Am. J. Hum. Genet. 73:86–94, 2003

Minimum Description Length Block Finder, a Method to Identify Haplotype Blocks and to Compare the Strength of Block Boundaries

H. Mannila,[1,2] M. Koivisto,[1,2] M. Perola,[4] T. Varilo,[4] W. Hennah,[4] J. Ekelund,[4] M. Lukk,[1] L. Peltonen,[3,4] and E. Ukkonen[1]

[1]Department of Computer Science, [2]Helsinki Institute for Information Technology Basic Research Unit, and [3]Department of Medical Genetics, University of Helsinki, and [4]Department of Molecular Medicine, National Public Health Institute, Helsinki

We describe a new probabilistic method for finding haplotype blocks that is based on the use of the minimum description length (MDL) principle. We give a rigorous definition of the quality of a segmentation of a genomic region into blocks and describe a dynamic programming algorithm for finding the optimal segmentation with respect to this measure. We also describe a method for finding the probability of a block boundary for each pair of adjacent markers: this gives a tool for evaluating the significance of each block boundary. We have applied the method to the published data of Daly and colleagues. The results expose some problems that exist in the current methods for the evaluation of the significance of predicted block boundaries. Our method, MDL block finder, can be used to compare block borders in different sample sets, and we demonstrate this by applying the MDL-based method to define the block structure in chromosomes from population isolates.

Introduction

Haplotype blocks (Daly et al. 2001; Patil et al. 2001; Gabriel et al. 2002; Zhang et al. 2002*b*) define fascinating microscale geography of the human genome. Although several studies have confirmed that some type of haplotype blocks define genome geography, the recent data about haplotype blocks in the human genome have left multiple uncertainties concerning block boundaries and their variation. If the hypothesis of the ancient origin of the haplotype blocks is true, this variation should be seen across different human populations. In principle, the block boundaries might vary in strength, the precise location depending on the history of the chromosomes studied. A block boundary observed in one population might not be observed in another. On the basis of the currently available data, many of the blocks and the block boundaries seem to be shared to some extent across populations, but there are also distinct differences in the lengths of the blocks (Gabriel et al. 2002). Consequently, the strength of the block boundaries should also vary in a mixed population, and reliable determination of these boundaries is important when one considers the putative use of the haplotype blocks in various population genetic applications.

Although recent studies have elegantly described the concept of human haplotype blocks, the reliability of block structure has not been addressed (Daly et al. 2001; Patil et al. 2001; Gabriel et al. 2002). The published methods have applied segmentation algorithms with relatively ad hoc criteria for block quality. Also, the existing methods produce a segmentation without any clear indication of how strong or weak the evidence for predicted block boundaries is.

We describe here a new method for finding haplotype blocks that is based on the use of the MDL principle. We give a rigorous definition of the quality of the segmentation of a genomic region into blocks and describe a dynamic programming algorithm for finding the optimal segmentation with respect to this measure. We also describe a method for finding the probability of a block boundary for each pair of adjacent markers; this provides a statistical tool for evaluating the significance of each block boundary.

To test our method, the MDL block finder, we have reanalyzed the published data of Daly et al. (2001). Our results are in relatively good agreement with the published conclusions, but they also reveal clear differences in the predicted block boundaries and in their strengths. We have also applied the method to analyze the haplotype block boundaries in study samples of isolated populations in Finland.

Material and Methods

Samples

The samples of the Finnish populations are described in detail by Varilo et al. (2003). The families were ascertained originally for a nationwide schizophrenia study.

Received January 30, 2003; accepted for publication April 11, 2003; electronically published May 20, 2003.

Address for correspondence and reprints: Dr. Leena Peltonen-Palotie, Department of Molecular Medicine, National Public Health Institute, P.O. Box 104 (Haartmaninkatu 8), FIN-00251, Helsinki, Finland. E-mail: leena.peltonen@ktl.fi

0002-9297/2003/7301-0009$15.00

After obtaining informed consent, a tube of EDTA blood was drawn from participants, and the DNA was extracted by standard methods.

From the family collection, anonymized parent-offspring trios were randomly selected for this study. The trios were selected to be representative for the settlement history of Finland, inhabited by two periods of immigration, 4,000 and 2,000 years ago, and consequently resulting in early and late settlement regions as well as small internal subisolates of the population (Peltonen et al. 2000; Paunio et al. 2001).

MDL Principle and Coding of Haplotype Data

Let D be an $n \times p$ matrix of n observations over p markers. We refer to the jth allele of observation i as D_{ij}. For simplicity, we first assume that $D_{ij} \in \{0,1\}$.

A marker interval $[a,b] = \{a,a+1,\ldots,b\}$ is defined by two marker indices: $a,b \in \{1,\ldots,p\}$. A "segmentation" is defined as a set of nonoverlapping, nonempty marker intervals. A segmentation is defined as "full" if the union of the intervals is $[1,p]$. The data matrix limited to interval $[a,b]$ is denoted by $D(a,b)$, and the values of the ith observation are denoted by $D(i,a,b)$.

The MDL principle (Rissanen 1978, 1987) considers the description of the data through use of two parts: the model, B, and the description of the data, D, given the model. The description length for the data and the model is $L(B,D) = L(B) + L(D \mid B)$, where $L(B)$ is the length of the description of the model and $L(D \mid B)$ is the length of the description of the data, when the data are described using the model B.

The MDL principle states that the desired descriptions of the data are ones having the minimum length $L(B,D)$ of the total description. (For a good survey of the [sometimes intricate] connections between MDL, Bayesian statistics, and machine learning, see Li and Vitanyi [1997].) The MDL principle has successfully been used in various applications (Quinlan and Rivest 1989; Kilpelainen et al. 1995; Domingos 1999; Hansen and Yu 2001).

The haplotype data set D can be described by specifying first how many blocks there are and where the blocks start and end. For each block, we have to specify how many typical haplotypes ("class centers") there are and what they are. For each observation and each block, we indicate which of the typical haplotypes the observation comes from.

This representation balances the complexity of the model, measured by $L(B)$, and the accuracy of the model in describing the data, measured by $L(D \mid B)$.

More formally, a block model B consists of the following components:

1. A segmentation, S—that is, the start and end markers s_h and e_h for each block ($h = 1,\ldots,\ell$). (Of course, $s_1 = 1$ and $e_\ell = p$. In some of our models, we allow parts of the data to be uncoded, so $s_{h+1} = e_h + 1$ does not necessarily hold.) Implicitly, the segmentation specifies the number of blocks, ℓ.
2. For each block h, the class centers $\theta_h = (\theta_{hc})$ ($c = 1,\ldots,k_h$), specifying the coordinates θ_{hcj} for each marker $j = s_h,\ldots,e_h$. Implicitly, each θ_h also specifies the number of centers k_h.

The coordinates θ_{hcj} are real numbers, encoding the probability of seeing 1 in marker j of an observation stemming from class center c of block h. So, strictly speaking, a class center is not a typical haplotype but a mean vector of the haplotypes associated with the class.

Given block model $B = [(s_h,e_h),(\theta_{hcj})]$, the data can be encoded as follows. For each observation $i = 1,\ldots,n$ and for each block $h = 1,\ldots,\ell$, we first have to specify which of the k_h class centers the observation $D(i,s_h,e_h)$ belongs to; let this center be c. This takes $\log k_h$ bits per observation.

Then we have to describe $D(i,s_h,e_h)$, through use of the center coordinates θ_{hcj}, for $j = s_h,\ldots,e_h$. This is done by assuming independence of marker values, given the class center. Thus, the probability is

$$P[D(i,s_h,e_h) \mid (\theta_{hcj})] = \prod_{j=s_h}^{e_h} \theta_{hcj}^{D_{ij}}(1-\theta_{hcj})^{1-D_{ij}} . \qquad (1)$$

Using the relation of coding lengths and probabilities, we get a code of length $-\log P[D(i,s_h,e_h)]$ for the data D, given the segmentation model B.

Under the assumption that the segmentation in the block model B is full, it can be coded using $\ell \log p$ bits for encoding the block boundaries, $\ell \log n$ bits for the number of centers in the block, and $\alpha k_h(e_h - s_h + 1)$ bits for coding the centers, where α is the number of bits needed for the coding of a real number. Theoretical arguments (Rissanen 1978, 1987; Hansen and Yu 2001) indicate that the appropriate accuracy is obtained by choosing $\alpha = (\log n)/2$. Thus, the length of the description of the block model is

$$L(B) = \ell \log p + \ell \log n + \sum_{h=1}^{\ell} k_h \alpha (e_h - s_h + 1) ,$$

and the length of the description of the data is

$$L(D \mid B) = \sum_{h=1}^{\ell} \sum_{i=1}^{n} \{\log k_h - \log P[D(i,s_h,e_h)]\} .$$

Thus, the goal of the segmentation procedure is to find a block model B such that the overall coding length $L(B,D) = L(B) + L(D \mid B)$ is minimized.

The description method is easily extended to handle missing or unknown data values. If the values D_{ij} are interpreted as a degree of certainty that the correct value

is 1, the expression in equation (1) can be used directly. For instance, one can assign the value 0.5 for each missing allele in the data. To obtain a proper probability model, a normalizing factor should be included in equation (1). However, the factor behaves as an irrelevant constant and therefore can be ignored.

Formal comparisons between existing methods and the above definition of block quality are somewhat difficult, since the methods are based on different approaches. For example, in the study by Gabriel et al. (2002), the approach was to define as a block a region in which only a small fraction of SNP pairs show evidence of historical recombinations. This, in turn, is quantified by having the upper confidence value of the D' linkage disequilibrium (LD) measure be <0.9. Our approach puts more emphasis on the reconstruction of actual haplotypes for the blocks.

Dynamic Programming Algorithm

We use a dynamic programming algorithm to compute an optimal block structure and then estimate the probabilities of each block boundary. Similar methods have recently been used by Zhang et al. (2002*a*, 2002*b*).

The MDL cost function is, as defined above, a function of the whole segmentation. However, it is straightforward to see that it can be decomposed into a sum of the costs of the blocks of the segmentation. Given a marker interval $[a,b]$, let $\hat{k}$ be the optimum number of centers and let $\hat{\theta}_{ij}$ be the corresponding center coordinates associated with the jth allele of ith observation, such that the cost,

$$f(a,b) = \log p + \log n + n \log \hat{k} + \frac{1}{2}\hat{k}(b-a+1)\log n + \sum_{i=1}^{n}\sum_{j=a}^{b}\left[-D_{ij}\log\hat{\theta}_{ij} - (1-D_{ij})\log(1-\hat{\theta}_{ij})\right], \quad (2)$$

is minimized. Then, for the MDL optimal block model B_{mdl}, we have

$$L(B_{mdl},D) = \min_{S}\sum_{[a,b]\in S} f(a,b),$$

where S runs through all full segmentations on $[1,p]$. Thus, the minimum description length of haplotype data can be defined as the sum of costs of coding of individual blocks.

Denote by $F(b)$ the cost of the optimal segmentation of the haplotypes from marker 1 to marker b. We have the typical dynamic programming equation

$$F(b) = \min_{1\le a\le b}[F(a-1) + f(a,b)];$$

in addition, $F(0)$ is defined to be 0. Namely, the coding from marker 1 to marker b is produced either by coding all the markers in one block (with cost $F[0] + f[1,b]$), or by coding for some a from marker 1 to marker $a-1$ optimally (cost $F[a-1]$) and then coding from marker a to marker b in one block (cost $f[a,b]$). Given the costs $f(a,b)$, the computation can be performed in $O(p^2)$ time (that is, in an amount of time proportional to p^2).

The cost $f(a,b)$ is computed by using k-means clustering on the data set $D(a,b)$. The number of cluster centers is varied from 1 to 10, and, for each number, we produce five different clusterings. For each clustering, the coding cost of $D(a,b)$ is computed, and, as the cost $f(a,b)$, we select the smallest cost. (The problem of finding the best cluster centers is NP-hard—that is, it is probably intractable to solve exactly; thus, the approach does not guarantee that the shortest description for the single block from a to b is found.) The computation of $f(a,b)$ takes time $O[n(b-a+1)]$ for a fixed number of iterations in the k-means algorithm. Thus, the total amount of time needed for computing the costs $f(a,b)$ is $O(np^3)$.

In many cases, it is interesting to see how optimal segmentations behave when some (inconsistent) markers are allowed to be ignored in the data. We call such ignored markers "gaps" between haplotype blocks. A natural extension of the problem of finding the optimum segmentation is to find a segmentation that gives the shortest description length and includes, at most, u gaps. Denoting by $F(b,u)$ the cost of optimal segmentation from marker 1 to marker b using, at most, u gaps, we have

$$F(b,u) = \min\left\{F(b-1,u-1), \min_{1\le a\le b}[F(a-1,u) + f(a,b)]\right\}.$$

Namely, if a gap is used at the bth marker, then the prefix segmentation of $[1,b-1]$ is allowed to contain, at most, $u-1$ gaps. Otherwise, a block $[a,b]$ is introduced, and the maximum allowed number of gaps from marker 1 to marker $a-1$ is still u. The computation of $F(b,u)$ can be arranged to take $O(np^3)$ time.

Computing the Probability of a Block Boundary

The dynamic programming algorithm finds the best segmentation. It is not obvious, however, how strongly or weakly the data support the existence of a block boundary between two markers. We next describe a way of computing the probability that there is a block boundary between markers j and $j+1$. These probabilities can be used as indicators of the solidity of the block structure.

Denote by $\mathcal{S}_{j,j+1}$ the set of all full segmentations having a boundary between markers j and $j+1$. Then we are

interested in the probability of any segmentation from $\mathcal{S}_{j,j+1}$, given the data D:

$$P(\mathcal{S}_{j,j+1} \mid D) = \sum_{S \in \mathcal{S}_{j,j+1}} P(S \mid D) \ .$$

Denoting by $\mathcal{S}[1,p]$ the set of all full segmentations on $[1,p]$, this can be written as

$$P(\mathcal{S}_{j,j+1} \mid D) = \frac{\sum_{S \in \mathcal{S}_{j,j+1}} P(S,D)}{\sum_{S' \in \mathcal{S}[1,p]} P(S',D)} \ . \qquad (3)$$

The probabilities $P(S,D)$ can be obtained in a natural way from our description method. For any segmentation S and data set D, we define

$$P(S,D) = Z^{-1} 2^{-\Sigma_{[a,b] \in S} f(a,b)} \ ,$$

where $f(a,b)$ are the minimum description lengths for the corresponding blocks as described in equation (2) and Z is a normalization constant that does not depend on S and D. Note that the normalization constant cancels out when substituted into equation (3).

Define $q(a,b) = 2^{-f(a,b)}$, and, for any interval $[j,j']$,

$$Q(j,j') = \sum_{S \in \mathcal{S}[j,j']} \prod_{[a,b] \in S} q(a,b) \ ,$$

where $\mathcal{S}[j,j']$ denotes the set of all full segmentations on marker interval $[j,j']$. Then, since $\mathcal{S}_{j,j+1}$ is equal to the Cartesian product $\mathcal{S}[1,j] \times \mathcal{S}[j+1,p]$, we have

$$P(\mathcal{S}_{j,j+1} \mid D) = \frac{Q(1,j)Q(j+1,p)}{Q(1,p)} \ .$$

(For a similar development, see eq. [3.14] of Durbin et al. [1998], as well as Liu and Lawrence [1999].) To obtain numbers with moderate size, we report the log odds of $P(\mathcal{S}_{j,j+1} \mid D)$. The log odds values are obtained by considering all possible segmentations, and they should not be considered as significance values.

To compute the quantities $Q(i,j)$, we can again apply dynamic programming. The equations are

$$Q(1,b) = \sum_{1 \le a \le b} Q(1,a-1)q(a,b)$$

and

$$Q(a,p) = \sum_{a \le b \le p} q(a,b)Q(b+1,p) \ .$$

Here, of course, we define $Q(1,0) = Q(p+1,p) = 1$. Thus, the probabilities $P(\mathcal{S}_{j,j+1} \mid D)$ can be computed for all j in time $O(p^2)$.

Results

When MDL is applied to analyze haplotype data, we need to specify a way of describing the data so that, in the presence of distinct haplotype blocks, the description is shorter than it is in the absence of distinct blocks. The haplotype structure yielding the shortest description is chosen as the best model.

This can be achieved as follows: We first describe the block structure of the genomic region by indicating where the block boundaries are and what representative examples are for each class of every block—that is, the class centers. We next describe, for each observed haplotype and block, (i) to which class of the block the haplotype, defined by the markers, belongs and (ii) how (if in any way) the haplotype differs from the class center. If the data can be described succinctly through use of such a coding, then the data provide evidence of a distinct haplotype structure; otherwise, the haplotype over the given region is not accepted. The length $L(D|B)$ in bits of the description of data D, given block boundaries B, can be considered to be a negative logarithm of the likelihood $\Pr(D|B)$ of the data, given the block boundaries.

Given this measure of probability $\Pr(D|B)$ for each block structure B, we can compute the posterior probability $\Pr(B|D)$ for each block structure B by using Bayes' rule. Further, the probability of a specific boundary between, say, markers k and $k+1$ in the data can be computed as the sum of posterior probabilities $\Pr(B|D)$ of any block structure having a boundary in that gap.

A very important characteristic of this approach is that it allows haplotype blocks to be discovered even in the presence of genotyping errors, since an observation can be viewed to stem from a class center even in the case of small variation; this just increases the coding length (and thus decreases the probability) of the block structure. Note that the description of the haplotype data in itself is not very interesting; rather, the length of the description indicates the strength of the evidence of the data for the distinct haplotype structure.

We have tested our MDL block finder method on simulated and real genotype data. The results obtained from simulated data show that the method finds the block structure that has been used to generate the data and that the method is quite robust against noise (data not shown). This resistance to the noise was also demonstrated when the method was used to analyze real data (see below).

For real data, figure 1 shows the results obtained when the MDL method was applied to the data from the study by Daly et al. (2001). In addition to the identified haplotype blocks, we have also indicated the optimal block structures for the case in which some mark-

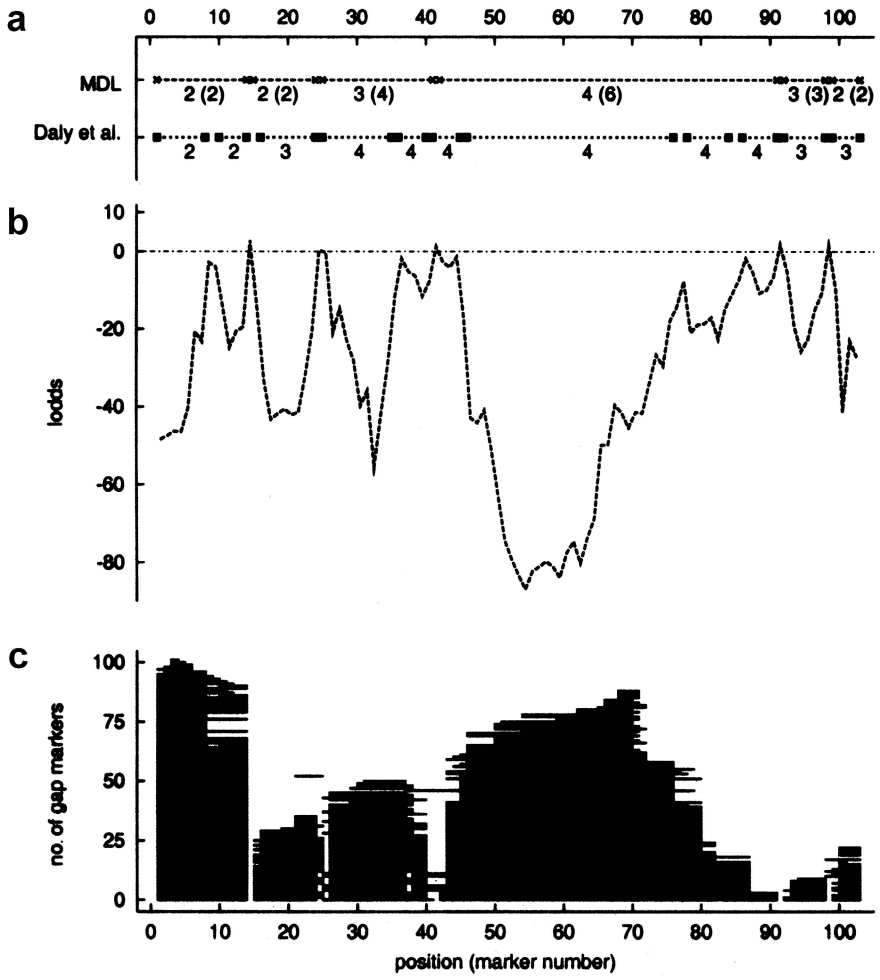

Figure 1 Haplotype block structure observed in the data of Daly et al. (2001). Marker number is given on the *X*-axis. *a,* The optimal block structure produced by the MDL scoring function, compared against the block boundaries reported by Daly et al. (2001). The numbers associated with the MDL blocks give the number of haplotype classes that suffice to cover at least 85% of the block and, in parentheses, the total number of the classes. The numbers associated with the Daly et al. (2001) blocks give the number of haplotypes in the block that suffice to cover at least 90% of the block. *b,* The log odds of the probability of block boundaries for each pair of adjacent markers. *c,* The optimal segmentation when k markers are allowed to be left outside the blocks, for varying k.

ers are allowed to be left outside the blocks. Some of the blocks (15–23) in the data of Daly et al. (2001) seem to be very strong, as indicated by the deep wells in the log probability plot. These most probably represent real haplotype block boundaries. On the other hand, for example, the block structure around marker number 40 seems to be far less well defined, as exemplified by the relatively flat probability curve in the region. This would challenge the existence of the haplotype block boundary. Overall, the segmentations reported by Daly et al. (2001) and those produced by the MDL block finder were in good agreement; however, an interesting difference can also be found at the block around marker 20. Since Daly et al. (2001) count exact matches, they report three distinctive common haplotypes. The MDL block finder suggests two haplotype classes, since it allows variability within a class. Table 1 shows an example of the class centers and the number of haplotypes from each class identified by the MDL block finder.

We added 5% or 10% of random noise to the SNP marker data of Daly et al. (2001) and performed a random permutation of the markers. Figure 2 shows the results, with marker locations plotted using physical distances. We observed that the MDL block finder is very tolerant of noise; the block structure obtained from the noisy data is almost identical to the structure obtained with the original data. As expected, the strength of the block boundaries decreases when noise is added. However, for noise levels up to 5%, the structure remains

Table 1

Haplotype Classes of Block 2 (Markers 15–24), Found by the MDL Method for the Data of Daly et al. (2001)

Class	No. of Associated Haplotypes	Center Coordinates (Major Allele of the Class)	No. of Haplotypes That Differ from the Most Commonly Occurring Haplotype at			
			0 Markers	1 Marker	2 Markers	⩾3 Markers
1	190	.16 (0), .02 (0), .96 (1), .64 (1), .06 (0), .15 (0), .05 (0), .02 (0), .94 (1), .02 (0)	115	44	20	11
2	68	.91 (1), .88 (1), .18 (0), .20 (0), .82 (1), .20 (0), .82 (1), .92 (1), .15 (0), .91 (1)	34	18	4	12

the same. When the markers are randomly permuted, the block structure disappears, which adds to the reliability of the results.

We tested the MDL method with SNP haplotypes obtained from samples of three Finnish subpopulations (Peltonen et al. 2000; Paunio et al. 2001). Five haplotype blocks were identified in the three Finnish subpopulations representing early settlement, late settlement, and a regional subisolate of the late settlement (fig. 3). The identified blocks varied in size from 12 kb to 361 kb. As expected, the haplotype blocks do not differ in different subpopulations of Finland, most probably reflecting the limited set of original founder chromosomes shared by all analyzed populations and the relatively short time since their first introduction to Finland. The log odds curve for estimating the probability of boundaries shows that, in general, the boundaries are stronger for larger values of observations; this was also obvious in our analyses of sampled data sets from Daly et al. (2001). We tested the significance of block boundaries by using bootstrap methods to investigate whether the optimal segmentation revealed a block boundary in the resampled data sets. The results were similar to the probabilities produced by the probabilistic approach (results not shown).

Discussion

We have described here a method, MDL block finder, for defining and finding haplotype blocks through use of the MDL principle. The more distinct the haplotype block structure is, the shorter is the description that can be obtained for the data. The coding cost function facilitates the use of dynamic programming to solve the problem, yielding an efficient algorithm for the problem.

We have also shown how the MDL principle can be used to obtain probabilities for block boundaries for all pairs of adjacent markers, providing a clear way to evaluate the significance of block boundaries. Experiments on simulated and real data have shown that the method produces useful results.

Haplotype blocks describe the history of the alleles in the population. They do not define the relationship between the populations but rather reflect the number of meioses in the history of the populations. Totally unrelated populations—one old with stable growth and the other young with a high expansion rate—could share identical haplotype blocks, at least if blocks are monitored using relatively common, old SNPs. When such common SNPs are used to construct haplotype blocks, the framework block pattern seems to be ancient and shared between different populations. Under such a concept, the recombinations would be the major force behind the haplotype blocks. In *Saccharomyces cerevisiae,* the recombination hotspots occur at regular intervals of ~50 kb; in humans, the detected haplotype blocks might indicate some level of regularity of recombination events, reflected by block boundaries. However, the relative contribution of random recombination and recombination hotspots for block borders is yet unknown (Stumpf 2002; Wang et al. 2002; Cardon and Abecasis 2003).

Further, chance and selective sweeps can also affect the haplotype structure in different populations (Zhang et al. 2002*a*). If we accept the concept of the binary recombination behavior of human chromosomes, the markers between the recombinational hotspots or within the haplotype blocks are in LD, whereas the markers across the blocks and flanking recombinational hotspots are not. Thus, we could use the block information for cost-beneficial design of initial mapping efforts of human traits and diseases. Since blocks are different in populations with different genetic histories, the determination of population-specific block structure would be beneficial, and the reasonable estimates of the significance of block borders should precede genotyping efforts utilizing the haplotype-block concept in marker selection. The logistics behind the utilization of block structure in disease gene mapping would be to determine the variance of LD blocks in disease alleles versus nondisease alleles and, on the basis of the deviation in the variance, to target the more detailed structural analyses to the particular region defined by LD block.

Here, MDL block finder would help in determining the statistical power of the study sample to identify the block borders. Moreover, population-specific analyses are most likely needed, for example, to find "haplotype-tagging SNPs" (Johnson et al. 2001). Discrete measures

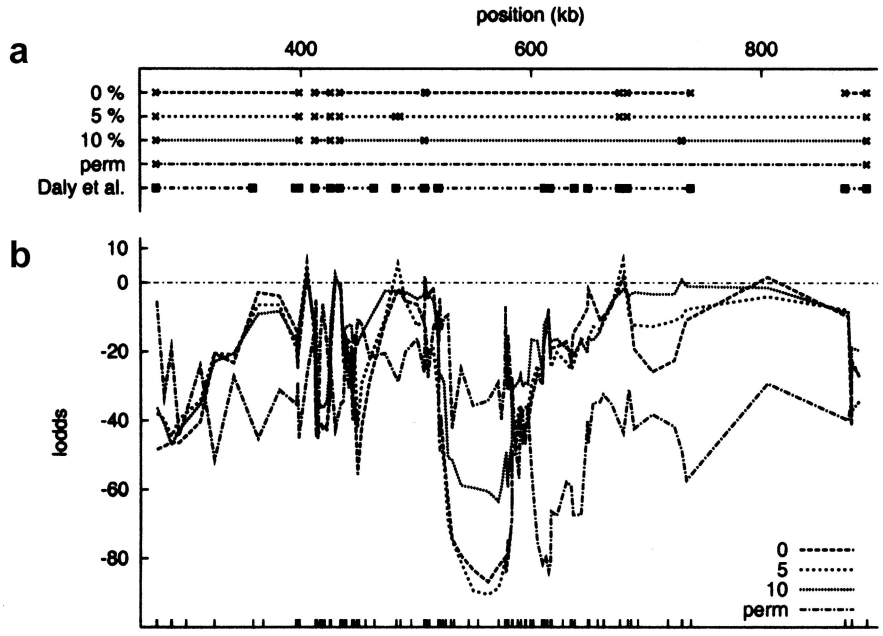

Figure 2 Haplotype block structure observed in the data of Daly et al. (2001) with added noise. The physical location of the marker is given on the *X*-axis. *a,* The block boundaries reported by Daly et al. (2001) and the optimal block structures produced by the MDL scoring function when 0%, 5%, and 10% of random noise was added to the data and when the order of markers was randomly permuted. *b,* The log odds of the probability of block boundaries for each pair of adjacent markers, after adding noise and permuting the order.

of block borders are somewhat risky, especially in population-specific analyses, since evidence for a nonexisting border might come from a small sample of informative chromosomes, the size of which also probably varies along the genome in a population-specific way, because of varying allele frequencies. The MDL block method should provide some statistical measures in the block recognition process. It seems likely that the map of equally spaced SNPs with solid statistical evidence of LD across the region would be the most efficient way to proceed in disease locus mapping. This approach should also eliminate some of the reports of few SNPs, chosen in an ad hoc manner, showing LD in disease alleles of common traits without a solid statistical framework.

Most of the current discussion concerning haplotype blocks has concentrated on their application in reducing genotyping costs. However, it is tempting to suggest the utilization of haplotype blocks in the final selection of the SNPs for functional studies. Restricting the SNPs to be analyzed to belong in a haplotype block associated with a phenotype would greatly enhance (and reduce cost) of these difficult analyses. Of course, this would require not only the background knowledge of block structure but also some novel biostatistical methods to distinguish the associated blocks from their surroundings. In any case, to be able to make this kind of important decision, it is essential for the investigator to evaluate the block structure within the population under study—and, again, to evaluate whether it is reasonable to trust the evidence of a block border in the samples studied.

Most studies concerning LD patterns and haplotype block structure have included only SNPs with a relatively high frequencies—that is, >10%–15%. It will be interesting to see how the block structure determined by MDL behaves when genetic markers with low allelic frequencies are built into the models in reasonably sized populations. If a hierarchic structure of haplotype blocks is seen, so that LD patterns seem to vary after inclusion of these low-frequency alleles, the need for solid statistical judgment of block borders becomes even more evident.

Intuitively, a haplotype block can be considered to represent a sequence of ordered markers such that, for those markers, most of the haplotypes in the population cluster into a small number of classes. Each class consists of identical or almost identical haplotypes. This notion can be formalized by considering the problem of describing the haplotypes in a succinct way. This approach is an instance of the MDL principle (Rissanen 1978, 1987), widely used in statistics, machine learning, and data mining (see, e.g., Li and Vitanyi 1997; Hansen

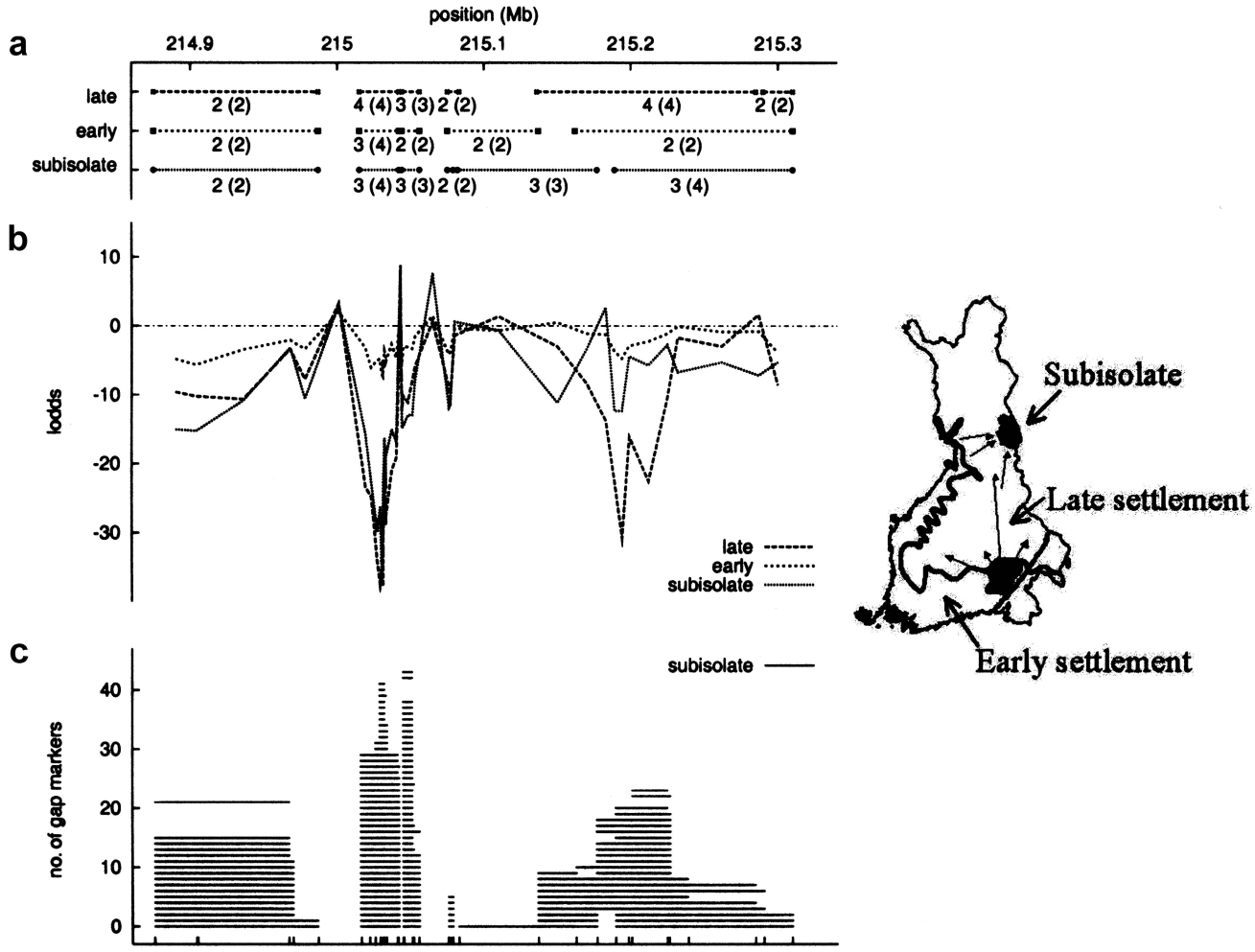

Figure 3 Haplotype block structure in the data from three subpopulations in Finland. Sample sizes are as follows: late settlement, $n = 108$; early settlement, $n = 32$; and subisolate of late settlement, $n = 108$. The physical location of the markers are given on the *X*-axis. *a,* The optimal block structure produced by the MDL scoring function in the three subpopulations. The numbers refer to the number of haplotype classes covering at least 85% of haplotypes and, in parenthesis, the full block. *b,* The log odds of the probability of block boundaries for each pair of adjacent markers. *c,* The subisolate; the optimal segmentation when k markers are allowed to be left outside the blocks, for varying k.

and Yu 2001). Similar ideas have also been applied to partitioning homogeneous DNA domains (Li 2001).

The MDL block finder described here provides a sound and simple way of assigning significance of haplotype block boundaries and has many obvious applications. The reports of identified haplotype blocks have used samples that vary greatly in size (e.g., 42, 258, or 550 chromosomes; see Patil et al. [2001], Daly et al. [2001], and Gabriel et al. [2002], respectively), and the significance of haplotype borders still has not been properly addressed. Further, the methods used to search for haplotype blocks described so far in the literature can be very vulnerable to genotyping errors, whereas the MDL block finder method seems to be quite resistant to this type of noise.

Our MDL method offers an alternative for estimating haplotype blocks and for comparing the blocks and their boundaries to each other. In addition to an obvious application in the human haplotype project (Couzin 2002), the method could also be applied in population genetics—for example, in estimating the genetic admixture of a population, or, because of its resistance to noise, in analyzing incomplete data sets, which inevitably occur in any large-scale genotyping efforts.

Computationally, the MDL block finder can be extended into many directions. The clustering approach and k-means algorithm could be replaced by closely related but directly probabilistic mixture models and by the usual expectation-maximization algorithm, respectively. The modifications of the method would facilitate

analyses of microsatellite markers. A challenging open problem is to develop the method further so that it could discover haplotype block structure through use of genotypic data without phase information.

The software for the MDL block finder is available from the authors.

Acknowledgments

The funding of the Academy of Finland and the Center of Excellence in Disease Genetics of the Academy of Finland is appreciated. L.P. is Gordon and Virginia MacDonald Distinguished Chair in Human Genetics in the David Geffen School of Medicine at the University of California Los Angeles, Los Angeles, California.

References

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

Couzin J (2002) Genomics: new mapping project splits the community. Science 24:1391–1393

Daly M, Rioux J, Schaffner S, Hudson T, Lander E (2001) High-resolution haplotype structure in the human genome. Nat Genet 29:229–232

Domingos P (1999) The role of Occam's razor in knowledge discovery. Data Mining Knowl Discov 3:1–19

Durbin R, Eddy S, Krogh A, Mitchison G (1998) Biological sequence analysis: probabilistic models of proteins and nucleic acids. Cambridge University Press, Cambridge

Gabriel S, Schaffner S, Nguyen H, Moore J, Roy J, Blumenstiel B, Higgins J, DeFelice M, Lochner A, Faggart M, Liu-Cordero S, Rotimi C, Adeyemo A, Cooper R, Ward R, Lander E, Daly M, Altshuler D (2002) The structure of haplotype blocks in the human genome. Science 296:2225–2229

Hansen M, Yu B (2001) Model selection and the principle of minimum description length. J Am Stat Assoc 96:746–774

Johnson G, Esposito L, Barratt B, Smith A, Heward J, Di Genova G, Ueda H, Cordell H, Eaves I, Dudbridge F, Twells R, Payne F, Hughes W, Nutland S, Stevens H, Carr P, Tuomilehto-Wolf E, Tuomilehto J, Gough S, Clayton D, Todd J (2001) Haplotype tagging for the identification of common disease genes. Nat Genet 29:233–237

Kilpelainen P, Mannila H, Ukkonen E (1995) MDL learning of unions of simple pattern languages from positive examples. In: P Vitanyi (ed) Proceedings of the Second European Conference on Computational Learning Theory (EuroCOLT). Springer-Verlag, Berlin, pp 252–260

Li M, Vitanyi P (1997) An introduction to Kolmogorov complexity and its applications. Springer-Verlag, New York

Li W (2001) New stopping criteria for segmenting DNA sequences. Phys Rev Lett 86:5815–5818

Liu J, Lawrence C (1999) Bayesian inference on biopolymer models. Bioinformatics 15:38–52

Patil N, Berno A, Hinds D, Barrett W, Doshi J, Hacker C, Kautzer C, Lee D, Marjoribanks C, McDonough D, Nguyen B, Norris M, Sheehan J, Shen N, Stern D, Stokowski R, Thomas D, Trulson M, Vyas K, Frazer K, Fodor S, Cox D (2001) Blocks of limited haplotype diversity revealed by high-resolution scanning of human chromosome 21. Science 294:1719–1723

Paunio T, Ekelund J, Varilo T, Parker A, Hovatta I, Turunen J, Rinard K, Foti A, Terwilliger J, Juvonen H, Suvisaari J, Arajarvi R, Suokas J, Partonen T, Lonnqvist J, Meyer J, Peltonen L (2001) Genome-wide scan in a nationwide study sample of schizophrenia families in Finland reveals susceptibility loci on chromosomes 2q and 5q. Hum Mol Genet 10:3037–3048

Peltonen L, Palotie A, Lange K (2000) Use of population isolates for mapping complex traits. Nat Rev Genet 1:182–190

Quinlan J, Rivest R (1989) Inferring decision trees using the minimum description length principle. Inf Comput 80:227–248

Rissanen J (1978) Modeling by shortest data description. Automatica 14:465–471

——— (1987) Stochastic complexity. J Roy Stat Soc [Ser B] 49:223–239

Stumpf M (2002) Haplotype diversity and the block structure of linkage disequilibrium. Trends Genet 18:226–228

Varilo T, Paunio T, Parker A, Perola M, Meyer J, Terwilliger J, Peltonen L (2003) The interval of linkage disequilibrium (LD) detected with microsatellite and SNP markers in chromosomes of Finnish populations with different histories. Hum Mol Genet 12:51–59

Wang N, Akey J, Zhang K, Chakraborty R, Jin L (2002) Distribution of recombination crossovers and the origin of haplotype blocks: the interplay of population history, recombination, and mutation. Am J Hum Genet 71:1227–1234

Zhang K, Calabrese P, Nordborg M, Sun F (2002*a*) Haplotype block structure and its applications to association studies: power and study designs. Am J Hum Genet 71:1386–1394

Zhang K, Deng M, Chen T, Watermanm M, Sun F (2002*b*) A dynamic programming algorithm for haplotype block partitioning. Proc Nat Acad Sci USA 99:7335–7339