Selection of Multiple SNPs in Case-Control Association Study Using a Discretized Network Flow Approach *

Shantanu Dutt†, Yang Dai‡, Huan Ren†, and Joel Fontanarosa‡

†Department of Electrical and Computer Engineering
‡Department of Bioengineering
SEO, 851 South Morgan Street, Chicago, IL 60607, USA

Abstract. Recent large scale genome-wide association studies have been considered to hold promise for unraveling the genetic etiology of complex diseases. It becomes possible now to use these data to assess the influence of interactions from multiple SNPs on a disease. In this paper we formulate the multiple SNP selection problem for determining genetic risk profiles of certain diseases by formulating novel 0/1 IP formulations for this problem, and solving them using a new near-optimal and efficient discrete optimization technique called *discretized network flow* that has recently been developed by us. One of the highlights of our approach to solving the multiple SNP selection problem is recognizing that there could be different genetic profiles of a disease among the patient population, and it is thus desirable to classify/cluster patients with similar genetic profiles of the disease while simultaneously selecting the right genetic marker sets of the disease for each cluster. This approach coupled with the DNF technique has yielded results for several diseases with some of the highest sensitivities seen so far and specificities that are higher or comparable to state-of-the art techniques, at a fraction of the runtime of these techniques.

Key words: Multiple SNPs, Case-Control Study, Optimization, Discretized Network Flow

1 Introduction

Recent large-scale, high-density genome-wide association (GWA) studies have improved our understanding of the genetic basis of many complex traits. Various published associations have not been replicated in comparable GWA studies, possibly due in part to the omission of interactions among disease-associated loci (called *epistasis*) from many statistical models [13],[9]. Increasing empirical evidence suggests that interactions among loci contribute broadly to complex human diseases. The task in epistatic study is identification of a set of k single nucleotide polymorphisms (SNPs) and the corresponding allele types that are associated with the disease. We call it the *k-SNP marker selection problem*. Much of the recent statistical work has focused on interaction models that have small or no marginal effects at each locus. Since the number of possible interaction combinations among the genotyped markers is astronomical for a large scale case-control association study, it is prohibitive to search one or a very few disease-related interactions among all these combinations. Several methods based on brute-force search have been developed, including the combinatorial partitioning method (CPM) [15], and multifactor-dimensionality reduction (MDR) [17].

* S. Dutt and Y. Dai are corresponding authors for this paper. This work was supported in part by NSF grants CCR-0204097 and CCF-0811855.

While these techniques have been used to effectively analyze data sets of small scale, they cannot be scaled-up for large data sets. More methods can be found in a review paper [14]. A result based on a Bayesian statistical model has indicated feasibility of genome scale epistatic analysis [20]. The model was applied to an association study set with 96,932 markers genotyped from 146 individuals (96 affected and 50 controls) for 2-SNP and 3-SNP mark set selections The run time was about 5 hours on a Pentium M 1.6GHz laptop with 512 Mb memory.

Brinza and co-authors considered the problems of searching for the most disease-associated and the most disease-resistant k-SNP marker sets [3],[4]. Combinatorial methods, such as greedy algorithms, were proposed to search these k-SNP marker sets. Using the selected marker set the authors further considered the disease susceptibility prediction problem, i.e., predict an individual's disease status based on the marker set. The algorithms have been shown to outperform other machine learning methods on three small-scale data sets. Recently, the above approach was further extended to searching for a k-SNP marker set which has the best odds ratio, a criterion often used in disease association studies [5].

In this paper, we propose a novel optimization-based approach to the k-SNP marker selection problem for large-scale SNP data. The detection of k-SNP mark set is formulated as 0/1 integer programming problems. Our formulation can simultaneously discover subgroups of cases and their corresponding best marker sets. The core technique for fast near-optimal solutions of the 0/1 IP problem is a recently developed new methodology called *discretized network flow* (DNF). DNF is a general computing framework for obtaining high-quality discrete optimization problems (DOPs) solutions in tractable run-times. The efficacy of DNF has been established in the realm of VLSI CAD for many hard DOPs [7],[8],[16]. DNF combines the computational efficiency of continuous optimization methods, in that it uses network flow, an optimal continuous optimization technique, as its core algorithmic process, with novel discretization techniques so that near-optimal legal discrete solutions are efficiently obtained.

The proposed approaches to epistasis analysis in GWAs SNP data sets are based on novel 0/1 IP formulations of multi-locus marker detection and use DNF to solve these formulations. The effectiveness of the approach are evaluated using 5 data sets in a previous study [5]. The new proposed method significantly outperformed previous methods in most cases: sensitivity[1] and specificity[2] in a 5-fold cross-validation test increased by 81% and 38.8%, respectively, from the results of MDR, and by 14% and -9.2% (the negative value indicates deterioration), respectively, from the results of the combinatorial method CPS [4], with a runtime that is a fraction of the runtimes of these methods.

2 Method

2.1 0/1 Integer Programming (IP) Formulations

We describe here our 0/1 IP models for optimizing the selection of a set of SNPs and their associated allele so that the chosen SNP-allele pairs are strong distinguishing markers between case and control. We consider un-phased genotype data of an experiment involving m_1 and m_2 individuals in case and control groups respectively. For each SNP there are 3 allele types. Given a pair $p_{i,j}$ of

[1] The percentage of cases correctly identified with the selected SNP marker set.
[2] The percentage of controls correctly identified with the selected SNP marker set.

SNP i and allele j (henceforth we will use the terms "SNP-allele pair" or just "allele" to refer to a $p_{i,j}$), we define $c_{i,j}(x) = 1$ if the x'th individual P_x in the case group has allele j at SNP i, and $c_{i,j}(x) = 0$ otherwise. Similarly, we also define $h_{i,j}(z) = 1$ if the z'th individual NP_z in the control group has allele j at SNP i, and $h_{i,j}(z) = 0$ otherwise. The presence of allele $p_{i,j}$ in an individual is denoted by marker $p^1_{i,j}$, while its absence is denoted by marker $p^0_{i,j}$.

We define the per-case *benefit* $b_{i,j}(x)$ of a SNP-allele pair $p_{i,j}$ for an individual x as:

$$b_{i,j}(x) = |(c_{i,j}(x) - \frac{\sum_{z=1}^{m_2} h_{i,j}(z)}{m_2}| \tag{1}$$

$b_{i,j}(x)$ is a good indicator of discriminative ability between case individual x and the control group for allele j at SNP i. Furthermore, $b_{i,j}(x)$ is also a correct indicator of the specificity of allele $p_{i,j}$ as we show below.

Claim 1 *The $b_{i,j}(x)$ definition is consistent with the specificity of allele $p_{i,j}$.*

Proof: There are two cases.
Case 1: $c_{i,j}(x) = 1$, i.e., $p_{i,j}$ is present in P_x. Then the second term in Eqn. 1 is the fraction of controls that also contain $p_{i,j}$. Thus higher $b_{i,j}(x)$ is, lower is this fraction, which means high specificity for $p_{i,j}$.
Case 2: $c_{i,j}(x) = 0$, i.e., $p_{i,j}$ is absent in P_x. The second term in Eqn. 1 can be re-stated as: [1 − fraction of controls in which $p_{i,j}$ is absent], which is also the definition of $b_{i,j}(x)$, since $c_{i,j}(x) = 0$. Thus higher $b_{i,j}(x)$ is, smaller is the fraction of controls for the $p^0_{i,j}$ marker, and thus higher is the specificity for this marker.

Note that in each case, $b_{i,j}(x)$ is, in fact, exactly the specificity of $p^{c_{i,j}(x)}_{i,j}$. ◊

A good target set of SNP-allele combinations for the entire case group can be one in which the selected SNP-allele pairs $p_{i,j}$s have the same value of $c_{i,j}()$ for all case individuals and whose sum of benefits is the maximum (greatest distinction from the controls). We thus formulate the following 0/1 integer programming (IP) for the k-SNP selection problem.

We first define the benefit-based similarity metric $s(x, y, i, j, val)$ between two individuals in the case group P_x and P_y for a SNP-allele pair marker $p^{val}_{i,j}$ ($val \in \{0, 1\}$) as:

$$s(x, y, i, j, val) = \begin{cases} (b_{i,j}(x))^\alpha + (b_{i,j}(y))^\alpha & \text{if } c_{i,j}(x) = c_{i,j}(y) = val \\ -\infty & \text{otherwise.} \end{cases} \tag{2}$$

The α parameter above magnifies (when $\alpha > 1$) or shrinks (when $\alpha < 1$) the ratio of benefits of different alleles. This is useful when there are constraints on the selection of alleles and $\alpha > 1$ can be used to give a magnified priority to the selection of alleles with high benefit. Further, let $d_{i,j}(val)$ be a 0/1 variable which indicates if $p^{val}_{i,j}$ is selected as a marker in the SNP-allele set ($d_{i,j}(val) = 1$) or not ($d_{i,j}(val) = 0$). The 0/1 IP formulation for the problem of the maximum-benefit SNP/allele set election is:

$$\begin{array}{ll} \text{Maximize :} & \sum_{p^{val}_{i,j}} \sum_{1 \le x \le m_1} \sum_{1 \le y \le m_1, y \ne x} s(x, y, i, j, val) \cdot d_{i,j}(val) \\ \text{Subject to :} & (i) \sum_{j=1}^{3} d_{i,j}(0) + d_{i,j}(1) \le 1, \forall \text{ SNPs } i. \\ & (ii) \sum_{p_{i,j}} (d_{i,j}(0) + d_{i,j}(1)) \le k. \end{array} \tag{3}$$

It is easy to see from the definition of $s(x, y, i, j, val)$ ($-\infty$ value for mismatched alleles between a pair of individuals in the case group), that in order

to maximize the objective function, no SNP-allele pairs $p_{i,j}$ will be selected that differ in $c_{i,j}$ value among any pair of individuals in the case group. Thus only common SNP-alleles across all individuals in the group will be selected.

Simultaneous Patient Clustering and k-SNP Marker Selection One issue that complicates the marker selection problem is that the genetic reasons of a disease for patients with different ethnic backgrounds can be different, and it can also be different within each ethnic group. Therefore, in order to improve the accuracy of selected markers, it is desirable to incorporate a clustering/classification step for individuals of the case group simultaneously with the marker selection process so that clusters are automatically formed based on similarities of genetic disease markers. Different markers can thus be selected for different clusters to best match their genetic disease profile. The IP problem given in Eqn. 3 put all individuals in the case group in one cluster and tries to find the best marker for this cluster, which may not be very strong since we are forcing marker selection into patients with potentially different genetic disease profiles. The formulation below is for partitioning individuals in the case group into up to G clusters, each with similar genetic disease profiles and simultaneously finding their best SNP-allele markers.

$$\begin{array}{ll} \text{Maximize :} & \sum_{1\le g\le G}\sum_{p^{val}_{i,j}}\sum_{1\le x\le m_1}\sum_{1\le y\le m_1} s(x,y,i,j,val)\cdot b^g_x\cdot b^g_y\cdot d^g_{i,j}(val) \\ \text{Subject to :} & (i)\sum_{j=1}^{3} d^g_{i,j}(0)+d^g_{i,j}(1)\le 1, \forall \text{ SNPs } i \text{ and } \forall \text{ clusters } g. \\ & (ii)\sum_{1\le g\le G} b^g_x = 1, \forall x.\ \ (iii)\sum_{p_{i,j}}(d^g_{i,j}(0)+d^g_{i,j}(1))\le k, \forall g. \end{array} \tag{4}$$

where G is an upper bound on the number of patient groups/clusters, b^g_x is a 0/1 variable that is 1 if P_x is chosen to be in cluster g, and 0 otherwise, $d^g_{i,j}(val)$ is a 0/1 variable that is 1 if $p^{val}_{i,j}$ is chosen as a SNP-allele marker for cluster g and is 0 otherwise, and k is an upper bound on the number of $p_{i,j}$'s selected in the marker set for each cluster. We note that an individual can belong to only one cluster, while $p^{val}_{i,j}$ may be chosen to be in the SNP-allele marker set for multiple clusters (both $p^0_{i,j}$ and $p^1_{i,j}$ of course cannot be in the same marker set).

The optimization problems formulated above involve large numbers of 0/1 integer variables that are indicative of problems which even state-of-the art MIP solvers (e.g. CPLEX) could fail to solve within a reasonable time frame. We propose network flow formulations of the problems, which can be solved approximately and efficiently by using the discretized network flow method.

2.2 The Discretized Network Flow Technique

The class of problems to which discretized network flow (DNF) can be applied are DOPs that can be modeled as mixed integer programming (MIP) problems with linear and non-linear (polynomial, min and max) objective functions and constraints; these encompass a very large class of DOPs. The basic idea of this technique can be described as follows.

A network flow graph G is a directed graph in which each arc e has a capacity $cap(e)$ and a cost $cost(e)$; see Fig. 1. The capacity of an arc indicates the maximum amount of flow that can pass through the arc; the cost of an arc is the cost incurred per unit flow through the arc. The *minimum cost flow problem* in G is to find a way to pass a certain amount of flow through G from a source node S to a sink node T that has minimum cost. Network flow is an elegant

continuous optimization technique that has found applications in many domains [1].

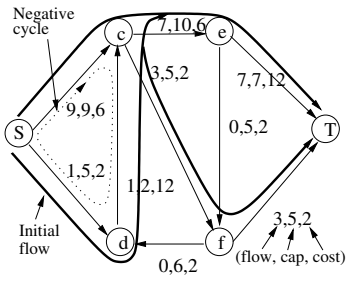

Fig. 1. *A feasible flow for sending 10 units of flow from source (S) to sink (T) is shown by curved dark lines. Arc labels are arranged as* $(flow, capacity, cost)$.

In standard network flow [1], the flow through an arc can be any continuous (i.e., real) value, and if there is incoming flow into a node u, there can be outgoing flow into any subset of outgoing arcs from u. In order to solve DOPs using a network flow approach (due to its time efficiency), we need flows in the graph G to have certain discrete properties. Briefly, the two most important of these are:

(1) *Discrete Arcs* : Some arcs e will need to have have a "binary-valued" discrete cost and flow amount structure, i.e., such a *discrete arc* can only have two (flow amount, cost incurred) pairs: $\{(0,0),(cap(e),cost(e))\}$. This means that any initial flow amount f through e will incur a cost of $cost(e)$ (irrespective of f), and, if $f < cap(e)$, then any subsequent flow through e will incur 0 cost. Such arcs are called *discrete arcs*, and their cost structure makes the incurred cost a concave function of the flow amount f. We thus use the near-optimal concave min-cost flow algorithm of [10] in all our network flow computations.

(2) *Mutual Exclusiveness* : For some nodes u at most one of their output arcs and/or one of their input arcs can have flow in them; see Fig. 2. We call this the *mutually exclusive output arcs (MEA)* requirement.

Of these, the MEA requirement is central in the solution of the 0/1 IPs for multiple allele selection discussed in Subsec. 2.1. Our DNF technique uses special algorithms and arc cost formulations using non-objective-function based costs (in addition to, of course, objective-function based costs) in order to achieve the above types of discretizations without sacrificing much in optimality [7], [8], [16]. We briefly discuss below our MEA satisfaction technique.

Satisfying MEA Constraints in DNF In general, the cost of arcs in a n/w graph G is determined based on the objective function being minimized. However, for the purpose of MEA satisfaction (and, in fact, for some other flow constraints as well), for each arc e in an MEA set, we add to its function based cost $C(e)$, a discrete *base cost* $C'(e)$, which is a constant for all edges e to which it is applied; thus its total cost $cost(e) = C'(e) + C(e)$. For various flow constraints, including MEA, that we considered, an invalid flow will always incur at least an extra C' cost. The $C'(e)$ cost is thus kept large enough so that a min-cost invalid flow always incurs more total cost than a valid min-cost flow (even though the objective-function part of this cost, the C-cost, incurred by an invalid flow may be less than that incurred by a valid flow) [16]. The correctness of this technique for MEA satisfaction has been established in [16], and is re-stated below using the terminology of this paper.

Theorem 1 *[16] Any min-cost flow in G with added C' costs on MEA arcs will satisfy all MEA constraints.*

2.3 Application of DNF to *k*-SNP Selection

In our DNF model, we approximate the 0/1 IP of Eqn. 4 by obtaining all clusters via simultaneous recursive bi-partitioning and allele marker set selections. We

start with the set of all cases as the initial single cluster C_1, and then recursively bi-partition each cluster C_i at the current level of bi-partitioning into two clusters $C_{i,1}$ and $C_{i,2}$, and simultaneously select their allele marker sets. The process continues as long as the overall specificity, by so doing, increases, and the upper bound of G clusters is not violated[3].

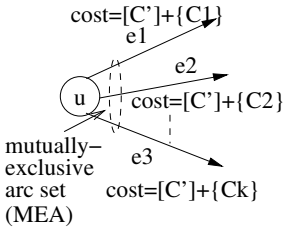

Fig. 2. *An MEA set and DNF structure for satisfying MEA requirements. The C' costs shown are discrete, indicated by square brackets; the other costs indicated by curly brackets are continuous.*

The network flow model for the bi-partition plus k-SNP selection problem is shown in Fig. 3. For each cluster C_i, the network flow graph G_i consists of two subgraphs $G_{i,1}, G_{i,2}$ corresponding to the two clusters to be formed, as shown in Fig. 3(a). In each subgraph, there are two levels of duplicated case "meta nodes" for each case individual. These two levels in each $G_{i,r}$ $(r = 1, 2)$ are connected by a complete meta bipartite graph, where a "meta edge" connects each pair of meta nodes P_x, P_y. As shown in Fig. 3b, each meta edge is a collection of allele-to-allele connections between alleles of P_x and P_y with the same allele values (0 or 1) that have costs $-s(x, y, i, j, val)$. A case P_x is selected to be in $C_{i,r}$ if there is flow through its two meta nodes in the two levels of $G_{i,r}$. An allele $p_{i,j}$ is selected as a marker in $C_{i,r}$ if flow passes through the $p_{i,j}$ nodes in $G_{i,r}$ corresponding to each P_x selected in $C_{i,r}$ (we shortly show that if a min-cost flow passes through any $p_{i,j}$ node of any selected P_x, it will pass through all $p_{i,j}$ nodes of all selected P_x's in a cluster; this implies that a common set of (up to k) alleles will be selected for each selected P_x in a cluster, and hence a correct allele marker set is selected for each cluster). The min-cost flow[4] through subgraphs $G_{i,1}$ and $G_{i,2}$ will select those P_x's in each subgraph that minimizes the total cost of all selected allele-to-allele arcs among all the selected P_x's (this corresponds to maximizing the objective function of Eqn. 4).

In order for the flow to partition C_i into two valid clusters with valid allele marker selections for them, two sets of constraints need to be satisfied: (1) each P_x is selected in only one of $C_{i1,}$ and $C_{i,2}$. (2) In each $G_{i,r}$, the flow goes through the same set of allele nodes in each P_x meta node that the flow selects.

The first constraint is satisfied via MEA requirements on input arcs to case meta nodes in the first level of each $G_{i,r}$, and on the outgoing arcs from case meta nodes in the second level of each subgraph; see Fig. 3. This causes the flow to pass through P_x meta nodes in only one subgraph.

Fig. 3(b) shows the detailed structure of the meta-node of a case P_x, as well as the details of connection between the meta nodes of cases P_x and P_y in the two levels. The structure in each individual meta-node is a tree with allele nodes for each SNP at the leaf level. Let m be the number of cases in C_i to be partitioned. An incoming flow of amount km to each meta-node in the 1st level is distributed through the tree structure to k leaf allele nodes (this indicates that the corresponding k SNP-allele pairs are selected as markers). Note that the incoming arcs to leaf allele nodes are discrete arcs with capacity m and objective function independent cost C'. Hence, to minimize total cost, flow is only distributed on k such arcs with full flow of amount m on each arc, i.e., to k leaf allele nodes.

[3] In our DNF modeling, we have not explicitly set an upper bound on the number of clusters; however, the number of clusters determined naturally turns out to be a small constant in the range $[6, 16]$.

[4] We solve the maximization problem of Eqn. 4 by using min-cost network flow by setting the cost of arcs for $p_{i,j}$ selection among node pairs P_x, P_y to be the negative of $s(x, y, i, j, val)$, which is the contribution of this selection to the maximization objective of Eqn. 4.

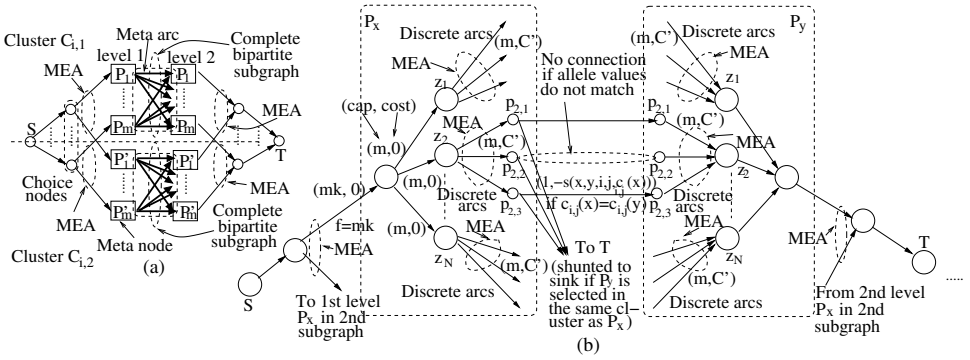

Fig. 3. *(a) High level network flow model for the bipartition problem. (b) Detailed patient meta node structure, and connection between patient meta nodes.*

The meta arcs between P_x meta nodes are each a set of connections between corresponding alleles, i.e., the SNP-allele pair $p_{i,j}$ present in the P_x meta-node is connected to that in the P_y meta node if and only if $c_{i,j}(x) = c_{i,j}(y)$; the cost of this arc is $-s(x, y, i, j, val)$. When both P_x and P_y are selected in one cluster, the allele to allele connections make sure that the flow passes through (i.e., select) the same k alleles as markers for the two cases. The cost of the arcs between the two P_x meta nodes in the two levels of a subgraph is $-\infty$. Hence, if a meta node P_x has a flow through it in level 1 (meaning it is selected to be in the corresponding cluster), then part of this flow will also go through the level-2 P_x meta node in that subgraph; the allele markers selected by the flow are also the same in both P_x meta nodes in the two levels.

The second constraint mentioned above is satisfied as follows. Due to the complete bipartite connection between case meta nodes at the two levels of each subgraph, all cases selected in a cluster subgraph will also select the same k alleles as markers. Let us assume that this is not the case, so that the sets of k alleles selected for two meta nodes P_x, P_y that are selected in level 1 to be in the same cluster are different. Consider the P_y meta node in level 2. The flow through the k alleles of P_x will be forced into the same k alleles of the 2nd-level P_y via the allele-to-allele connections, and the flow through each allele incurs a discrete C' cost. The flow from the k alleles of the 1st-level P_y will also be forced through the corresponding alleles in the 2nd-level P_y. Since the two sets of k alleles are not the same, there will be flow through at least $(k + 1)$ alleles in the 2nd level P_y. This thus incurs $(k + 1)$ C' costs, instead of only k C' costs for a valid flow—in which a consistent set of k alleles are selected for all meta-nodes in the same cluster. Thus the invalid flow that goes though at least $(k+1)$ alleles in the 2nd-level P_y meta-node cannot be a min-cost flow; in other words, a min-cost flow will always select a common set of k alleles across all selected meta-nodes in each subgraph.

2.4 Improved 0/1 IP Formulation and DNF Model with Explicit Specificity Consideration

As explained in Sec. 2.1, the formulation of Eqn. 4 is geared toward choosing a common set of allele markers for each cluster, which implies a sensitivity of 1 for its cluster. This in turn implies a high sensitivity for the entire set of allele marker

sets (one marker set per cluster)[5]. As established in Claim 1, the definition of $b_{i,j}(x)$ is its specificity. Thus the selection of high-benefit markers (as per the objective function of Eqn 4) is conducive to high specificity for each allele marker set M_i, and thus also to high specificity for the set $\mathcal{M} = M_1, \ldots, M_G$ of allele marker sets, assuming a uniform distribution of mismatching controls across alleles. For example, consider a 2-element allele marker set $M_i = (p^0_{i1,j1}, p^1_{i2,j2})$, and let each allele marker have a specificity of 0.6, i.e., the probability of a control mis-matching each allele marker is 0.6. Assuming an uniform distribution of controls that mismatch each allele marker, the probability of a control mismatching marker set M_i is the probability that it mismatches either allele marker = 1 - (the probability that it matches both allele markers) $= 1 - (1 - 0.6)^2 = 0.84$. However, the mismatching controls may not be uniformly distributed, and if there is some clustering of the controls that mismatch the alleles of M_i, then the specificity of M_i could be much lower than 0.84. For example, if the set of controls that mismatch both allele markers are the same, then the specificity of M_i is only 0.6. Thus a more direct method is needed for identifying allele markers so that the specificity of each M_i and that of $\mathcal{M}$ is high. Toward that end, we formulate a modified 0/1 IP and then discuss its DNF model below.

We first define the function $dis(z, i, j)$ between a control individual NP_z and all case individuals that mismatch the allele value of $p_{i,j}$ in NP_z as:

$$dis(z, i, j) = k \cdot \frac{m_1}{m_2} \cdot m_{avg} \cdot Avg_{\{(P_x | c_{i,j}(x) \neq c_{i,j}(z)\}} 2(b_{i,j}(x))^{\alpha} \tag{5}$$

where Avg_S is the average function over the set S and m_{avg} is the average number of cases in a cluster. The above function is the benefit associated with NP_z mismatching the $p_{i,j}$ value present in a subset of cases, and is tuned so that the total benefit associated with a unit percentage contribution to specificity (due to controls mismatching alleles selected in marker sets) is approximately equal to the total benefit (see objective functions of Eqns. 4 and 6) associated with a unit percentage contribution to sensitivity.

Let $\{C_g\}$ be the set of clusters, and $d^g_{i,j}(*)$ be a 0/1 variable that is 1 if and only if $p_{i,j}$ is not selected as an allele marker in cluster C_g. Then a control NP_z does not match any $p_{i,j}$ selection in any of the allele marker sets $\{M_g\}$ if $\prod_{g=1}^{G}[d^g_{i,j}(not((c_{i,j}(z))) + d^g_{i,j}(*)] = 1$. We thus include these terms for each $p_{i,j}$ in the maximization objective function of the new 0/1 IP given below.

$$\begin{array}{ll} \text{Maximize :} & \sum_{1 \leq g \leq G} \sum_{p^{val}_{i,j}} \sum_{1 \leq x \leq m_1} \sum_{1 \leq y \leq m_1} s(x, y, i, j, val) \cdot b^g_x \cdot b^g_y \cdot d^g_{i,j}(val) \\ & + \sum_{p_{i,j}} \sum_{1 \leq z \leq m_2} \prod_{g=1}^{G} [d^g_{i,j}(not((c_{i,j}(z))) + d^g_{i,j}(*)] \cdot dis(z, i, j) \\ \text{Subject to :} & (i) \sum_{j=1}^{3} d^g_{i,j}(0) + d^g_{i,j}(1) + d^g_{i,j}(*) = 1, \forall \text{ SNPs } i \text{ and } \forall \text{ clusters } g. \\ & (ii) \sum_{1 \leq g \leq G} b^g_x = 1, \forall x. \;\; (iii) \sum_{p_{i,j}} (d^g_{i,j}(0) + d^g_{i,j}(1)) \leq k, \forall g. \end{array} \tag{6}$$

[5] For a case-clustered solution with multiple allele marker sets $\mathcal{M} = M_1, \ldots, M_G$, each marker set M_i corresponding to cluster C_i, a case has a match with the set of marker solutions $\mathcal{M}$ if it has a match with *any* allele marker set M_i; it then contributes to the sensitivity of $\mathcal{M}$ (i.e., to the true positive [TP] number; see Eqn. 7). Conversely, a control has a mismatch with $\mathcal{M}$ if the individual has a mismatch with *all* marker sets in $\mathcal{M}$, and only then does it contribute to the specificity of $\mathcal{M}$ (i.e., to the true negative [TN] number; see Eqn. 7).

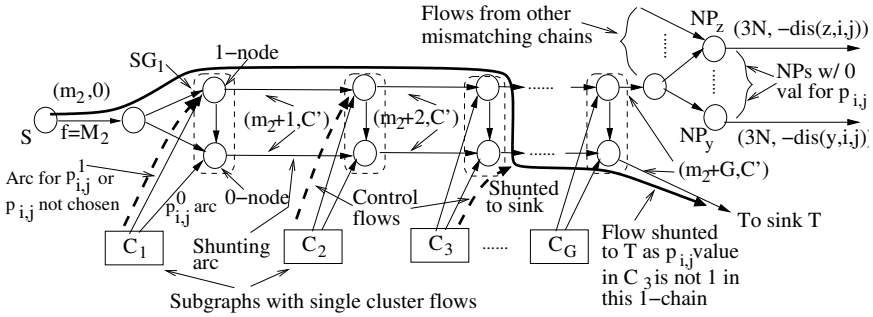

Fig. 4. *DNF model for computing the product of 0/1 variables: a 1-chain for allele $p_{i,j}$. The main flow (from S) is shown by a dark solid curve, and control flows are shown by dark dashed curves.*

The term $\prod_{g=1}^{G}[d_{i,j}^{g}(not((c_{i,j}(z))) + d_{i,j}^{g}(*)] \cdot dis(z,i,j)$ in the objective function forces a selection of alleles $p_{i,j}$ in marker sets of clusters that most controls will mismatch in their values (recall that value=1 indicates presence of $p_{i,j}$ and value=0 indicates absence). The corresponding DNF structure corresponding to this term for each $p_{i,j}$ are two chain structures, one for marker $p_{i,j}^{1}$ called the 1-chain and the other for marker $p_{i,j}^{0}$ called the 0-chain. Each chain has G sequentially connected "gateway" subgraphs SG_g's through which a flow amount of m_2 coming from the source S can potentially pass; see Fig. 4. Each SG_g is controlled by a flow coming into it from the network subgraph in which cluster C_g is being formed. Each SG_g has a 0-node and a 1-node, and if the chain is a 1-chain, and either $p_{i,j}^{1}$ is selected as a marker for C_g or no $p_{i,j}$ (i.e., neither $p_{i,j}^{1}$ nor $p_{i,j}^{0}$) is selected as its marker, a unit control flow will come in from C_g into the 1-node of SG_g. There it incurs a C' cost on the arc leading to SG_{t+1}, which means a min-cost flow will choose to push the flow from S along this arc and into SG_{t+1}. If, however, $p_{i,j}^{0}$ is selected as a marker for C_g, then the control flow from C_g goes into the 0-node of SG_g, which incurs a C' cost in the shunting arc of SG_g that leads to the sink T. This means that a min-cost flow will choose to divert the flow from S to this shunting arc and the flow thus does not go through the 1-nodes of this 1-chain, and is finally shunted to the sink as shown in Fig. 4. The structure for a 0-chain for $p_{i,j}$ is analogous. It is thus clear that the flow from S will pass through this chain iff the term $\prod_{g=1}^{G}[d_{i,j}^{g}(not((c_{i,j}(z))) + d_{i,j}^{g}(*)] = 1$.

If flow reaches the end of, say, a 1-chain for $p_{i,j}$, it is then diverted to a structure for each NP_z that has marker $p_{i,j}^{0}$ where it incurs a $-dis(z,i,j)$ cost. Thus a min-cost flow choosing such a path means that the corresponding NP_z's through which the flow finally passes do not match the $p_{i,j}$ marker selection, if any, in any of the clusters. This implies a contribution toward high specificity of the set of marker sets $\{M_t\}$. 1- and 0-chains for each $p_{i,j}$ supplement the structure of Fig. 3 to solve the 0/1 IP formulation of Eqn. 6. Compared to solving the formulation of Eqn. 4, the solutions to the formulation of Eqn. 6 provided an 11.5% average improvement in specificity with an 8.4% average deterioration in sensitivity, for the data sets for the five diseases discussed in Sec. 3.

3 Computational Study

3.1 Data Sets

We consider 5 data sets which were used in [5] for the evaluation of our method:

1) Crohn's disease: This data set consists of genotypes of 103 SNPs from 144 Crohn's disease patients and 243 healthy controls. The 103 SNPs locate on a 616 KB region of human Chromosome 5q31 that may contain a genetic variant responsible for the disease. The cases and controls are individuals from 129 trios [6].
2) Autoimmune disorder: This data set consists of genotypes of 108 SNPs from 384 cases of autoimmune disorder and 652 controls. The SNPs are selected from a 330KB of human CAN containing gene CD28, CTLA4 and ICONS, which are proved related to autoimmune disorder [19].
3) Tick-borne encephalitis: This data set consists of genotypes of 41 SNPs of 21 patients with tick-borne encephalitis virus and 54 patients with mild disease [3].
4) Rheumatoid arthritis: This data set consists of genotypes of 2300 SNPs from 460 patients with rheumatoid arthritis and 460 controls. The SNPs are selected by Illumina for an approximately 10KB region of chromosome 18q that showed evidence for linkage in the U.S. and French linkage scans [3].
5) Lung cancer: This data set consists of genotypes of 141 SNPs from 322 German male smokers with lung cancer and 273 age-matched healthy smokers. The 141 SNPs are selected from a total of 83,715 SNPs that had been screened using genome-wide DNA pooling strategy, because they showed putative allelic imbalance between case and control DNA pools [18].

3.2 Results

We use the 5-fold cross-validation procedure to evaluate our method. The criteria for evaluation of the predictive ability of the algorithm are sensitivity (Sens) and specificity (Spec). Sens and Spec are defined as follows.

$$\text{Sens} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad \text{and} \quad \text{Spec} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{7}$$

where TP, FP, TN and FN are the numbers of true positives, false positives, true negatives and false negatives, respectively, from the 5-fold CV test (see footnote 4, p. 7, for an explanation of how these numbers are determined for our clustering-based multiple marker sets).

	# of Clusters	**Sens (%)**	**Spec (%)**	**Runtime (sec)**	**# of Clusters**	**Sens (%)**	**Spec (%)**	**Runtime (sec)**
Data set	$k = 4$, $\alpha = 1.5$				$k = 5$, $\alpha = 1.5$			
Autoimmune disorder	11	71.1	57.1	2941	12	84.1	67.5	3458
Crohn's disease	9	65.5	67.3	3204	12	84.7	71.2	3651
Tick-borne encephalitis	6	83.3	69.2	801	6	100.0	96.9	720
Lung cancer	8	72.3	69.1	3972	14	84.2	83.2	5240
Rheumatoid arthritis	13	75.0	63.4	19440	13	85.2	71.7	21240
	$k = 7$, $\alpha = 1.5$				$k = 10$, $\alpha = 1.5$			
Autoimmune disorder	16	86.5	72.4	3971	16	86.5	74.6	4025
Crohn's disease	14	84.7	74.0	4044	16	86.1	75.3	4400
Tick-borne encephalitis	6	100.0	96.9	824	6	100.0	96.9	820
Lung cancer	16	85.1	83.9	5354	16	84.4	85.0	5517
Rheumatoid arthritis	13	83.4	75.0	22320	14	85.0	77.8	23060

Table 1. *Sensitivity, specificity and runtimes of the 5-fold CV procedure using the DNF model to solve the 0/1 IP given in Eqn. 6.*

The results of our model using the 5-fold CV procedure are shown in Table 1. We also used the MDR method [17] with $k = 4, 5$ for comparison, since MDR

could not finish for larger k's. For $k = 5$ a random search option was used for the rheumatoid arthritis data set, due to the large number of SNPs in it.

Our model achieved the best sensitivity and specificity when using $k = 10$. Even for $k = 5$, our proposed model outperformed MDR significantly. The percentages of sensitivity (resp. specificity) increase reached 70.8 (67.1), 54.9 (12.1), 100.0 (42.5), 107.3 (31.3), and 72.5 (41.3) for Crohn's disease, autoimmune disorder, tick-borne encephalitis virus, rheumatoid arthritis, and lung cancer, respectively. Note also that except for tick-borne encephalitis (which has very few cases), the numbers of clusters formed for the other four diseases are very similar, even though the number of cases across these disease data sets vary from 144 to 460. This seems to indicate that natural clusters that are independent of the number of cases are being formed. Thus assuming that the cases in our data sets are representative of the general case population, the number of clusters for any of these diseases would probably remain unchanged even if the number of cases increases significantly.

	k=4		k=5	
Data set	**Sens(%)**	**Spec(%)**	**(Sens)(%)**	**Spec(%)**
Autoimmune	50.9	55.3	54.3	60.2
Crohn's	48.6	50.0	49.6	42.5
Tick-borne	52.1	67.2	50.0	67.5
Lung cancer	50.0	57.8	48.8	58.9
Rheumatoid	49.7	54.9	41.1	54.6

Table 2. *Sensitivity and specificity of the MDR method [17] obtained from the 5-fold CV.*

We also compared the results of our method and with those of the CPS method. Our method increased sensitivity by 7.6%, 7.7%, 26.6% for Crohn's disease, autoimmune disorder, and tick-borne encephalitis virus, respectively. However, the specificity was decrease by 16.3% and 16.3% for the first two data sets, but increased by 4.9% for the last data set. Our methods are significantly faster than the CPS. For example, it only took our method 820 seconds for tick-borne data, but it took the CPS method 6.3 hours, even though we used a somewhat slower CPU than that used by CPS (a 1.8 GHz Pentium M vs. a 3.2 Ghz Pentium 4).

	DNF			**CPS**		
Data set	**Sens (%)**	**Spec (%)**	**runtime (sec)**	**Sens (%)**	**Spec (%)**	**Runtime (hour)**
Crohn's	86.1	75.3	4400	80.0	89.9	1189
Tick-borne	100.0	96.9	820	80.2	92.4	6.3
Autoimmune	86.5	74.6	4025	79.0	89.1	17400

Table 3. *Sensitivity, specificity and runtimes obtained for the DNF method and the CPS method [4] for different data sets. The DNF results are from the 5-fold cross-validation procedure, and the CPS results are reported from [4], which were obtained from a leave-out cross-validation procedure.*

4 Conclusions

We proposed 0/1 IP problems to identify k-SNP marker sets that predict an individual's disease status from their genotype data. Our method demonstrated significant improvement in performance compared with several existing methods. Using the novel DNF technique our model can simultaneously detect multiple k-SNP marker sets which correspond to different genetic subgroups from data of large scale case-control studies, and do so very time-efficiently. With further refinement of the model, our method has promise for the analysis of genome-wide association study data.

Acknowledgments. We would like to thank Dr. Dumitru Brinza for providing the data sets used in this work.

References

1. Ahuja, R.K.K., Magnanti, T.L., Orlin, J.B.: Network Flows: Theory, Algorithms, and Applications. Pearson Education (1993).
2. Atamurk, A., Savelsbergh, M.: Integer Programming Software Systems. Annals of Operations Research. 140(1), 67-124 (2005).
3. Brinza, D., Zelikovsky, A.:Combinatorial Analysis of Disease Association and Susceptibility for Rheumatoid Arthritis SNP Data. in Proc. of 15th Genetic Analysis Workshop (GAW15), pp. 6–11 (2006).
4. Brinza, D., Zelikovsky, A.: Combinatorial Methods for Disease Association Search and Susceptibility Prediction In: Bucher, P., Moret, B.M.E.(Eds.): WABI 2006. LNCS, vol. 4175, pp. 286–297. Springer, Heidelberg (2006).
5. Brinza, D., Zelikovsky, A.: Design and Validation of Methods Searching for Risk Factors in Genotype Case-Control Studies. Journal of Computational Biology 15, 81–90 (2008).
6. Daly, M.J., Rioux, J.D., Schaffner, S.F., Hudson, T.J., Lander, E.S.: High-resolution haplotype structure in the human genome. Nat. Genet. 29, 229–232 (2001).
7. Dutt, S., Ren, H.: Discretized Network Flow Techniques for Timing and Wire-Length Driven Incremental Placement with White-Space Satisfaction. accepted for publication in IEEE Trans. of VLSI (2008).
8. Dutt, S., Ren, H., Suthar, V.: A Network-Flow Approach to Timing-Driven Incremental Placement for ASICs. Proc. IEEE Int'l Conf. CAD (ICCAD), pp. 375–382 (2006).
9. Hirschhorn, J. N., Daly, M. J.: Genome-wide association studies for common diseases and complex traits. Nature Rev. Genet. 6, 95 (2005).
10. Nahapetyan A., Pardalos P.: Adaptive Dynamic Cost Updating Procedure for Solving Fixed Charge Network Flow Problems. Computational Optimization and Appl. Jour. , 39(1), pp. 37-50, 2008.
11. Li, J.: A novel strategy for detecting multiple loci in Genome-Wide Association Studies of complex diseases. International Journal of Bioinformatics Research and Applications 4, 150–163 (2008).
12. Li, J.: Prioritize and Select SNPs for Association Studies with Multi-Stage Designs. Journal of Computational Biology 15, 241-257 (2008).
13. McCarthy, M. I. et al:. Genome-wide association studies for complex traits: consensus, uncertainty and challenges. Nat Rev Genet 9, 356–369 (2008).
14. Musani, S. K. et al. Detection of Gene Gene Interactions in Genome-Wide Association Studies of Human Population Data. Hum Hered 63, 67–84 (2007).
15. Nelson, M., Kardia, S., Ferrell, R.: A combinatorial partitioning method to identify multi-locus genotypic partitions that predict the quantitative trait variation. Genome Res. 11, 2115 (2001).
16. Ren, H., Dutt, S.: Algorithms for Simultaneous Consideration of Multiple Physical Synthesis Transforms for Timing Closure. Accepted for publication, Proc. IEEE Int'l Conf. CAD (ICCAD) (2008).
17. Ritchie, M. D., Hahn, L. W., Moore, J. H.: Software for MDR and MDR Permutation Testing module Power of multifactor dimensionality reduction for detecting gene-gene interactions in the presence of genotyping error, missing data, phenocopy, and genetic heterogeneity. Genet Epidemiol 24, 150 - 157 (2003).
18. Spinola, M. et al: Association of the PDCD5 Locus With Lung Cancer Risk and Prognosis in Smokers. J Clin Oncol 24, 1672-1678 (2006).
19. Ueda, H. et al.: Association of the T-cell regulatory gene CTLA4 with susceptibility to autoimmune disease. Nature 423, 506–511 (2003).
20. Zhang, Y., Liu, J. S.: Bayesian inference of epistatic interactions in case-control studies. Nat Genet 39, 1167 (2007).