Optimal haplotype reconstruction in half-sib families

Aurélie Favier, Jean-Michel Elsen, Simon de Givry, Andrés Legarra
INRA, Toulouse, France
{afavier,jean-michel.elsen,degivry,andres.legarra}@toulouse.inra.fr

Abstract

In the goal of genetic improvement of livestock by marker assisted selection, we aim at reconstructing the haplotypes of sires from their offspring. We reformulated this problem into a binary weighted constraint satisfaction problem. Our results showed these problems have a small treewidth and can be solved optimally, improving haplotype reconstruction compared to previous approaches especially for medium-size half-sib families.

1 Introduction

Haplotype-based analysis plays an important role in genetics, including study of a population, association mapping, and linkage / association analysis. However haplotypes of diploid individuals cannot easily be acquired and only unphased genotype data can be obtained through application of experimental techniques. It is therefore necessary to propose efficient haplotype reconstruction methods from genotype data, able to cope with a large number of dense markers such as *single nucleotide polymorphisms* (SNPs). Di-allelic SNPs are mutations at single nucleotide positions taking two values (e.g., *allele A* or *B*), and are the most prevalent sequence variations between individuals of all species. The combination of marker alleles on a single chromosome is called a *haplotype*. The combination of unordered pairs of alleles on homologous chromosomes is called a *genotype*.

There are two main sources of genotype data for haplotype inference: coming either from a population of unrelated individuals, or from a *pedigree*, which gives the parental relationships between individuals [13]. We are interested in the latter case where large pedigrees of livestock are available. Two categories of methods exist: statistical methods [1, 12, 18, 8, 17, 7] and rule-based methods [10, 14, 6]. The latter often assume zero recombinants or are more appropriate for pedigree data with a small expected number of recombinations, such as high density marker data in a short chromosomal region. The problem is NP-hard, even in the case of tree pedigree and no missing data [6]. Exact (i.e., complete) methods [1, 6, 8] have their worst-case time complexity exponential in the minimum between the number of individuals and the number of markers. Another option is the use of approximate methods such as greedy and iterative search methods [10, 14, 18, 7] or Monte-Carlo methods [17].

There is a need in animal genetics for exact and fast methods for haplotype reconstruction: current data in cattle genetics consists of thousands of individuals and tens of thousands of markers. We propose a new statistical exact method for haplotype inference from genotypes on such large pedigree data under the Mendelian laws of inheritance and the probability of recombination events.

Mendel's laws involved are very simple: there is one marker allele coming from each of the parents, and, for a given marker, the copy that the parent transmits to its progeny is picked up at random. So, in some cases the determination of allele origin is very simple. For example, if a father/*sire* has an *homozygous* (i.e., same alleles) genotype AA at one marker, and his son has an *heterozygous* (i.e., different alleles) genotype AB, then with certainty allele A in the son came from the father.

The second law involved is the probability of *recombination events*. Recombination events are produced by meiosis, which is a complicated biological phenomenon. However, *genetic maps* have been built that condense the probability of recombination (or recombination fraction) between any two points in a chromosome, aka two *loci*, into a linear metric, usually the Haldane's mapping function, assuming no interference in the formation of crossing-overs. For instance, if the mother/*dam* haplotypes on

three loci are AA and BB, then the probability of the transmitted AB gamete to her son is simply the recombination fraction between loci 1 and 2 divided by 2.

Now, haplotype inference is explained on a simple example. Assume a family of two parents and one offspring. The genotype of the father for two SNPs is AB AB (recall that the allele order in a pair doesn't matter), and the genotype of the mother is BB AA. The mother haplotypes are trivial (both haplotypes are BA); however the father has two possible sets of haplotypes (AA and BB, or AB and BA). Assume that one son has genotype AB AB. For this son, B in the first locus and A in the second came from the mother (because there is no other possibility) and they form a first haplotype BA. Thus, AB constitutes the other haplotype that came from the father. Now, if the recombination fraction between the two loci is less than 0.5 (roughly if they are on the same chromosome), probably the father transmitted haplotype AB with no recombination; thus, its haplotypes are AB and BA.

In Section 2, we present an efficient method to reconstruct the haplotypes of the sire from the information of its genotype and of its offspring genotypes in a *half-sib family* (each son has a different dam). This particular pedigree is common in livestock genetics for marker assisted selection. Further, once the sire haplotypes are reconstructed, and conditionally to this configuration, the haplotypes of its sons are easy to compute [7]. Assuming *linkage equilibrium* (i.e., random association of alleles at two or more loci) and equal allele frequencies at every locus, our method reformulates the likelihood of genotype data in a compact way, resulting in a binary weighted constraint satisfaction problem [11], which can be maximized later by a systematic search method or by a dynamic programming algorithm, exploiting the small treewidth of the resulting instances.

Section 3 gives experimental results on simulated and real datasets. In this study, we assumed no missing data (except the dams) and no erroneous genotypes. However, we could impute missing sire genotype data from its offspring, removing beforehand Mendelian errors [15].

2 Method

Assume a single half-sib family, the sire and its n descendants are genotyped at L loci but not the dams. Let $\mathbf{M}$ be a matrix such that $M^i_{l,1}, M^i_{l,2}$ are the observed genotype information of individual i ($i \in \{0,1\ldots,n\}$, with index 0 for the sire) at locus l ($l \in \{1,\ldots,L\}$) for its two alleles ($M^i_{l,j} \in \{A,B\}, j \in \{1,2\}$) with an arbitrary order. For convenience, in the following examples, the genotype of an individual i is given by a list of pairs of alleles, e.g., $\mathbf{M}^i =$ AB AA BA means $M^i_{1,1} = A, M^i_{1,2} = B, M^i_{2,1} = A, M^i_{2,2} = A, M^i_{3,1} = B, M^i_{3,2} = A$.

Let now define vector $\mathbf{h}$ ($h_l \in \{-1,1\}, l \in \{1,\ldots,L\}$) as the indicator of allele origin for the sire haplotypes. h_l has two possible states: $h_l = 1$ (resp. $h_l = -1$) if the first haplotype has allele $M^0_{l,1}$ (resp. $M^0_{l,2}$) and the second haplotype has allele $M^0_{l,2}$ (resp. $M^0_{l,1}$) at locus l. For instance, a sire genotype observed at three loci such that $\mathbf{M}^0 =$ AB AA BA and $\mathbf{h} = (1,1,-1)$ implies that the first sire haplotype is AAA and the second one is BAB. The problem to solve is to find the most probable assignment of $\mathbf{h}$ given the observed genotypes. Note that the assignment of h_l with homozygous sire locus l does not matter, it is set to 1.

Instead of using the observed genotypes in our probabilistic model directly, we will use an intermediate data that is sufficient to model meiosis events. Let $\mathbf{T}$ be a matrix such that indicator variable T^i_l, called the *transmission value*, defines the origin of the paternal allele at locus l ($l \in \{1,\ldots,L\}$) in the i-th descendant ($i \in \{1\ldots,n\}$). This origin is referred to the genotype information in the sire ($M^0_{l,1}, M^0_{l,2}$), not to its haplotypes. T^i_l has three possible states: $T^i_l = 1$ (resp. $T^i_l = -1$) if the paternal allele of the i-th descendant comes from the first allele $M^0_{l,1}$ (resp. the second allele $M^0_{l,2}$), or $T^i_l = \star$ if the origin of the paternal allele is unknown. Let $\mathbf{T}^i$ be the transmission vector $(T^i_1,\ldots,T^i_L)$.

Variable T_l^i is known with certainty (i.e., T_l^i is -1 or 1) if and only if the i-th descendant is homozygous and the sire is heterozygous at locus l. Otherwise, T_l^i is unknown ($T_l^i = \star$). Remember that the dam genotypes are assumed to be unknown. For example, if the descendant is AA and the sire BA, it is necessary that A in the descendant came from the second allele of the sire, so $T_l^i = -1$.

A locus l such that $T_l^i \neq \star$ is called an *informative locus* for the i-th descendant. A *preceding informative locus k of l* is the first informative locus found in the order from $l-1$ to 1. The set of pairs of consecutive informative loci is composed of all the pairs of informative loci with their corresponding preceding informative locus.

Example 1. *Consider a sire with three sons from three different dams. Only the sire and the sons are genotyped on seven loci such that* $\boldsymbol{M}$ *is given by:*

$\boldsymbol{M}^0$:	*AB*	*BB*	*AA*	*BA*	*BA*	*AA*	*AB*
$\boldsymbol{M}^1$:	*BB*	*BA*	*AA*	*AA*	*BB*	*AB*	*BB*
$\boldsymbol{M}^2$:	*BA*	*BB*	*AB*	*AA*	*BB*	*AA*	*AA*
$\boldsymbol{M}^3$:	*AA*	*BB*	*AA*	*AB*	*AA*	*AB*	*AA*

Construction of transmission vectors. *The sire is homozygous at loci* 2,3 *and* 6*, so for these loci the transmission value is* $\star$ *for each son. For the other loci, the sire is heterozygous, so we study the genotypes of the sons to complete the transmission vectors. We detail for son* 2*. At locus* 1*, the son is heterozygous as the sire so we do not know with certainty which of the alleles (*$M^0_{1,1}$ *or* $M^0_{1,2}$*) was transmitted:* $T_1^2 = \star$*. At locus* 4*, the son is AA and the sire is BA, so it is certain that the sire transmits the second allele (*$M^0_{4,2}$*) :* $T_4^2 = -1$*. At locus* 5*, the son is homozygous BB and the sire is BA, so it is certain that the sire transmits the first allele (*$M^0_{5,1}$*) :* $T_5^2 = 1$*. It is the same reasoning at locus 7.*
Finally matrix $\boldsymbol{T}$ *is equal to:*

$\boldsymbol{T}^1$:	*-1*	$\star$	$\star$	*-1*	*1*	$\star$	*-1*
$\boldsymbol{T}^2$:	$\star$	$\star$	$\star$	*-1*	*1*	$\star$	*1*
$\boldsymbol{T}^3$:	*1*	$\star$	$\star$	$\star$	*-1*	$\star$	*1*

The set of informative loci of son 1 *is* $\{1,4,5,7\}$ *and its set of pairs of consecutive informative loci is* $\{(1,4),(4,5),(5,7)\}$*. It is* $\{4,5,7\}$ *(resp.* $\{1,5,7\}$*) and* $\{(4,5),(5,7)\}$ *(reps.* $\{(1,5),(5,7)\}$*) for son* 2 *(resp. son* 3*).*

To summarize, $\mathbf{h}$ are the decision variables and $\mathbf{T}$ the observations used in our model. The posterior probability of the haplotypes is given by

$$p(\mathbf{h}|\mathbf{T}) = \frac{p(\mathbf{T}|\mathbf{h}).p(\mathbf{h})}{\sum_{\mathbf{h}'} p(\mathbf{T}|\mathbf{h}').p(\mathbf{h}')}$$

In the absence of prior information for $\mathbf{h}$, i.e., assuming *linkage equilibrium*, $p(\mathbf{h}|\mathbf{T}) \propto p(\mathbf{T}|\mathbf{h})$ and the most likely haplotype configuration is the one that maximizes $p(\mathbf{T}|\mathbf{h})$.

Because meiosis events producing each descendant are independent,

$$p(\mathbf{T}|\mathbf{h}) = \prod_{i=1}^{n} p(\mathbf{T}^i|\mathbf{h})$$

Applying the chain rule, we obtain also

$$p(\mathbf{T}^i|\mathbf{h}) = \prod_{i=1}^{n} p(T_1^i|\mathbf{h}).p(T_2^i|\mathbf{h},T_1^i).p(T_3^i|\mathbf{h},T_1^i,T_2^i)\dots p(T_L^i|\mathbf{h},T_1^i,\dots,T_{L-1}^i)$$

These probabilities are defined in an iterative way starting from $l=1$. For the first position, $p(T_1^i|\mathbf{h}) = 0.5$ if T_1^i equals to either -1 or 1 (i.e., $p(\mathbf{T}|\mathbf{h}) = p(\mathbf{T}|-\mathbf{h})$), and $p(T_1^i|\mathbf{h}) = 1$ if $T_1^i = \star$. The same applies for a series of $\star$'s up to the first $T_l^i \neq \star$. For any next position, two cases can be distinguished. For $T_l^i = \star$, $p(T_l^i|\mathbf{h}, T_1^i, \ldots, T_{l-1}^i) = 1$ because it is a complete set of events. For $T_l^i \neq \star$, assuming *no interference in the formation of crossing-overs* and *equal allele frequencies at each locus*, only the current informative locus l and the preceding informative locus k of l are used, $p(T_l^i|\mathbf{h}, T_1^i, \ldots, T_{l-1}^i) = p(T_l^i|h_k, h_l, T_k^i)$ (if l is the first informative locus, $p(T_l^i|\mathbf{h}, T_1^i, \ldots, T_{l-1}^i) = 0.5$). This is so because, assuming independence of crossing-over, the probability of recombination between k and l does not depend on the presence or not of previous recombinations between 1 and k. And because any $\star$ between $k+1$ and $l-1$ does not modify the likelihood, assuming all SNPs have equal allele frequencies, and so, transmitted alleles from the dams to their sons do not matter. Thus, only informative loci (transmission values) in $\mathbf{T}^i$ are used.

Let r_{kl} denote the recombination fraction between k and l, obtained by the Haldane mapping function from the known marker map ($r_{kl} \in [0, 0.5]$). A pair of alleles placed on the same chromosome in the sire at locus k and l will be transmitted together (no recombination) with a probability $1 - r_{kl}$; the opposite (transmitted alleles come from a recombination between homologous chromosomes) occurs with frequency r_{kl}.

Thus, $p(T_l^i|h_k, h_l, T_k^i) = (1 - r_{kl})$ in two cases: if $T_l^i = T_k^i$ and $h_l = h_k$, or if $T_l^i \neq T_k^i$ and $h_l \neq h_k$. Both indicate the same sire haplotype origins for these two loci in the i-th descendant. In any other case (different origins), $p(T_l^i|h_k, h_l, T_k^i) = r_{kl}$. An algebraic form of $p(T_l^i|h_k, h_l, T_k^i)$ is $r_{kl}^{1-a} \times (1 - r_{kl})^a$, where a measures the same origin ($a = 1$) or not ($a = 0$). We have $a = a_{kl}^i(\mathbf{h}) = \frac{1}{2} + \frac{1}{2}\frac{h_k h_l}{T_k^i T_l^i}$.

The log-likelihood of $\mathbf{h}$ can be expressed as

$$\begin{aligned} V =& \log\left[p(\mathbf{T}|\mathbf{h})\right] = \sum_{i=1}^{n} \log\left[p(\mathbf{T}^i|\mathbf{h})\right] = \sum_{i=1}^{n}\sum_{l=1}^{L} \log\left[p(\mathbf{T}_l^i|\mathbf{h}, T_1^i, \ldots, T_{l-1}^i)\right] \\ =& n\log\left(\frac{1}{2}\right) + \sum_{i=1}^{n}\sum_{l\in I_i}\left[\left(1 - a_{kl}^i(\mathbf{h})\right)\log(r_{kl}) + a_{kl}^i(\mathbf{h})\log(1 - r_{kl})\right] \end{aligned} \tag{1}$$

where I_i is the set of informative loci for the i-th descendant (except the first informative locus the contribution of which is $\log(\frac{1}{2})$), and k the preceding informative locus of l. A rewriting of equation 1 as a quadratic form in $\mathbf{h}$ allows a sparse representation, which is computationally easier to manipulate:

$$V = K + \sum_{l=1}^{L}\sum_{k<l} \frac{1}{2} h_k h_l \log\left(\frac{1 - r_{kl}}{r_{kl}}\right) \sum_{i\in\{1,n\} \text{ s.t. } (k,l)\in F_i} \frac{1}{T_k^i T_l^i} \tag{2}$$

where $K = n\log\left(\frac{1}{2}\right) + \sum_{i=1}^{n}\sum_{l\in I_i}\frac{1}{2}\log\left[(1 - r_{kl})r_{kl}\right]$ and F_i is the set of pairs of consecutive informative loci in the i-th descendant.

Therefore V can be expressed as a quadratic form: $V = K + \mathbf{h}'\mathbf{W}\mathbf{h}$ with a symmetric $L \times L$ matrix $\mathbf{W}$ such that

$$W_{ll} = 0 \text{ and } W_{kl} = W_{lk} = \frac{1}{4}\log\left(\frac{1 - r_{kl}}{r_{kl}}\right) \sum_{i\in\{1,n\} \text{ s.t. } (k,l)\in F_i} \frac{1}{T_k^i T_l^i}$$

Let N_{kl}^+ (respectively N_{kl}^-) be the number of descendants such that each descendant i has $T_l^i = T_k^i$ (resp. $T_l^i \neq T_k^i$) and (k,l) is a pair of consecutive informative loci for this descendant.
Finally,

$$W_{kl} = \frac{1}{4}(N_{kl}^+ - N_{kl}^-)\log\left(\frac{1 - r_{kl}}{r_{kl}}\right) \tag{3}$$

Example 2. *Consider Example 1, we now compute N_{kl}^+ and N_{kl}^- for every pair of consecutive informative loci occuring in at least one descendant. We obtain $N_{1,4}^+ = 1$ due to son 1 ($T_1^1 = T_4^1$), $N_{1,4}^- = 0$, $N_{1,5}^+ = 0$,*

$N^-_{1,5} = 1$ due to son 3 ($T^3_1 \neq T^3_5$), $N^+_{4,5} = 0$, $N^-_{4,5} = 2$ due to sons 1 ($T^1_4 \neq T^1_5$) and 2 ($T^2_4 \neq T^2_5$), $N^+_{5,7} = 1$ due to son 2 ($T^2_5 = T^2_7$), and finally, $N^-_{5,7} = 2$ due to sons 1 ($T^1_5 \neq T^1_7$) and 3 ($T^3_5 \neq T^3_7$). Others N^+_{kl} and N^-_{kl} are all equal to zero.

2.1 Weighted constraint satisfaction formulation

This quadratic form can be directly translated into a *binary Weighted Constraint Satisfaction Problem* (WCSP) [11].

A binary WCSP is a pair (X,F) where $X = \{1,\ldots,m\}$ is a set of m variables and F a set of binary cost functions. Each variable $i \in X$ has a finite domain D_i of values than can be assigned to it. A binary cost function $f_{ij} \in F$ is a function $f_{ij} : D_i \times D_j \mapsto \mathcal{N}$ where $\mathcal{N}$ is the set of non-negative integers. The *constraint graph* of a binary WCSP is a graph $G = (X,E)$ with one vertex for each variable and one edge $(i,j) \in E$ for every cost function $f_{ij} \in F$.

The weighted constraint satisfaction problem is to find a complete assignment t of all the variables minimizing the total cost function $\sum_{f_{ij} \in F} f_{ij}(t[i],t[j])$ where $t[i]$ is the value assigned to variable i in t. This problem is NP-hard.

State-of-the-art WCSP exact (i.e., complete) solving methods are either *Depth-First Branch and Bound* (DFBB) exploiting local consistency techniques [11] or dynamic programming algorithms such as *bucket elimination*, aka *Variable Elimination* (VE) [5] or a combination of both approaches such as *Backtrack with Tree Decomposition* (BTD) [9, 3].

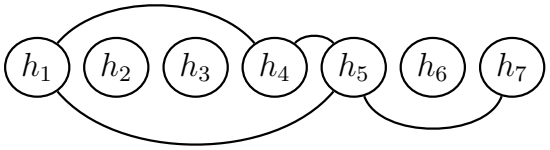
We have the following WCSP formulation of our haplotyping problem. We define $X = \{1,\ldots,L\}$ the set of $m = L$ variables with domain $D_i = \{-1,1\}, i \in \{1,\ldots,L\}$. Each WCSP variable i corresponds to a decision variable h_i of our problem. The set of binary cost functions is defined by $F = \{f_{kl} | W_{kl} \neq 0, k < l\}$. Each cost function f_{kl} represents two terms $-W_{kl}$ and $-W_{lk}$ of the symmetric matrix $\mathbf{W}$ ($W_{kl} = W_{lk}$) in the quadratic form $\mathbf{h}'\mathbf{W}\mathbf{h}$ (we use opposite terms for minimization). Because cost functions must be positive, where as $-W_{kl}$ may be negative, a constant term $2|W_{kl}|$ is added to each cost function[1]. Thus, $f_{kl}(h_k,h_l) = -2W_{kl}h_kh_l + 2|W_{kl}|$, or equivalently,

$$\begin{array}{rl} f_{kl}(-1,-1) \\ f_{kl}(1,1) \end{array} = \begin{cases} -4W_{kl} & \text{if } W_{kl} < 0 \\ 0 & \text{otherwise} \end{cases} \qquad \begin{array}{rl} f_{kl}(-1,1) \\ f_{kl}(1,-1) \end{array} = \begin{cases} 4W_{kl} & \text{if } W_{kl} > 0 \\ 0 & \text{otherwise} \end{cases}$$

Notice that these functions are soft versions of disequality (if $W_{kl} < 0$) and equality (if $W_{kl} > 0$) constraints.

For any complete assignment $\mathbf{h}$, we have $\sum_{f_{kl} \in F} f_{kl}(h_k,h_l) \propto V$ (see Equation 2). Thus, an optimal solution of WCSP (X,F) corresponds to the most likely haplotype configuration.

Consider Example 1 again, the constraint graph is given below.



2.2 Extension to the case of genotyped dams

In the case we know the genotypes of the dams, then we can take into account this extra information, by extending our definition of transmission values (without changing anything else). Variable T^i_l is known

[1] In order to get integer costs, we also multiply each cost function by a sufficiently large number and take the smallest following integer, such that it does not change the set of optimal solutions.

with certainty if and only if the i-th descendant is homozygous and the sire is heterozygous at locus l *or* the i-th descendant and the sire are heterozygous and the dam is homozygous at locus l (i.e., T_l^i is -1 or 1); otherwise T_l^i is unknown ($T_l^i = \star$). For example, if the descendant is AB, the sire BA and the dam AA, it is necessary that B in the descendant came from the first allele of the sire, and thus $T_l^i = 1$.

Example 3. *Consider Example 1 again, we add the information of genotyped dams. Let* $\boldsymbol{M}^{di}$ *be the genotypes of the i-th dam of the i-th descendant.*

$\boldsymbol{M}^{d1}$:	*AA*	*AB*	*AB*	*BB*	*AA*	*BA*	*AB*	$\boldsymbol{T}^1$:	*-1*	$\star$	$\star$	*-1*	*1*	$\star$	*-1*
$\boldsymbol{M}^{d2}$:	*AB*	*BA*	*BB*	*AB*	*AB*	*AA*	*AB*	$\boldsymbol{T}^2$:	$\star$	$\star$	$\star$	*-1*	*1*	$\star$	*1*
$\boldsymbol{M}^{d3}$:	*AA*	*BA*	*AB*	*AA*	*AB*	*BA*	*AA*	$\boldsymbol{T}^3$:	*1*	$\star$	$\star$	***1***	*-1*	$\star$	*1*

For son 2 *at locus* 1*, the transmitted allele from the sire is not identifiable (the son, sire, and dam are heterozygous). So,* T_1^2 *is still equal to* $\star$*. For son* 3 *at locus* 4*, the transmitted allele from the sire can be identified: the son and sire are heterozygous AB and BA respectively, and the dam is homozygous AA, so it is certain that the dam transmitted allele A and the sire transmitted allele B, and thus* $T_4^3 = 1$*. For this example, it is the only modification of the transmission values with respect to Example 1.* N^+ *and* N^- *are kept unchanged, except for* $N_{1,4}^+ = 2$*, due to sons* 1 *(*$T_1^1 = T_4^1$*) and* 3 *(*$T_1^3 = T_4^3$*). Finally, the constraint graph is the same as in Example 1.*

3 Experimental Results

3.1 Datasets and methods

A first dataset[2] consists of half-sib families which were simulated by considering either linkage disequilibrium at the sire/dams or not. In the former case, disequilibrium was generated first by simulating a Wright-Fisher scenario with 100 individuals mating at random during 100 generations; the sires and the dams haplotypes were sampled from the last generation. In both cases, the founders were simulated in linkage equilibrium and using a Beta distribution ($\alpha = 2, \beta = 2$) of allele frequency similar to the one observed in bovine livestock. Recombination events on a single chromosome of $S \in \{1, 2\}$ Morgan were simulated using Haldane's mapping function, producing sons haplotypes. Genotypes were obtained by randomly permuting the two alleles at every locus of every pair of haplotypes. The number of SNPs L varied from 100 to 10000. These markers were evenly-spaced on the chromosome. The number of descendants n varied from 1 to 1000. 50 families were simulated for each set of parameters.

A second dataset[2] was built in the same way, but taking 44 real haplotypes of the father chromosome X in 44 trios of CEU population (see HAPMAP phase 3 release 2 project at `www.hapmap.org`) as initial sire/dams haplotypes. Because only 1 copy of chromosome X is present in males, its haplotype is known with certainty. This dataset provides a real pattern of linkage disequilibrium, contrary to simulated datasets. We selected $L = 36000$ SNPs such that for each locus the two alleles occured in our data. These markers were evenly-spaced on the chromosome of $S = 1.64$ Morgan.

Five haplotyping methods/softwares were studied. Exact methods are `Merlin` [1] version 1.1.2 ; `Superlink` [8] version 1.6 ; and our approach implemented in WCSP solvers `toulbar2` version 0.9.2 (for DFBB [4] used by default, and BTD [3]) and `toolbar` version 3.1 (for VE [5])[3]. Approximate methods are `W&M` [18] (implemented by us in R language) and `LinkPhase` [7] with parameters recommended by the authors and with unreconstructed loci fixed arbitrarily in a post-processing step. All the tested

[2] `carlit.toulouse.inra.fr/cgi-bin/awki.cgi/HaplotypeInference`

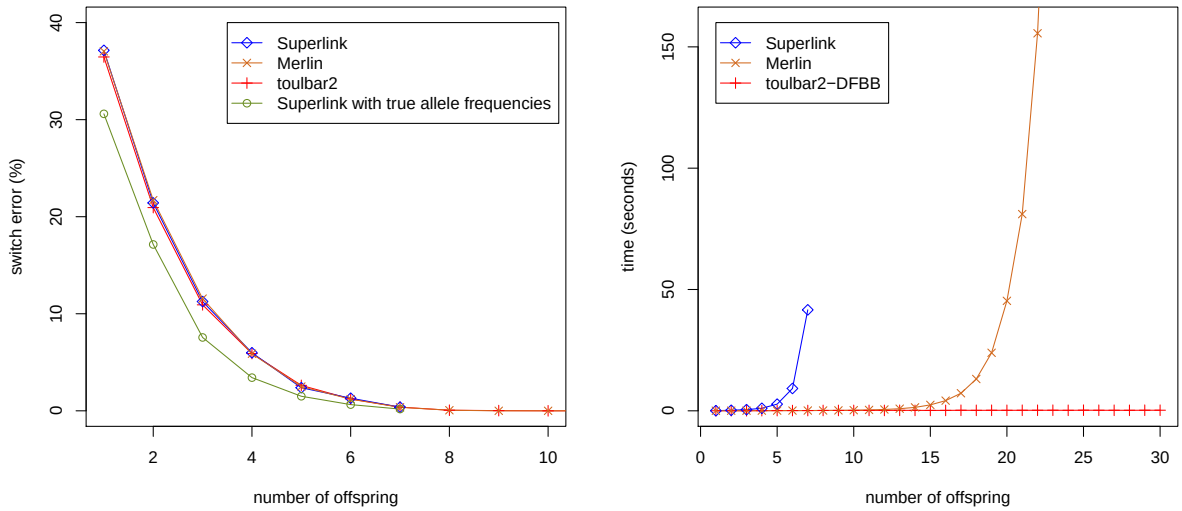
[3] `carlit.toulouse.inra.fr/cgi-bin/awki.cgi/ToolBarIntro`

methods except ours reconstruct all the individuals haplotypes. However, knowing the sire haplotypes, it is easy to find the most probable haplotypes for each son and its dam in linear time $O(L)$.

The experimentations were performed on a 2.6GHz Intel Xeon computer running Linux 2.627-11-server with 64 GB. These methods were compared in terms of the percentage of switch error [16], which measures the proportion of heterozygous loci whose allele origin (first or second sire haplotype) is wrongly inferred relative to the previous heterozygous locus ; and the CPU solving time in seconds. Reported results are mean over 50 families.

3.2 Comparison with exact methods

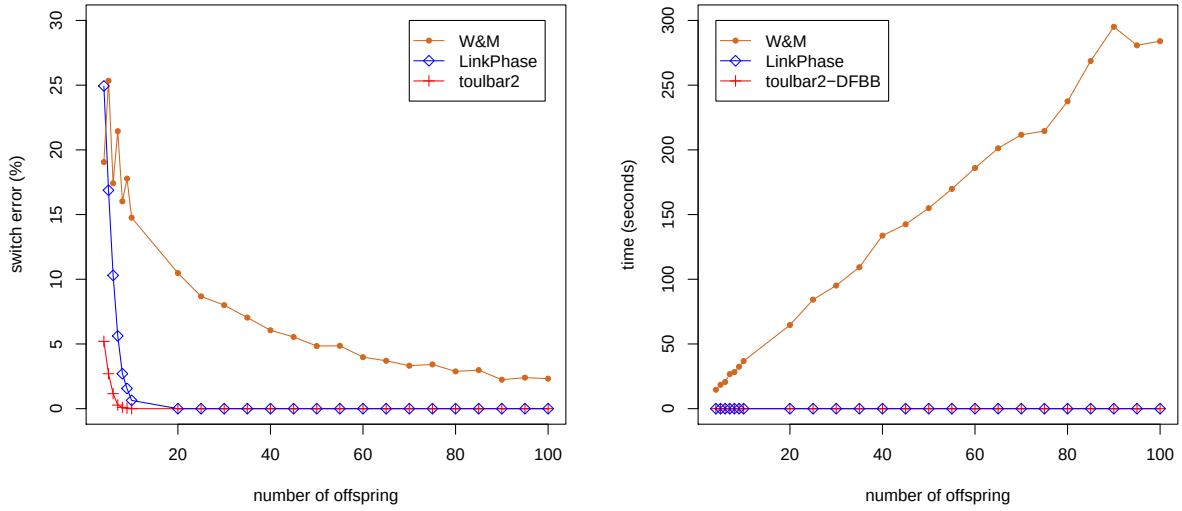
We compared our approach `toulbar2` with two exact methods, `Merlin` [1] and `Superlink` [8], varying the number of descendants in the first dataset without linkage disequilibrium and without genotyped dams. Figure 1(a) shows experimentally that these three methods find the same optimal sire haplotype configuration if we assumed all SNPs have equal allele frequencies for all the methods (option *-fe* in `Merlin` and given as input in .DAT file for `Superlink`)[4]. The switch error (Fig. 1(a)) decreases rapidly with the number of descendants. It is less than 1% (resp. 6%) for $n = 7$ (resp. 4) descendants. If we provide the true allele frequencies, `Superlink` found better sire haplotypes for small families and `Merlin` did not improve its results (because it does not fully reconstruct ungenotyped dam haplotypes). `Merlin` and `Superlink` are dynamic programming algorithms which have their time and space complexity increasing exponentially with the number of descendants. `Superlink` ran out of memory for more than 7 descendants. `Merlin` took more than 150 seconds for 22 descendants whereas `toulbar2` using default depth-first branch and bound (DFBB) took less than a second (Fig. 1(b)).



(a) (b)

Figure 1: Comparison with exact methods for $S = 1, L = 1500, n \in [1, 30]$.

[4] In fact, there may be several optimal solutions and each method can find a different one resulting in small differences in terms of switch error (`Merlin` may output two solutions and we took the first one in our results).



(a) (b)

Figure 2: Comparison with approximate methods for $S = 1, L = 1500, n \in [4, 100]$.

3.3 Comparison with approximate methods

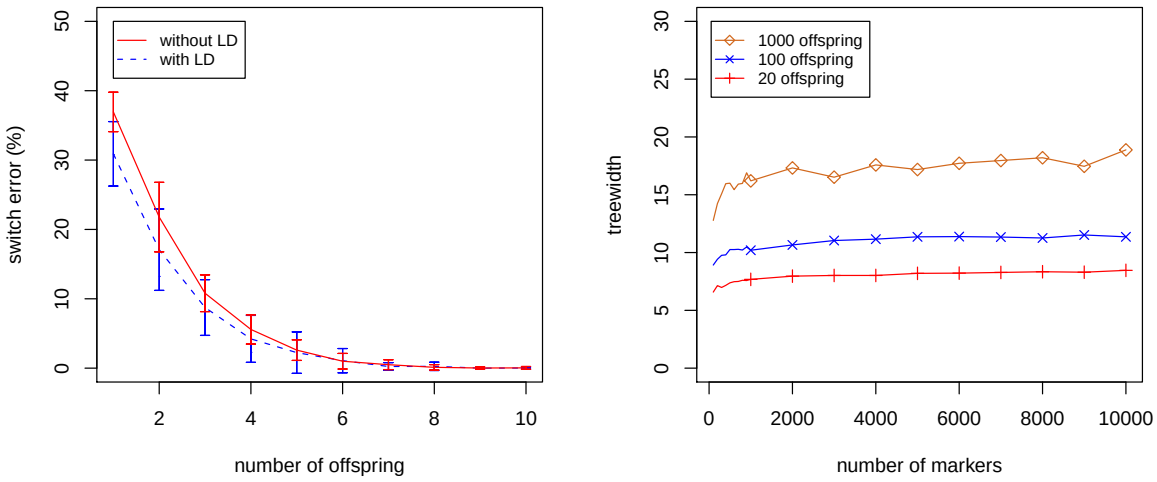
We compared `toulbar2` with two approximate methods, `W&M` [18] and `LinkPhase` [7], on the same dataset as in the previous section. `LinkPhase` required families of about twenty individuals to reconstruct entirely the sire haplotypes and to find the true haplotypes (Fig. 2(a)). For instance, with 4 descendants, `LinkPhase` did not reconstruct one third of the heterozygous loci; instead, `toulbar2` reconstructed all the sire haplotypes with 73% less of switch errors compared to `LinkPhase`. Moreover, `toulbar2` is guaranteed to find an optimal haplotype configuration. The convergence of `W&M` towards the true haplotypes was much slower compared to the two other methods. Furthermore, while `LinkPhase` and `toulbar2` (DFBB) solved every family within one second, `W&M` computing time grew linearly with the number of descendants (Fig. 2(b)).

3.4 Comparison with and without linkage disequilibrium

If we consider linkage disequilibrium, the mean switch error (Fig. 3(a)) is slightly better than without linkage disequilibrium, but the variance is much higher. This phenomenon may be due to the reduced number of different (sire and dams) haplotypes, resulting in less heterozygous markers ($\approx 40\%$ less than wout LD).

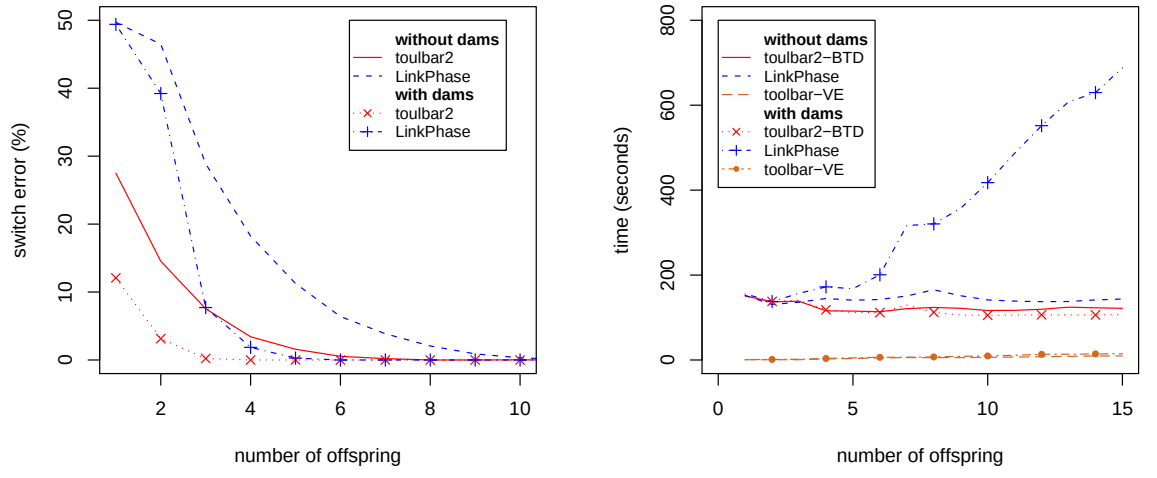
3.5 Study of the treewidth of our WCSP formulation

In order to assess the difficulty of the resulted WCSP instances of our first dataset (without linkage disequilibrium), we measured the treewidth of their constraint graph [2]. The treewidth of a graph gives an idea of its acyclicity (a tree as a treewidth of 1). Dynamic programming algorithms (VE [5] and BTD [3], but not DFBB) exploiting the WCSP formulation have their time and space complexity exponential in the treewidth (DFBB being exponential in the number of variables). Figure 3(b) shows the average treewidth obtained by a variable elimination order following the chromosome order. We noticed the



(a) (b)

Figure 3: (a) Comparison with/w. out linkage disequilibrium for $S = 1, L = 1500, n \in [1,10]$ using `toulbar2`. (b) Constraint graph analysis of our WCSP formulation ($S = 2$).



(a) (b)

Figure 4: Human chr. X dataset w/wout genotyped dams ($L = 36000, n \in [1,15]$).

treewidth remains relatively small (the maximal treewidth found in all our simulations was 30) and it seems to increase logarithmically with the number of individuals and the number of markers. We can conclude that the resulting WCSP instances are easy to solve by any dynamic programming algorithm. Therefore, we used VE and BTD instead of DFBB on very large datasets as done in the next section.

3.6 Comparison with and without genotyped dams on human chromosome X dataset

Using our second real dataset, we found the switch error was less than 1% (resp. 4%) for $n = 6$ (resp. 4) descendants (Fig. 4(a)), which is similar to our first dataset. By exploiting the additional information of genotyped dams, only 3 descendants are needed to reconstruct the sire haplotypes with less than 1% of switch error. Concerning performance (Fig. 4(b)), `toolbar` VE and `toulbar2` BTD performed similarly with or without the genotyped dams, although VE was much faster but needed more space than BTD. On the contrary, `LinkPhase` time increased linearly with the number of descendants in the case of genotyped dams. The treewidth was 11 in average.

4 Conclusion

In this paper, we have proposed a sparse representation (with a small treewidth) of sire haplotype reconstruction in half-sib families and a method which finds an optimal haplotype configuration. This method obtained good results, in terms of accuracy and time, on simulated and real datasets.

In the future, we will improve our results for small families with linkage disequilibrium and study other kinds of pedigrees.

References

[1] G. Abecasis, S. Cherny, W. Cookson, and L. Cardon. Merlin – rapid analysis of dense genetic maps using sparse gene flow trees. *Nature Genetics*, 30:97–101, 2002.

[2] H. Bodlaender. Discovering treewidth. In *Theory and Practive of Computer Science - SOFSEM'2005*, pages 1–16, 2005.

[3] S. de Givry, T. Schiex, and G. Verfaillie. Exploiting Tree Decomposition and Soft Local Consistency in Weighted CSP. In *Proc. of AAAI-06*, Boston, MA, 2006.

[4] S. de Givry, M. Zytnicki, F. Heras, and J. Larrosa. Existential arc consistency: Getting closer to full arc consistency in weighted CSPs. In *Proc. of IJCAI-05*, pages 84–89, Edinburgh, Scotland, 2005.

[5] Rina Dechter. Bucket elimination: A unifying framework for reasoning. *Artificial Intelligence*, 113(1–2):41–85, 1999.

[6] K. Doi, J. Li, and T. Jiang. Minimum recombinant haplotype configuration on tree pedigrees. In *WABI'03*, pages 339–353, 2003.

[7] T. Druet and M. Georges. A Hidden Markov Model Combining Linkage and Linkage Disequilibrium Information for Haplotype Reconstruction and Quantitative Trait Locus Fine Mapping. *Genetics*, 184, 2010.

[8] M. Fichelson, N. Dovgolevsky, and D. Geiger. Maximum Likelihood Haplotyping for General Pedigrees. *Human Heredity*, 59(1):41–60, 2005.

[9] P. Jégou and C. Terrioux. Hybrid backtracking bounded by tree-decomposition of constraint networks. *Artificial Intelligence*, 146:43–75, 2003.

[10] S. Knott, J.-M. Elsen, and C. Haley. Methods for multiple-marker mapping of quantitative trait loci in half-sib populations. *Theoretical and Applied Genetics*, 93(1-2):71–80, 1996.

[11] J. Larrosa and T. Schiex. Solving weighted CSP by maintaining arc consistency. *Artificial Intelligence*, 159(1-2):1–26, 2004.

[12] S. Lauritzen and N. Sheehan. Graphical Models for Genetic Analyses. *Statistical Science*, 18(4):489–514, 2003.

[13] T. Niu. Algorithms for inferring haplotypes. *Genetic Epidemio.*, 27:334–347, 2004.

[14] D. Qian and L. Beckmann. Minimum-Recombinant Haplotyping in pedigrees. *American journal of human genetics*, 70(6):1434–1445, 2002.

[15] M. Sanchez, S. de Givry, and T. Schiex. Mendelian error detection in complex pedigrees using weighted constraint satisfaction. *Constraints*, 13(1):130–154, 2008.

[16] M. Stephens and P. Donnelly. A comparison of bayesian methods for haplotype reconstr. from population genotype data. *Am J Hum Genet*, 73:1162–1169, 2003.

[17] E. Wijsman, J. Rothstein, and E. Thompson. Multipoint linkage analysis with many multiallelic or dense diallelic markers: MCMC provides practical approaches for genome scans on general pedigrees. *Am J Hum Genet*, 79:846–858, 2006.

[18] J. Winding and T. Meuwissen. Rapid haplotype reconstruction in pedigrees with dense marker maps. *J. of Animal Breeding and Genetics*, 121:26–39, 2004.