent or the progeny had a missing genotype which was reconstructed by application of Mendelian rules, are denoted type 2. A pedigree of more than two generations is required to detect a type 2 conflict. Sequences of more than one marker in which there is a shared allele between parent and progeny for each marker, but neither of the progeny haplotypes correspond to either of the parental haplotypes, are denoted type 3. All three types of conflicts can be due to mutations, incorrect parentage recording, or genotyping errors. Type 3 conflicts can also be due to recombination.

When the program identifies a conflict, it marks the locus and individuals in conflict. The haplotypes of the two individuals in conflict are considered unresolved, and the user is prompted to resolve the conflict by changing or deleting a genotype. If the user does not resolve the conflict, the algorithm will proceed under the assumption that both genotypes in conflict are correct, and these genotypes will be used to resolve haplotypes of other relatives.

A haplotype \( H(s, t) \) and a genotype \( G(s, t) \) are denoted a “match” if there exists the string \( \bar{H} \in \prod_{i=s}^{t} A_i \), such that \( \{ H(s, t), \bar{H} \} \) is consistent with \( G(s, t) \). Using the declarations above, the “haplotype reconstruction problem” is defined as follows; given a set \( \bar{G} \) of an individual’s genotypes for several closely linked markers with all possible haplotype configurations, the objective is to derive the correct haplotype configurations, \( G \), where \( G \in \bar{G} \).

Reconstruction of missing genotypes: Animals without parents recorded in the data file are denoted “founders.” The sire, dam, and a single direct offspring are denoted a “nuclear family.” In actual data, genotypes for some markers are missing, and some genotypes will be incorrect. Some missing genotypes can be inferred using Mendelian rules and the data of relatives. We propose that reconstruction of missing genotypes should precede haplotype resolution, since genotype (or allele) reconstruction is more fundamental phase in comparison to haplotype resolution, which requires determination not only of the alleles at a locus, but also of their parental associations. Nevertheless, during haplotype resolution a missing genotype might also be
resolved due to its flanking alleles. In the proposed algorithm the entire pedigree is first scanned from bottom to top, considering each nuclear family. If either the parent or the offspring of the individual with unknown genotype is homozygous, then the algorithm determines that the individual with the unknown genotype must have at least one copy of this allele. If information from a homozygous progeny and its parents is insufficient to resolve both alleles of the individual with unknown genotype, the algorithm then checks if there is an allele in any of the offspring of this individual that does not appear in its mate. If so, the algorithm determines that this allele was derived from the parent with the missing genotype. Any contradiction to these two rules is evidence of marker conflict.

**Resolving haplotypes:** In every nuclear family there is an offspring–parent pair with a maximum number of parental unresolved markers that are resolved in its offspring. It can be formulated by \( \max (|\{ i | G_2(i) \text{ resolved and } G_1(i) \text{ not resolved} \}|) \), where \( G_2(i) \) is the parental genotype of marker \( i \), and \( G_1(i) \) is the offspring genotype for this marker. The offspring with the maximum number of resolved markers that are not resolved in its parent can potentially contribute the most information to resolving its parents’ markers. Haplotype resolution is divided into two sequential steps: solving for the parent by separation of its two haplotypes, \( G(1, l) = \{H_1(1, l), H_2(1, l)\} \), and assigning the progeny haplotype \( (H_1 \text{ or } H_2) \) to the parent from the offspring whose haplotype was most complete. This haplotype is then denoted as the resolved haplotype (RHap). In the second step, the RHap is assigned to each of the parent’s progeny, where possible. The criterion of which haplotype to assign when either solving for the parent by its progeny or solving for the progeny by its parent depends on the existence of a resolved marker that is heterozygous in the parental genotype and has been resolved in the progeny. This criterion is denoted the “separation rule.” An example is given in Figure 1.

Although it is not possible to determine haplotype source (paternal or maternal) for founders, it is possible to determine both haplotypes. Therefore, when assigning RHap for founders, the algorithm determines the founder’s offspring with maximal contribution to the determination of the parental haplotypes, i.e., the offspring with the maximum number of resolved markers not resolved in its parent, and applies the separation rule. Haplotype determination can be improved in nonfounder families by using information from other offspring of the same parent. The more resolved loci found in offspring that are not resolved in the parent, the more the offspring contributes to the composition of RHap. The offspring can then be utilized as the source for RHap, as long as the separation rule can be applied, as in the example in Figure 1, thus gathering haplotype information not just from a single offspring, but also from its siblings.

In a parent with heterozygous loci, it is then possible to identify which haplotype was passed to a progeny, provided that the progeny genotypes differ from the parent in at least one locus heterozygous in the parent. In this case, it is possible to improve RHap iteratively each time an offspring is found to have a resolved marker that is not resolved in his parent, which fulfills the separation rule. This can be done only in nonfounder families, because in founders it is not possible to determine paternal or maternal haplotypes, except in the case where all the founder’s genotypes are homozygous.

After maximum RHap resolution is achieved, RHap then can be used as a haplotype source. For example, in the case given in Figure 1, resolution of parental genotype can be used to better resolve the offspring’s genotype, which then can be used to resolve its siblings and its own offspring, which in turn can be used to resolved
the offspring haplotype and its own parent and vice versa. Offspring without resolved markers can be resolved only to the level of the best-resolved offspring in the same nuclear family. Resolving other families is then processed recursively, with the offspring of the first family considered as the parent of a new family.

We implemented the algorithm using C++ object oriented programming. The program, Large Scale Pedigree Haplotyper (LSPH), can be downloaded at http://cowry.agri.huji.ac.il/LSPH.exe.

Generation and analysis of simulated data: Ten independent populations were simulated. The chromosomal segment analyzed consisted of five tightly linked biallelic markers. Genotypes were assumed known only for males. Population allelic frequencies were determined for each marker by simulation from a uniform distribution. Each population consisted of five founder males. Genotypes and haplotypes of founders were generated by random sampling of alleles on the basis of simulated population allelic frequencies. Each sire passed one of his two haplotypes intact to each progeny, each with a probability of 0.5. The dam haplotype of each progeny was determined by random sampling of alleles, on the basis of the population allelic frequencies. Five male progeny were generated for each male founder, and five male progeny were generated for each male of the second generation, for a total of 155 males with recorded genotypes. Genotypes and haplotypes of the grandsons of the founders were determined using the same procedure used to determine the genotypes of their sires. That is, for each grandson, one complete haplotype of his sire was selected as the paternal haplotype, and the maternal haplotype was determined by random selection of alleles on the basis of population frequencies. Thus zero recombination from sires to sons was assumed throughout. In addition, 10 more simulated populations were generated under the same conditions, but with 20% of genotypes randomly designated as missing.

Simulated data sets were also analyzed by the block-extension algorithm (Li and Jiang 2003) and integer linear programming (ILP) algorithm (Li and Jiang 2004) of the haplotype resolution package PedPhase2 and by SimWalk2 version 2.91 (Sobel and Lange 1996). Results of these programs and LSPH were compared. PedPhase2 and SimWalk2 required generating “dummy” records for individuals listed as parents. Therefore it was necessary to generate dummy dam records for all non-founders, increasing the number of individuals in each data set to 305. Simwalk2 could not run on the simulated data set, because the number of founders exceeded the maximum allowed. Therefore an abbreviated data set consisting of 155 individuals was generated for this program by deleting 75 grandsons and their 75 dams.

Analysis of actual data: A QTL affecting milk production traits located in the central region of chromosome 6 was detected independently by different research groups in several cattle populations (summarized by Ron et al. 2001). Twelve diallelic markers were detected within a 2-cM chromosomal segment distal to BM143, a multiallelic microsatellite, as described by Cohen-Zinder et al. (2005). Details of the polymorphisms genotyped on BTA6 are given in Table 1. These markers are located in 10 genes and included SNP or variations in simple sequence repeats.

Semen was obtained for 418 artificial insemination sires of the Israeli Holstein population with genetic evaluations. These bulls were genotyped for the 12 diallelic markers and BM143. Of these bulls, 12 had three or more genotype conflicts between their genotype and

| Gene     | Location | Polymorphism* | No. of valid genotypes | Allele frequency (%) | Heterozygous genotypes (%) |
|----------|----------|---------------|------------------------|----------------------|---------------------------|
| BM143    | Intergenic region | Microsatellite (GT)_n | 350 | 55.1 | 83.1 |
| MLR1     | Exon 7   | Insertion (TGAT) | 296 | 50.5 | 50.7 |
| MED28    | Exon 4   | Substitution (C/T) | 310 | 57.2 | 40.0 |
| LAP3     | Exon 12  | Substitution (C/T) | 334 | 57.3 | 56.9 |
| IBSP     | Exon 7   | Substitution (A/G) | 333 | 61.3 | 50.9 |
| SPP1(1)  | Intron 5 | Substitution (A/G) | 361 | 57.0 | 48.7 |
| SPP1(2)  | Exon 7   | Substitution (T/G) | 302 | 72.9 | 41.4 |
| PKD2     | Promoter | Insertion (A)    | 322 | 67.1 | 41.0 |
| ABGG2    | Intron 3 | Substitution (A/T) | 342 | 55.4 | 48.5 |
| PPMIK    | Exon 2   | Substitution (GC/AT) | 363 | 73.6 | 43.0 |
| HERC6    | Intron 5 | Insertion (C)    | 324 | 67.9 | 37.6 |
| FAM13A1(1) | Intron 9 | Substitution (A/G) | 372 | 81.8 | 31.1 |
| FAM13A1(2) | Exon 12 | Substitution (C/A) | 365 | 58.9 | 53.4 |

* For insertions, the inserted DNA segment is given in parentheses. For substitutions, the two forms are given in parentheses separated by a slash.

Table 1: Description of the polymorphisms analyzed in the actual data
the genotype of their putative sires. We therefore con-
cluded that either paternity was incorrect or that our
semen sample was mislabeled. These 12 bulls were dis-
carded from further analysis, leaving 406 bulls. The num-
bers of bulls genotyped for each marker are also listed in
Table 1. There were 4374 valid genotypes of 5278 pos-
sible genotypes (406 animals × 13 markers). Thus geno-
typing efficiency was 83%. Of the valid genotypes, 2111
(48%) were heterozygous. Frequencies of heterozygotes
by marker are also given in Table 1. The frequencies of
the bulls’ year of birth are given in Figure 2. More than
95% of the bulls were born between 1980 and 1996. All
known parents and grandparents of bulls with geno-
types were included in the pedigree file for a total of
1149 animals. Of these, 372 were “founders,” that is,
cows or bulls with parents not included in the pedigree
file, and 584 were females. The maximum number of
generations from founders to the youngest genotyped
animals was six. None of the females or founders was
genotyped, because DNA was not available for these
animals. Nearly all of the females had only a single male
progeny.

Thirteen alleles ranging in fragment length from 90
to 118 bp were observed for BM143 and their allelic
frequencies are given in Table 2. Most allele frequencies
were quite low. Most of the conflicts detected by ap-
plication of the algorithm involved BM143. Therefore
the data were also analyzed with this marker deleted.
The number of individuals analyzed and the genotypes
with and without BM143 included are given in Table 3.
The actual data results were analyzed on the basis of the
number and types of conflicts detected and the propor-
tion of haplotype-resolved heterozygous marker loci.

RESULTS AND DISCUSSION

CPU time for the analysis of either simulated or real
data sets by LSPH was never >2 sec on an Intel Pentium
1.4 GHz M processor computer with 1 GB RAM. The
computing time of LSPH with relatively large data sets
was much shorter than, or at least as short as, other
tested rule-based methods. The results from the analysis
of the simulated data are summarized in Table 4. The
average rate of haplotype resolution with no missing
data was 91.5% with a standard deviation of 2.0%.
Haplotypes were resolved for 74% of the heterozygous
markers with a standard deviation of 8.6%. With 20% miss-
ing data, the rate of haplotype resolution was 67.3%
with a standard deviation of 1.3%, and resolution for the
heterozygous markers was 54.2% with a standard de-
viation of 6.1%. None of the haplotype resolutions were
incorrect. Unlike other algorithms (e.g., Li and JIANG
2004), the amount of missing data has no effect on CPU
time.

LSPH results for the real data are presented in Table
5. Results are presented separately, including BM143,
the only multiallelic marker, and with BM143 deleted. In
both cases the program was able to resolve only ~2% of
the unrecorded genotypes. The fraction of resolved
markers for real data was much smaller than that for the
simulated data. This result was due chiefly to the greater

| TABLE 2 |
The allelic frequencies for BM143 in the bulls genotyped

| Allele length (bp) | Frequency |
|-------------------|-----------|
| 90                | 1         |
| 92                | 20        |
| 94                | 73        |
| 100               | 33        |
| 102               | 116       |
| 104               | 62        |
| 106               | 1         |
| 108               | 8         |
| 110               | 89        |
| 112               | 183       |
| 114               | 31        |
| 116               | 68        |
| 118               | 15        |
| Total             | 700       |

| TABLE 3 |
Statistics of the actual data

| Including BM143 | Without BM143 |
|-----------------|---------------|
| Animals         |               |
| Total           | 1,149         | 1,142         |
| Founders        | 372           | 372           |
| Ancestors, not founders, without genotypes | 371 | 379 |
| Animals with genotypes | 406 | 401 |
| Genotypes      |               |
| Markers        | 13            | 12            |
| No. of possible genotypes | 14,937 | 13,704 |
| (total animals × markers) |     |     |
| Total recorded genotypes | 4,374 | 3,945 |
| Recorded heterozygous genotypes | 2,111 | 1,795 |

* No founders had genotypes.
† Initially known genotype data.
‡ Initially known heterozygous genotype data.
genotype resolution with and without origin was resolved for only 22% and 17% of the heterozygous genotypes, which are by definition 'resolved.' Allele known genotypes, but this includes all homozygous genotypes. In addition, real data may contain recombination events as well as incorrect genotypes. These were marked as 'type 3' conflicts and could not be resolved. LSPH was able to resolve haplotypes for 60% of the known genotypes, but this includes all homozygous genotypes, which are by definition "resolved." Allele origin was resolved for only 22% and 17% of the heterozygous genotypes with and without BM143 included, respectively. With BM143 included, the program was able to completely resolve at least one heterozygous marker for 205 (18%) animals of the total. Assuming no recombination within this chromosomai segment, this is also the number of animals with resolved haplotypes. As noted previously, the analysis included 584 female ancestors that were not genotyped. No female haplotypes were resolved. Thus haplotypes were resolved for 36% of the males. With BM143 deleted, the number of animals with resolved haplotypes was reduced to 150, or 13% of the total animals and 27% of the males.

With BM143 included, 75 conflicts were detected. Of these, 22 were denoted type 1, 3 were denoted type 2, and 50 were denoted type 3. Only 36 conflicts were detected with BM143 deleted, and none were type 2. Thus inclusion of a microsatellite with multiple alleles increased the number of animals with resolved haplotypes by 50%, but doubled the number of conflicts.

As noted, type 1 and 2 conflicts are due to genotyping mistakes, incorrect parental assignments, sample switching, or mutations. Genotyping error rates for microsatellites are on the order of 1%, while mutation rates are much lower (Well et al. 2004). Previous results indicate that the rate of incorrect paternity recording of cows during the 1990s was 11.7% (Well et al. 2004), although it is expected that the incorrect paternity rate for AI bulls should be lower, and paternity was verified by genetic markers for at least 80% of the bulls (Y. Zaron, personal communication). In addition, incorrect paternity should lead to multiple conflicts, as was the case for 12 bulls, which were discarded from the analysis. Although type 3 conflicts could also be due to recombination, it is very unlikely that this is the case for most of these conflicts. Haplotypes were resolved for only 203 and 150 animals with and without BM143 included. Assuming that the chromosomal segment spans 2 cM, only three to four recombinations per generation should have occurred among the animals with resolved haplotypes, and most would be undetectable. In conclusion, genotyping mistakes are still the most likely explanation for most of the conflicts.

SNP have the advantages that they are more frequent throughout the genome and genotyping errors are less frequent than for microsatellites, but have the disadvantage that they are less polymorphic than microsatellites, which reduces the likelihood that allele origin can be unequivocally determined. Clearly genotyping females, especially dams of AI sires, could increase the fraction of haplotypes resolved; but in most cases this will not be a viable option, because unlike AI sires, DNA of females is not collected and stored.

Studies based on the minimum recombination principle have been previously published. Tapadar et al. (1999) implemented their model using likelihood function, but could not deal with missing data. O’Connell (2000) estimated the haplotype frequencies for biallelic markers using ZAPLO, but had difficulties in completing missing data and analysis of markers with more than two alleles. The methods proposed by Qian and Beckmann (2002) and Li and Jiang (2003) deal with missing and inaccurate genotype data, but these methods can be applied only to pedigrees of several tens of individuals. The implementation package PedPhase 2.0 for the minimum recombinant haplotype configuration

---

**TABLE 4**

Means and standard deviations for haplotype resolution in the analysis of the simulated data

| Markers with BM143 | Missing data (%) | Haplotypes resolved | Mean (%) | SD (%) |
|--------------------|------------------|----------------------|----------|--------|
|                    | 0                | Total                | 91.5     | 2.0    |
|                    | Heterozygous     |                      | 74.4     | 8.6    |
|                    | 20               | Total                | 67.3     | 1.3    |
|                    | Heterozygous     |                      | 54.2     | 6.1    |

Each population included 155 males of three generations. Ten simulations were analyzed for each set of conditions.

---

**TABLE 5**

Results of the actual data

| Genotype resolution*                        | Including BM143 | Without BM143 |
|--------------------------------------------|-----------------|---------------|
| Total genotypes                            | 4,608           | 4,141         |
| Heterozygous genotypes                      | 2,347           | 1,991         |
| Alleles                                    | 12,628          | 11,445        |
| Phase resolution                            |                 |               |
| Completely resolved genotypes, including homozygotes | 2,786 | 2,502 |
| Resolved alleles, including homozygotes     | 6,716           | 6,023         |
| Resolved heterozygous genotypes             | 525             | 352           |
| Animals with at least one marker completely resolved | 203 | 150 |
| Conflicts‡                                 |                 |               |
| Type 1                                     | 22              | 18            |
| Type 2                                     | 3               | 0             |
| Type 3                                     | 50              | 20            |

*Including recorded genotypes.

‡The conflict types are described in the text.
Inference of Haplotypes 1763

| Comparison criterion | LSPH | SimWalk2a | PedPhase (block extension)b | PedPhase (ILP)c |
|----------------------|------|-----------|-----------------------------|---------------|
| Average running speed | <1 sec | —         | <1 sec                      | <8 sec        |
| Limitation on the no. of individuals | No | Yes | No | No |
| Requires generation of “dummy” parents | No solution is given | A single solution is given and flagged | A single solution is given, but not flagged | A single solution is given, but not flagged |
| Multiple solutions are possible | Missing parents | Reports | Reports | Terminates with error status |
| Individuals with incorrect gender | Reports | Reports | Terminates with error status | Ignores |
| Redundant individuals | Reports | Reports | Terminates with error status | Ignores |
| Percentage of incorrect allele determination of heterozygotes, no missing data | 0 | — | 24.4 | 10.7 |
| Percentage of incorrect allele determination of heterozygotes 20% missing data | 0 | — | — | 15.9 |
| Percentage of incorrect genotype determination, 20% missing data | 0 | — | — | 8.5 |
| Percentage of correct allele determination of heterozygotes, no missing data | 74.4 | — | 75.6 | 89.3 |
| Percentage of correct allele determination of heterozygotes 20% missing data | 54.2 | — | — | 84.1 |

*Sobel and Lange (1996), distribution sites: http://watson.hgen.pitt.edu/register and http://www.genetics.ucla.edu/software. SimWalk could not run on the simulated data sets of 155 individuals. Thus values referring to this data set are missing.

Li and Jiang (2003) distribution site: http://www.cs.ucr.edu/˜jili/haplotyping. Percentage of errors is not given for data sets with missing genotypes because the block-extension algorithm of PedPhase terminated with error status in this case.

Li and Jiang (2004) distribution site: http://www.cs.ucr.edu/˜jili/haplotyping.

The results of LSPH, SimWalk2, and two functions of PedPhase—block-extension and ILP—on the simulated data sets are compared in Table 6. As noted previously, SimWalk2 could not run on the simulated data sets of 305 individuals, including dams. Running time for SimWalk2 for the truncated set of 155 individuals, including dams, was 50 min, as compared to <1 sec for LSPH or either PedPhase function (although for some pedigree files, the ILP algorithm running time was >8 sec). Of the three programs, LSPH is the only one that does not provide a solution for haplotypes that cannot be unequivocally resolved. Although both SimWalk2 and PedPhase present a single-haplotype resolution of all possible combinations, only SimWalk2 differentiates between these cases and the cases in which haplotypes are unequivocally determined. For both PedPhase

---

**TABLE 6**

Comparison of LSPH, PedPhase, and SimWalk2 on simulated data, 10 sets of 155 animals

problem (Li and Jiang 2004) contains five functions for inferring haplotypes from genotypes for members of a general pedigree. Two functions [locus-based dynamic programming (DP) and member-based DP] are exponential in the size of the input and are not recommended by the authors for a large number of markers or individuals in the pedigree. The ILP function computes all possible solutions whenever it cannot unequivocally determine the haplotype. Moreover, the number of all possible haplotype solutions with zero recombinants, even for a moderately sized data set, can be huge. The constrained-finding function produces similar results to LSPH on simulated data with no missing data, but cannot deal with missing data. The last function of the PedPhase package, the block-extension algorithm, is intended for analysis of large pedigrees. This is a heuristic algorithm (Li and Jiang 2003, 2004) and as such provides one possible solution out of many, which may or may not be correct. In real data sets, the founders’ haplotypes are generally unknown, which increases the number of possible solutions. Not all of the functions in the PedPhase package perform complete validity checks. LSPH, on the contrary, performs several different types of validity checks from the basic level of pedigree structure (such as duplicate individuals or missing individual–family relationship) to the Mendelian consistency check and data completion (when possible) for each locus for each individual. In case of inconsistency, LSPH also determines the mismatch type, as described.

The results of LSPH, SimWalk2, and two functions of PedPhase—block-extension and ILP—on the simulated data sets are compared in Table 6. As noted previously, SimWalk2 could not run on the simulated data sets of 305 individuals, including dams. Running time for SimWalk2 for the truncated set of 155 individuals, including dams, was 50 min, as compared to <1 sec for LSPH or either PedPhase function (although for some pedigree files, the ILP algorithm running time was >8 sec). Of the three programs, LSPH is the only one that does not provide a solution for haplotypes that cannot be unequivocally resolved. Although both SimWalk2 and PedPhase present a single-haplotype resolution of all possible combinations, only SimWalk2 differentiates between these cases and the cases in which haplotypes are unequivocally determined. For both PedPhase
and SimWalk2, nonexistent recombination events were reported. That is, haplotype solutions requiring recombinations in previous generations were derived, even though the simulated population was generated without recombinations.

Of the heterozygous loci, 24.4 and 10.7% were incorrectly resolved by the block-extension and ILP algorithms of PedPhase with no missing data. It should be noted that since there are only two alternatives with equal probabilities, random haplotype determination would result in 50% correct decisions. Only the ILP algorithm was able to run on the data sets with 20% missing data, and in this case 8.5% of the genotype determinations were incorrect. Since only 20% of genotypes were missing, 42.5% of the missing genotypes were incorrectly determined. Of the correctly determined genotypes, including both known and reconstructed, 15.9% of the heterozygous loci were incorrectly resolved.

The percentage of correct allele determinations of heterozygotes for LSPH and the two PedPhase algorithms are also presented, with and without missing data. The results for LSPH are the same as those given in Table 4. With no missing data, the block-extension algorithm value is only marginally higher than LSPH, even though 24.4% of all the determinations are incorrect. The ILP algorithm was able to correctly resolve allelic origin for 15% more of the heterozygote genotypes, but this is still only marginally better than could be obtained by LSPH, if the 25.6% unresolved heterozygotes are randomly assigned. On the basis of this comparison, the “resolution” rate for LSPH can be assumed to be 87.2%, with the advantage that the program indicates which haplotypes are known with certainty. With missing data, the ILP algorithm is able to correctly resolve 84.1% of the heterozygotes, as compared to the 54.2% of heterozygotes that are resolved with certainty by LSPH. Again, another 22.9% of the unresolved heterozygotes would be “correctly resolved” by random assignment.

LSPH was more robust than the other programs with respect to the input requirements. It could handle pedigree data with different types of missing information and discrepancies, such as only a single parent is known, only one allele of a genotype is recorded, gender conflicts, or the same individual is recorded twice. PedPhase terminated with error status if there were missing parents, redundant individuals, or Mendelian conflicts in genotype. PedPhase terminated normally if there were incorrect gender determinations, but did not note the discrepancy. Although all the programs were designed to run on the Windows operating system, only LSPH has the capability to produce output in Microsoft Excel XML format (among others), which was discovered to be extremely helpful in manipulating large amounts of data.

As noted in the Introduction, information on haplotype frequencies can be used to determine the most probable haplotype in situations in which more than a single solution is possible. The aim of statistics-based programs, such as SimWalk2, is to find a haplotype configuration with the maximum likelihood under the assumed model. Exact algorithms for finding the most probable haplotype configuration can work only for small data sets, where the number of consistent haplotype configurations for a given pedigree is in a manageable range.

We developed and implemented an algorithm for inference of haplotypes suitable for analysis of a large population without a well-defined pedigree structure, with individuals genotyped for many closely linked genetic markers under the assumption of no recombination. This was useful for inferring missing genotype, as well as resolving haplotypes by using the information from closely linked markers. During the development of this algorithm, effort was directed toward handling data from a large pedigree with missing and erroneous genotypes. LSPH does not perform an exhaustive search, and does not make any assumptions on the underlying population, and therefore can also be applied to analysis of human populations.

The algorithm is rule based, and as such the reconstructed haplotypes are error free, provided that there are no genotyping mistakes or recombinations. Under these restrictions, the optimal solution can then be defined as resolution of the maximum number of haplotypes. We cannot prove that LSPH is optimal in this sense, although on simulated data 91% of the haplotypes were resolved with no missing data.

Kerr and Kinghorn (1996) developed a segregation analysis method to determine genotype probabilities for individuals that were not genotyped on the basis of the known genotypes of their relatives. This method can be applied to a very large population, but considers only a single locus and does not resolve haplotypes. Windig and Meuwissen (2004) developed a rapid algorithm that determines the most probable haplotypes for short chromosomal segments with many markers and large families. Further study is suggested to combine these methods with our algorithm for further resolution of genotypes and haplotypes.

We thank L. Jing for providing us with the implementation package, PedPhase 2.0. Funding for this study was provided by a grant from the Israel Milk Marketing Board.

LITERATURE CITED

Clark, A. G., 1990 Inference of haplotypes from PCR-amplified samples of diploid populations. Mol. Biol. Evol. 7: 111–122.

Cohen-Zinder, M., E. Seroussi, D. M. Larkin, J. J. Loor, A. Everts-van der Wind et al., 2005 Identification of a missense mutation in the bovine ABCG2 gene with a major effect on the QTL on chromosome 6 affecting milk yield and composition in Holstein cattle. Genome Res. 15: 936–944.

Daly, M. J., J. D. Rioux, S. F. Schaffner, T. J. Hudson and E. S. Lander, 2001 High-resolution haplotype structure in the human genome. Nat. Genet. 29: 229–232.

Eronen, L., F. Geerts and H. Toivonen, 2004 A Markov chain approach to reconstruction of long haplotypes. Pac. Symp. Bio- comput. 9: 104–115.
Eskin, E., E. Halperin and R. M. Karp, 2003 Large scale reconstruction of haplotypes from genotype data, pp. 104–113 in Proceedings of the 7th Annual International Conference on Research in Computational Molecular Biology. ACM Press, New York.

Excoffier, L., and M. Slatkin, 1995 Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population. Mol. Biol. Evol. 12: 921–927.

Gabriel, S. B., S. F. Schaffner, H. Nguyen, J. M. Moore, J. Roy et al., 2002 The structure of haplotype blocks in the human genome. Science 296: 2225–2229.

Gusfield, D., 2001 Inference of haplotypes from samples of diploid populations: complexity and algorithms. J. Comp. Biol. 8: 305–324.

Helmuth, L., 2001 Genome research: map of the human genome 3.0. Science 293: 583–585.

Kerr, R. J., and B. P. Kinghorn, 1996 An efficient algorithm for segregation analysis in large populations. J. Anim. Breed. Genet. 113: 457–469.

Li, J., and T. Jiang, 2003 Efficient inference of haplotypes from genotypes on a pedigree. J. Bioinform. Comput. Biol. 1: 41–69.

Li, J., and T. Jiang, 2004 An exact solution for finding minimum recombinant haplotype configurations on pedigrees with missing data by integer linear programming, pp. 101–110 in Proceedings of the 8th Annual International Conference on Research in Computational Molecular Biology. ACM Press, New York.

Mackinnon, M. J., and M. A. J. Georges, 1998 Marker assisted pre-selection of young dairy sires prior to progeny testing. Livest. Prod. Sci. 54: 229–250.

Meuwissen, T. H. E., and M. E. Goddard, 2000 Fine mapping of quantitative trait loci using linkage disequilibria with closely linked marker loci. Genetics 155: 421–430.

Niu, T. H., Z. H. S. Qin, X. P. Xu and J. S. Liu, 2002 Bayesian haplotype inference for multiple linked single-nucleotide polymorphisms. Am. J. Hum. Genet. 70: 157–169.

O’Connell, J. R., 2000 Zero-recombinant haplotyping: applications to fine mapping using SNPs. Genet. Epidemiol. 19 (Suppl. 1): S64–S70.

Qian, D., and L. Beckmann, 2002 Minimum-recombinant haplotyping in pedigrees. Am. J. Hum. Genet. 70: 1434–1445.

Qin, Z. H. S., T. H. Niu and J. S. Liu, 2002 Partition-ligation-expectation maximization algorithm for haplotype inference with single-nucleotide polymorphisms. Am. J. Hum. Genet. 71: 1242–1247.

Ron, M., D. Kliger, E. Feldmesser, E. Seroussi, E. Ezra et al., 2001 Multiple QTL analysis of bovine chromosome 6 in the Israeli Holstein population by a daughter design. Genetics 159: 727–735.

Sobel, E., and K. Lange, 1996 Descent graphs in pedigree analysis: applications to haplotyping, location scores, and marker-sharing statistics. Am. J. Hum. Genet. 58: 1323–1337.

Stephens, M., N. J. Smith and P. Donnelly, 2001 A new statistical method for haplotype reconstruction from population data. Am. J. Hum. Genet. 68: 978–989.

Tapadar, P., S. Ghosh and P.P. Majumder, 1999 Haplotyping in pedigrees via a genetic algorithm. Hum. Hered. 50: 43–56.

Weller, J. I., 2001 Quantitative trait loci analysis in animals. CABI, New York.

Weller, J. I., E. Feldmesser, M. Golik, I. Tager-Cohen, R. Domochovsky et al., 2004 Factors affecting incorrect paternity assignment in the Israeli Holstein population. J. Dairy Sci. 87: 2627–2640.

Windig, J. J., and T.H.E. Meuwissen, 2004 Rapid haplotype reconstruction in pedigrees with dense marker maps. J. Anim. Breed. Genet. 121: 26–39.

Communicating editor: R. W. Doerge