Optimization Methods and Software
Vol. 00, No. 00, Month 200x, 1–18

Set Covering Approach for Reconstruction of Sibling Relationships

W.A. CHAOVALITWONGSE†‡*, T.Y. BERGER-WOLF‡§, B. DASGUPTA§ and M.V. ASHLEY¶

†Department of Industrial and Systems Engineering, Rutgers University, NJ, USA,
‡DIMACS, Rutgers University, Piscataway, NJ, USA,
§Department of Computer Science, University of Illinois, Chicago, IL, USA,
¶Department of Biological Sciences, University of Illinois, Chicago, IL, USA

(Received 00 Month 200x; In final form 00 Month 200x)

A new combinatorial approach for modeling and reconstructing sibling relationships in a single generation of individuals without parental information is proposed in this paper. Simple genetic constraints on the full-sibling groups, imposed by the Mendelian inheritance rules, are employed to investigate data from codominant DNA markers. To extract the minimum number of biologically consistent sibling groups, the proposed combinatorial approach is employed to formulate this minimization problem as a set covering problem, which is a well-known *NP*-hard combinatorial optimization problem. We conducted a simulation study of a relaxed version of the proposed algorithm to demonstrate that our combinatorial approach is reasonably accurate and the exact version of the sibling relationship construction algorithm should be pursued. In addition, the results of this study suggest that the proposed algorithm will pave our way to a new approach in computational population genetics as it does not require any a priori knowledge about allele frequency, population size, mating system, or family size distributions to reconstruct sibling relationships.

Keywords: Sibling Relationships, Set Covering Problems, Combinatorial Optimization, DNA markers, Linear Assignment Problems

*Corresponding author. Department of Industrial and Systems Engineering, Rutgers University, 96 Frelinghuysen Rd., Piscataway NJ 08854, USA. Tel: (732) 445-5469. Fax: (732) 445-5647. Email: wchaoval@rci.rutgers.edu

Optimization Methods and Software ISSN 1055-6788 print/ ISSN 1029-4937 online ©200x Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: doi

1 Introduction

1.1 *Set Covering Problem: Preamble*

The Set Covering Problem (SCP) is one of the most studied mathematical programming models in combinatorial optimization for several important real world applications (see the survey by Balas [4] and the annotated bibliography by Ceria et al. [21]). Practical applications of the SCP include crew scheduling [3,9,20,23,39,41], emergency facility location [43,45], assembly line balancing [40], production flow-lines optimization [17], political redistricting and boolean expression simplification [18]. The SCP has been proven to be *NP*-complete [29]. A more detailed discussion on the problems of this class can be found in [19].

The SCP is a problem of covering the rows of a $m \times n$ zero-one matrix by a subset of the columns at the minimal cost. In other words, given a $m \times n$ zero-one matrix A, the SCP is to select the minimum weight subset of columns while ensuring that every row has at least one non-zero entry in the submatrix induced by the columns. Each entry of the matrix A is represented by a_{ij}, where $a_{ij} = 1$ when column i covers row j, $a_{ij} = 0$ otherwise. In the SCP, the decision variables $x_i = 1$ if column i is selected to be in the solution and $x_i = 0$ otherwise. The SCP can be formulated as the following:

$$\min \quad \sum_{i=1}^{n} c_i x_i \tag{1}$$

$$\text{s.t.} \sum_{i=1}^{n} a_{ij} x_i \geq \overline{1} \qquad j = 1, \dots m \tag{2}$$

$$x_i \in \{0, 1\} \qquad i = 1, \dots n. \tag{3}$$

The constraint set (2) guarantees that each row j must be covered by one column. The set partitioning problem is a special case of the SCP when the inequalities in the constraint set (2) are replaced by equalities, which ensure that each column can only cover one row (no over-covering). When $c_i = 1 \ \forall i$, the problem is called the unicost SCP.

There have been numerous optimal and heuristic solution approaches for the SCP presented in the literature [5–8, 13, 19, 36]. Many exact solution techniques for the SCP can be found the literature, including branch-and-bound approach [5, 13, 19, 36] and cutting plane algorithm [6–8]. As the SCP is an *NP*-hard problem, the proposed exact solution techniques to large-scale instances are very time-consuming. A large number of heuristic approaches have been developed for solving large-scale problems with relatively short computational time. Greedy construction algorithms with a randomized technique were proposed in [22], in which the approximation ratio of the greedy algorithm is $\ln m + 1$. Later on in the 80's, most of optimal algorithms for SCP proposed are typically based on tree-search procedures, which have been used to solve SCP's with up to 50 rows and 500 columns at considerable computational cost [6, 10]. An approach based on a dual heuristic algorithm was proposed and used to solve SCP's with up to 200 rows and 2,000 columns [28]. A new solution technique based on Lagrangian heuristic, feasible solution exclusion constraints, and an improved branching strategy was proposed to solve SCP's with up to 400 rows and 4,000 columns [11, 13]. Recently, a simulated annealing heuristic approach was proposed to solve SCP's with up to 1,000 rows and 10,000 columns [32]. Other robust heuristic approaches found in the literature include Lagrangian heuristics [11, 19], neural networks [30], genetic algorithms [12, 26] and ant colony algorithms [1]. The performance of nine different heuristic algorithms on the unicost SCP was studied in [30]. Most of the algorithms were based on LP rounding techniques, construction-based approaches, and neural network algorithms.

1.2 *SCP: Complexity Issues*

The SCP can be shown to be *NP*-complete by polynomially reducing the well known *NP*-complete Vertex Cover Problem (VCP) to the SCP. The VCP (often called Node Cover Problem) is considered to be a special case of the SCP, where each node in a graph covers the edges incident to it. The difference is that in VCP every element (edge) appears in exactly two sets (its endpoints), which yields better approximation guarantees. In other words, if $k = \max_i \sum_{j=1}^{n} a_{ij}$ is defined as the maximum number of ones appearing in any row, then the SCP having exactly $k = 2$ ones in every row becomes the VCP. The VCP can be defined as follows. Let G be an undirected graph. A set S of nodes covers an edge if at least one of its endpoints lies in S. The set S is a vertex cover if it covers every edge. The VCP is to find a vertex cover of minimum weight, given a graph G and weights $w_i \geq 0$ on vertices. It is well known that, given a graph G, a clique of G, an independent set of $\bar{G}$, and a node-cover of $\bar{G}$ are equivalent. As we all know that clique is *NP*-complete, node cover is also *NP*-complete. It is simple to show that a graph G has a clique of size k if and only if $\bar{G}$ has a node-cover of size $|V| - k$, if and only if $\bar{G}$ has an independent set of size k. Therefore, all three problems can be polynomially transformed to each other [38].

Another proof of *NP*-completeness of the SCP can be derived from a 3-exact cover problem, which can be defined as follows. Given a family $F = \{S_1, \ldots, S_n\}$ of n subsets of $S = \{u_1, \ldots, u_{3m}\}$, each of cardinality three, is there a subfamily of m subsets that covers S? 3-exact cover is a special case of set cover. The 3-exact cover problem is proven to be NP-complete by demonstrating that tripartite matching is a special case of 3-exact cover (as well as set cover). It is worth noting that 3-exact cover can be polynomially transformed to 0-1 knapsack problem. This fact has been used to illustrate the NP-completeness of the 0-1 knapsack problem.

The optimization of the SCP, Minimum Set Covering problem (MSCP), is NP-hard and not approximable within a constant factor. In other words, there is no constant factor. The MSCP is approximable within $1+\ln|S|$ and is hard to approximate within $(1-\epsilon)\ln n$ time (only when $NP \subseteq DTIME(n^{O(\log\log n)})$) [27]. Subsequently, for the MSCP with the maximum set size of B, the lower bound has been proven to be $\ln B - O(\ln\ln B)$ [44]. The Minimum Node Cover problem is also NP-hard and approximable within $2-(\log\log n/2\log n)$ [35].

1.3 *MSCP: Application in Sibling Group Reconstruction*

In this paper, we consider applications of the MSCP that have relevance for genetic epidemiology, molecular ecology, population genetics, and conservation biology. Specifically, we have employed combinatorial optimization techniques to solve MSCPs for reconstructing sibling relationships based on single generation genetic data without parental information. This problem is extremely important because knowledge of familial relationships is needed for many biological applications including the estimation of heritabilities of quantitative characters, studies of mating systems and fitness, and managing populations of endangered species. Typically, biologists have used parental data to establish sibling groups indirectly through parentage assignments (e.g., see [33]). Reconstructing sibling relationships without parental data is a much more difficult problem, but one that faces many investigators who sample and genotype cohorts of offspring rather than parent/offspring groups. In these cases, a reliable method of reconstructing family structure in the population would be extremely useful for studying inheritance patterns, natural selection, breeding biology, and gene flow parameters [14].

In recent years, there has been a growing interest in developing computational methods for reconstructing sibling relationships without the parental data [15]. Most of those use statistical population parameters to find maximum likelihood clusters [16, 25, 37, 42, 46]. There are, thus far, only two methods in the literature developed to incorporate combinatorial optimization approaches to solve the sibling relationship reconstruction problem. Graph clustering algorithms were used to form groups from pairwise likelihood distance graph [14]. The Mendelian inheritance rules were used to enumerate all possible potential full-sibling groups and subsequently a heuristic approach was used to construct a maximal (but not necessarily optimal) partition of the individuals into those groups to reconstruct sibling relationships [2].

Despite the increasing research efforts in sibling group reconstruction, to our knowledge, the approach based on the MSCP has not been addressed elsewhere. In this paper, we present a systematically combinatorial optimization approach to solve the sibling relationship reconstruction problem based on single generation genetic data with no parental information by reducing it to a MSCP. Specifically, our proposed approach is the proper formalization of the algorithm (proposed by Almudevar and Field in [2]) by using the Mendelian inheritance rules to impose constraints on the genetic content possibilities of a sibling group. From single generation genetic data, we formulate MSCP's based on the inferred combinatorial constraints and use a provably correct algorithm to construct the smallest number of groups of individuals that satisfy these constraints. The advantages of our approach are as follows: the proposed algorithm allows half-sibling relationships to exist in the population, and it does not require a priori knowledge of the allele frequency, number of loci sampled, mating system, or the size of the family groups. This method can be easily extended to incorporate null-allele type errors. In this experiment, we tested our algorithm on simulated data with known parents and sibling groups. After our algorithm reconstructs sibling groups, we assess the accuracy of our approach by using an extension of an accuracy measure (partitioning distance) presented in [31]. The accuracy assessment of our algorithm can be accomplished by solving a maximum linear assignment problem.

This paper is organized as follows. Section 2 discusses the problem of how to reconstruct sibling relationships based on single generation genetic data. Section 3 introduces the 4-allele algorithm used to formulate the sibling group reconstruction as a MSCP. Section 4 describes the design of simulation experiments to assess the accuracy of the proposed algorithm. Section 5 discusses the results obtained from our experiments. The final section provides a discussion of the implications and results of our methods, difficulties still remaining, and plans to address remaining issues in future research.

2 Reconstruction of Sibling Relationships

2.1 *Basic Definitions*

A group of individuals is called *full siblings* if they have the same parents. A group of individuals is called *half siblings* if exactly one of the two parents is the same for every individual in the group. A group of individuals is called *siblings* if they are either full or half siblings. *Gene* is the fundamental physical and functional unit of heredity, which carries information from one generation to the next. *Locus* is a specific location on a chromosome. *Allele* is one of the different versions of the same gene found at the same locus on homologous chromosomes or in different individuals. *Allele frequency* is the fraction of all the alleles of a gene in a population that are of one type. *Genetic markers* are loci where individuals can be experimentally sampled. A *Homozygous* individual is one having two identical alleles at a particular genetic locus. A *Heterozygous* individual is one having two different alleles at a particular genetic locus. A *diploid* individual has two sets of chromosomes; one set from the father and one from the mother.

2.2 *Sibling Relationship Reconstruction Problem*

The sibling relationship reconstruction problem can be formally stated as follows. Given a set U of n diploid individuals of the same generation, the goal is to reconstruct the existing sibling relationships among them. Each individual $1 \leq i \leq n$ is represented by a genetic marker of l loci $\langle (a_{ij}, b_{ij}) \rangle_{1 \leq j \leq l}$. The numbers a_{ij} and b_{ij} represent a specific allele. Mendelian inheritance rules impose two necessary (but not sufficient) constraints on a group of diploid individuals $S \subseteq U$ to be full siblings:

Definition 2.1 A set $S \subseteq U$ has the *4-allele property* if $|\bigcup\limits_{i \in S} a_{ij} \cup b_{ij}| \leq 4$ for $1 \leq \forall j \leq l$.

Definition 2.2 A set $S \subseteq U$ has the *2-allele property* if $|\bigcup\limits_{i \in S} a_{ij}| \leq 2$ and $|\bigcup\limits_{i \in S} b_{ij}| \leq 2$ for $1 \leq \forall j \leq l$.

Clearly, the 2-allele property in the definition 2.2 is stronger (tighter) than the 4-allele property in the definition 2.1. Assuming that the order of the parental alleles is always the same in the offspring (i.e., the maternal is always on the same side), the 2-allele property is theoretically equivalent to a biologically consistent full sibling relationship. In addition, we propose the following theorem to present a relationship between the definition 2.2 and the definition 2.1.

Theorem 2.3 *Let a be the number of distinct alleles presented in a given locus and R be the number of distinct alleles that either appear with three different alleles in this locus or are homozygous (appear with itself). Then, given a set of individuals with the 4-allele property, there exists a series of allele switches within some of the loci resulting in a set that satisfies the 2-allele property if and only if for all the loci in the set*

$$a + R \leq 4.$$

Proof Necessity: Let us assume that there exists a series of switch operations that result in a set that satisfies the 2-allele property. Consider any locus j in the set after the switches have been performed: $\langle a_{ij}, b_{ij} \rangle_{i \in S}$. Since the 2-allele property is satisfied, each allele a_{pj} appears with no more than two different alleles b_{qj} among all the individuals and vice versa.

- If the number of different alleles $a \leq 2$ then, clearly, the number of distinct alleles $R \leq 2$ and $a + R \leq 4$.
- If $a = 3$ and there is no allele $a_{pj} = b_{pj}$, then either $|\cup a_{ij}| = 1$ or $|\cup b_{ij}| = 1$ and each allele appears with no more than two other alleles; that is, $R = 0$ and $a + R < 4$. If there is $a_{pj} = b_{pj}$, then this allele may appear with three other alleles in the individuals (itself and the remaining two alleles). The two other alleles appear with at most two alleles each. Thus, we have $R = 1$; therefore, $a + R = 4$.
- Consider the loci with $a = 4$. In this case, if all a_{ij} differ from all b_{ij} then $R = 0$. Therefore, $a + R \leq 4$ is satisfied. If some allele $a_{pj} = b_{qj}$, then there is only one more allele that can be on the left side and one more allele that can be on the right side. Therefore, $a = 3$, which contradicts the assumption that $a = 4$.

Sufficiency: Assume, for all the loci, $a + R \leq 4$. We consider each locus independently at different values of a for a given locus j.

- If $a = 2$ then the 2-allele property is satisfied trivially.
- If $a = 3$, then $R \leq 1$. Note that if there is an allele appearing with three other alleles, then it necessarily appears with itself. Let a_{pj} and a_{qj} be the alleles that do not appear with itself. We can switch the alleles so that a_{pj} is on the left and a_{qj} is on the right in all the individuals that contain them. If all the individuals contain either of the two alleles ($R = 0$), then he union on the left side is $\{a_{pj}, b_{ij}\}$ while the unions on the right side is $\{a_{qj}, b_{ij}\}$ (where b_{ij} is the third allele). Consequently, the 2-allele property is satisfied. If there are individuals that do not contain a_{pj} or a_{qj}, then they must contain the pair (b_{ij}, b_{ij}), which does not change the unions.
- If $a = 4$, then $R = 0$ implying that no allele appears with more than three other alleles in this locus and no allele can appear with itself. Consider the case of individuals with allele a_{pj}. As $R = 0$, a_{pj} appears with at most two other alleles. Therefore, there must exist another allele a_{qj} that does not co-appear with a_{pj} in any individuals. Each individual must have either a_{pj} or a_{qj} and $\bigcup\limits_{i \in S} a_{ij} = \{a_{pj}, a_{qj}\}$. As the rest must contain the remaining two other alleles, the 2-allele property is then satisfied. Consider the case of individuals with neither allele a_{pj} or a_{qj}. All the remaining individuals must contain another allele pair (b_{sj}, b_{tj}). Since $R = 0$, these alleles b_{sj} and b_{tj} could not appear with both a_{pj} and a_{qj}. Then the 2-allele property is also satisfied.

□

To reconstruct such a sibling relationship for each genetic dataset of a single generation, we can formulate the sibling relationship reconstruction problem as a Minimal 2-allele Set Covering Problem (M2SCP). The M2SCP is defined as follows: Given a collection U of n l-tuples $\left\{A_i = \langle (a_{ij}, b_{ij}) \rangle_{\substack{1\leq i\leq n\\ 1\leq j\leq l}}\right\}$, find a minimum number of subsets $S_1, ..., S_m$ in U that satisfy the 2-allele property and whose union is U. The mathematical formulation of the M2SCP is given by

$$
\begin{aligned}
& \min m && \qquad\qquad (4)\\
\text{s.t.}\quad & \bigcup_{i=1}^{m} S_t = U, && \text{for } 1 \leq \forall t \leq m, S_t \subseteq U, \qquad\qquad (5)\\
& |\bigcup_{A_i \in S_t} \{a_{ij}, b_{ij}\}| + R_j \leq 4, && \text{for } 1 \leq \forall t \leq m, 1 \leq \forall j \leq l. \qquad\qquad (6)
\end{aligned}
$$

where R_j is the value of x such that either $(x, x) \in \bigcup_{A_i \in S_t} \{(a_{ij}, b_{ij})\}$ or $|\{y : (x, y) \in \bigcup_{A_i \in S_t} \{(a_{ij}, b_{ij})\} \vee (y, x) \in \bigcup_{A_i \in S_t} \{(a_{ij}, b_{ij})\}\}| = 3$. We then propose the following algorithm to solve the M2SCP:

(i) For each locus, independently, create all possible 2-allele sets of individuals. Note that for a alleles in the locus of R distinct alleles, there are at most $\binom{a}{4} + \binom{R}{3} + \binom{R}{2} = O(a^4)$ sets.

(ii) Find the sibling sets that are consistent with all the loci. Note that such sibling sets must exist as each pair of individuals forms a consistent sibling set.

(iii) Find a smallest size set cover of all the individuals from the sets found in previous step.

Although the proposed M2SCP algorithm will generate a sibling relationship that is biologically consistent, it is computationally expensive. In this paper, we postulate that we can under-approximate the sibling relationships by exploiting the 4-allele property. The proposed approximation scheme is considered to be as a heuristic approach to solve the M2SCP. Note that the 4-allele property is computationally much cheaper than the M2SCP. To under-approximate the solution to the M2SCP, we propose a novel and much faster algorithm using the 4-allele property to approximate the solution to the sibling reconstruction problem. This algorithm, considered to be a relaxation version of M2SCP, is described in the next section.

3 Proposed Technique: Minimal 4-allele Set Covering Problem (M4SCP)

We exploit the 4-allele property to identify sibling groups among a given group of juveniles. First, we assume that the relationships may be promiscuous and half siblings may be both paternal and maternal. Thus, an individual may be in more than one sibling group. Note that we can consider the M4SCP as a relaxation of the M2SCP. The M4SCP can defined as follows: Given a collection U of n l-tuples $\left\{A_i = \langle (a_{ij}, b_{ij}) \rangle_{\substack{1\le i\le n \\ 1\le j\le l}}\right\}$, the M4SCP is to find a minimum number of subsets $S_1, ..., S_m$ in U that satisfy the 4-allele property and whose union is U. The formulation of the M4SCP is given by

$$\min m \tag{7}$$

$$\text{s.t.} \quad \bigcup_{i=1}^{m} S_t = U, \qquad \text{for } 1 \le \forall t \le m, S_t \subseteq U, \tag{8}$$

$$|\bigcup_{A_i \in S_t} \{a_{ij}, b_{ij}\}| \le 4, \ \text{ for } 1 \le \forall t \le m, 1 \le \forall j \le l. \tag{9}$$

It is worth noting that the M4SCP always underestimates the M2SCP because in the M2SCP, the last constraint will be $|\bigcup_{A_i \in S_t} \{a_{ij}, b_{ij}\}| + R_j \le 4$, for $1 \le \forall t \le m, 1 \le \forall j \le l$. As we mentioned earlier, the difficulty of solving the M2SCP is finding R_j for all the combinations (tuples) of alleles. In order to make the problem more precisely defined, the M4SCP can be stated as follows: Given a universe $U = \{1, 2, ..., n\}$ and a collection of sets $\mathcal{S} = \{S_1, S_2, ..., S_m\}$ such that $S_i \subseteq U$, find the smallest number of sets in $\mathcal{S}$ whose union is the universe:

$$\min_{I \subseteq [m]} |I| \text{ s.t. } \bigcup_{i \in I} S_i = U. \tag{10}$$

Let M is an $n \times m$ matrix whose elements $m_{ij} = 1$ if $i \in S_j$ and $m_{ij} = 0$ otherwise. After stating the M4SCP above, we then formulate the M4SCP as a mixed-integer 0-1 programming problem given by

$$\min \quad \sum_{i=1}^{m} x_i \tag{11}$$

$$\text{s.t.} \quad Mx \ge \overline{1} \tag{12}$$

$$x_i \in \{0, 1\}. \tag{13}$$

We propose the following algorithm to solve the M4SCP:

(i) For each pair of individuals A_p and A_q, form a set S_{pq} that represents their 4-allele property. That is, S_{pq} is a collection of l loci where each locus is a union of alleles of the corresponding locus for p and q.

(ii) An individual belongs to a set S_{pq} if, for each locus, the set of the alleles of that individual for that locus is in the the corresponding locus set of S_{pq}.

(iii) Find the minimum set covering of S. For each set in S define the corresponding set of individuals covered by that set as a sibling group. Return the group structure induced by S.

We can state the pseudo-code of the algorithm for the M4SCP as in Figure 1.

```
**Algorithm** 4-alleleSets
FOR $i = 0 \dots n-1$ DO
    FOR $j = i+1 \dots n-1$ DO
        $S_{ij} = \emptyset$
        FOR $t = 0 \dots k-1$ DO
            $S_{ij} = S_{ij}(\{a_{it}, b_{it}\} \cup \{a_{jt}, b_{jt}\})$
FOR $p = 1 \dots n^2 - n$ DO
    FOR $i = 0 \dots n-1$ DO
        FOR $t = 0 \dots k-1$ DO
            IF $(a_{it} \notin S_p \vee b_{it} \notin S_p)$ THEN
               break
        individual$_i \in S_p$
Find a minimum set cover $S = \{S_1, ..., S_p\}$
RETURN the $p$ sibling groups defined by $S$.
```

Figure 1. Pseudo-code of the proposed algorithm for M4SCP.

Proposition 3.1 *Any set covers defined above induces a valid collection of 4-allele groups.*

Proof Each set S_{pq} is defined by a locus-by-locus union of two individuals. Each individual has at most two alleles per locus; therefore, each set has no more than four alleles per locus (4-allele property). Every 4-allele combination for a collection of individuals is represented by some sets as there is a set corresponding to every pair of individuals. In any feasible solutions, the set cover contains all the individuals; that is, every individual belongs to some sets induced by the set cover. Each set in the set cover does not violate the 4-allele property; therefore, the entire sets in the set cover are a valid collection of 4-allele groups. □

4 Experimental Design

In this study, we developed and assessed the accuracy of the M4SCP algorithm on simulated data. To create a set of simulation data, we first generated a number of adults (parents) with the full genetic information. Based on this parentage information, a single generation of juveniles was then generated. Note that as the parentage information is retained, the true sibling groups are known. After developing a computer algorithm for the M4SCP, the M4SCP algorithm was then used to solve SCP's and reconstruct the sibling groups. To assess the accuracy of the proposed M4SCP algorithm, we use the extended partition distance proposed in [31] to measure the precision of the reconstruction with respect to the true sibling groups (see Section 4.2 for more details). In this study we assume that organisms are diploid and we simulate diploid organisms.

4.1 *Experiment Protocol and Parameters*

The genetic data were simulated with a given number of adult males M and females F, a given number of loci l, and a specified number of alleles per locus a. Each individual was created by randomly choosing from an independent identical uniform distribution (IID) $2l$ number of alleles from among a alleles, which are matched up into l loci. Then, $j \times F$ juveniles are created, where j is the factor of the number of juveniles as the number of females. A male and a female is chosen randomly, independently and uniformly from the adult population. A couple has a random number of offspring, up to a specified maximum number of offspring o. Each offspring randomly gets one of the mother's and one of the father's alleles per locus, which are assembled randomly. In short, this protocol creates a population of juveniles with known parents and siblings that is biologically consistent.

The parameter ranges for the study are as follows:

- The number of adult females $F = 10$ and adult males $M = 10$.
- The number of loci sampled $l = 2, 4, 6, 10$.
- The number of alleles per locus $a = 2, 5, 10, 20$.
- The factor of the number of juveniles as the number of females $j = 1, 2, 5, 10$.
- The maximum number of offspring per couple $o = 2, 5, 10, 30, 50$.

Each offspring was created with parental alleles in 2 manners: recombined and non-recombined alleles. Using the above parameter settings, we generated a series of 640 test instances for every combination of each setting, whose characteristics are described in Table 1.

Based on the juvenile population of the simulated test instances, the M4SCP algorithm was used to find the smallest number of 4-allele sets, which are

Parameter Settings	Rows (m)	Columns (n)	Max-number of ones per row	Density (%)	Number of Problems
l=2	1602.5	45	993.14	58.7%	160
l=4	1602.5	45	825.21	51.9%	160
l=6	1602.5	45	764.27	51.7%	160
l=10	1602.5	45	721.54	51.8%	160
a=2	1602.5	45	1357.72	72.8%	160
a=5	1602.5	45	715.99	49.0%	160
a=10	1602.5	45	611.14	45.3%	160
a=20	1602.5	45	619.30	47.0%	160
j=1	45.0	10	36.54	72.7%	160
j=2	190.0	20	135.39	60.5%	160
j=5	1225.0	50	717.56	45.7%	160
j=10	4950.0	100	2405.67	34.7%	160
o=2	1602.5	45	576.29	26.5%	128
o=5	1602.5	45	629.26	38.1%	128
o=10	1602.5	45	711.13	51.1%	128
o=30	1602.5	45	999.40	71.1%	128
o=50	1602.5	45	1214.13	80.8%	128

Table 1. Characteristics of Simulated Instances

postulated to be the full sibling groups. Although the MSCP is NP-hard, modern Mixed Integer Programming (MIP) solvers can solve our simulation instances to optimality (even the largest instance of 1000 × 4950) in timely manner. We formulated the M4SCP test instances as MIP problems. To solve MSCP's, we used CPLEX 9.0 MIP solver by ILOG to obtain optimal solutions to the M4SCP test instances. After the optimal solutions to the M4SCP test instances were obtained, the groups reconstructed by the M4SCP algorithm were compared with the true sibling groups. In recent years, several methods have been used to assess the accuracy of the reconstructed sibling groups comparing with the true groups. However, most of them are mathematically inconsistent. To assess the accuracy of the solutions obtained from the M4SCP algorithm, we use an extension of the partition distance measure described in [31].

4.2 *Solution Accuracy Measure*

The minimum partition distance used to assess the accuracy of our algorithm has been shown to be equivalent to the Maximum Linear Assignment Problem (MLAP) [31] (also called maximum bipartite weighted matching problem), which is a well known linear programming problem [24,34]. The MLAP can be defined as follows: given two collections of sets $\{A_1, ..., A_n\}$ and $\{B_1, ..., B_m\}$, let C be $n \times m$ cost matrix where c_{ij} is the cost of the assignment of A_i to B_j. Then the MLAP is to find an assignment of the set A to the set B at the maximum cost such that each element in set A is assigned to at most one in set B vice versa. The MLAP can be formulated as a MIP problem given by

$$\max \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} x_{ij} \tag{14}$$

$$\text{s.t.} \quad \sum_{j=1}^{m} x_{ij} \leq 1 \qquad \text{for } i = 1, \ldots, n \tag{15}$$

$$\sum_{i=1}^{n} x_{ij} \leq 1 \qquad \text{for } j = 1, \ldots, m \tag{16}$$

$$x_{ij} \in \{0, 1\}.$$

Generally, the accuracy measure can be formulated as a MLAP as follows: given two partitions in U, $\{P_1, ..., P_n\}$ and $\{Q_1, ..., Q_m\}$, define $c_{ij} = |P_i \cap Q_j|$. Then, $|U|-$(maximum assignment) represents the minimum number of elements to be deleted so that these two partitions are identical. This distance measure will give the accuracy of our set covers reflected by the distance between two sets.

5 Computational Results

In this section, we present computational results of the M4SCP algorithm on the simulated test instances described in Table 1. After the M4SCP algorithm had found the set cover of the single generation for each test instance, the solution to the M4SCP (the set cover) were then compared with the real sibling relationships (already known when the test instances were generated) by formulating the problem as a MLAP to find the minimum distance (deletions) between the two sets. The solution to the MLAP is considered to be an error rate of the predicted sibling relationships constructed by the M4SCP algorithm. Subsequently, we calculate the accuracy measure based on the error rate (accuracy = 1-error rate). All the instances of the set cover problem in our simulation were solved optimally under 0.3 seconds. The average CPU time to solve the test instances and the average accuracy of the predicted sibling relationships are presented in Table 2.

We examine the accuracy behavior as a function of the number of loci, alleles per each locus, juvenile population size, and maximum number of offspring, illustrated in Figures 2-5. Figure 2 demonstrates that the accuracy increases as the number of offspring per couple increases. On the other hand, Figure 3

Parameter Settings	Avg. CPU Time (seconds)	Avg. Accuracy (%)
l=2	0.256	54.18%
l=4	0.211	52.71%
l=6	0.191	54.78%
l=10	0.185	55.28%
a=2	0.155	36.98%
a=5	0.161	58.34%
a=10	0.387	60.71%
a=20	0.185	60.91%
j=1	0.010	70.50%
j=2	0.017	62.88%
j=5	0.113	49.56%
j=10	0.750	34.00%
o=2	0.218	18.19%
o=5	0.272	36.66%
o=10	0.218	53.98%
o=30	0.225	75.17%
o=50	0.237	87.17%

Table 2. Performance Characteristics of the M4SCP Algorithm on Simulated Instances

illustrates that the accuracy decreases as the total number of juveniles increases. Figure 4 shows that the accuracy is not strongly influenced by the number of alleles. Similarly, Figure 5 shows that the accuracy is not strongly influenced by the number of sampled loci. These observations demonstrate that the numbers of offspring and juveniles heavily influence the accuracy of our M4SCP algorithm. In contrast, these results surprisingly showed that the number of alleles per locus and the number of sampled loci are not strong factors for the M4SCP algorithm to reconstruct sibling relationships (except when there are only 2 alleles per locus and the algorithm assumes that all the juveniles are siblings). It is worth noting that the M4SCP algorithm has found fewer sibling groups than those in the real population, merging the true families into a reconstructed one. This observation supports the theoretical definition of the M4SCP algorithm as an under-approximation algorithm for the M2SCP as mentioned in Section 3. This study shows the tightness of the solution by the M4SCP and the real solution. The accuracy of the prediction of the sibling relationships can be improved with the M2SCP algorithm, which we have shown to yield biologically consistent sibling relationships.

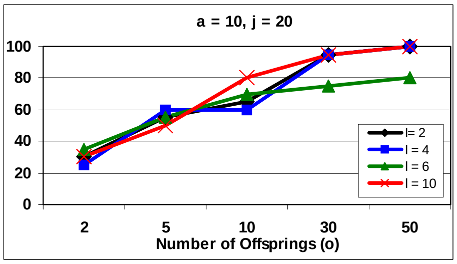

Figure 2. Accuracy (%) of the M4SCP as a function of the number of offspring (o) while fixing the number of alleles ($a = 10$) and juveniles ($j = 20$).

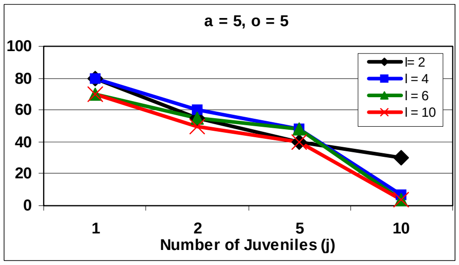

Figure 3. Accuracy (%) of the M4SCP as a function of the number of juveniles (j) while fixing the number of alleles ($a = 5$) and offspring ($o = 5$).

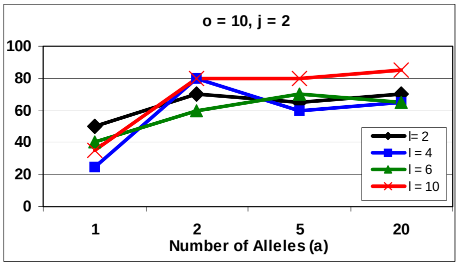

Figure 4. Accuracy (%) of the M4SCP as a function of the number of alleles (a) while fixing the number of juveniles($j = 2$) and offspring ($o = 10$).

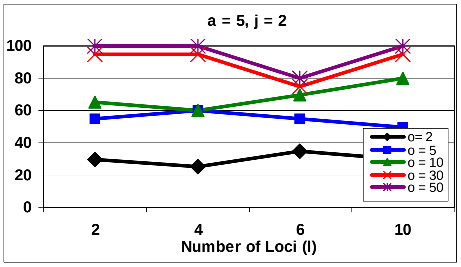

Figure 5. Accuracy (%) of the M4SCP as a function of the number of loci (l) while fixing the number of juveniles($j = 2$) and alleles ($a = 5$).

6 Conclusions and Prospects

In this paper, we present a set covering approach based on the M4SCP for reconstructing sibling relationships in the absence of parental data. In contrast to other existing methods in the literature, our approach does not require the implementation of any statistical estimates of the relatedness among the individuals. On the other hand, our approach directly employs a Mendelian constraint on the possible genetic content of a sibling group. Such a constraint based on Mendelian rules turns out to be sufficiently powerful to reconstruct the sibling relationships fairly accurately in our simulations. The stronger version of this constraint from the M2SCP algorithm has the potential to accurately reconstruct sibling groups without any prior knowledge of the population structure and its genetic characteristics. Nonetheless, to validate this approach and its applicability more extensive simulations and experiments are required, as well as comparison to other known methods. In the future, we need to investigate the computational complexity and better algorithmic solutions to the M2SCP and M4SCP. We need to conduct simulations with the M2SCP algorithm and run these for a wider range of parameters and parameter distributions, as well as allow for errors in the data. We need to validate the results on real biological datasets, especially where the sibling groups have been established using other methods. Last but not least, we need to compare the performance of our method to other methods of sibling reconstruction. Nevertheless, the overall outcome of this study suggests that the proposed algorithm will pave our way to a new approach in computational population genetics as it does not require any a priori knowledge about allele frequency, population size, mating system, or family size distributions to reconstruct sibling relationships.

References

[1] D. Alexandrov and Y. Kochetov. Behavior of the ant colony algorithm for the set covering problem. *Proceedings of Symposium on Operations Research*, pages 255–260, 2000.

[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] E. K. Baker, L. D. Bodin, W. F. Finnegan, and R. J. Ponder. Efficient heuristic solutions to an airline crew scheduling problem. *AIIE Transactions*, (11):79–85, 1979.

[4] E. Balas. A class of location, distribution and scheduling problems: Modeling and solution methods. In P. Gray and L. Yuanzhang, editors, *Proceedings of the Chinese-U.S. Symposium on Systems Analysis*. J. Wiley and Sons, 1983.

[5] E. Balas and M.C. Carrera. A dynamic subgradient-based branch and bound procedure for set covering. *Operations Research*, 44(6):875–890, 1996.

[6] E. Balas and A. Ho. Set covering algorithms using cutting planes, heuristics, and subgradient optimization: A computaional study. *Mathematical Programming Study*, (12):37–60, 1980.

[7] E. Balas and S.M. Ng. On the set covering polytype: I all facets with coefficients in {0,1,2}. *Mathematical Programming*, (43):57–69, 1989.

[8] E. Balas and S.M. Ng. On the set covering polytype: Ii lifting the facets with coefficients in {0,1,2}. *Mathematical Programming*, (43):1–20, 1989.

[9] J.J. Bartholdi. A gauranteed-acuracy round-off algorithm for cyclic scheduling and set covering. *Operations Research*, (29):501–510, 1981.

[10] J.E. Beasley. An algorithm for set covering problems. *European Journal of Operations Research*, (31):85–93, 1987.

[11] J.E. Beasley. A lagrangean heuristic for set-covering problems. *Naval Research Logistics*, (37):151–164, 1990.

[12] J.E. Beasley and P.C. Chu. A genetic algorithm for the set covering problem. *European Journal of Operations Research*, (94):392–404, 1996.

[13] J.E. Beasley and K. Jornsten. Enhancing an algorithm for set covering problems. *European Journal of Operations Research*, (58):293–300, 1992.

[14] J. Beyer and B. May. A graph-theoretic approach to the partition of individuals into full-sib families. *Molecular Ecology*, 12:2243–2250, 2003.

[15] 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.

[16] M.S. Blouin, M. Parsons, V. Lacaille, and S. Lotz. Use of microsatellite loci to classify individuals by relatedness. *Molecular Ecology*, 5:393–401, 1996.

[17] N. Brauner, C. Dhaenens-Flipo, M.-L. Espinouse, F. Finke, and H. Gavranovic. Decomposition into parallel work phases with application to the sheet metal industry. *Proceedings of International Conference on Industrial Engineering and Production Management*, (1):389–396, 1999.

[18] M.A. Breuer. Simplification of the covering problem with application to boolean expressions. *Journal of the Association of Computing Mchinery*, (17):166–181, 1970.

[19] A. Caprara, M. Fischetti, and P. Toth. Algorithms for the set covering problem. *Annals of Operations Research*, (98):353–371, 2000.

[20] A. Caprara, M. Fischetti, P. Toth, D. Vigo, and P.L. Guida. Algorithms for railway crew management. *Mathematical Programming*, (79):125–141, 1997.

[21] S. Ceria, P. Nobili, and A. Sassano. Set covering problem. In M. Dell'Amico, F. Maffioli, and S. Martello, editors, *Annotated Bibliographies in Combinatorial Optimization*, pages 415–428. J. Wiley and Sons, 1997.

[22] V. Chvatal. A greedy heuristic for the set covering problem. *Math. of Operations Research*, 3(4):233–235, 1979.

[23] T.E. Combs and J.T. Moore. A hybrid tabu search/set partitioning approach to tanker crew scheduling. *Military Operations Research Society Journal*, (9):43–57, 2004.

[24] W. Cook, W. Cunningham, W. Pulleyblank, and A. Schrijver. *Combinatorial Optimization*. Wiley-Interscience Publications, 1998.

[25] Stuart C.Thomas and William G.Hill. Sibship reconstruction in hierarchical population structures using markov chain monte carlo techniques. *Genet. Res., Camb.*, 79:227–234, 2002.

[26] A.V. Eremeev. Algorithm with a non-binary representation for the set covering problem. *Proceedings of Symposium on Operations Research*, pages 175–181, 1999.

[27] U. Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45:634–652, 1998.

[28] M.L. Fisher and P. Kedia. Optimal solution of set covering/partitioning problems using dual heuristics. *Management Science*, (36):674–688, 1990.

[29] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of* NP-*Completeness*. W.H. Freeman and Co., 1979.

[30] T. Grossman and A. Wool. Computational experience with approximation algorithms for the set covering problem. *European Journal of Operations Research*, 101(1):81–92, 1997.

[31] Dan Gusfield. Partition-distance: A problem and class of perfect graphs arising in clustering. *Information Processing Letters*, 82(3):159–164, May 2002.

[32] L.W. Jacobs and M.J. Brusco. Note: A local-search heuristic for large set-covering problems. *Naval Research Logistics*, 42(7):1129–1140, 1995.

[33] A. G. Jones and W. R. Ardren. Methods of parentage analysis in natural populations. *Molecular Ecology*, (12):2511–2523, 2003.

[34] E.L. Lalwer. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, USA, 1976.

[35] Francesco Maffioli and Giulia Galbiati. Approximability of hard combinatorial optimization problems: an introduction. *Annals of Operations Research*, 96:221–236, 2000.

[36] C. Mannino and A. Sassano. Solving hard set covering problems. *Operations Research Letters*, 18:1–5, 1995.

[37] I. Painter. Sibship reconstruction without parental information. *Journal of Agricultural, Biological, and Environmental Statistics*, 2:212–229, 1997.

[38] Christos H. Papadimitriou and Kenneth Seiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications, 1998.

[39] J. Rubin. A technique for the solution of massive set-covering problems with application to

airline crew scheduling. *Transportation Science*, (7):34–48, 1973.

[40] M.R. Salveson. The assembly line balancing problem. *Journal of Industrial Engineering*, (6):18–25, 1955.

[41] F. Shepardson and R.E. Marsten. A lagrangean relaxation algorithm for the two duty period scheduling problem. *Management Science*, (26):274–281, 1980.

[42] Bruce R. Smith, Christophe M. Herbinger, and Heather R. Merry. Accurate partition of individuals into full-sib families from genetic data without parental information. *Genetics*, 158:1329–1338, 2001.

[43] C. Toregas, R. Swain, C. Revelle, and L. Bergman. The location of emergency service facilities. *Operations Research*, (19):1363–1373, 1971.

[44] L. Trevisan. Non-approximability results for optimization problems on bounded degree instances. *Proceedings of the 33rd annual ACM symposium on Theory of computing*, pages 453–461, 2001.

[45] W. Walker. Using the set-covering problem to assign fire companies to fire houses. *Operations Research*, (22):275–277, 1974.

[46] Jinliang Wang. Sibship reconstruction from genetic data with typing errors. *Genetics*, 166:1968–1979, April 2004.