A Phylogenetic Approach to Genetic Map Refinement

Denis Bertrand[1], Mathieu Blanchette[2], and Nadia El-Mabrouk[3]

[1] DIRO, Université de Montréal, H3C 3J7, Canada, bertrden@iro.umontreal.ca
[2] McGill Centre for Bioinformatics, McGill University, H3A 2B4, Canada.
blanchem@mcb.mcgill.ca
[3] DIRO, mabrouk@iro.umontreal.ca

Abstract. Following various genetic mapping techniques conducted on different segregating populations, one or more genetic maps are obtained for a given species. However, recombination analyses and other methods for gene mapping often fail to resolve the ordering of some pairs of neighboring markers, thereby leading to sets of markers ambiguously mapped to the same position. Each individual map is thus a partial order defined on the set of markers, and can be represented as a Directed Acyclic Graph (DAG). In this paper, given a phylogenetic tree with a set of DAGs labeling each leaf (species), the goal is to infer, at each leaf, a single combined DAG that is as resolved as possible, considering the complementary information provided by individual maps, and the phylogenetic information provided by the species tree. After combining the individual maps of a leaf into a single DAG, we order incomparable markers by using two successive heuristics for minimizing two distances on the species tree: the breakpoint distance, and the Kemeny distance. We apply our algorithms to the plant species represented in the Gramene database, and we evaluate the simplified maps we obtained.

1 Introduction

Similarly to a road map indicating landmarks along a highway, a genetic map indicates the position and approximate genetic distances between markers along chromosomes. Genetic mapping using DNA markers is a key step towards the discovery of regions within genomes containing genes associated with particular quantitative traits (QTLs). This is particularly important in crops and other grasses, where the localization of markers linked to genes playing major roles in traits such as yield, quality, and disease resistance, can be harnessed for agricultural purposes [3].

In order to fulfill their purpose of locating QTLs as precisely as possible, ideal genetic maps should involve as many markers as possible, evenly distributed over the chromosomes, and provide precise orders and distances between markers. In reality, recombination analysis, physical imaging and the other methods used for genetic mapping only give an approximate evaluation of genetic distances between markers, and often fail to order some pairs of neighboring markers, leading to partial orders, with sets of incomparable markers, that is, set of markers

affected to the same locus. Moreover, to identify a specific marker locus, one requires polymorphisms at that locus in the considered population. As different populations do not contain polymorphisms for all the desired loci, the different genetic maps obtained for the same species on the basis of different segregating populations generally contain different markers. However, as long as some common markers are used, individual maps can be combined into a single one.

Various approaches have been considered to integrate different maps of a single species. As genetic distances are poorly comparable between maps, a standard approach has been to reduce each map to the underlying partial order between markers. This simplification allows representing a map as a Directed Acyclic Graph (DAG), where nodes correspond to markers, and paths between nodes to the ordering information [18]. Combining DAGs from different maps may lead to cycles, corresponding to conflicts (two markers A and B that are ordered $A \rightarrow B$ in one map, and $B \rightarrow A$ in another). Different approaches have been considered to cope with such conflicts. In [18], a DAG is recovered by simply "condensing" the subgraph corresponding to a maximum subset of "conflicting" vertices into a single vertex. In [6, 7] the authors find a median by removing a minimum number of conflicts.

In contrast to the work that has been done for combining information of different maps of a single species, no similar effort has been expended to improve the markers' partial order information on one species on the basis of the genetic information of related species. In this context, the only comparative genomic study for genetic mapping is the one that we have conducted [1] for linearizing a DAG representing the map of a given species, with respect to a related species for which a total order of markers is known. In the context of computing the rearrangement distance between two maps, a more general study has been conducted by Zheng *et. al* [19] for inferring the minimal sequence of reversals transforming one DAG into another. Another study by the same authors [20] has considered the problem of reconstructing synthenic blocks between two gene maps by eliminating as few noisy markers as possible.

In this paper, starting from a species tree and a set of DAGs (individual maps) labeling each leaf (species), the goal is to infer, at each leaf, a single combined DAG that is as resolved as possible, considering the complementary information provided by individual maps, and the phylogenetic information provided by the species tree. Ideally (assuming sufficient complementary information between maps and sufficient phylogenetic information), a complete linearization of the DAGs is desirable. However, as ideal situations are rarely encountered, we will consider the more restricted, but more biologically relevant problem, of integrating maps and reducing pairs of incomparable and conflicting markers of each DAG, given the phylogenetic signal.

We proceed as follows. After combining the individual maps of a leaf into a single DAG by using a method similar to that of Yap *et al.* [18], we resolve incomparable pairs of markers by using two successive heuristics for minimizing two distances on the species tree: the breakpoint distance, and the Kemeny

distance [10] defined as the total number of pairwise ordering conflicts over the branches of the tree.

The second heuristic is based on the previous work of Ma *et.al* [11] for reconstructing ancestral gene orders. The developed algorithm is guaranteed to identify a most parsimonious scenario for the history of each incomparable pair of markers, although it provides no guarantee as to the optimality of the global solution. The paper is organized as follows. We introduce all concepts and notations in Section 2. We then describe our methodology in Section 3, and present our two heuristics in Section 4. In Section 5, we apply our method to the Gramene database [8] and evaluate the simplified maps we obtain.

2 Gene order data and representation as graphs

Experimental methods used for genetic mapping give rise to individual maps, generally represented by lines upon which are placed individual loci (Figure 1, Map1 and Map2). Each locus represents the position of a specific marker that might appear at several positions in the genome. However, in this paper, we assume that each marker exhibits a single polymorphism along the genome, allowing to treat the concept of marker and locus synonymously. In other words, marker duplications are not allowed.

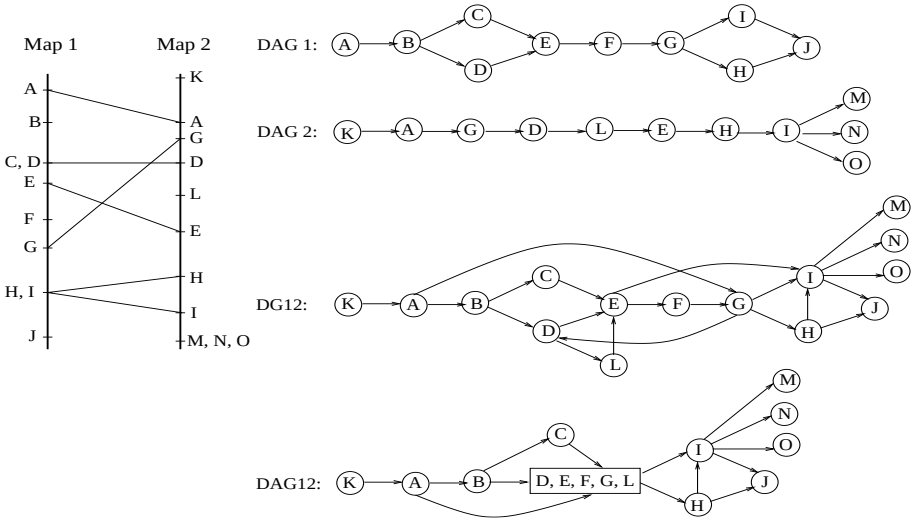

Fig. 1. Modeling maps as Directed Acyclic Graphs (DAGs). Left: maps as they are represented in a gene map database such as Gramene. Map1 and Map2 correspond to two mapping studies of the same chromosome. Letters correspond to markers placed at positions proportional to their distance from each other. Common markers between the two maps are linked. Right, from up to bottom: DAG representing Map1; DAG representing Map2; Directed Graph (DG) corresponding to the union of DAG1 and DAG2; DAG corresponding to map integration, after simplifying the strongly connected components.

Modeling a map as a DAG: Following the notations of [18], maps may be represented as Directed Acyclic Graphs (DAGs), where each marker is represented by a vertex, and each pair of adjacent markers are connected by an edge (Figure 1, DAG1 and DAG2). Often due to the lack of recombination between two loci, a number of different markers may appear at the same position on the map (for example, markers C and D in Map1). It follows that a single marker may be connected to a set of other markers.

Such DAGs represent partial orders between markers. Two markers A and B are *comparable* iff there is a directed path from A to B (in which case we write $A < B$) or from B to A (we write $A > B$), and *incomparable* otherwise. A *conflict* between two maps is a pair of markers A and B that are ordered $A < B$ on one map, and $A > B$ on the other. The *Kemeny distance* [10] between two DAGs (or partial orders) is the number of conflicts between them. For example, in Map1 (Figure 1), (A, C) is a pair of comparable markers ($A < C$, or similarly $C > A$) and (C, D) is a pair of incomparable markers. Moreover, the Kemeny distance between Map1 and Map2 is 20 since markers D, E, F, G and L are in conflict with each other.

Map integration: Different studies on the same species conducted on different populations give rise to different maps involving different markers. As long as some identical markers are shared between studies, maps can be merged to a Directed Graph (DG) with a single connected component, by performing the union of the individual maps [18]. More precisely, let $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_n$ be n DAGs corresponding to n maps, and $\mathcal{M}$ be the set of markers represented in at least one $\mathcal{D}_i$, for $1 \leq i \leq n$. Then the *union DG* is the directed graph $\mathcal{G}$ defined as follows: a vertex is in $\mathcal{G}$ iff it is in at least one $\mathcal{D}_i$, and an edge is in $\mathcal{G}$ iff it is in at least one $\mathcal{D}_i$, for $1 \leq i \leq n$ (Figure 1, DG12).

Due to conflicts between maps, such union DG may contain cycles (for example, (D, L, E, F, G) is a cycle in DG12). Markers involved in such cycles cannot be ordered relative to each other without yielding a contradiction. Two main approaches have been used in the literature to cope with cycles.

1. A *Strongly Connected Component* (SCC) of a DG $\mathcal{G}$ refers to a maximum subset $\mathcal{V}$ of vertices of $\mathcal{G}$ such that, for each $(v_1, v_2) \in \mathcal{V}^2$, there is a directed path in $\mathcal{G}$ from v_1 to v_2 and from v_2 to v_1. For example, $\{D, E, F, G, L\}$ is an SCC in DG12. A number of very efficient algorithms are able to find the SCCs in a graph [12, 16]. This yields to the possibility of simplifying a DG to a DAG by "condensing" the subgraph that comprises an SCC into a single vertex (DAG12 in Figure 1). Markers belonging to such a vertex are considered pairwise incomparable.

2. Based on the hypothesis that conflicts are due to mapping errors, Jackson *et.al* [7] considered the problem of inferring a consensus map leading to a minimum number of such errors. Their method is based on finding a median order for the Kemeny distance, which is an NP-hard problem [17]. They proved that inferring a median order according to this distance is equivalent

to finding an acyclic subgraph of minimum weight in a weighted directed graph (i.e. a minimum feedback arc set), and designed an exact algorithm and a heuristic to solve it.

3 Methodology

Given a phylogenetic tree T for a set of n species and a set of DAGs (set of individual maps) at each leaf of T, our goal is to produce a single DAG at each leaf of T that is as resolved as possible considering the shared information between maps and the phylogenetic information provided by T.

In the rest of this paper, *resolving a pair of incomparable markers* will refer to fixing an order between the two markers, and *simplifying a DAG* will refer to resolving a number of pairs of incomparable markers in the DAG.

3.1 Integrating maps

We integrate the set of DAGs labeling each leaf of T into a single DAG as follows. We first construct the DAG's union DG as described in Section 2. Then, in contrast to [7], we do not try to solve *conflicts of the union DG*, that is the pairs of markers involved in a cycle of the DG, at this stage. Rather, conflicts are reduced to SCCs, as in [18], and resolving such SCCs is delayed to the next phase considering the phylogenetic information of the species tree.

3.2 Marker content of internal nodes

We would like to account for the phylogenetic information represented by the species tree T. Considering a most parsimonious model of evolution, the goal is to infer marker orders that minimize a given distance on T. Preliminary to computing any distance on T is the assignment of marker content at each internal node. We will proceed as follows, assuming a model with no convergent evolution. Let M be a marker, $\mathcal{L}$ be the set of leaves that contain the marker M, and v be the node of T representing the least common ancestor of $\mathcal{L}$. Then, we assign M to each node belonging to a path from v to an element of $\mathcal{L}$.

3.3 Minimizing an evolutionary distance

In contrast to gene order data, maps do not provide information on adjacencies, but rather on relative orders between markers: an edge $A \rightarrow B$ in a DAG does not mean that A is adjacent to B, but rather that A precedes B on the chromosome. Indeed another DAG for the same species may contain an edge $A \rightarrow C$, leading to two possible total orders for the three markers: $A\,B\,C$ or $A\,C\,B$. Therefore, a classical gene order distance such as the inversion distance, or its reduction to the breakpoint distance, is not directly applicable to such data. In this case, a more natural distance is the number of conflicts between two maps, that is the Kemeny distance. In the case of a species tree, the Kemeny distance can be generalized as follows:

Definition 1. *Given a species tree T with a total order assigned to each node, the Kemeny distance on T is the sum of Kemeny distances of each pair of adjacent nodes (nodes connected by an edge of T).*

For the purpose of introducing our optimization problems, we recall the classical notion of a linear extension.

Definition 2. *Let $\mathcal{D}$ be a DAG on a set $\mathcal{M}$ of markers. A linear extension of $\mathcal{D}$ is a total order $\mathcal{O}$ of $\mathcal{M}$ such that if $A < B$ in $\mathcal{D}$ then $A < B$ in $\mathcal{O}$.*

Now, consider the following optimization problems, where "Given" should be replaced by either Kemeny, Breakpoint or Inversion:

Minimum-"Given" Linearization Problem
Given: A species tree T with a DAG at each leaf and a set of markers at each internal node;
Find: A total order at each internal node of T, and at each leaf of T, a linear extension of its DAG, minimizing the "Given" distance on T.

Notice that this problem is proved to be NP-hard for the breakpoint distance [13], for the inversion distance [2], and for the Kemeny distance [17].

The Minimum-Kemeny Linearization Problem is the one most directly applicable to partial orders. This problem is equivalent to the Minimum-Breakpoint Linearization Problem and the Minimum-Inversion Linearization Problem in the case of a marker set restricted to the same two markers $\mathcal{M} = \{A, B\}$ at each node and leaf of T. Moreover, it is equivalent to the Minimum-Inversion Linearization Problem with inversions restricted to segments of size 2 [14]. However, in the general case, a solution to the Minimum-Kemeny Linearization Problem is not guaranteed to minimize the inversion or breakpoint distance. Using this distance only allows combining the information obtained on closely related species, in case of no large genome rearrangements.

Simplifying DAGs: Following the above observations, we will present, in the next section, two algorithms aiming to simplify each leaf's DAG as follows:

1. Simplify the DAG based on the breakpoint distance. Although the resulting DAG $\mathcal{D}$ is not a total order, the developed algorithm can be seen as a heuristic for the Minimum-Breakpoint Linearization Problem, as any linear extension of $\mathcal{D}$ can be seen as a (possibly suboptimal) solution to this problem;

2. Simplify the resulting DAG based on the Kemeny distance. Similarly to the above step, the developed algorithm can be seen as a heuristic for the Minimum-Kemeny Linearization Problem.

4 Algorithms

Our two heuristics are inspired from the general methodology used by Ma *et. al* [11] for inferring ancestral gene orders, which in turn is inspired by the Fitch algorithm for substitution parsimony [4].

4.1 A heuristic for the Minimum-Kemeny Linearization Problem

Considering the assumption of no convergent mutation, the Fitch algorithm infers the DNA sequences at the internal nodes of a phylogenetic tree based on the DNA sequences at the leaves [4]. The sequences are treated site-by-site. Although nucleotide assignment is not unique, any assignment gives an evolutionary history with the minimum number of substitutions.

A similar idea has been considered in [11] for inferring ancestral gene orders on the basis of minimizing the number of breakpoints (or maximizing the number of adjacencies). The Ma *et. al* algorithm [11] proceeds in two steps. First, using a bottom-up traversal, it determines the potential adjacencies of each individual gene. This step results in a graph at each internal node, potentially with cycles. Then, in a top-down traversal, the information obtained on a node's parent is used to simplify the node's graph. The whole algorithm is guaranteed to identify a most-parsimonious scenario for the history of each individual adjacency. However, in contrast to the case of DNA sequences for which individual nucleotides are independent, adjacencies are not, and thus the whole-genome prediction is not guaranteed to minimize the number of breakpoints.

As DAGs provide information on relative orders between markers, rather than immediate adjacencies, we aim at inferring the ancestral order ($A < B$ or $B < A$) for each pair of markers (A, B). The *Kemeny-Simplification* algorithm described in Figure 2 is guaranteed to identify a most-parsimonious scenario for the history of each individual pair of markers. However, as pairs of markers are not independent, this does not guarantee the optimality of whole-map predictions.

Algorithm Kemeny-Simplification (T)
1. In a bottom-up traversal of T,
For each internal node v of T *do*
  For each pair (A, B) of markers of $\mathcal{M}_v^2$ *do*
    If $A < B$ (resp. $A > B$) in both children of v *then*
      Set $A < B$ (resp. $A > B$) in v;
    Else If $A > B$ in one child and $A < B$ in the other, *then*
      Set (A, B) incomparable in v;
    Else If $A < B$ (resp. $A > B$) in one child and incomparable in the other, *then*
      Set $A < B$ (resp. $A > B$) in v;
    Else If (A, B) are incomparable in both children of v *then*
      Set (A, B) incomparable in v;

2. In a top-down traversal of T,
For each node v of T that is not the root *do*
  For each pair (A, B) of markers of $\mathcal{M}_v^2$ *do*
    If (A, B) are incomparable in v but ordered $A < B$ (resp. $A > B$)
    in v's parent *then*
      Set $A < B$ (resp. $A > B$) in v;

Fig. 2. For each node v, $\mathcal{M}_v$ is the marker content at v.

During the second step of the algorithm (top-down traversal), conflicts may be created. For example, let A, B, C be three markers such that $A > B$ and (A, C) and (B, C) are incomparable. Resolving this two pairs by $B > C$ and $C > A$ results in transforming the comparable pair (A, B) into a conflicting pair. This may lead to a loss of order information at the leaves of T. To avoid this problem, we weight each order between two markers based on the number of times it appears in all species. Then for each leaf v, orders between pairs of markers in the parent of v are sorted according to their weight, and added successively in the DAG of v if they do not create a conflict.

4.2 A heuristic for the Minimum-Breakpoint Linearization Problem

As the Kemeny distance is not guaranteed to provide a good evaluation of the evolutionary distance in the case of large inversions, before applying the *Kemeny-simplification* algorithm on T, we first simplify DAGs by using a heuristic for the Minimum-Breakpoint Linearization Problem. This heuristic is based on the third step of the Ma *et al.* [11] algorithm aiming to recover a "partially linearized" gene order at a particular node of the tree. This step proceeds by first weighting each edge by an estimate of the likelihood of its presence in the ancestor, and then choosing adjacency paths of maximum weight.

Based on this idea, we develop the *Breakpoint-Simplification* algorithm that proceeds as follows:

For each leaf v of T,

1. Convert v's DAG into an *extended DG* $\mathcal{G}$ (possibly containing cycles) as follows: (1) expand each vertex corresponding to an SCC into the set of vertices of this SCC; (2) add an edge between each vertex connected to an SCC and each vertex of this SCC; (3) add an edge between each pair of markers that are potentially adjacent in a linear extension of the DAG (i.e. between all incomparable markers). For example, in DAG12 of Figure 1, the SCC $\mathcal{S} = \{D, E, F, G, L\}$ is replaced by five vertices labeled D, E, F, G, L; each pair of vertices (X, Y) belonging to $\{B, C\} \times \mathcal{S}$, $\mathcal{S} \times \{I, H\}$ and $\mathcal{S}^2$ is connected by an edge.

2. Weight each edge (A, B) of $\mathcal{G}$ by an estimate $w(A, B)$ of the probability of having B following A in the species f. This estimate is computed as follows:

$$w(A, B) = \frac{\sum_{i=i_1}^{i_k} \frac{1}{ADJ(A,i)}}{n}$$

where $i_1, \ldots, i_k$ represent the k leaves of T (including v) containing (A, B) as an edge in their corresponding extended DG, and $ADJ(A, i)$, for $i_1 \leq i \leq i_k$, is the number of edges adjacent to A in the extended DG of i. Recall that n is to the number of leaves (species) of T.

3. Construct a set of paths of maximum weight that cover all nodes of $\mathcal{G}$. This problem is known to be NP-hard [5] and we propose a simple greedy heuristic to resolve it. Our heuristic proceeds by sorting all the edges of $\mathcal{G}$ by weight, and then adding them in order to a new graph, initially restricted to the set of vertices of $\mathcal{G}$ and no edges, until each vertex has a unique predecessor and successor.

4. Incorporate the obtained set of adjacency paths into the original v's DAG. This is done by applying the heuristic that we have developed in [1] for simplifying a DAG with respect to a given total order. In our case, the heuristic is applied successively to the total order represented by each adjacency path.

4.3 The general method

In summary, our methodology can be subdivided into three main steps:

- **Step 1:** Perform map integration at each leaf of T;
- **Step 2:** Apply the Breakpoint-Simplification algorithm on T;
- **Step 3:** Apply the Kemeny-Simplification algorithm on T.

In the following section, we will analyse the efficiency of each step of the general method.

5 Experiments on the Gramene database

Gramene [8] is an important comparative genomics mapping database for crop grasses. It uses the completely sequenced rice genome to organize information on maize, sorghum, wheat, barley, and other gramineae (see Figure 3 for a phylogenetic tree of the species present in Gramene, excluding rice). It provides curated information on genetic and genomic datasets related to maps, markers, genes, genomes and quantitative trait loci, as well as invaluable tools for map comparison.

Correlating information from one map to another and from one species to another requires to have common markers, i.e. markers that are highly polymorphic among several populations. Such markers, also called "anchor markers" are typically SSRs (Simple Sequence Repeats, or microsatellites) or RFLPs (Restriction Fragment Length Polymorphism). In our study, we selected exclusively RFLP markers, as they appeared to be the most shared among all crop species present in Gramene, and thus those most likely to gain additional order information following a phylogenetic analysis. Moreover, they represent the largest family of DNA markers present in Gramene (17,715 different markers).

In order to consider only non-duplicated markers, we select, in each species, those appearing at a single locus. Moreover, as only markers shared between species may gain additional order information from a phylogenetic study, we further restrict ourselves, for each species s, to the set of "valid markers" defined

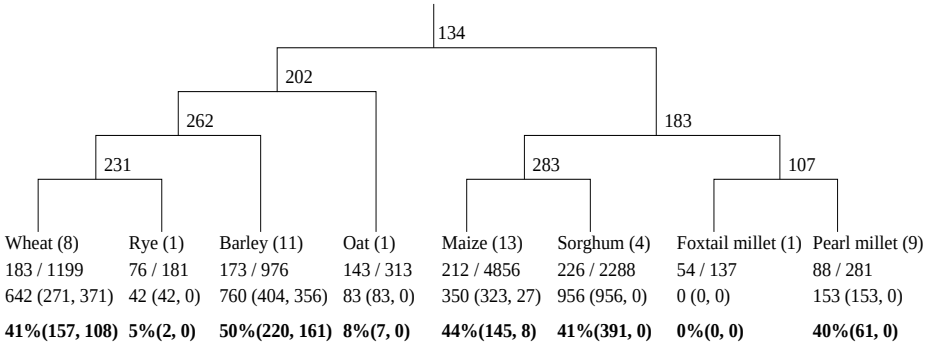

Fig. 3. Species included in the Gramene database (excluding the rice genome), with the phylogeny provided by [9]. Each internal node is labeled by the cardinality of its marker set. Labels of each leaf are defined from line 1 to line 4 as follows: (1) the species name followed, in brackets, by the number of map sets used in our study (each map set involves one map for each chromosome); (2) the number of valid RFLPs followed by the total number of RFLPs; (3) the total number of incomparable and conflicting pairs of markers in the union DG resulting from map integration (in brackets, the number of incomparable pairs, followed by the number of conflicting pairs); (4) the percentage of resolved incomparable and conflicting pairs of markers (in brackets, the number of resolved incomparable pairs , followed by the number of resolved conflicting pairs).

as follows: a *valid marker* in s is a non-duplicated marker in s that appears as a non-duplicated marker in at least one other species.

Figure 3 gives the distribution of total and valid RFLPs among species, and also the number of incomparable and conflicting (markers involved in a cycle) pairs of markers in the union DG obtained after the first step of of map integration. The total set of incomparable and conflicting pairs are those we hope to resolve following a phylogenetic analysis.

Results of applying our methodology (Section 4.3) to the Gramene database are given in the last line of leaf labels in Figure 3. The percentage of resolved incomparable and conflicting pairs of markers is given, followed in brackets by the actual number of resolved pairs. Overall, for species with a number of map set greater than one, the resolution rate ranges from 40% to 50%.

Results evaluation: To test the efficiency of our methodology, we perform the following experiments. We randomly choose 50 segments of two or three adjacent genes, each from a randomly chosen genetic map; the markers of each segment are made incomparable. We then apply our methodology, and check the percentage of incomparable pairs correctly resolved after each step (Section 4.3). This process is repeated 500 times.

Results are presented in Table 1. Performing the union of individual maps allows the integration, in a single map, of the complementary information interspersed in these maps. As conflicts between individual maps are usually due to mapping errors rather than to real rearrangement events that would have

	Segment size 2				Segment size 3			
	% Resolution		% Errors		% Resolution		% Errors	
INTEG	36.7		2.3		37.4		2.6	
INTEG+KEM	51.8	(15.1)	12.1	(36.0)	52.9	(15.6)	11.7	(34.0)
INTEG+BP	48.5	(11.8)	9.0	(30.0)	50.1	(12.7)	8.7	(26.7)
INTEG+BP+KEM	54.4	(17.7)	11.5	(30.6)	54.9	(17.5)	10.6	(27.5)

Table 1. "Segment size 2" (resp. "Segment size 3"): simulations done with segments of two markers (resp. three markers); % Resolution: percentage of introduced artificial incomparable pairs of markers that are resolved by the considered method; % Errors: percentage of errors (incomparable pairs incorrectly ordered) among the number of resolved artificial incomparable pairs; Results are presented for the following application of the general methodology steps (Section 4.3). INTEG: Step 1; INTEG+KEM: Step 1 followed directly by Step 3; INTEG+BP: Step 1 followed by Step 2; INTEG+BP+KEM: Final results (after applying Step 1, Step 2 and Step 3). Numbers in brackets are the percentage of resolution and error, for incomparable pairs remaining after INTEG.

affected one particular population, they are expected to be rare. This observation is confirmed by our results. Indeed, the step of integrating maps (INTEG in Table 1) allows to resolve a large proportion of incomparable pairs, with high resolution power ($\sim 2\%$ errors).

Following this step, the Kemeny-Simplification algorithm (KEM) has a higher resolution rate than the Breakpoint-Simplification algorithm (BP), but with a lower level of efficiency ($\sim 12\%$ errors for KEM, versus 9% for BP). Applying the complete methodology (BP followed by KEM) leads to a good compromise. However, it should be noted that less confidence should be given to incomparable pairs resolved from the phylogenetic information in comparison to those resolved from combining individual maps of a given species. This is indicated by the percentage of error ($\sim 30\%$) for incomparable pairs remaining after step INTEG (number in brackets in Table 1).

6 Conclusion

This paper is a first effort towards accounting for the phylogenetic information of a species tree to increase the resolution of genetic maps. The main assumption is that individual maps of one species may gain additional order information by considering the complementary information obtained from closely related species. In the case of species that are close enough to preserve a high degree of gene order conservation, minimizing the Kemeny distance on the species tree is an appropriate way of increasing the resolution of individual maps. However, the Kemeny distance is not appropriate anymore for species that have diverged from each other by large rearrangement events. In this case, using a genomic rearrangement measure (e.g. inversions or breakpoints) is more appropriate. Based

on this idea, we have designed a two-step methodology: resolve a number of incomparable markers by considering a rearrangement distance (namely the breakpoint distance), and then increase the resolution rate by considering the Kemeny distance.

Another more accurate heuristic for the Minimum-Breakpoint Linearization Problem may be designed by using a Median Branch-and-Bound approach, similar to the one developed for inferring ancestral gene orders of a species tree [15]. The general idea would be to begin with an arbitrary order at each internal node of the species tree, and then, in a bottom-up traversal, consider each triplet, and improve the order of the median by minimizing the breakpoint or inversion distance. However, as leaves are labeled by partial orders, instead of a linear-time algorithm for computing the breakpoint distance between two orders, an exponential-time algorithm, as the one that we have developed in [1], would be needed for computing a distance between a partial and a total order. The resulting complete heuristic is therefore likely to be intractable for reasonably large datasets. Moreover, as the number of possible solutions is likely to be huge, evaluating the obtained resolutions may be much more difficult.

Results obtained on the Gramene database are encouraging, as a high level of resolution is reached. However, our preliminary simulations performed to evaluate the method reveal a lack of specificity. These simulations may be improved, for example by removing an individual map and checking whether the order information it contains can be recovered by our methodology. Additional work should also be done to improve the various steps of the methodology, and better adapt it to the gramineae species.

References

1. G. Blin, E. Blais, D. Hermelin, P. Guillon, M. Blanchette, and N. El-Mabrouk. Gene maps linearization using genomic rearrangement distances. *Journal of Computational Biology*, 14(4):394–407, 2007.
2. A. Caprara. The reversal median problem. *Journal on Computing*, 15(1):93–113, 2003.
3. B.C.Y. Collard, M.Z.Z. Jahufer, J.B. Brouwer, and E.C.K. Pang. An introduction to markers, quantitative trait loci (QTL) mapping and marker-assisted selection for crop improvement: The basic concepts. *Euphytica*, 142:169–196, 2005.
4. W.M. Fitch. Toward defining the course of evolution: Minimum change for a specific tree topology. *Systematic Zoology*, 20(406– 416), 1971.
5. M.R. Garey and D.S. Johnson. In *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, CA, 1979.
6. B.N. Jackson, S. Aluru, and P.S. Schnable. Consensus genetic maps: a graph theoretic approach. In *IEEE Computational Systems Bioinformatics Conference (CSB'05)*, pages 35–43, 2005.
7. B.N. Jackson, P.S. Schnable, and S. Aluru. Consensus genetic maps as median orders from inconsistent sources. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 5(2):161–171, 2008.
8. P. Jaiswal and *others*. Gramene: a bird's eye view of cereal genomes. *Nucleic Acids Research*, 34:D717– D723, 2006.

9. E.A. Kellogg. Relationships of cereal crops and other grasses. *Proceedings of the National Academy of Sciences of USA*, 95(5):2005–2010, 1998.
10. J.P. Kemeny. Mathematics without numbers. *Daedelus*, 88:577–591, 1959.
11. J. Ma, L. Zhang, B.B. Suh, B.J. Raney, R.C. Burhans, W.J. Kent, M. Blanchette, D. Haussler, and W. Miller. Reconstructing contiguous regions of an ancestral genome. *Genome Research*, 16(12):1557–1565, 2006.
12. E. Nuutila and E. Soisalon-Soininen. On finding the strongly connected components in a directed graph. *Information Processing Letters*, 49:9–14, 1993.
13. I. Pe'er and R. Shamir. The median problems for breakpoints are NP-complete. *Electronic Colloquium on Computational Complexity (ECCC),* Report 71, 1998.
14. D. Saari and V. Merlin. A geometric examination of Kemeny's rule. *Social Choice and Welfare*, 7:81–90, 2000.
15. D. Sankoff and M. Blanchette. Multiple genome rearrangement and breakpoint phylogeny. *Journal of Computational Biology*, 5:555–570, 1998.
16. R. E. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal of Computing*, 1(2):146–160, 1972.
17. Y. Wakabayashi. The complexity of computing medians of relations. *Resenhas*, 3:323–349, 1998.
18. I.V. Yap, D. Schneider, J. Kleinberg, D. Matthews, S. Cartinhour, and S. R. McCouch. A graph-theoretic approach to comparing and integrating genetic, physical and sequence-based maps. *Genetics*, 165:2235–2247, 2003.
19. C. Zheng, Aleksander Lenert, and D. Sankoff. Reversal distance for partially ordered genomes. *Bioinformatics*, 21(Supp. 1):i502–i508, 2005.
20. C. Zheng, Q. Zhu, and D. Sankoff. Removing noise and ambiguities from comparative maps in rearrangement analysis. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4(4):515–522, 2007.