Reconstructing Sibling Relationships from Microsatellite Data

Saad Sheikh[1], Tanya Y. Berger-Wolf[1], Wanpracha Chaovalitwongse[2], Bhaskar DasGupta[1], and Mary V. Ashley[3]

[1] Department of Computer Science, University of Illinois at Chicago, 851 South Morgan Street, Chicago, IL 60607, {tanyabw,ssheikh,dasgupta@cs.uic.edu}
[2] Department of Industrial Engineering, Rutgers University, PO Box 909, Piscataway, NJ 08855, wchaoval@rci.rutgers.edu,
[3] Department of Biological Sciences, University of Illinois at Chicago, 840 West Taylor Street, Chicago, IL 60607, ashley@eeb.uic.edu

1 Introduction

New technologies for collecting genotypic data from natural populations open the possibilities of investigating many fundamental biological phenomena, including behavior, mating systems, heritabilities of adaptive traits, kin selection, and dispersal patterns. The power and potential of genotypic information often rests in our ability to reconstruct genealogical relationships among individuals. These relationships include parentage, full and half-sibships, and higher order aspects of pedigrees [5, 6, 10]. In this paper we are only concerned with full sibling relationships.

Some areas of genealogical inference, such as parentage, have been studied extensively [10]. While several methods for sibling reconstruction have been proposed [13, 8, 2, 1, 4, 12, 14], most have not been 'ground-truthed' (but see [6]) and have received relatively limited application. We build on our earlier work [3, 7] and propose a new algorithm for sibship reconstruction using combinatorial optimization. There have been no truly combinatorial methods for kinship reconstruction problems [2, 4]. Combinatorial methods have been very successful in closely related molecular genetics questions, such as haplotype reconstruction [9, 11]. Our approach uses the simple Mendelian inheritance rules to impose constraints on the genetic content possibilities of a sibling group. We formulate the inferred combinatorial constraints and, under the parsimony assumption, use a provably correct algorithm to construct the smallest number of groups of individuals that satisfy these constraints. We test our approach on both simulated and real biological data.

1.1 Problem Statement

We now define the sibling reconstruction problem. Given a genetic (microsatellite) sample from a population of n diploid individuals of the same generation, U, the goal is to reconstruct the full sibling groups (groups of individuals with the same parents). We assume no knowledge of parental information.

Formally, we are given a set U of n individual microsatellite samples from l genetic loci

$$U = \{X_1, ...X_n\}, \text{ where } X_i = (< a_{i1}, b_{i1} >, ..., < a_{il}, b_{il} >)$$

and a_{ij} and b_{ij} are the two alleles of the diploid individual i at locus j. The goal is to find a partition of individuals $P_1, ...P_m$ such that

$$\forall 1 \leq k \leq m, \ \forall X_u, X_v \in P_k : \ Parents(X_u) = Parents(X_v)$$

Notice, here that we have not defined the function $Parents(x)$. This is a biological objective. We will discuss computational approaches to achieve a good estimate of the biological sibling relationship.

1.2 2-Allele Property

Mendelian genetics introduce two overlapping necessary (but not sufficient) constraints on full siblings groups: 4-allele property and 2-allele property [3]. The 4-allele property states that a group of siblings can have no more than 4 alleles at any locus and the 2-allele property takes into account the fact that an individual must inherit one allele from each parent. It was proven in [3] that the 2-allele property is equivalent to a simple constraint that we restate here without the proof.

Theorem 1 (Berger-Wolf et al [3]). *Let R be the number of alleles that are homozygous or appear with 3 other distinct alleles and A be the total number of distinct alleles at a locus. Then $A + R \leq 4$.*

The constraint above reduces the possible combinations of alleles at a locus in a group of siblings down to a few canonical options. Assuming the alleles are numbered 1 through 4, Table 1 lists all different types of sibling groups possible with the 2-allele property. We do this by listing all possible pairs of parents whose alleles are among 1,2,3, and 4 and all the offspring they can produce. However, in any sibling group with a given set of parents only a subset of the offspring types from the table may be present.

2 Minimum 2-Allele Set Cover Algorithm

We now present our algorithm for solving the sibling reconstruction problem. As we have mentioned, the biological function $Parents(x)$ cannot be defined mathematically. We model the objective of reconstructing the sibling relationships by assigning individuals parsimoniously into the smallest number of (possibly overlapping) groups that satisfy the necessary 2-allele constraint. To achieve this, our algorithm uses the 2-allele and 4-allele properties (specifically, Table 1 see Appendix A) to generate all maximal potential sibling sets. We then restate the problem as a minimum set cover to find the minimum number of sibling sets containing all the individuals.

Data Structures.

An *ItemSet* (a potential sibling group) is a set of individuals. It maintains the individuals assigned to a potential sibling group as well as the corresponding set of canonical possibilities at each locus from Table 1. *ItemSet*s are maintained uniquely (using a hash table) in *ItemSetsMap*. *ItemSetsMap* implicitly tracks what can *possibly* be accommodated in the set without changing the corresponding canonical possibilities from Table 1. This is done by comparing the *AlleleMap* and *PossibilitiesSet* at each locus. *AlleleMap* is a mapping of alleles encountered at each locus onto numbers 1 through 4. *PossibilitiesSet* is the set of entries from Table 1 with which the mapped alleles at the locus conform.

Algorithm 2-allele.

It can be shown that any pair of individuals necessarily satisfies the 2-allele property. Thus, initially we use all $\binom{n}{2}$ pairs of n individuals to generate the candidate sets in the form of *ItemSet*s. Each *ItemSet* is generated with the initial possible canonical sets from Table 1 for each locus j. Each allele is assigned a number between 1 and 4 based on the order of its occurrence and stored in the *AlleleMap*. Then, for each pair of individual alleles we search for all matching canonical sets in Table 1 to determine the $PossibilitiesSet_j$. To avoid duplicate potential sets we query *ItemSetsMap* hash table. After generating the initial sets based on pairs of individuals, the algorithm repeatedly iterates through all the individuals, testing each set for a possible assignment of the individual to the set. In each iteration, only the sets that were present at the beginning of the cycle are considered for each individual. An individual is assigned to a set if its alleles match the possibilities of the set as defined by the extended Table 1 (see Appendix A).

The validity of the new *ItemSet* is determined by the Table 1. The alleles at every locus of the new individual must match at least one of the canonical patterns that collectively satisfy all the previous individuals assigned to the set. Once we determine that the *ItemSet* can be expanded (and its set of possible matching parents reduced) to accommodate the new individual in a valid way, we create a modified copy of the set. The individual is then checked against this new set for all the remaining loci. After we have verified that the new individual does not violate the 2-allele property of the new *ItemSet* at every locus, *ItemSetsMap* is checked to verify that the new set has not been previously created and the set is finally added to the *ItemSetsMap*. However, for the remainder of the iteration cycle all the individuals are checked only against the sets that were present in the *ItemSetsMap* at the beginning of the cycle. This ordering ensures that each individual is checked against each *ItemSet* that could accommodate it exactly once.

We repeat this process, cycling through all the individuals in the population. Once a set present at the beginning of the cycle has been inspected against all the individuals, the set is marked as *done* and is not revisited. This ensures that all sibling pairs that could possibly occur are evaluated, and that no sibling sets are generated that never occur in data.

The cycles of iterations over the individuals continues until all sets are marked as *done*. As the last step a set for each of the elements is added containing just that element.

After all the potential sibling sets are generated we apply the minimum set cover to find the minimum number of sibling groups whose union contains all the individuals. While the minimum set cover problem is NP-complete, modern mixed integer program solvers can solve it to optimally in most instances. Thus, it is not meaningful to discuss the theoretical computational complexity of the algorithm.

In order to solve set cover we use a standard integer programming solver, MIP solver CPLEX 9.0 [1] to solve the set cover problem to optimality.

[1] CPLEX is a registered trademark of ILOG

3 Results

We now present the experimental results of the evaluation of the 2-allele algorithm on simulated and biological data. We generated simulated data using the methodology from [3].

3.1 Biological Data

We use *Scaptodrosophila hibisci* dataset [15] for testing our algorithm on real data. The sibling groups for this dataset were reconstructed optimally, with 0 error. This accurate reconstruction despite the some missing data shows the robustness of our approach. However, more tests are necessary to accurately assess the sensitivity of the algorithm to missing and mistyped data.

3.2 Random Data

Random uniform data provides a baseline for the accuracy estimate of our algorithm. With the given number of alleles, loci, and offspring, the uniform distribution is the worst case scenario for the set cover algorithm. The uniform distribution of alleles makes many sets equivalent for the set cover problem. Therefore, the optimal solution may contain one of the equivalent sets rather than the "true" set. Any addition of structure to the genetics of the set of individuals only improves the performance of the algorithm.

We measure the reconstruction error of the 2-allele algorithm as the function of the number of loci, alleles per each locus, juvenile population size, and the maximum family size. First, we compare the error rate of the 2-allele algorithm to that of the 4-allele algorithm. Figure 1 shows selected results of the comparison. As expected, the 2-allele algorithm is more accurate than the 4-allele algorithm. In most cases, the error rate is reduced by about 10 percent, achieving a maximum of about 30 percent for families of size 5. The improvement is better in the lower ranges of the number of offspring per female since those are the parameter value ranges where the 4-allele algorithm did not perform well. No other clear trend of the accuracy improvement as a function of any data parameters emerges. Thus, we expect that the overall trends of the 4-allele and the 2-allele algorithms are similar, while the latter is more accurate.

Figure 2 (Appendix B) shows representative results for the accuracy of the 2-allele algorithm. As is the case with the 4-allele algorithm, the main factor for the decrease of the error of the 2-allele algorithm is the increase of the number of offspring per female. Similar to the 4-allele algorithm, the error rate decreases slightly as the number of alleles per locus increases. The number of sampled loci for the random data is not a strong factor in the accuracy of reconstruction. However, we believe that for real biological data where the allele distributions within a locus are much more structured, the number of sampled loci will be important. As the total number of juveniles increases, while the number of adults and the maximum family size remain the same, many more of the juveniles are half-siblings sharing many of their alleles. Thus, when the rest of the parameters are fixed, as the number of juveniles increases the error rate increases as well.

4 Conclusions

We have proposed a new combinatorial algorithm for the problem of reconstruction of sibling relationships from single generation genetic data. We have implemented and tested our approach on both real biological and simulated data. The simulated data provides a base line for the accuracy estimate of our algorithm, with real biological data likely to have better reconstruction accuracy.

The main advantage of the combinatorial approach is its lack of reliance on a priori knowledge about various population parameters, such as allele frequency and the degree of inbreeding. However, Mendelian constraints do not provide any basis for distinguishing between family groups when the groups are small, or when individuals share many common alleles. Additional information, such as *relative* allele frequency within the sample can be easily added to generate combinatorial constraints on the potential sibling sets. Unlike likelihood methods, combinatorial approaches use that information only for comparison purposes, and do not require a background data model or an accurate estimate value for any of the parameter. Thus, we believe that combinatorial approaches are particularly appropriate for analysis of natural animal and plant populations where background information is difficult to obtain.

Acknowledgments

This research is supported by the following grants: NSF IIS-0612044 (Berger-Wolf, Ashley, Chaovalitwongse, DasGupta), Fullbright Scholarship (Saad Sheikh), NSF CCF-0546574 (Chaovalitwongse).

References

1. A. Almudevar. A simulated annealing algorithm for maximum likelihood pedigree reconstruction. *Theoretical Population Biology*, 63:63–75, 2003.
2. A. Almudevar and C. Field. Estimation of single generation sibling relationships based on DNA markers. *Journal of Agricultural, Biological, and Environmental Statistics*, 4:136–165, 1999.
3. T. Y. Berger-Wolf, B. DasGupta, W. Chaovalitwongse, and M. V. Ashley. Combinatorial reconstruction of sibling relationships. In *Proceedings of the 6th International Symposium on Computational Biology and Genome Informatics (CBGI 05)*, pages 1252–1255, Utah, July 2005.
4. J. Beyer and B. May. A graph-theoretic approach to the partition of individuals into full-sib families. *Molecular Ecology*, 12:2243–2250, 2003.
5. M. S. Blouin. DNA-based methods for pedigree reconstruction and kinship analysis in natural populations. *TRENDS in Ecology and Evolution*, 18(10):503–511, October 2003.
6. K. Butler, C. Field, C.M. Herbinger, and B.R. Smith. Accuracy, efficiency and robustness of four algorithms allowing full sibship reconstruction from DNA marker data. *Molecular Ecology*, 13:1589–1600, 2004.
7. A. Cahovalitwongse, T. Y. Berger-Wolf, B. Dasgupta, and M. V. Ashley. Set covering approach for reconstruction of sibling relationships. *Optimization Methods and Software*, 2006.
8. S. C.Thomas and W. G.Hill. Sibship reconstruction in hierarchical population structures using Markov chain Monte Carlo techniques. *Genet. Res., Camb.*, 79:227–234, 2002.
9. E. Eskin, E. Haleprin, and R. M. Karp. Efficient reconstruction of haplotype structure via perfect phylogeny. *Journal of Bioinformatics and Computational Biology*, 1(1):1–20, 2003.
10. A. G. Jones and W. R. Ardren. Methods of parentage analysis in natural populations. *Molecular Ecology*, (12):2511–2523, 2003.
11. J. Li and T. Jiang. Efficient inference of haplotypes from genotype on a pedigree. *Journal of Bioinformatics and Computational Biology*, 1(1):41–69, 2003.
12. I. Painter. Sibship reconstruction without parental information. *Journal of Agricultural, Biological, and Environmental Statistics*, 2:212–229, 1997.
13. B. R. Smith, C. M. Herbinger, and H. R. Merry. Accurate partition of individuals into full-sib families from genetic data without parental information. *Genetics*, 158:1329–1338, 2001.
14. J. Wang. Sibship reconstruction from genetic data with typing errors. *Genetics*, 166:1968–1979, April 2004.
15. A.A.C. Wilson, P. Sunnucks, and J.S.F. Barker. Isolation and characterization of 20 polymorphic microsatellite loci for Scaptodrosophila hibisci. *Molecular Ecology Notes*, 2:242–244, 2002.

5 Appendix A: Possibilities Table

Parents	Offspring allele a	allele b
Set parents (1 , 2) (3, 4)		
	1	3
	2	4
	1	4
	2	3
	3	1
	4	2
	4	1
	3	2
Set parents (1, 2) (1, 3)		
	1	1
	2	3
	1	3
	2	1
	3	2
	3	1
	1	2
Set parents (1 , 2) (1 , 2)		
	1	1
	1	2
	2	1
	2	2

Parents	Offspring allele a	allele b
Set parents (1, 1) (1 , 1)		
	1	1
Set parents (1 , 1) (1 , 2)		
	1	1
	1	2
	2	1
Set parents (1, 1) (2, 3)		
	1	2
	1	3
	2	1
	3	1
Set parents (1, 1) (2, 2)		
	1	2
	2	1

Table 1. Canonical possible combinations of parent alleles and all resulting offspring allele combinations

6 Appendix B: Results

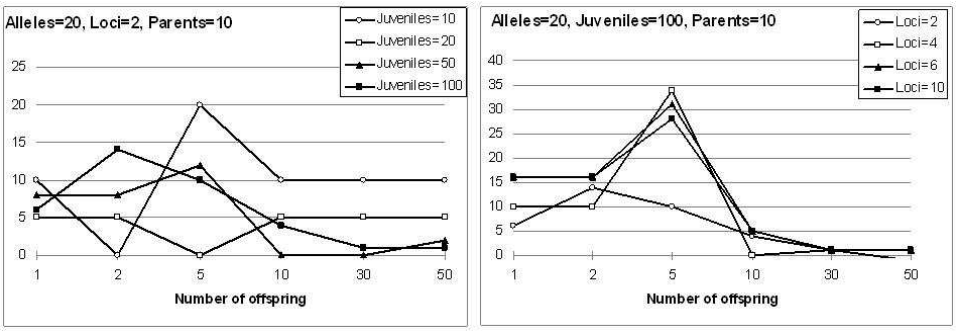

Fig. 1. Error rate comparison between the 4-allele algorithm from [3] and the 2-allele algorithm implemented in this paper. The y-axis shows the decrease of the error rate, or the improvement in the accuracy rate, from the 4-allele algorithm to the 2-allele algorithm on the same random data.

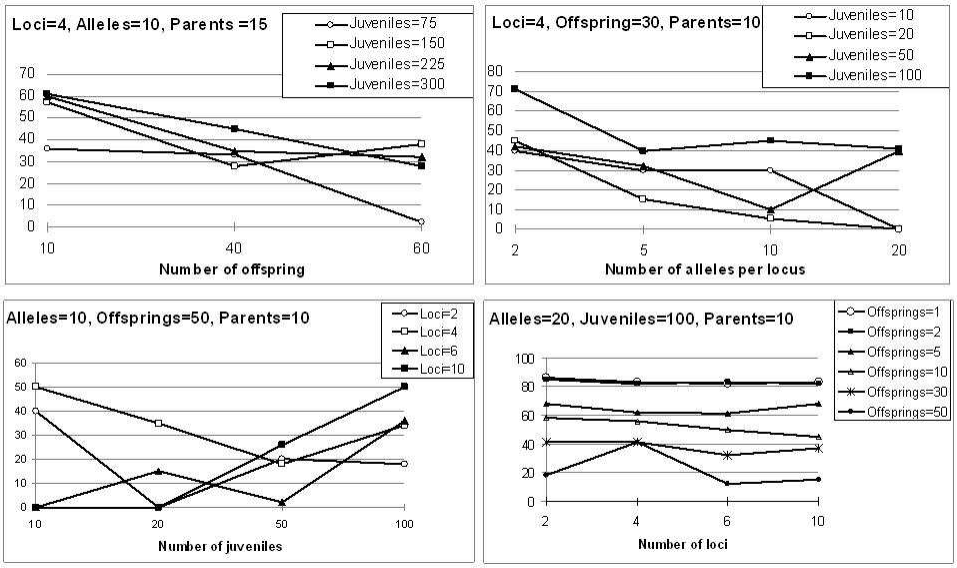

Fig. 2. Accuracy of the sibling group reconstruction using the 2-allele algorithm on randomly generated data. The y-axis shows the error rate as a function of various simulation parameters.