Bioinformatics, 32(6), 2016, 801–807
doi: 10.1093/bioinformatics/btv669
Advance Access Publication Date: 14 November 2015
Original Paper

OXFORD

Genome analysis

Uncovering multiloci-ordering by algebraic property of Laplacian matrix and its Fiedler vector

Mookyung Cheon[1], Choongrak Kim[2,*] and Iksoo Chang[1,*]

[1]Creative Research Initiatives Center for Proteome Biophysics, Department of Brain and Cognitive Sciences, Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu 711-873, Korea and [2]Department of Statistics, Pusan National University, Busan 609-735, Korea

*To whom correspondence should be addressed.

Associate Editor: Igor Jurisica

Received on March 31, 2015; revised on October 16, 2015; accepted on November 9, 2015

Abstract

Motivation: The loci-ordering, based on two-point recombination fractions for a pair of loci, is the most important step in constructing a reliable and fine genetic map.

Results: Using the concept from complex graph theory, here we propose a Laplacian ordering approach which uncovers the loci-ordering of multiloci simultaneously. The algebraic property for a Fiedler vector of a Laplacian matrix, constructed from the recombination fraction of the loci-ordering for 26 loci of barley chromosome IV, 846 loci of *Arabidopsis thaliana* and 1903 loci of *Malus domestica*, together with the variable threshold uncovers their loci-orders. It offers an alternative yet robust approach for ordering multiloci.

Availability and implementation: Source code program with data set is available as supplementary data and also in a software category of the website (http:// biophysics.dgist.ac.kr)

Contact: crkim@pusan.ac.kr or iksoochang@dgist.ac.kr.

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

A locus is the chromosome location of a gene, a genetic marker or any specific DNA sequence, and can be regarded as a point on a line. The loci-ordering is a linear arrangement of genes or genetic markers in a linkage group which is a group of genes with their loci located on the same chromosome. For these genes, the segregation ratio for the genotypes and phenotypes departs from the Mendelian law. Therefore, the loci-ordering is the most important step in constructing a reliable and fine genetic map. The loci-ordering also includes localizing a new locus, a disease, or a marker locus, on an existing map. However computational estimation of loci-ordering for dense genes is rate-limiting step in genetic mapping procedures including linkage grouping or distance spacing between genes.

Genetic recombination is estimated by the recombination fraction, which is the ratio of recombinant gametes to total gametes (Goss and Harris, 1975; Kanaar and Hoeijmakers, 1998; Ott, 1999; White *et al.*, 1985). For homologous recombination, we expect that the greater the physical distance between two loci on a chromosome, the greater the chance they will recombine. Let $l_1, l_2, \ldots, l_n$ denote n loci in a linkage group, then our interest is ordering n loci to construct genomic map based on two-point recombination fractions r_{ij}, $1 \le i < j \le n$ for a pair of loci i and j, which usually ranges from 0 to 0.5. If two loci i and j are closely located, i.e. tightly linked, then r_{ij} is close to 0, and if not it is away from 0. With n loci, there are $n!/2$ possible orderings if the orientation of the orders is ignored. Locus ordering searches for the best locus order among the possible orders and it amounts to evaluating the maximum likelihood for each order. With only 10 loci, for example, there are 1 814 000 possible orders, and therefore, the loci-ordering entails a formidable computational problem and is practically unfeasible.

In overcoming such difficulties, the loci-ordering minimizing the number of crossovers was regarded as the best ordering, and this

method has been shown to be the maximum likelihood ordering under the full penetrance assumption, i.e. lack of interference (Thompson, 1987). Also, several other approaches, closely related with this idea, have been suggested. Examples for objective score functions used in other approaches are the minimum of adjacent recombination fractions criterion (Falk, 1989), the maximum sum of adjacent lod scores criterion (Weeks and Lange, 1987), minimum sum of the probability of double recombinants (Knapp *et al.*, 1990), maximum likelihood (Lander and Green, 1987), and minimum obligatory crossovers (Thomson, 1988). Most searching procedures for optimizing those objective score functions are exhaustive and the efficient techniques for searching strategies are necessary. To reduce the exhaustive searching procedures, seriation (Lander *et al.*, 2009) and branch-and-bound (Buetow and Chakravarti, 1987) searching techniques have been suggested. Comparisons among minimized or maximized score functions described above have been made (Cheema and Dicks, 2009; Hackett and Broadfood 2003; Kammerer and MacCluer, 1988; Olson and Boehnke, 1990). Softwares for computing genetic maps by using various objective score functions and optimizing techniques have been developed and distributed such as MAPMAKER (Lander, *et al.*, 2009), JoinMap (Criscione *et al.*, 2009; Van Ooijen, 2006), CarthaGene (de Givry, *et al.*, 2005), Neighbour mapping (Ellis, 1997), MapManger QTX (Manly *et al.*, 2001), RECORD (Van Os *et al.*, 2005), AntMap (Iwata and Ninomiya, 2006), MSTMAP (Wu *et al.*, 2008), MadMapper (West *et al.*, 2006).

Here we suggest a conceptually different yet a simple Laplacian ordering method for the loci-ordering via the Fiedler vector (Fiedler, 1973; Fiedler, 1975), taking the close analogy with the recent development for characterizing complex network phenomenon in broad areas (Amaral *et al.*, 2000; Buldyrev *et al.*, 2010; Chen *et al.*, 2006; Getz *et al.*, 2000; Liljeros *et al.*, 2001), which circumvents most of the difficulties in the previous approaches (Falk, 1989; Kammerer and MacCluer, 1988; Knapp *et al.*, 1990; Lander and Green, 1987; Olson and Boehnke, 1990; Thompson, 1987; Weeks and Lange, 1987). We view that the elements of a correlation matrix, constructed from two-point recombination fractions r_{ij}, $1 \le i < j \le n$ for a pair of loci i and j, share a complex correlation for n loci in a linkage group. The Fiedler vector from the graph theory has been used for several graph manipulations such as partitioning (Pothen *et al.*, 1990), linear labelling (Juvan and Mohar, 1992) and envelope minimization (Barnard *et al.*, 1993; Juvan and Mohar, 1992). Recently, the Fiedler vector was successfully employed in clustering and classification problem of complex networking phenomena (Kim *et al.*, 2008).

2 Methods

2.1 Laplacian matrix and Fiedler vector

The adjacency matrix A of a graph G with n vertices is defined as an $n \times n$ symmetric matrix with components a_{ij}, where the diagonal elements a_{ii} are equal to zero for all $i = 1, 2, \ldots, n$. The Laplacian matrix of a graph G is defined as $L = D - A$, where D, called the degree matrix, is a diagonal matrix with the ith diagonal element $d_i = \sum_{j=1}^{n} a_{ij}$. Note that the Laplacian matrix is symmetric and positive semidefinite. The exact algebraic property of the Laplacian matrix is the following. If the graph is one connected cluster as a whole, the rank of the Laplacian matrix is n - 1, so that the smallest eigenvalue of $L = L(G)$ is always zero with constant eigenvector and all other eigenvalues are positive. Let $0 = \lambda_1 < \lambda_2 < \cdots < \lambda_n$ be n eigenvalues of L in an increasing order, then an eigenvector corresponding to λ_2, the nonzero smallest eigenvalue, is called the Fiedler vector and an eigenvector corresponding to λ_n, the largest eigenvalue, is called the Frobenius vector. The sum of eigenvector elements for each of nonzero eigenvalue is always zero. If the graph, however, consists of M disconnected clusters, the number of zero eigenvalue of the Laplacian matrix $L(G)$ for the whole graph is equal to M. For each disconnected cluster, the exact algebraic property for eigenvalues and eigenvectors mentioned above still holds.

2.2 Eigenvalue equation for a Laplacian matrix

Let a_{ij} denote a closeness in the sense of genetic distance such as the Haldane distance or the Kosambi distance between two loci i and j, which are proportional to r_{ij} for very small distance. That is to say, if two loci i and j are closely located, then a_{ij} is large, and if they are far away from each other, then a_{ij} is small. Also let $\mathbf{z} = (z_1, \ldots, z_n)'$, where z_i denotes the relative order of the ith locus among n loci. Then, the motivation on loci ordering is as follows; if two loci i and j are closely located (i.e. large a_{ij}), then z_i and z_j should be very close. On the other hand, if two loci i and j are located far away from each other (i.e. small a_{ij}), then z_i and z_j should be very different. Hence, our goal can be achieved by minimizing the weighted sum of squares

$$Q = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(z_i - z_j)^2 a_{ij}$$

To avoid the trivial solution $z_i = 0$ for all i, the constraint $\mathbf{z}'\mathbf{z} = 1$ is imposed. Also, the constraint $\mathbf{z}'\mathbf{1} = 0$, where $\mathbf{1} = (1, \ldots, 1)'$, is imposed since the minimum is invariant under translations.

Therefore, the problem can be rewritten as

$$\arg\min_{z} \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(z_i - z_j)^2 a_{ij}$$

subject to $\mathbf{z}'\mathbf{z}=\mathbf{1}$ and $\mathbf{z}'\mathbf{1}=0$

To solve the problem, note that

$$\begin{aligned} Q &= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(z_i^2 - 2z_i z_j + z_j^2)a_{ij} \\ &= \sum_{i=1}^{n} z_i^2 d_i - \sum_{j=1}^{n}\sum_{i\neq j}^{n} z_i z_j a_{ij} \\ &= \mathbf{z}'\mathbf{L}\mathbf{z} \end{aligned}$$

To minimize Q subject to $\mathbf{z}'\mathbf{;z} = 1$, use Lagrangian method, i.e. for a Lagrangian multiplier λ,

$$\begin{aligned} \mathbf{T} &= \mathbf{z}'\mathbf{L}\mathbf{z} - \lambda(\mathbf{z}'\mathbf{z} - 1) \\ \frac{\partial \mathbf{T}}{\partial \mathbf{z}} &= 2\mathbf{L}\mathbf{z} - 2\lambda\mathbf{z} = 0 \\ &\Rightarrow (\mathbf{L} - \lambda\mathbf{I})\mathbf{z} = 0 \end{aligned}$$

which yields a nontrivial solution $\mathbf{z}$ if and only if λ is an eigenvalue of L and $\mathbf{z}$ is the corresponding eigenvector. By multiplying $\mathbf{z}'$ on both sides, we have

$$\mathbf{z}'\mathbf{L}\mathbf{z} = \lambda$$

Therefore, the nonzero smallest eigenvalue and the associated eigenvector, which is just the Fiedler vector, yield the optimal solution.

3 Results

3.1 Strategy of Laplacian ordering method for loci-ordering

We describe how to use the Fiedler vector to the loci-ordering based on estimates of recombination fraction r_{ij} between two loci i and j,

$1 \leq i < j \leq n$. Let the adjacency between two loci i and j be $a_{ij} = 1 - r_{ij}$ because if two genes are closely located the a_{ij} will be close to 1. On the other hand, if these are far away from each other, a_{ij} will be small because r_{ij} becomes large. Also let $\mathbf{z} = (z_1, \ldots, z_n)'$, where z_i denotes the relative order of the ith gene among n genes. For example, if $\mathbf{z} = (-1/\sqrt{6}, 0, 1/\sqrt{6}, -\sqrt{2}/\sqrt{6}, \sqrt{2}/\sqrt{6})$ then the resulting order is (2, 3, 4, 1, 5) or (4, 3, 2, 5, 1). Therefore, estimating the order of n loci corresponds to finding $\mathbf{z}$. It can be shown that $\mathbf{z}$ is the eigenvector associated with the nonzero smallest eigenvalue of the Laplacian matrix of a_{ij}, and is just the Fiedler vector.

For an illustrative example, we consider 26 loci of barley chromosome IV generated by the North American Barley Genome Mapping Project (NABGMP) (Liu, 1998). The adjacency matrix for 26 loci is represented by a heat map in Figure 1a when they are randomly ordered. The simple subjection of Laplacian ordering method to the adjacency matrix for 26 loci produced loci-ordering results which are contrary to our expectation because raw data are quite often contaminated by noise which might give rise to poor results. Several methods are available to remove the noise, and the idea of shrinkage (James and Stein, 1961) is one of the widely used methods among them and further developed (Bickel, 1983; Donoho and Johnston, 1994; Efromovich, 1985; Efromovich and Pinsker, 1982; Nussbaum, 1985). Among methods of shrinkage, the hard-threshold, given by $t_H(x) = xI(|x| > \delta)$ where $\delta > 0$ is a threshold parameter to be estimated, is often used. We note that the hard-threshold is a fixed threshold in the sense that it annihilates all the components, which are less than δ, of the adjacency matrix. In the Fiedler vector, the ordering is given when a small δ is used, and the clustering given when a large δ is used, however, it is not easy to choose an appropriate δ for either ordering or clustering (Kim *et al.*, 2008).

Here we propose a new method of shrinkage, called the variable threshold which turns out to be especially useful in the multiloci-ordering problem. The idea of variable threshold is intuitive and very simple. In each row (or column) of the adjacency matrix a_{ij}, we replace each value by a very small value Δ, for example $\Delta = 0.01$ in order to keep the whole cluster as a connected one, except k largest values, where k is a parameter to be estimated. Consequently, the degree of threshold for each row (or column) of the adjacency matrix will be different, and that's why we call this method the variable threshold. Therefore, it is apparent that the hard-threshold does not consider the characteristic aspects of each row (or column) of the adjacency matrix whereas the variable threshold is flexible because it keeps characteristics of at least k adjacent loci in networks. Determination of k is based on the following arguments. Note that for large k, there will be a lot of non-small components still keeping the noisy contributions in the adjacency matrix. On the other hand, for small k, there will be a lot of very small components in the adjacency matrix. In general, the components of Fiedler vector corresponding to the non-zero smallest eigenvalue manifest relative weight hence ordering among them, and contain different values for large k. For small k, they contain several groups of same values, and each group corresponds to a different cluster. These different clusters could also be identified easily if we take $\Delta = 0.0$ since in this case the number of zero eigenvalues upon diagonalizing a Laplacian matrix a_{ij} is equal to the number of disconnected hence different clusters. To be more specific, the Fiedler vector shows the clustering pattern for small k, and the ordering pattern for large k. We choose k at which the pattern of the Fiedler vector changes from the clustering to the ordering. Therefore, we need to increase k from 2 to the integer at which the clustering pattern disappears and the ordering behaviour is maximally achieved since the noisy components of the adjacency matrix are minimally included.

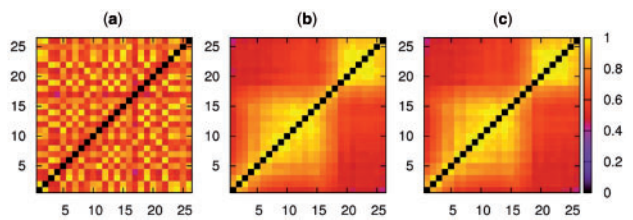

Fig. 1. The heat map representation of an adjacency matrix $\mathbf{A} = (a_{ij}) = (1 - r_{ij})$, where r_{ij} is the recombination fraction between loci i and j for 26 barley data by (**a**) the random loci-order, (**b**) the loci-ordering via a Laplacian ordering method with a variable threshold $k = 3$ and (**c**) the true loci-order. It shows the closeness of (b) to (c)

To see how variable threshold affects estimation of loci-ordering, we first define three measures for the performance of estimators. Let $\mathbf{t} = (t(1), \ldots, t(n))$ and $\mathbf{e} = (e(1), \ldots, e(n))$, where $t(i)$ denotes the true order of the ith gene and $e(i)$ denotes the estimated order of the ith gene, respectively. Also, let $\mathbf{c} = (c(1), \ldots, c(n))$, where $c(i)$ denotes the bootstrap confidence interval of 95% for the locus of ith gene, implying that among 1000 bootstrap gene orders made by resampling technique the locus is located at the map position 95% of the time (Efron and Tibshirani, 1998; Liu, 1998). Therefore, the best possible order of the 26 loci is 1, 2, …, 26, where each number represents the relative locus of each gene. If $\mathbf{t}$ and $\mathbf{c}$ are known, the accuracy of the estimated order $\mathbf{e}$ can be measured by NCL= $\sum_{i=1}^{n} I(t(i) = e(i))$, NCLCI= $\sum_{i=1}^{n} I(e(i) \in c(i))$, and SAD= $\sum_{i=1}^{n} |t(i) - e(i)|$, where NCL denotes the number of correctly ordered loci, NCLCI denotes the number of correct loci contained in the bootstrap confidence interval and SAD denotes the sum of absolute difference between the true order and the estimated order. Note that NCLCI is a relaxed accuracy measure compared to NCL. Therefore, good estimates should have both the large value of NCL and NCLCI and the small value of SAD simultaneously.

3.2 Loci-ordering of 26 loci of barley chromosome IV data

Having a simple and general conceptual picture for both the clustering and the ordering of any given adjacency matrix, guaranteed by the exact algebraic property of a Laplacian matrix and its Fiedler vector, we apply the Laplacian ordering method with the variable threshold to the 26 loci of barley chromosome IV data. Firstly, we take $\Delta = 0.0$ and diagonalize Laplacian matrices L for $k = 2, 3, 4, \ldots, 25$ in order to identify a value k^* which separates the clustering behaviour from the ordering behaviour of the Fiedler vector. It turns out that the number of zero eigenvalues is 2 for $k = 2$, and 1 for $k > 2$. It immediately implies that k^* is 3 at which the ordering behaviour of components of the Fiedler vector is best envisaged compared to those for $k > 3$. Secondly, we also take $\Delta = 0.01$ and repeat diagonalizing L. In this case the number of zero eigenvalues is 1 for all $k = 2, 3, 4, \ldots, 25$. The Fiedler vector for $k = 2$, however, also shows a clustering pattern of two distinct region of loci whereas those for $k > 2$ show an ordering pattern as listed in the second row of Supplementary Table S1. Since the components of Fiedler vector manifest the relative weight of each of 26 loci, we rank order them for $k > 2$ in the ascending order, and list both the component of Fiedler vector and the corresponding rank next to it in Supplementary Table S1.

The quality of the predicted loci-order compared to the true loci-order is estimated for each k by the values of NCL, NCLCI and SAD. Supplementary Figure S1 illustrates the values of these three

estimators as k changes from 25 down to 3. The best loci-ordering occurs at $k = 3$ at which NCL, NCLCI (SAD) attains the maximum (minimum) value of 18, 26 (12) simultaneously. The fact that NCLCI is also 26 for $k = 16$–19 does not necessarily imply the occurrence of the best loci-order because NCL and SAD for these k values are much worse than those for $k = 3$. Based on the predicted loci-order for $k = 3$ from our Laplacian ordering method, we also illustrate the closeness of a predicted heat map (see Fig. 1b) of the adjacency matrix, which is constructed from the random heat map (see Fig. 1a) without any loci-ordering a priori, to that (see Fig. 1c) from the true loci-order.

3.3 Practical application to a high-density genetic map

We apply the Laplacian ordering method to data with a high-density genetic map. We use an experimental data with single feature polymorphism (SFP) makers and a recombinant inbred lines (RIL) genotype population derived from *Arabidopsis thaliana*, which is available in ATGC website served by Kozik (http://www.atgc.org/XLinkage/MadMapper/ath_sfp_map_example). This website provides detail procedures constructing the linkage maps by the MadMapper software scripts (West *et al.*, 2006). The RIL population with 846 genetic markers is employed in this work, which is believed to be high quality data and obtained by filtering steps such as non-redundant scores and 'parental min-max method', 'RIL distribution' algorithms for removing ambiguous genotypes and reducing missing genotype scores (West *et al.*, 2006).

The recombination fractions from the RIL population for 846 makers are constructed by using a MadMapper script. The initial adjacency matrix given by randomized index-headers is constructed (see Fig. 2a). We diagonalize the Laplacian matrix with $\Delta = 0.00$ at different variable threshold k by increasing from $k = 3$–42. It takes 533 seconds over total 39 diagonalizing processes based on a Linux machine with 3.3GHz Xeon CPUs. The numbers of clusters (linkage groups) versus k in Figure 2b are identified by counting the number of zero eigenvalue. Interestingly we find five clusters at very wide range of variable threshold from $k = 12$ to $k = 41$. Grouping of 846 genetic markers into 5 chromosomes are made by the clustering pattern (grouping elements with same values) in the Fiedler vector at $k = 12$ and with $\Delta = 0.01$. Classification into five linkage groups exactly correspond to the separation of genetic markers on the five chromosome (209, 122, 156, 140, 219 markers for chromosome 1, 2, 3, 4, 5, respectively) in Arabidopsis. The heat map in Figure 2c shows not ordering but clustering of genetic markers by the Fiedler vector (five block sizes are 209, 122, 219, 140, 156).

Loci-ordering of 846 genetic markers should be performed separately on five chromosomes. And the quality of the resulting loci-order from Laplacian approach in this work should be compared with that

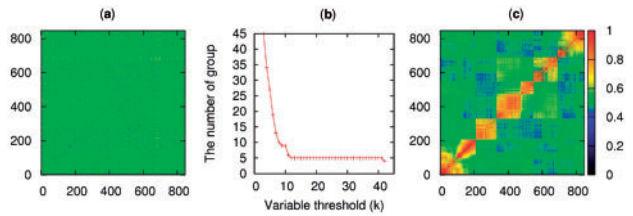

Fig. 2. (**a**) The heat map representation of an initial adjacency matrix given by randomized index-headers of total 846 genetic markers in Arabidopsis thaliana data. (**b**) The number of linkage group obtained by counting the number of zero eigenvalues versus k. (**c**) The heat map representation of the adjacency matrix given by clustering pattern of the Fiedler vector at $k = 12$ and with $\Delta = 0.01$

from the existing other method, such as MadMapper (West *et al.*, 2006). The RIL populations of five linkage groups (see Fig. 3a–e) are plotted by the MadMapper script for chromosome 1, 2, 3, 4, 5, respectively. We generate the initial adjacency matrix (see Fig. 3f–j) given by randomized index-headers of markers and the heat maps ordered by the MadMapper software scripts (see Fig. 3k–o). The Laplacian loci-ordering method also provides heat maps (see Fig. 3p–t) which are very similar to the heat maps by the MadMapper method. The variable threshold k^* at which clustering pattern disappears are 8, 12, 11, 7, 11 for chromosome 1, 2, 3, 4, 5, respectively. For the chromosome 1 having 209 markers, we checked the computation time. By the MadMapper script which uses the usual optimization algorithm minimizing total scores, it takes 1 h and 32 min based on a Linux machine with 3.3GHz Xeon CPUs. By using the Laplacian loci-ordering method, it takes only 3.6 s for performing total 11 Laplacian procedures from $k = 3$ to $k = 12$ with $\Delta = 0.00$ and at $k^* = 8$ with $\Delta = 0.01$. The comparison plots between two approaches in loci-ordering (see Fig. 3u–y) show globally similar but slight differences in dense regions where recombination fractions between neighboring loci are very small.

3.4 Missing data and reliability score for each locus

The 148 RIL population with 209 genetic markers on the chromosome 1 has 0.6% missing genotypes through filtering process. All recombination fractions between loci (genetic markers) can be evaluated even through filtering process. Now we check how the Laplacian loci-ordering method works well even with missing genotypes. We randomly replace genotypes to be missed artificially from 1% to 10% composition more in the RIL data. For missing replacements from 1% (total 1.6%) to 4% (total 4.6% missing genotypes), all recombination fractions can be evaluated. However values of recombination fractions are somewhat changed so that the loci-ordering is also changed (SAD between two loci-orders with 0.6% and 4.6% missing genotypes is 148) in Figure 4a and b. With total 5.6% missing genotypes, we observe missing recombination fractions. In constructing adjacency matrix, we assign the recombination fraction to be 0.5 for those missing data. While we see a little missing data (0.096%) in the adjacency matrix for total 5.6% missing genotypes, we see many missing data (18.2%) in Figure 4c for total 8.6% missing genotypes, but the loci-ordering is not significantly changed (SAD between two loci-orders is 290) in Figure 4d. For further missing recombination fractions (37.6%) in Figure 4e for total 9.6% missing genotypes, the loci-ordering is not working as shown in Figure 4f. The heat map shows bad ordering due to many missing data. The comparison in loci-ordering between 0.6% and 9.6% missing genotypes shows no more linear correlation (SAD between two loci-orders is 8058). Therefore we recognize that our Laplacian approach is working even with 8.6% missing genotypes and 18.2% missing recombination data for 209 loci on the chromosome 1.

The input data of our software using the Laplacian matrix are not the RIL populations or f2 intercross populations directly determined by experiments but the recombination fractions evaluated from the RIL or f2 intercross data. Hence we do not present any criteria in filtering process for reducing missing genotype scores and removing uncertain genotypes. Instead our program provides a key algorithm for loci-ordering based on recombination fractions not providing whole packages from processing experimental data to mapping linkage maps. Even though we do not have the filtering steps, our method provides excellent loci-ordering among high quality loci data even under the condition that low quality loci data were slightly embedded.

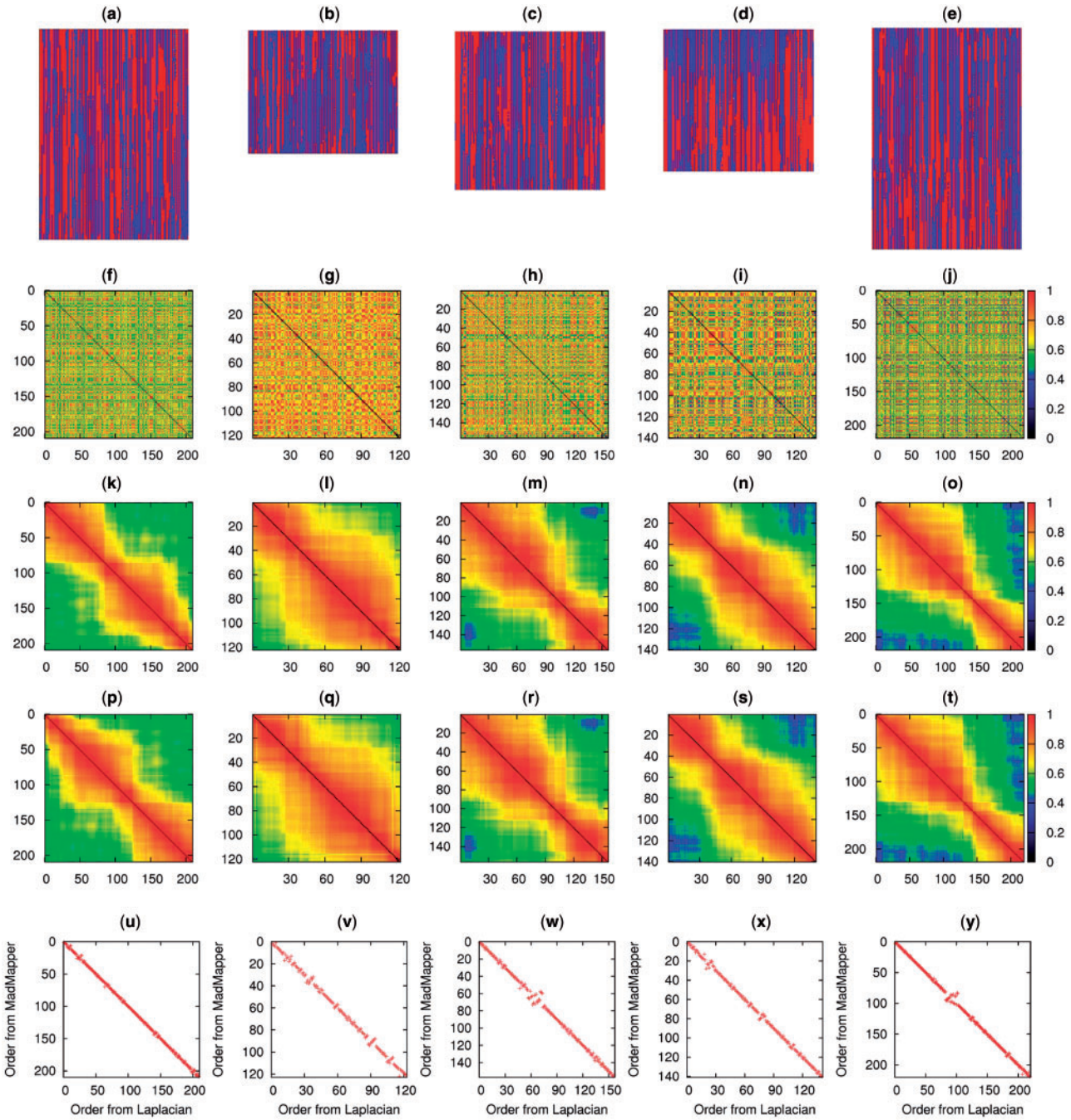

Fig. 3. (**a**)–(**e**) The RIL populations of five linkage groups (chromosome 1, 2, 3, 4, 5, respectively). (**f**)–(**j**) The initial adjacency matrix given by randomized index-headers of markers. (**k**)–(**o**) The heat maps ordered by the MadMapper software scripts. (**p**)–(**t**) The heat maps ordered by the Laplacian loci-ordering method in this work. The variable threshold parameters k^* are 8, 12, 11, 7, 11 for chromosome 1, 2, 3, 4, 5, respectively. (**u**)–(**y**) The comparison of the resulting loci-order from MadMapper approach with that from our Laplacian approach

The Laplacian loci-ordering method could not add any weighting factor on specific loci of previously known maps or recombination fractions derived from high quality statistics. It means that the elements of Laplacian matrix after filtering the variable threshold process have equal weights in diagonalizing L, so that we could not assign any reliability scores for each locus at a given k. Instead we have to determine the reliability scores for each locus analyzing over several loci-orders at variable threshold $k > k^*$. As we see in the previous sections and Supplementary Figure S1, we obtain the best loci-ordering at k^* where clustering pattern disappears in increasing k. At $k > k^*$, a bit poor orders might be given at some markers due to a little noisy data. But we get reliable placements of markers checking over loci-orders at $k > k^*$. Reliability scores for 26 loci of barley chromosome and 209 loci of Arabidopsis are given in the program package of the supplementary data.

3.5 Application to F1 populations in outbreeding species

We apply the Laplacian ordering method to additional complex data with F1 apple populations (Gardner *et al*, 2014). This

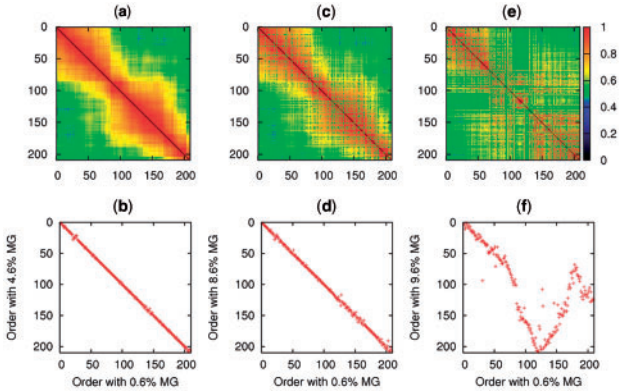

Fig. 4. The heat maps ordered by Laplacian loci-ordering method and the loci-order comparison plots between 0.6% missing genotype (MG) in the RIL data for the chromosome 1 and (**a, b**) 4.6%, (**c, d**) 8.6% and (e, f) 9.6% missing genotypes. Missing genotypes are assigned randomly in the 148 RIL population with 209 genetic markers

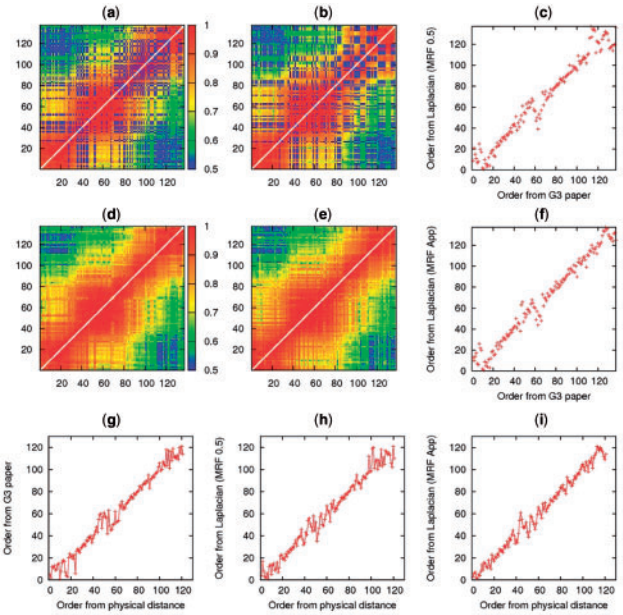

Fig. 5. The heat maps ordered by (**a, d**) JoinMap package, (**b, e**) Laplacian method and (**c, f**) loci-order comparison plots between two methods. The adjacency matrix was built under assigning the missing data as (a–c) $r_{ij} = 0.5$ (MRF 0.5) and (d–f) approximate r_{ij} (MRF App). The loci-order comparison plots of physical map to orders by using (**g**) JoinMap package published in G3 paper, Laplacian ordering with (**h**) $r_{ij} = 0.5$ missing data (MRF 0.5) and (**i**) approximate r_{ij}

outbreeding heterozygous system is analyzed by the next-generation DNA sequencing method, which are cost-effective but provide relatively low quality data with many missing data. Initially 273 835 SNP markers were identified by sequence alignments but after many filtering processes only 1903 markers could be considered in constructing genetic linkage maps. We constructed two-point recombination fractions r_{ij} by counting recombinant events and using the Newton-Raphson algorithm to find out the maximum LOD score (Zhang *et al*, 2015). The Laplacian clustering procedure for 1903 markers results in 17 linkage groups which are exactly consistent with published results by Gardner *et al*. Loci-ordering by Laplacian method was performed for all 17 chromosomes. The heat maps and comparison plots of orders for the chromosome 3 are shown in Figure 5. We observe moderate correlations between two orders by using the JoinMap package and the Laplacian loci-ordering method. The comparison plots of physical map to genetic maps constructed by different methods reveal that both correlations are moderate since the original data were not good enough to provide high quality genetic maps. But we can advocate that our Laplacian loci-ordering can be applicable to the further complex genotyping populations. The detailed descriptions and results for other chromosomes are given in the supplementary data.

3.6 Software for the Laplacian loci-ordering method

We provide three key files (main fortran program, input parameter data file, input recombination fraction data file) and peripheral files (Manual.txt, Score.f, GeneMarker.dat, Aij_three.gnu, etc) for describing software package and analyzing output data. Here we present the brief procedures for making use of the loci-ordering approach. The detailed step-by-step procedures for using the loci-ordering and clustering linkage groups are presented in manual files of software package.

1. Run a main program (executable file for Laplacian_Loci.f) with $\Delta = 0.0$ and at variable threshold k=2, 3, 4, 5, . . .
2. Check the number of clusters by counting the number of zero eigenvalues in an output file (Output file: Laplacian_Eigenvalues.dat). Determine k^* (when the number of zero eigenvalue is 1 in increasing k).
3. Run a main program (executable file for Laplacian_Loci.f) with $\Delta = 0.01$ and at k^*.
4. Sorting the elements of the Fiedler vector. We provide two orders (ascending order and descending order in sorting the elements. Output files: Loci_ordering_ascending.dat and Loci_ordering_descending.dat)

4 Conclusions

In this work we present a simple Laplacian loci-ordering method which enables one to uncover the loci-order of multiloci simultaneously, which has been known as a challenging problem. This approach is made possible by taking the close analogy with the important conceptual development recently achieved for characterizing various complex network phenomena. Once constructing a Laplacian matrix from the recombination fraction of multiloci, the exact properties for the Fiedler vector of Laplacian matrix together with the variable threshold are directly applied to uncover the loci-order of 26 loci of barley chromosome IV data, 846 loci of Arabidopsis chromosome data and 1903 loci of heterozygous apple data. It offers both a simple method without any assumptions a priori and a very fast one computationally for uncovering not only the clustering behaviour but also the ordering behaviour of any number of loci simultaneously.

Funding

This work was supported by the Creative Research Initiatives (Center for Proteome Biophysics, Grant No. 2011-0000041) (M.C. and I.C.) and Basic Science Research Program (2013012741) through the National Research Foundation, Korea (C.K.). It is also supported by DGIST MIREBrain program of Ministry of Science, ICT and Future Planning (2015010013) (M.C. and I.C.).

Conflict of Interest: none declared.

References

Amaral,L.A.N. *et al*. (2000) Classes of small-world networks. *Proc. Natl. Acad. Sci. USA*, **97**, 1149–1152.

Barnard,S.T. *et al*. (1993) A spectral algorithm for envelope reduction of sparse matrices. In: *Proceedings of the Supercomputing '93,* pp. 493–502.

Bickel,P.J. (1983) Minimax estimation of a normal mean subject to doing well at a point. In: Rizvi,M.H. Rustagi,J.S. and Siegmund,D. (eds.) *Recent Advances in Statistics*. Academic Press, New York, pp. 511–528.

Buetow,K.H. and Chakravarti,A. (1987) Multipoint Gene-Mapping Using Seriation .2. Analysis of Simulated and Empirical-Data. *Am. J. Hum. Genet.*, **41**, 189–201.

Buldyrev,S.V. *et al*. (2010) Catastrophic cascade of failures in interdependent networks. *Nature*, **464**, 1025–1028.

Cheema,J. and Dicks,J. (2009) Computational approaches and software tools for genetic linkage map estimation in plants. *Brief. Bioinf.*, **10**, 595–608.

Chen,Y. *et al*. (2006) Universal behaviour of optimal paths in weighted networks with general disorder. *Phys. Rev. Lett.*, **96**, 068702.

Criscione,C.D. *et al*. (2009) Genomic linkage map of the human blood fluke *Schistosoma mansoni*. *Genome Biol.*, **10**, R71.

de Givry,S. *et al*. (2005) CAR(H)(T)AGene: multipopulation integrated genetic and radiation hybrid mapping. *Bioinformatics*, **21**, 1703–1704.

Donoho,D.L. and Johnston,I.M. (1994) Ideal spatial adaptation by wavelet shrinkage. *Biometrika* **81**, 425–455.

Efron,B. and Tibshirani,R.J. (1998) *An Introduction to the Bootstrap*. CRC Press, New York.

Efromovich,S.Y. (1985) Nonparametric estimation of a density with unknown smoothness. *Theory Prob. Appl.*, **30**, 557–568.

Efromovich,S.Y. and Pinsker,M.S. (1982) Estimation of square-integrable probability density of a random variable. *Problems Inf. Trans.*, **18**, 175–189.

Ellis,T.H.N. (1997) Neighbour mapping as a method for ordering genetic markers. *Genet. Res.*, **69**, 35–43.

Falk,C.T. (1989) A simple scheme for preliminary ordering of multiple loci: application to 45 CF families. In: Elston,R.C., Spence,M.A., Hodge,S.E. and MacCluer,J.W. (eds.) *Multipoint Mapping and Linkage Based Upon Affected Pedigree Members*. Genetic Workshop 6. Liss, pp. 17–22.

Fiedler,M.A. (1973) Algebraic connectivity of graphs. *Czech. Math. J.*, **23**, 298–305.

Fiedler,M.A. (1975) Property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. *Czech, Math. J.*, **23**, 298–305.

Gardner,K.M. *et al*. (2014) Fast and cost-effective genetic mapping in apple using next-generation sequencing. *G3-Genes Genom Genet.*, **4**, 1681–1687.

Getz,G. *et al*. (2000) Coupled two-way clustering of gene microarray data. *Proc. Natl. Acad. Sci. USA*, **97**, 12079–12084.

Goss,S.J. and Harris,H. (1975) New method for mapping genes in human chromosomes. *Nature*, **255**, 680–684.

Hackett,C.A. and Broadfoot,L.B. (2003) Effects of genotyping errors, missing values and segregation distortion in molecular marker data on the construction of linkage maps. *Heredity*, **90**, 33–38

Iwata,H. and Ninomiya,S. (2006) AntMap: Constructing genetic linkage maps using an ant colony optimization algorithm. *Breeding Sci.*, **56**, 371–377.

James,W. and Stein,C. (1961) Estimation with quadratic loss. In: Proceedings of the 4th Berkeley Symposium in Mathematical Statistics and Probability., **1**, 361–379.

Juvan,M. and Mohar,B. (1992) Optimal linear labelling and eigenvalues of graphs. *Disc. Appl. Math.*, **36**, 153–168.

Kammerer,C.M. and MacCluer,J.W. (1988) Empirical power of three preliminary methods for ordering loci. *Am. J. Hum. Genet.*, **43**, 964–970.

Kanaar,R. and Hoeijmakers,J.H.J. (1998) Genetic recombination: From competition to collaboration. *Nature*, **345**, 477–478.

Kim,C. *et al*. (2008) A simple and exact Laplacian clustering of complex networking phenomena: Application to gene expression profiles. *Proc. Natl. Acad. Sci. USA*, **105**, 4083–4087.

Knapp,S.J. *et al*. (1990) Mapping quantitative trait loci using molecular marker linkage maps. *Theor. Appl. Genet.*, **79**, 583–592.

Lander,E.S. *et al*. (2009) MAPMAKER: an interactive computer package for constructing primary genetic linkage maps of experimental and natural populations (vol. 1 pp. 174, 1987), *Genomics*, **93**, 398–398.

Lander,E.S. and Green,P. (1987) Construction of multilocus genetic linkage maps in human. *Proc. Natl. Acad. Sci. USA*, **84**, 2363–2367.

Liljeros,F. *et al*. (2001) The web of human sexual contacts. *Nature*, **411**, 907–908.

Liu,B.H. (1998) *Statistical Genomics*. CRC Press, New York.

Manly,K.F. *et al*. (2001) Map manager QTX, cross-platform software for genetic mapping. *Mamm. Genome*, **12**, 930–932.

Mohar,B. (1991) In: Alavi,Y. *et al*. (ed.) *Graph Theory, Combinatories and Applications*. John Wiley, New York, pp. 871–898.

Mohar,B. (1992) Laplace eigenvalues of graphs – a survey. *Disc. Math.*, **109**, 171–183.

Nussbaum,M. (1985) Spline smoothing in regression models and asymptotic efficiency in L_2. *Ann. Statist.*, **13**, 984–997.

Olson,J.M. and Boehnke,M. (1990) Monte Carlo comparison of preliminary methods for ordering multiple genetic loci. *Am. J. Hum. Genet.*, **47**, 470–482.

Ott,J. (1999) *Analysis of Human Genetic Linkage*. 3rd edn. The Johns Hopkins University Press, London.

Pothen,A. *et al*. (1990) Partitioning sparse matrices with eigenvectors of graph. *SIAM J. Matr. Anal. Appl.*, **11**, 430–452.

Thompson,E.A. (1987) Crossover counts and likelihood in multipoint linkage analysis. *IMA J. Math. Appl. Med. Biol.*, **4**, 93–108.

Thompson,E.A. (1988) Two-locus and three-locus gene identity by descent inpedigrees. *IMA J. Math. Appl. Med. Biol.*, **5**, 261–279.

Van Ooijen,J.W. (2006) JoinMap 4, Software for the calculation of genetic linkage maps in experimental populations. *Kyazma B.V.*, Wageningen, The Netherlands.

Van Os,H. *et al*. (2005) RECORD: a novel method for ordering loci on a genetic linkage map. *Theor Appl. Genet.*, **112**, 30–40.

Weeks,D.E. and Lange,K. (1987) Preliminary ranking procedures for multilocus ordering. *Genomics*, **1**, 236–242.

West,M.A.L. *et al*. (2006) High haplotyping with microarray-based expression and single feature polymorphism markers in Arabidopsis. *Genome Res.*, **16**, 787–795.

White,R. *et al*. (1985) Construction of linkage maps with DNA markers for human chromosome. *Nature*, **313**, 101–105.

Wu,Y.H. *et al*. (2008) Efficient and accurate construction of genetic linkage maps from the minimum spanning tree of a graph. *Plos Genet.*, **4**, e1000212.

Zhang,L.Y. *et al*. (2015) Linkage analysis and map construction in genetic populations of clonal F-1 and double cross. *G3-Genes Genom Genet.*, **5**, 427–439.