

An efficient comprehensive search algorithm for tagSNP selection using linkage disequilibrium criteria

Zhaohui Qin¹, Shyam Gopalakrishnan² and Gonçalo Abecasis¹

¹ Center for Statistical Genetics, Department of Biostatistics, School of Public Health, University of Michigan, Ann Arbor, MI 48109-2029;

² Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2122.

Running title: *LD based comprehensive tagSNP selection*

Corresponding author:

Dr. Zhaohui S. Qin, Center for Statistical Genetics, Department of Biostatistics, School of Public Health, University of Michigan, 1420 Washington Heights, Ann Arbor, MI 48109-2029, Tel: (734) 763-5965, Fax: (734) 763-2215, E-mail: qin@umich.edu.

Abstract

Selecting SNP markers for genome wide association studies is an important and challenging task. The goal is to minimize the number of markers selected for genotyping in a particular platform and therefore reduce genotyping cost while simultaneously representing information provided by all other markers. Depending on the platform, it is also important to select sets that are robust against occasional genotyping failure. An array of methods has been proposed to effectively select these tag SNPs using various criteria. In this study, we devised an improved algorithm for tagSNP selection using the pairwise r^2 criteria. We first break down large marker sets into many disjoint pieces, where more exhaustive searches can replace the greedy algorithm for tagSNP selection. These exhaustive searches lead to smaller tagSNP sets being generated for any given threshold. In addition, we evaluated multiple solutions that are equivalent according to the LD criteria to accommodate additional constraints such as platform specific SNP assay characteristics. We have written a computer program named FESTA (Fragmented Exhaustive Search for TAGging SNPs) based on this algorithm, and evaluated its performance using HapMap data from Chromosome 2 and the ENCODE regions. We found that in densely typed regions, using a moderate pairwise r^2 of 0.5 as the threshold for defining tagSNP sets, about 10% of all markers are selected as tags. At a more stringent threshold ($r^2 \geq 0.8$), about 20% of all markers are selected as tags.

Introduction

With the rapid improvement of high-throughput genotyping technologies, genome-wide association studies are emerging as a promising approach to detect genetic variants that contribute to human diseases. Initially, genome-wide association studies will focus on single nucleotide polymorphisms (SNPs) because of their high abundance in the human genome, their low mutation rates and their accessibility to high-throughput genotyping (Collins et al. 1997). There are more than 10 million verified SNPs in dbSNP (build 124) (Sachidanandam et al. 2001), but typing all available SNP markers is inefficient and not necessary since many will provide redundant information due to linkage disequilibrium (LD). A better strategy is to select a subset of representative SNPs (tagging SNPs or tagSNPs) and remove the rest from consideration (Johnson et al. 2001, Cardon and Abecasis 2003). The objective is to have little information overlap among the selected SNPs while retaining much of the information in the original set. Using simulations, various authors (Zhang et al. 2002; Stram 2004) have shown that the power of detecting disease-causing variant using tagSNPs is only slightly lower than when the full set of available SNP markers are used.

The selection of tagSNPs has become a very active research topic and many strategies have been proposed. Some consider the haplotype block structure, and search for “haplotype tagging SNPs (htSNPs)” to capture the limited haplotype diversity within each block (Patil et al. 2001, Zhang et al. 2002, Gabriel et al. 2002, Johnson et al. 2001, Meng et al. 2003, Sebastiani et al. 2003, Avi-Itzhak et al. 2003, Ke and Cardon 2003, Lin

and Altman 2004). An alternative strategy is to select SNPs such that haplotypes consist of these markers can then be used to predict other SNPs (Goldstein et al. 2003, Stram 2003, Hampe et al. 2003, Holldórson et al. 2004, De Bakker 2005). Recently, Zhang and Jin (2003) and Carlson et al. (2004) introduced methods based on the LD measure r^2 . These methods search for a small set of SNPs that are in strong LD (measured through pairwise r^2) with other SNPs that are not selected for genotyping. Pairwise r^2 is an attractive criterion for tagSNP selection since it is closely related to statistical power for case control association studies, where a directly associated SNP is replaced with an indirectly associated tagSNP (Pritchard and Przeworski, 2001).

In this manuscript, we describe efficient algorithms for tagSNP selection that can find minimum tagSNP sets based on pairwise LD measure r^2 ; compare alternative solutions according to additional criteria, such as assay design scores; force certain markers in or out of the tagSNP set; and find double coverage tagSNPs. We replace a greedy search, where markers are added sequentially to the tagSNP set, with an exhaustive search where all marker combinations can be evaluated. In most settings, our method is guaranteed to find the optimal tagSNP set(s) defined by the r^2 criterion. Our computational improvements rely on the observation that LD to large extent is a local property (Reich et al. 2001; Patil et al. 2001; Daly et al. 2002; Jeffreys et al. 2001; Gabriel et al. 2002; Dawson et al. 2002). We arrange the genome into precincts of markers in high LD, such that markers in different precincts show only low pairwise disequilibrium. TagSNP selection can then be performed within each precinct independently, greatly reducing computation complexity. Using data from the HapMap project (The International

HapMap Consortium 2003), we showed that the majority of these precincts contain relatively small numbers of SNPs, especially when a stringent r^2 criterion is used. Our algorithm readily identifies equivalent tagSNP sets, so that additional selection criteria can be incorporated. For example, an investigator might focus on the set that contains more coding SNPs, or the one that contains SNPs that are easier to genotype. Other useful extensions are also discussed in this manuscript, such as the inclusion of mandatory tagSNPs and double coverage which can increase robustness against sporadic genotyping failures. A software program called FESTA (Fragmented Exhaustive Search for TAgSNPs) has been developed implementing these new algorithms, and we hope it will be useful for those designing large scale association studies.

Method

Consider a set \mathbb{S} of M bi-allelic SNP markers a_1, a_2, \dots, a_M . Further assume that all these markers have minor allele frequency (MAF) above a certain threshold (0.05 was used in this study). Two-SNP haplotype frequencies were first estimated (Hill 1974), and the pairwise LD measure r^2 (also referred to as “ Δ^2 ”) (Devlin and Risch 1995) was calculated for each pair of markers using the inferred haplotype frequencies (Hill and Robertson 1968). Two markers a_i and a_j are said to be in strong LD if the r^2 between them is greater than a pre-specified threshold value r_0 , denoted as $r^2(a_i, a_j) \geq r_0$ ($r_0 = 0.5$ or 0.8 in this study). Further, we consider markers in strong LD to be good tagSNPs for each other such that a_i can be used as surrogate for a_j , or vice versa.

Our aim is to find tagSNP set, denoted by T , a subset of \mathbb{S} such that $\forall a_i \in \mathbb{S} \setminus T, \exists a_j \in T$ satisfies $r^2(a_i, a_j) \geq r_0$. In our presentation, we introduce two intermediate SNP sets, P and Q . The candidate set P contains all the markers that are eligible to be chosen as tagSNPs and the target set Q contains all the markers that are yet to be tagged, that is, no marker in Q is in LD with any tagSNP in T . For each marker a_m in P , let $C(a_m) = \{a : a \in Q \text{ and } r^2(a, a_m) \geq r_0\}$ represent the subset of Q which contains markers that are in strong LD with a_m , and let $|C(a_m)|$ be the number of the elements in set $C(a_m)$. Typically, the candidate set P is the complement of the tagSNP set T , $P = \mathbb{S} \setminus T$ and $P = Q$. One exception occurs when some SNPs are excluded as tagSNPs because they can not be easily genotyped, but they still should be tagged by other markers if

possible. In this case, the candidate set is a subset of target set. We describe several different algorithms for updating P , Q and T starting with a greedy approach (Carlson et al. 2004). We then outline successive refinements of a partition and exhaustive search algorithm, designed to allow processing of very large number of markers. Finally, we discuss algorithms that allow mandatory tagSNPs to be included, and implement a robust “double coverage” strategy.

Greedy approach

The detailed algorithm is as follows (Carlson et al. 2004),

Algorithm 1 (greedy approach):

1. Set $T = \emptyset$ and $P = Q = \mathbb{S}$;
2. For each marker a_m in P , calculate $|C(a_m)|$;
3. For every marker a_m where $|C(a_m)| = 0$, add a_m into T , and remove it from Q ;
4. Find the marker in P that has the highest $|C(a_m)|$ value, denoted as a_{max} , and add a_{max} into T , removing it and all connected SNPs, i.e., $C(a_m)$ from Q ;
5. Repeat steps 2-4 until $Q = \emptyset$.

In Step 4, by removing associated markers from consideration, the coverage overlap among tagSNPs is greatly reduced. In addition, this approach speeds up computation by reducing the number of markers considered at each stage. Although it is simple to implement, the greedy procedure may miss more efficient solutions. Figure 1 gives a simple example, where markers a and b each tag half of all markers and together can tag

all the markers. However, marker c is connected to more than half of all markers, and it is the first marker selected by the greedy algorithm. In this example, the greedy algorithm produced a set with three tagSNPs, despite the fact that the optimal solution contains only a and b .

Exhaustive search and partitioning algorithm

An exhaustive search guarantees the minimum tagSNP set. Starting with a candidate set P , we evaluate all K -marker combinations in P , to see whether they cover the entire target set Q . To find the smallest tagSNP set we can start with $K = 1$, and increase K gradually until full coverage is achieved.

Theoretically, the exhaustive search solves any tagSNP selection problem. But in practice, genome-wide tagSNP selection requires consideration of thousands of SNP markers. For problem of this scale, exhaustive searches can not be directly applied due to prohibitive computation cost.

Since appreciable LD only occurs within clusters of nearby markers along chromosomes, a practical solution is to first decompose the set of markers into disjoint precincts, such that markers in different precincts are never in strong LD. Then, selecting tagSNPs using the r^2 criteria in the whole set is equivalent to first selecting tagSNPs in each precinct and then combining all the tagSNPs together.

Partitioning the markers into precincts can be achieved using standard algorithms in graph theory. In fact, if we represent each SNP marker by a node, and use an edge to connect a pair of nodes if two markers are in LD, then the SNP markers and their LD connections can be represented by a graph, which can be partitioned using the Breadth First Search (BFS) algorithm (Thomas et al. 1990). Starting from any node (a marker) in a new precinct, this algorithm adds all neighboring nodes (markers in LD) and all neighbors of the newly added nodes to the precinct, until there are no neighbors to be added to the precinct. This process is restarted from different nodes until all the nodes are assigned a precinct.

After the partitioning step, we perform the tagSNP selection within each precinct. Starting with $K = 1$, all k -marker combinations are searched to see if they cover all markers within this precinct. If not, K is increased by one and the search is repeated until a tagSNP set is found or a pre-specified search limit is reached. The detailed algorithm is as follows:

Algorithm 2 (exhaustive search):

1. Apply the *Breadth First Search* to decompose the entire set of markers into precincts \mathbb{S}_i such that high LD can only be observed within precincts.

$$\mathbb{S} = \bigcup_{i=1}^n \mathbb{S}_i, \text{ and } \mathbb{S}_i \cap \mathbb{S}_j = \emptyset \text{ for all } i \neq j;$$

2. Within each precinct \mathbb{S}_i , set $k_i = 1$,

- a. Enumerate all possible K_i -marker combinations. Both the candidate set P_i and the target set Q_i is \mathbb{S}_i . If no such combination can cover the entire precinct, set $K_i = K_i + 1$ and repeat this step;
 - b. Record all tagSNP sets that can cover the precinct. These form the complete minimum tagSNP sets $\{T_i^j : j = 1, \dots, J_i\}$, where J_i is the total number of such minimum tagSNP sets.
3. Any combination of tagSNP sets identified from all disjoint subsets forms a tagSNP set for the whole set \mathbb{S} , the overall size of such minimum tagSNP sets

is $\sum_{i=1}^n K_i$, and the total number of such minimum tagSNP sets is $\prod_{i=1}^n J_i$.

An alternative procedure for the Step 2a above is to conduct exhaustive search starting with a large K_i , using the greedy approach to set an upper bound for K_i within each precinct. If $K_i \leq 2$, no reduction in the size of the tagSNP set is possible. Otherwise, we search all $(K_i - 1)$ -marker combinations. If none is able to cover the whole set, K is already the minimum number of tagSNPs. Otherwise we let $K_i = K_i - 1$ and repeat the previous step. This strategy will save computation time if the optimal tagSNP size is close to the greedy result.

Greedy exhaustive hybrid approach

When evaluating all K_i -marker combinations (Step 2 above), the computation cost required for an exhaustive search might be too great in some precincts. In such cases, we propose a hybrid solution which reduces the computation cost, but retains a good chance

of finding optimal tagSNP sets. To decide whether an exhaustive search is feasible, we compare the computation cost for evaluating all K_i -marker combinations within a precinct-- $\binom{N_i}{K_i}$ to a computation cost limit L specified *a priori*, which can be chosen by the user based on available resources. In this study, we tried two different limits, 10^6 and 10^8 . Larger limits allow more comprehensive searching, but require additional computational effort. When this limit is exceeded, we apply the following hybrid algorithm. In turn, we remove each marker a_m where $|C(a_m)| \geq \frac{|Q_i|}{K_i}$ from candidate set P_i , and remove a_m together with all the markers in $C(a_m)$ from target set Q_i from consideration. Then we conduct an exhaustive search for all $(K_i - 1)$ -marker combinations in $P_i \setminus \{a_m\}$ to cover $Q_i = S_i \setminus (\{a_m\} \cup C(a_m))$. The total number of combinations becomes $\binom{N_i - |C(a_m)| - 1}{K_i - 1}$. When $N(a_m)$ is large, this reduction in number of combinations can be quite significant (e.g., when $N_i = 100$, $K_i = 8$, $\binom{100}{8} = 1.86 \times 10^{11}$. If $N(a_m) = 50$, $\binom{49}{7} = 8.59 \times 10^7$, a savings of more than 2165 fold).

An exhaustive search in the reduced set proceeds only if the number of combinations is less than L , otherwise, another greedy step can be taken by adding the most promising marker to the tagSNP set. If the exhaustive search is successful, the tagSNP set obtained in the reduced set plus the previously selected marker a_m together form a complete K_i -marker tagSNP set. The detailed algorithm is summarized in Algorithm 3:

Algorithm 3 (greedy-exhaustive hybrid):

In the i th precinct, for every marker a_m in candidate set P_i , determine $|C(a_m)|$, and

proceed if only if $|C(a_m)| \geq \frac{|Q_i|}{K_i}$ and $\binom{N_i - |C(a_m)| - 1}{K_i - 1} < L$, where L is the computation

limit specified *a priori*;

- 1 Let $T_{im} = \{a_m\}$, candidate set $P_i = \mathbb{S}_i \setminus \{a_m\}$ and target set $Q_i = \mathbb{S}_i \setminus (\{a_m\} \cup C(a_m))$;
- 2 Conduct an exhaustive search for all possible $(K_i - 1)$ --marker combinations in candidate set $P_i = \mathbb{S}_i \setminus \{a_m\}$. If successful, combine any successful combination and marker a_m to form a tagSNP set of size K_i .

When $\binom{|Q_i|}{K_i}$ is extremely large, we recursively apply Algorithm 3.

Mandatory tagSNP markers

Our algorithm readily allows users to force certain mandatory SNP markers to be included or excluded in the tagSNP set. There are several scenarios where such functionality is important. First, in candidate gene studies, previous knowledge may be available as to which SNPs are functionally important. These might include nonsynonymous coding region SNPs (cSNPs) as well as SNPs located in regulatory regions. Second, in genome wide studies, one might carry out multiple rounds of genotyping and tagSNP selection. In such cases, additional tagSNPs could be selected at each round to cover the markers not tagged by tagSNPs successfully genotyped in the previous round. We provide an example of this in the results section. In other settings, it

may be useful to exclude certain SNPs from consideration as tags. For example, some SNP markers may be difficult to genotype using a particular platform.

When there are mandatory markers t_1, t_2, \dots, t_r to be included, put these markers into the

tagSNP set T , and remove them from the candidate set, e.g., P becomes $P \setminus \bigcup_{i=1}^r \{t_i\}$. The

target set Q becomes $Q \setminus \bigcup_{i=1}^r (\{t_i\} \cup C(t_i))$. If there are SNPs u_1, u_2, \dots, u_s that need to be

excluded from the tagSNP set, remove them from the candidate set, the target set Q is unchanged.

Choosing between alternative solutions

Within a densely typed SNP set, redundant tagSNPs are common, which results in multiple tagSNP sets of the same size. All of these sets are equal in the sense of minimizing the number of tagSNPs. In order to choose one set for genotyping, additional criteria can be entertained. Here we evaluate several alternative criteria:

1. Maximize average r^2 between tagSNPs and untagged SNPs they represent;
2. Maximize the lowest r^2 between tagSNPs and the untagged SNPs they connect to;
3. Minimize the average r^2 among all pairs of tagSNPs within a precinct;
4. Maximize the average r^2 among all pairs of tagSNPs within a precinct;
5. Maximize the average minor allele frequencies among all tagSNPs.

In criteria 1 and 2, we try to identify the tagSNP sets that have the strongest connections with those untagged SNPs, which should increase power on average and in the worst case respectively. The purpose of using criteria 3 is to find a tagSNP set whose members are as independent as possible which minimizes overlap between tagSNPs and potentially increases the chance of linking to untyped SNPs. Criteria 4 may increase redundancy and robustness to genotype failure; and criteria 5 may improve genotyping success for some assays.

To evaluate the relationship between each tagSNP set identified by the aforementioned criteria, and more importantly, their potential of uncovering the disease causing mutations in association studies, we conducted some empirical evaluations, summarized in the Result section.

Other types of criteria may be of even greater interest in practice. For example, in many genotyping technologies, some SNPs are harder to genotype than others due to characteristics of surrounding genome sequence. We can use this information to select tagSNPs that are likely to have a high success rate, and to avoid SNPs that are prone to genotyping failure.

Double coverage

So far, both the greedy approach and the greedy-exhaustive hybrid search algorithm focus on finding a tagSNP set such that each SNP marker is either a tagSNP itself or is in

LD with at least one of the tagSNPs. This is a criterion aimed at minimizing the number of tagSNPs selected. In reality, random genotyping failure or genotyping error on these tagSNPs can result in loss of power to identify the true signal. To be more robust against such adverse events, we evaluated a more stringent criterion requiring that, if possible, every untyped marker should be in LD with at least two tagSNPs.

Here we describe a general strategy for finding tagSNP sets that will have double coverage on the SNP markers considered. As always, an exhaustive search is able to find such tagSNP sets when the marker set considered is not too large. When exhaustive searches are not feasible, we adopt the following strategy: we use each of the single coverage sets as a base, and select additional tagSNPs. Then, we collect all SNPs that are not yet double covered to build a target set Q (some SNPs will be double covered by the base set of tagSNPs). Similarly, all SNPs that are not in the starting base tagSNP set are combined into the candidate set P . With these starting points, we conduct an exhaustive or greedy exhaustive hybrid search using P and Q . Note that double coverage is not possible for singleton SNPs. In practice, it may be useful to consider double coverage only for large precincts, where the cost of losing a SNP to genotyping failure might be higher, and our software FESTA incorporates this option.

Results

To illustrate our proposed piecewise exhaustive search tagSNP selection strategy, and to compare it with the greedy approach, we applied both methods to two sets of data, the entire chromosome 2 and five ENCODE regions (ENr112, ENr131, ENr113, ENm010 and ENm013) genotyped by the HapMap project (release 16, March 2005). In these examples, we consider the HapMap CEU samples only. The first is in the context of a genome wide association study and the second is similar to the situation of a candidate region study.

Chromosome wide tagging

We applied the greedy algorithm and the greedy-exhaustive hybrid search algorithm to the HapMap Chromosome 2 data (release 16, March 2005, CEU plate only). There are 87882 SNP markers spanning a 243 Mb region. 25852 of them have minor allele frequency less than 0.05 and are not considered here, reducing the number of SNP markers to 62030, or about 1 SNP every 3.9 kb. Using an r^2 threshold of 0.5, these SNP markers are organized into 11691 precincts. 5236 precincts contain just one SNP (“singletons”), and must be included in any tagSNP set. The largest precinct contains 296 SNPs. The distribution of the number of SNPs in each precinct is shown in Fig 2A.

Tagging the 6455 non-singleton precincts requires 8895 tagSNPs using the greedy approach. That brings the total number of tagSNPs identified by the greedy approach to 14131, or about 22.8% of the 62030 SNPs considered. Our approach reduces this number

to 13873 (with $L=10^6$), a savings of 258 SNPs. The largest number of tagSNPs identified in any precinct is 21. The distribution of the number of tagSNPs identified in each non-singleton precinct is shown in Fig. 2B. Differences in tagSNP number between our algorithm and the greedy only occur in those precincts where more than two tagSNPs are required by the greedy approach. There are 563 of such precincts, in which the greedy approach identified 2251 tagSNPs, and our algorithm identified 1993 tagSNPs, a 11.5% of reduction (13.6% with $L=10^8$). Our algorithm required about 31 minutes (1852 seconds) of CPU time to partition the 60,000 markers, and select tagSNPs in each precinct.

We also considered other r^2 thresholds. For example, when the r^2 threshold is set to 0.8, our algorithm broke down the entire set of SNPs into 22921 precincts. 13100 of them are singletons. The largest precinct contains 110 SNPs. The distribution of the number of markers in each precinct is shown in Fig 2C. Among the 9821 precincts that contain more than one SNPs. The number of tagSNPs in each subset ranges from 1 to 6; the distribution of the number of tagSNPs identified in each non-singleton precinct is shown in Fig. 2D. With this stringent threshold, there are only 148 precincts that require more than two tagSNPs using the greedy approach. Among them, the greedy approach and our approach identified 500 and 400 tagSNPs respectively. The reduction rate is 20.0% with $L=10^6$ or 21.0% when $L=10^8$. Our algorithm was completed in 109 seconds (with $L=10^6$). All the detailed results were summarized in Table 1 and 2. When double coverage is required, 31% and 25% more tagSNPs are needed with r^2 thresholds of 0.5 and 0.8

respectively. The distributions of the physical sizes of each precinct using r^2 thresholds of 0.5 and 0.8 are shown in Figure 2E and 2F.

Among all the non-singleton precincts (6455 for r^2 threshold 0.5, and 17685 for r^2 threshold 0.8), most require only a small number of tagSNPs, so that the exhaustive search can be applied directly. With $L=10^6$, the hybrid greedy exhaustive approach was required for 95 precincts ($r^2 \geq 0.5$) or 1.5% of all precincts (11 precincts with $r^2 \geq 0.8$ threshold).

Densely typed region

A very dense SNP map was recently released by the HapMap project on the ENCODE regions. We used five such regions (ENr112, ENr131, ENr113, ENm010 and ENm013) to evaluate the performance of our algorithm in densely typed chromosomal regions. The detailed results are summarized in Table 3. Each of the five ENCODE regions are 500 kb in length, and the average number of SNPs in these regions is 642 (range from 424 to 849), corresponding to a SNP density about 1 SNP per 779 bps (1 SNP per 1180 bp to 1 SNP per 589 bp for individual regions).

In this set of densely typed SNPs, using our method with $L=10^6$, when the r^2 threshold of 0.5 was used, the average percentage of tagSNPs required to cover each of the five regions is 9.3% of all markers (range from 6.2% to 12.5%). For a robust double coverage, 36.5% more tagSNPs on average are required (range from 20.8% to 52.0%). With a more stringent r^2 threshold of 0.8, the average percentage of tagSNPs required increased to

18.4% of all markers (range from 17.3% to 32.5%). To double cover these regions, 58.1% more tagSNPs on average are required (range from 51.8% to 61.8%). The average savings of tagSNP numbers for precincts where the greedy algorithm required more than two tags is 5.4% with r^2 threshold of 0.5 and 9.0% with r^2 threshold of 0.8 ($L=10^6$).

In these data, there are many alternative tagSNP sets for each region. (1.1×10^{15} to 2.5×10^{44} when using r^2 threshold of 0.5, and 3.8×10^{29} to 6.3×10^{68} when using r^2 threshold of 0.8), and our exhaustive search method allows flexibility in choosing the final tagSNP set. The number of minimum tagSNP sets in each precinct is listed in Table 3.

Additional TagSNPs for denser SNP map

With the rapid advance of genotyping technologies, progressively denser SNP maps will become available. As more refined association studies are carried out, it will be useful to select new tagSNPs to “fill holes” in the initial sparse maps. With a good picking strategy for the first round of tagging, this staged approach should result in only a small to moderate increase in the total number of tagSNPs compared to a one stage strategy.

To evaluate this strategy, we constructed an artificial sparse SNP map for each of the five ENCODE regions. Specifically, we selected one in every five consecutive SNP markers. The density of this sparse map is about 1 SNP per 5kb, close to the density of the phase I HapMap. Then, five different tagSNP sets are identified using the five criteria described in the Method section, denoted by T_i , $i = 1, 2, 3, 4, 5$. Finally, we applied our approach to

the full ENCODE SNP set, using each of these tagSNP sets as a seed, so as to search for the optimal tagSNP set to cover the previously “hidden” SNP markers. The effectiveness of these tagSNP sets is evaluated by comparing the number of new tagSNPs needed to cover the “newly found” SNPs. In addition to the five criteria, we also compared three other tagSNP selection strategies: a picket fence strategy with one SNP every 5kb; the same number of random SNPs as the number of tagSNPs for the sparse map; or using all original SNPs as tagSNPs. The results are summarized in Table 4 (r^2 threshold of 0.5) and TableS1 (r^2 threshold of 0.8) in the Supplementary Material. From there, one can see that when the r^2 threshold is 0.5, only 11.6% more tagSNPs (range from 0% to 18.9%) are needed to fill holes in the original map and that number is 4.8% (range from 1.0% to 9.0%) when r^2 threshold is 0.8. The five tagSNP sets require a lesser number of tagSNPs to cover the holes, compared to tagSNPs picked a picket fence or randomly.

Discussion

In this manuscript, we developed an efficient computational framework for tagSNP selection using the r^2 criteria. Our algorithm can handle 100,000s of linked markers and can identify smaller tagSNP sets than the greedy approach (Carlson et al. 2004). Using both chromosome wide data and densely typed ENOCDE data from HapMap, we illustrated the utility of our approach and showed savings increase in more densely typed regions and inside large LD “blocks”. Computational effort required by our method can be tailored to available computing resources. Another important advance is the ability of our method to identify multiple equivalent tagSNP sets and use additional criteria, such as assay design scores, to choose among potential tagSNP sets.

For the HapMap chromosome 2 CEU data, r^2 thresholds of 0.5 and 0.8 resulted in 13873 (22.4%) and 23652 (38.1%) SNPs being selected as tagSNPs. When the five densely typed HapMap ENCODE regions were combined together, using r^2 threshold of 0.5 and 0.8, the total number of tagSNPs is 299 (9.3%) and 592 (18.4%) respectively. In places where the greedy approach does not guarantee optimal result, with $L=10^6$ and r^2 thresholds of 0.5 and 0.8, our method reduces tagSNP size by 11.5% and 20.0% on chromosome 2 respectively, and the reduction rate is 5.4% and 18.2% on average for the five ENCODE regions. We also tried to mimic the two stage association study design (Sobell et al. 1993, Satagopan and Elston 2003, Satagopan et al. 2004). First, tagSNPs were identified using a sparse map with about 1 SNP every 5 Kb, mimicking the current phase I HapMap data. Then using a denser map with 1 SNP per kb, additional SNPs were

picked to fill holes in the original map. We found that this staged tagSNP picking strategy works well; at r^2 threshold of 0.5, it requires only 11.6% more tagSNPs compared to the one stage tagSNP picking strategy using the dense SNP map only, no matter which of the five additional criterion was used in the first stage. At r^2 threshold of 0.8, it requires only 4.8% more tagSNPs compared to the one stage tagSNP picking strategy using the dense SNP map only.

The result here suggested that our new tagSNP selection method can be readily applied to association studies, it produces a smaller tagSNP set using the same criteria compared to the greedy approach, and the computation cost is reasonable and adjustable. It is flexible enough such that many practical issues such as preferences for certain SNPs and staged designs can be addressed.

Many of the existing tagSNP picking algorithms aim to capture haplotype diversity using the reduced set of markers and work within the boundary of the haplotype blocks. On the other hand, tagSNP selection using r^2 criteria does not require precaculation of block's and can easily be applied to cover the whole chromosome. Recently, several methods advocate multiple-marker tagging strategies (Stram 2004, de Bakker, 2004), which allow for multi-SNP tags. While these methods further reduce the number of tagSNPs selected, they may be sensitive to random genotyping failures. Thus, we prefer approaches for tagSNP selection based on pairwise LD.

Our approach is amenable to further computational improvements. For example, parallel programming could be used to search for tagSNPs in separate precincts, further speeding up the computation.

We have implemented the algorithms described here in a software program called FESTA. FESTA carries out partitioning of large marker sets, and implements both the greedy approach and our exhaustive search for picking tagSNPs. It allows inclusion and exclusion of certain markers as tagSNPs, and can accommodate study or platform specific SNP preferences such as assay design scores. It also includes comprehensive search for all optimal or close-to-optimal double coverage tagSNP sets. This program is freely available and can be downloaded at <http://www.sph.umich.edu/csg/qin/FESTA>.

Acknowledgement

This work is partially supported by NIH RO1-HG002651-01. We are grateful to Drs. Mike Boehnke and Randy Pruim for critical comments on an early version of this manuscript.

References

Avi-Itzhak HI, Su X, De La Vega FM (2003) Selection of minimum subsets of single nucleotide polymorphisms to capture haplotype block diversity. *Pac Symp Biocomput.* 466-477.

Cardon LR, Abecasis GR (2003) Using haplotype blocks to map human complex trait loci. *Trends Genet.* **19**, 135-140.

Carlson CS, Eberle MA, Rieder MJ, Yi Q, Kruglyak L and Nickerson DA (2004) Selecting a maximally informative set of single-nucleotide polymorphisms for association analysis using linkage disequilibrium. *Am. J. Hum. Genet.* **74**, 106-120.

Collins FS, Guyer MS, Charkravarti A (1997) Variations on a theme: cataloging human DNA sequence variation. *Science*, **278** 1580-1581.

Daly MJ, Rioux JD, Schaffner SF, Hudson TJ, Lander ES (2001) High-resolution haplotype structure in the human genome. *Nat Genet.* **29**, 229-232.

Dawson E, Abecasis GR, Bumpstead S, Chen Y, Hunt S, Beare DM, Pabial J, Dibbling T, Tinsley E, Kirby S (2002) A first generation linkage disequilibrium map of human chromosome 22. *Nature* **418**, 544-548.

Delvin B, Risch N (1995) A comparison of linkage disequilibrium measures for fine-scale mapping. *Genomics* **29**, 311-322.

Gabriel SB, Schaffner SF, Nguyen H, Moore JM, Roy J, Blumenstiel B, Higgins J, DeFelice M, Lochner A, Faggart M, Liu-Cordero SN, Rotimi C, Adeyemo A, Cooper R, Ward R, Lander ES, Daly MJ and Altshuler D (2002) The structure of haplotype blocks in the human genome. *Science*, **296**, 2225-2229.

Garey and Johnson (1979) Computers and Intractabilities: A Guide to the Theory of NP-Completeness. W. H. Freeman New York.

Goldstein DB, Ahmadi KR, Weale ME, Wood NW (2003) Genome scans and candidate gene approaches in the study of common diseases and variable drug responses. *Trends Genet.* **19**, 615-622.

Hampe J, Schreiber S, Krawczak M (2003) Entropy-based SNP selection for genetic association studies. *Hum Genet.* **114**, 36-43.

Hill WG (1974) Estimation of linkage disequilibrium in randomly mating populations. *Heredity* **33**, 229-239.

Hill WG, Robertson A (1968) The effects of inbreeding at loci with heterozygote advantage. *Genetics* **60**, 615-628.

Halldorsson BV, Bafna V, Lippert R, Schwartz R, De La Vega FM, Clark AG, Istrail S. (2004) Optimal haplotype block-free selection of tagging SNPs for genome-wide association studies. *Genome Res.* **14**, 1633-1640.

The International HapMap Consortium (2003) The International HapMap Project. *Nature* **426**, 789-796.

Johnson GC, Esposito L, Barratt BJ, Smith AN, Heward J, Di Genova G, Ueda H, Cordell HJ, Eaves IA, Dudbridge F (2001) Haplotype tagging for the identification of common disease genes. *Nat. Genet.* **29**, 233-237.

Jeffreys AJ, Kauppi L, Neumann R (2001) Intensely punctuate meiotic recombination in the class II region of the major of histocompatibility complex. *Nat. Genet.* **29**, 217-222.

Ke X, Cardon LR (2003) Efficient selective screening of haplotype tag SNPs. *Bioinformatics* **19**, 287-288.

Lin Z, Altman RB (2004) Finding haplotype tagging SNPs by use of principal components analysis. *Am. J. Hum. Genet.* **75**, 850-861.

Meng Z, Zaykin DV, Xu CF, Wagner M, Ehm MG. (2004) Selection of genetic markers for association analyses, using linkage disequilibrium and haplotypes. *Am. J. Hum. Genet.* **73**, 115-130.

Patil N, Berno AJ, Hinds DA, Barrett WA, Doshi JM, Hacker CR, Kautzer CR, Lee D H, Marjoribanks C, McDonough DP, et al. (2001) Blocks of limited haplotype diversity revealed by high-resolution scanning of human chromosome 21. *Science* **294**, 1719-1723.

Pritchard JK, Przeworski M (2001) Linkage disequilibrium in humans: models and data. *Am. J. Hum. Genet.* **69**, 1-14.

Reich DE, Cargill M, Bolik S, Ireland J, Sabeti PC, Richter DJ, Lavery T, Kouyoumjian R, Farhadian SF, Ward R, Lander ES (2001) Linkage disequilibrium in the human genome. *Nature* **411**, 199-204.

Sachidanandam R, Weissman D, Schmidt SC, Kakol JM, Stein LD, Marth G, Sherry S, Mullikin JC, Mortimore BJ, Willey DL, Hunt SE, Cole CG, Coggill PC, Rice CM, Ning Z, Rogers J, Bentley DR, Kwok PY, Mardis ER, Yeh RT, Schultz B, Cook L, Davenport R, Dante M, Fulton L, Hillier L, Waterston RH, McPherson JD, Gilman B, Schaffner S, Van Etten WJ, Reich D, Higgins J, Daly MJ, Blumenstiel B, Baldwin J, Stange-Thomann N, Zody MC, Linton L, Lander ES, Altshuler D; International SNP Map Working Group (2001) A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms. *Nature* **409**, 928-933.

Satagopan JM, Elston RC. (2003) Optimal two-stage genotyping in population-based association studies. *Genet. Epidemiol.* **25**, 149–157.

Satagopan JM, Venkatraman ES, Begg CB (2004) Two-stage designs for gene-disease association studies with sample size constraints. *Biometrics* **60**, 589–597.

Sebastiani P, Lazarus R, Weiss ST, Lunkel LM, Kohane IS and Romani MF (2003) Minimal haplotype tagging *Proc. Natl. Acad. Sci. USA* **100**, 9900-9905.

Sedgwick R (1998) Algorithms in C++, Parts 1-4: Fundamentals, Data Structure, Sorting, Searching (3rd Edition) Addison-Wesley.

Sobell JL, Heston LL, Sommer SS (1993) Novel association approach for determining the genetic predisposition to schizophrenia: case-control resource and testing of a candidate gene. *Am. J. Med. Genet.* **48**, 28–35.

Stram DO, Haiman CA, Hirschhorn JN, Altshuler D, Kolonel LN, Henderson BE and Pike MC (2003) Choosing haplotype-tagging SNPs based on unphased genotype data using preliminary sample of unrelated subjects with an example from the multiethnic cohort study. *Hum. Hered.* **55**, 27-36.

Cormen TH, Leiserson CE, Rivest RL (1990) Introduction to algorithms. Cambridge, Mass., MIT Press, New York, McGraw-Hill.

Zhang K, Deng M, Chen T, Waterman MS and Sun F (2002) A dynamic programming algorithm for haplotype partitioning. *Proc. Natl. Acad. Sci. USA* **99**, 7335-7339.

Zhang, K. and Jin, L. (2003) HaploBlockFinder: haplotype block analysis. *Bioinformatics* **19**, 1300-1301.

Zhang K, Qin Z, Ting C, Waterman MS, Liu JS and Sun F (2004) Haplotype Block Partitioning and Tag SNP selection using genotype data and their applications to association studies. *Genome Res.* **14**, 908-916

Figure 1. An example when the greedy approach does not provide the smallest tagSNP set.

Figure 2. Summary of non-singleton precincts identified on Chromosome 2. Left panel represents results obtained when 0.5 was used as the r^2 threshold, right panel represents results obtained when 0.8 was used as the r^2 threshold. (A-B) Distribution of the number of SNPs contained in each precinct (truncated). (C-D) Distribution of the number of tagSNPs identified in each precinct. (E-F) Distribution of the physical distance spanned by each precinct (truncated).

Figure 3. Summary of all non-singleton precincts identified in the five ENCODE regions. Left panel represents results obtained when 0.5 was used as the r^2 threshold, right panel represents results obtained when 0.8 was used as the r^2 threshold. (A-B) Distribution of the number of SNPs contained in each precinct (truncated). (C-D) Distribution of the number of tagSNPs identified in each precinct. (E-F) Distribution of the physical distance spanned by each precinct (truncated).

Table 1. Summary of Chromosome 2: Size of disjoint subsets and number of SNPs and tagSNPs in each subset.

	Total	Median	Max	Time
$r^2 \geq 0.5$ (11691 precincts)				
Physical distance of precincts (kb)	243405	31.0	1593.6	-
# of SNPs	62030	4	296	-
# of tagSNPs (greedy)	14131	1	21	29.7
# of tagSNPs (FESTA, cutoff: 1 M)	13873	1	21	1852.5
# of tagSNPs (FESTA, cutoff: 100 M)	13825	1	21	61482.4
# of tagSNPs (FESTA, double cover)	18177	2	37	2167.3
$r^2 \geq 0.8$ (22921 precincts)				
Physical distance of precincts (kb)	243405	2.95	925.1	-
# of SNPs	62030	2	110	-
# of tagSNPs (greedy)	23752	1	7	21.1
# of tagSNPs (FESTA, cutoff 1 M)	23652	1	6	109.0
# of tagSNPs (FESTA, cutoff: 100 M)	23647	1	6	2118.6
# of tagSNPs (FESTA, double cover)	29644	2	17	311.5

Table 2. Distributions of the size of the tagSNP sets using the greedy approach and the FESTA algorithm.

	$r^2 \geq 0.5$		$r^2 \geq 0.8$	
	Greedy	FESTA	Greedy	FESTA
Singleton ^a	5236	5236	13100	13100
1	5140	5140	9194	9194
2	752	881	479	554
3	304	255	109	53
4	131	76	27	13
5	56	44	8	3
6	28	19	3	4
7	15	16	1	0
8	18	15	0	0
9	5	3	0	0
10	3	3	0	0
11	2	2	0	0
21	1	1	0	0
Total	14131	13873	23752	23652

^aSingleton means the precinct only contains one SNP. In other words, singleton refers to a SNP marker that is not in LD (pairwise LD measure r^2 greater than a threshold) with any other SNP in the entire set. Such a SNP, by definition, is one of the tagSNPs.

Table 3. Summary of TagSNPs identified by the greedy approach in the five encode regions (CEU plate only).

Region	ENr112	ENr131	ENr113	ENm010	ENm013
# of SNPs	791	849	660	424	485
$r^2 \geq 0.5$					
# of subsets*	42	72	32	38	20
# of Singletons*	17	27	11	15	5
# of tagSNPs (Greedy)	66	101	54	54	30
# of tagSNPs (FESTA, cutoff: 1 M)	65	98	53	53	30
# of tagSNPs (FESTA, cutoff: 100 M)	64	98	51	53	29
# of tagSNPs (FESTA, cutoff: double)	79	149	64	75	41
$r^2 \geq 0.8$					
# of subsets*	114	171	102	102	62
# of Singletons*	52	73	49	49	21
# of tagSNPs (Greedy)	137	180	110	105	68
# of tagSNPs (FESTA, cutoff: 1 M)	132	178	110	104	68
# of tagSNPs (FESTA, cutoff: 100 M)	132	178	110	104	66
# of tagSNPs (FESTA, cutoff: double)	210	288	167	161	110

^a Singleton refers to a SNP marker that is not in LD (pairwise LD measure r^2 greater than the threshold) with any other marker in the entire set. Such a marker, by definition, is one of the tagSNPs.

Table 4. Performance comparison of tagSNP sets selected by five different criteria in terms of coverage on denser SNP maps (with r^2 threshold of 0.5).

Region	ENr112	ENr131	ENr113	ENm010	ENm013
SNPs in dense map ^a	791	849	660	424	485
SNPs in sparse map ^b	159	170	133	85	98
One Stage Picking					
TagSNPs in dense map	65	98	53	53	30
TagSNPs in sparse map	35	54	35	26	20
Two stage Picking					
Max average r^2 b/ tags & non-tags ^c	72	110	58	62	31
Min lowest r^2 b/ tags & nontags ^d	73	109	58	63	31
Min average r^2 among tags ^e	71	110	58	63	30
Max average r^2 among tags ^f	75	110	59	63	32
Max average MAF of tags ^g	70	110	57	63	30
Other Strategies					
Random Picking ^h	79.8	123.0	72.2	68.0	37.4
Picket Fence ⁱ	85	121	74	70	43
Use All Sparse ^j	187	216	152	116	105

^aDense map means the densely typed SNP sets obtained from the ENCODE region in the release 16 HapMap data. (CEU plate, March 2005). All tagSNPs in this table were identified using our algorithm with the computation limit set at 1 million.

^bSparse map means the SNP sets obtained by selecting the first SNP in every five consecutive SNPs in the dense maps.

^{c-g}Total number of tagSNPs needed to cover the SNPs in the dense map using the tagSNPs identified using different criteria on the sparse map as seeds. Criteria 1, tagSNP sets that maximize the average r^2 between tagSNP and SNPs that it connected to; Criteria 2, tagSNP sets that minimize the average r^2 between tagSNP and SNPs that it connected to; Criteria 3, tagSNP sets that minimize the average r^2 among all tagSNPs; Criteria 4, tagSNP sets that maximize the average r^2 among all tagSNPs; Criteria 5, tagSNP sets that maximize the average minor allele frequencies among all tagSNPs.

^hTotal number of tagSNPs needed to cover the SNPs in the dense map using T random SNPs in the sparse map as seeds. T is the number of tagSNPs identified by our algorithm on the sparse map. The number is obtained by repeating this procedure 100 times and taking the average.

ⁱTotal number of tagSNPs needed to cover the SNPs in the dense map using T equally-spaced SNPs in the sparse map as seeds.

^jTotal number of tagSNPs needed to cover the SNPs in the dense map using all the SNPs in the sparse map as seeds.

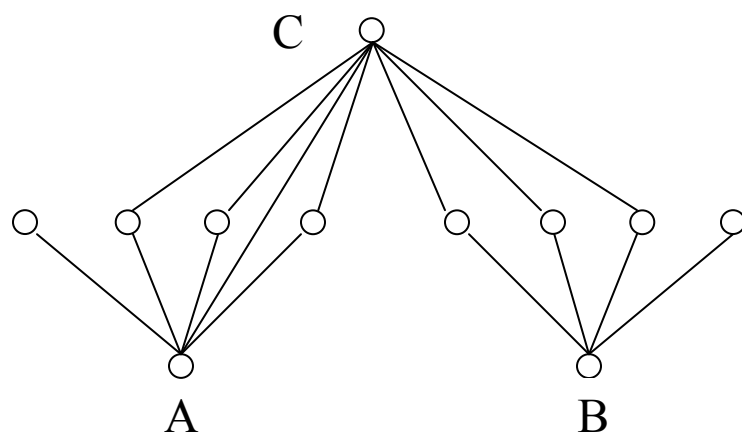


Figure 1.

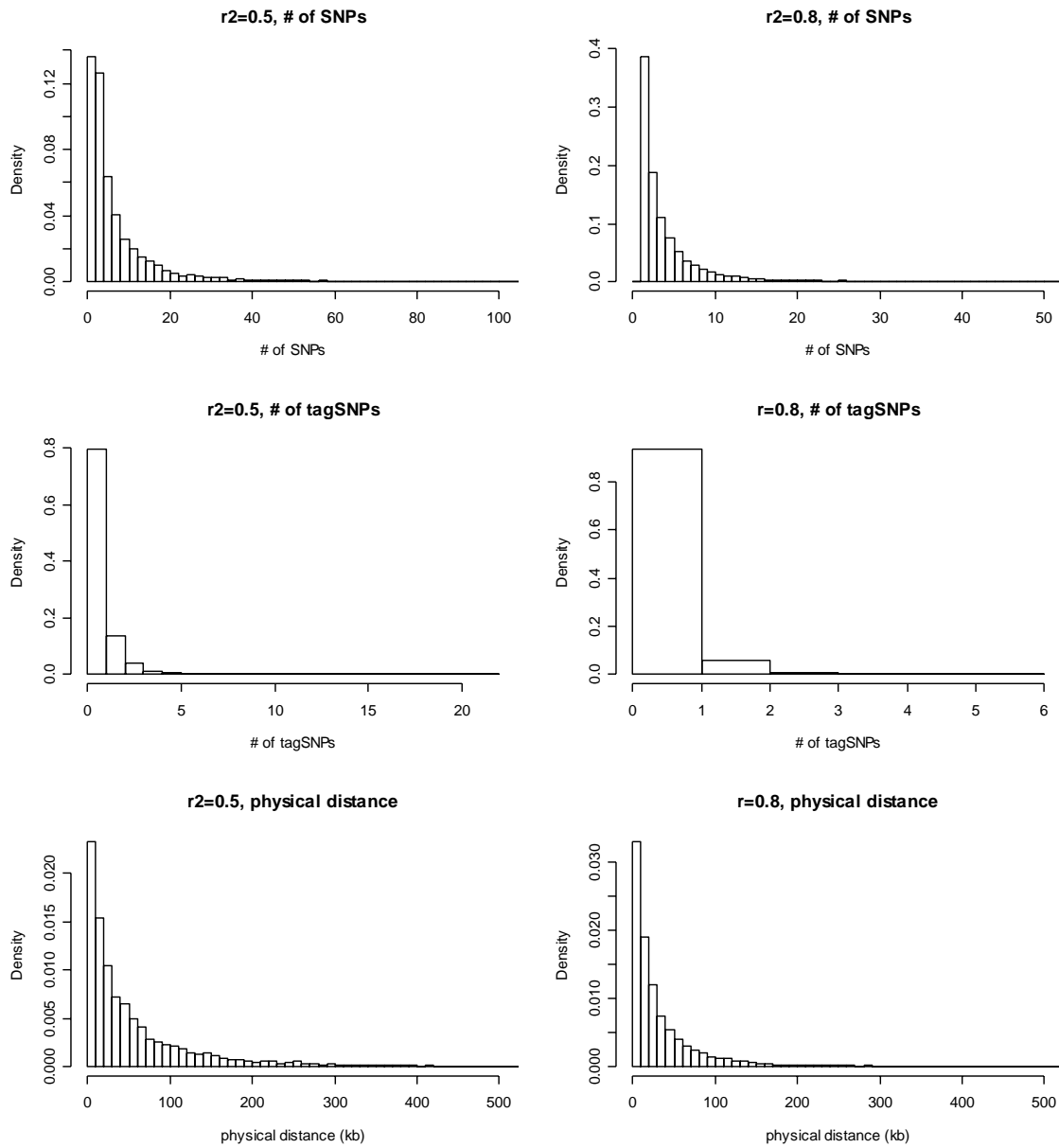


Figure 2.

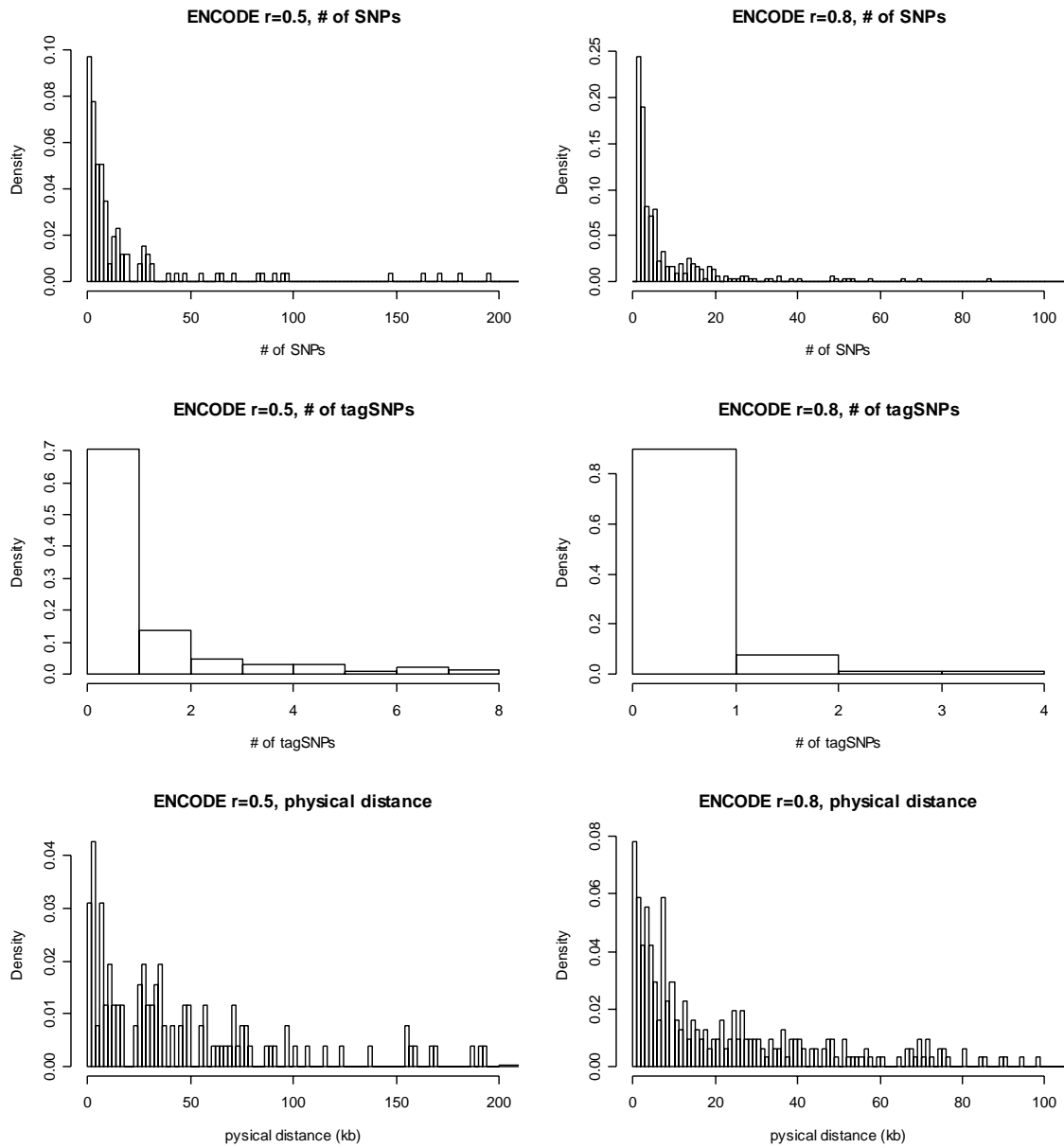


Figure 3.

Table S1. Performance comparison of tagSNP sets selected by five different criteria in terms of coverage on denser SNP maps (with r^2 threshold of 0.8).

Region	ENr112	ENr131	ENr113	ENm010	ENm013
SNPs in dense map ^a	791	849	660	424	485
SNPs in sparse map ^b	159	170	133	85	98
One Stage Picking					
TagSNPs in dense map	132	178	110	104	68
TagSNPs in sparse map	57	81	52	42	38
Two stage Picking					
Max average r^2 b/ tags & non-tags ^c	143	184	114	107	71
Min lowest r^2 b/ tags & nontags ^d	142	184	114	106	71
Min average r^2 among tags ^e	144	186	114	105	71
Max average r^2 among tags ^f	144	186	114	107	71
Max average MAF of tags ^g	144	186	113	109	72
Other Strategies					
Random Picking ^h	160.0	209.6	136.2	121.3	80.4
Picket Fence ⁱ	158	209	138	120	93
Use All Sparse ^j	234	271	192	148	127

^aDense map means the densely typed SNP sets obtained from the ENCODE region in the release 16 HapMap data. (CEU plate, March 2005). All tagSNPs in this table were identified using our algorithm with the computation limit set at 1 million.

^bSparse map means the SNP sets obtained by selecting the first SNP in every five consecutive SNPs in the dense maps.

^{c-g}Total number of tagSNPs needed to cover the SNPs in the dense map using the tagSNPs identified using different criteria on the sparse map as seeds. Criteria 1, tagSNP sets that maximize the average r^2 between tagSNP and SNPs that it connected to; Criteria 2, tagSNP sets that minimize the average r^2 between tagSNP and SNPs that it connected to; Criteria 3, tagSNP sets that minimize the average r^2 among all tagSNPs; Criteria 4, tagSNP sets that maximize the average r^2 among all tagSNPs; Criteria 5, tagSNP sets that maximize the average minor allele frequencies among all tagSNPs.

^hTotal number of tagSNPs needed to cover the SNPs in the dense map using T random SNPs in the sparse map as seeds. T is the number of tagSNPs identified by our algorithm on the sparse map. The number is obtained by repeating this procedure 100 times and taking the average.

ⁱTotal number of tagSNPs needed to cover the SNPs in the dense map using T equally-spaced SNPs in the sparse map as seeds.

^jTotal number of tagSNPs needed to cover the SNPs in the dense map using all the SNPs in the sparse map as seeds.