HAL
open science

An Approximation Algorithm for l∞-Fitting Robinson Structures to Distances

Victor Chepoi, Morgan Seston

To cite this version:

Victor Chepoi, Morgan Seston. An Approximation Algorithm for l∞-Fitting Robinson Structures to Distances. 26th International Symposium on Theoretical Aspects of Computer Science STACS 2009, Feb 2009, Freiburg, Germany. pp.265-276. inria-00359296

HAL Id: inria-00359296

https://inria.hal.science/inria-00359296v1

Submitted on 6 Feb 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Symposium on Theoretical Aspects of Computer Science 2009 (Freiburg), pp. 265–276
www.stacs-conf.org

AN APPROXIMATION ALGORITHM FOR l_∞-FITTING ROBINSON STRUCTURES TO DISTANCES

V. CHEPOI [1] AND M. SESTON [1]

[1] LIF, Université d'Aix-Marseille, Marseille Cedex 9, France
E-mail address: chepoi,seston@lif.univ-mrs.fr

Abstract. In this paper, we present a factor 16 approximation algorithm for the following NP-hard distance fitting problem: given a finite set X and a distance d on X, find a Robinsonian distance d_R on X minimizing the l_∞-error $||d - d_R||_\infty = \max_{x,y\in X}\{|d(x,y) - d_R(x,y)|\}$. A distance d_R on a finite set X is Robinsonian if its matrix can be symmetrically permuted so that its elements do not decrease when moving away from the main diagonal along any row or column. Robinsonian distances generalize ultrametrics, line distances and occur in the seriation problems and in classification.

1. Introduction

1.1. Seriation problem. Many applied algorithmic problems involve ordering of a set of objects so that closely coupled objects are placed near each other. These problems occur in such diverse applications as data analysis, archeological dating, numerical ecology, matrix visualization methods, DNA sequencing, overlapping clustering, graph linear arrangement, and sparse matrix envelope reduction. For example, a major issue in classification and data analysis is to visualize simple geometrical and relational structures between objects. Necessary for such an analysis is a dissimilarity on a set of objects, which is measured directly or computed from a data matrix. The classical *seriation problem* [16, 18] consists in finding of a simultaneous permutation of the rows and the columns of the dissimilarity matrix with the objective of revealing an underlying one-dimensional structure. The basic idea is that small values should be concentrated around the main diagonal as closely as possible, whereas large values should fall as far from it as possible. This goal is best achieved by considering the so-called *Robinson property* [20]: a dissimilarity matrix has this property if its values do not decrease when moving away from the main diagonal along any row or column. Experimental data usually contain errors, whence the dissimilarity can be measured only approximatively. As a consequence, any simultaneous permutation of the rows and the columns of the dissimilarity matrix gives a matrix which fails to satisfy the Robinson property, and we are led to the problem of finding a matrix reordering which is as close as

1998 ACM Subject Classification: Primary 68W25; Secondary 62H30 and 62-07.
Key words and phrases: Robinsonian dissimilarity, approximation algorithm, fitting problem.
We are grateful to Bernard Fichet for numerous insightful discussions during the work on this paper.
The authors were partly supported by the ANR grant BLAN06-1-138894 (projet OPTICOMB).

SYMPOSIUM ON THEORETICAL ASPECTS OF COMPUTER SCIENCE

© V. Chepoi and M. Seston
Creative Commons Attribution-NoDerivs License

possible to a Robinson matrix. As an error measure one can use the l_p-distance between two matrices. Several heuristics for seriation using Robinson matrices have been considered in the literature (the package seriation [14] contains their implementation). However, these methods either have exponential complexity or do not provide any optimality guarantee of the obtained solutions. In this paper, we provide a factor 16 algorithm for the NP-hard problem of optimally fitting a dissimilarity matrix by a Robinson matrix under the l_∞-error.

1.2. Definitions and the problem. Let X be a set of n elements to sequence, endowed with a *dissimilarity function* $d : X^2 \to \mathbb{R}^+ \cup \{0\}$ (i.e., $d(x, y) = d(y, x) \geq 0$ and $d(x, x) = 0$). A dissimilarity d and a total order $\prec$ on X are *compatible* if $d(x, y) \geq d(u, v)$ for any four elements such that $x \prec u \prec v \prec y$. Then d is *Robinsonian* if it admits a compatible order. Basic examples of Robinson dissimilarities are the *ultrametrics* and the standard *line-distance* between n points on the line. Denote by $\mathcal{D}$ and $\mathcal{R}$ the sets of all dissimilarities and of all Robinson dissimilarities on X. For $d, d' \in \mathcal{D}$, define the l_∞-error by $||d - d'||_\infty = \max_{x,y\in X}\{|d(x, y) - d'(x, y)|\}$. To formulate the corresponding fitting problem, we relax the notions of compatible order and Robinson dissimilarity. Given $\epsilon \geq 0$, a total order $\prec$ on X is called *ϵ-compatible* if $x \prec u \prec v \prec y$ implies $d(x, y) + 2\epsilon \geq d(u, v)$. An *$\epsilon$-Robinsonian dissimilarity* is a dissimilarity admitting an ϵ-compatible order, i.e., for each pair $x, y \in X$ one can pick a value $d_R(x, y) \in [d(x, y) - \epsilon, d(x, y) + \epsilon]$ so that the resulting dissimilarity d_R is Robinsonian. In this paper, we study the following NP-hard [8] optimization problem:

Problem l_∞-FITTING-BY-ROBINSON: *Given $d \in \mathcal{D}$, find a Robinson dissimilarity $d_R \in \mathcal{R}$ minimizing the l_∞-error $||d - d_R||_\infty$, i.e., find a least ϵ such that d is ϵ-Robinsonian.*

1.3. Related work. Fitting general distances by simpler distances (alias low-distortion embeddings) is a classical problem in mathematics, data analysis, phylogeny, and, more recently, in computer science. We review here only the results about l_∞-fitting of distances (this error measure is also known as the *maximum additive distortion* or the *maximum additive two-sided error* [5]). Farach et al. [13] showed that l_∞-fitting of a distance d by an ultrametric is polynomial. This result has been used by Agarwala et al. [1] to design a factor 3 approximation algorithm for l_∞-fitting of distances by tree-distances, a problem which has been shown to be strongly NP-hard [1]. A unified and simplified treatment of these results of [1, 13] using sub-dominants was given in [7]. A factor 2 approximation algorithm for the NP-hard problem of l_∞-fitting of a dissimilarity by a line-distance was given by Hstad et al. [15]. Bădoiu [4] proposed a constant-factor algorithm for l_∞-fitting of distances by l_1-distances in the plane.

Seriation is important in archeological dating, clustering hypertext orderings, numerical ecology, sparse matrix ordering, matrix visualization methods, and DNA sequencing [3, 6, 16, 18, 19, 20]. A package seriation implementing various seriation methods is described in [14]. The most common methods for clustering provide a visual display of data in the form of dendrograms. Dissimilarities in perfect agreement with dendrograms (i.e., ultrametrics) are Robinsonian. Generalizing this correspondence, [11, 12] establish that the Robinson dissimilarities can be visualized by hierarchical structures called pyramids.

1.4. Our result and techniques. The main result of the paper is a factor 16 approximation algorithm for the problem l_∞-FITTING-BY-ROBINSON. The basic setting of our algorithm goes as follows. First we show that the optimal error ϵ^* belongs to a well-defined list Δ of size $O(n^4)$. As in some other minmax problems, our approximation algorithm tests the entries of Δ, using a parameter ϵ, which is the "guess" for ϵ^*. For current $\epsilon \in \Delta$, the

algorithm either finds that no ϵ-compatible order exist, in which case the input dissimilarity d is not ϵ-Robinsonian, or it returns a 16ϵ-compatible order. Now, if ϵ is the least value for which the algorithm does not return the negative answer, then $\epsilon^* \geq \epsilon$, and the returned 16ϵ-Robinsonian dissimilarity has l_∞-error at most $16\epsilon^*$, establishing that we have a factor 16 approximation algorithm.

For $\epsilon \in \Delta$, a canonical binary relation $\preccurlyeq$ is computed so that any ϵ-compatible total order refines $\preccurlyeq$ or its dual. If $\preccurlyeq$ is not a partial order, then the algorithm halts and returns the negative answer. If $\preccurlyeq$ is a total order, then we are done. Otherwise, we select a maximal chain $P = (a_1, a_2, \ldots, a_p)$ of the partial order $\preccurlyeq$ and search to fit each element of $X^\circ := X \setminus P$ between two consecutive elements of P. We say that $a_i, a_{i+1} \in P$ form a *hole* H_i and that all elements $x \in X^\circ$ assigned between a_i and a_{i+1} are *located* in H_i. This distribution of the elements to holes is performed so that (a) all elements X_i of X° located in the same hole H_i must "fit" in this hole, i.e., for all $x, y \in X_i$ one of the orders $a_i \prec x \prec y \prec a_{i+1}$ or $a_i \prec y \prec x \prec a_{i+1}$ must be $c\epsilon$-compatible for some $c \leq 12$. Partitioning X° into sets X_i, $i = 1, \ldots, p-1$, is not obvious. Even if such a partition is available, we cannot directly apply a recursive call to each X_i, because (b) the elements located outside the hole H_i will impose a certain order on the elements of X_i and, since we tolerate some errors, (c) we cannot ensure that X_i is exactly the set of elements which must be located in H_i in some ϵ-compatible total order. To deal with (a), we give a classification of admissible and pairwise admissible holes for elements of X°. This allows to show that, if we tolerate a 12ϵ-error, then each element $x \in X^\circ$ can be located in the leftmost or rightmost admissible hole for x (we call them *bounding holes* of x). Both locations are feasible unless several elements have the same pair of bounding holes. For $i < j$, let X_{ij} be the set of all elements of X° having H_i and H_{j-1} as bounding holes. To deal with (b) and (c), on each set X_{ij} we define a directed graph $\mathcal{L}^{\rightarrow}_{ij}$. The strongly connected components (which we call *cells*) of $\mathcal{L}^{\rightarrow}_{ij}$ have the property that in any ϵ-compatible order all elements of the same component must be located in the same hole. In fact the cells (and not the sets X_i) are the units to which we apply the recursive calls in the algorithm. To decide in which hole H_i or H_{j-1} to locate each cell of $\mathcal{L}^{\rightarrow}_{ij}$ and to define the relative order between the cells assigned to the same hole, we define another directed graph $\mathcal{G}_{ij}$ whose vertices are the cells of $\mathcal{L}^{\rightarrow}_{ij}$ in such a way that (i) if some $\mathcal{G}_{ij}$ does not admit a partition into two acyclic subgraphs then no ϵ-compatible order exist and (ii) if $\mathcal{G}_{ij}$ has a partition into two acyclic subgraphs $\mathcal{G}^-_{ij}$ and $\mathcal{G}^+_{ij}$, then all cells of $\mathcal{G}^-_{ij}$ will be located in H_i, all cells of $\mathcal{G}^+_{ij}$ will be located in H_{j-1}, and the topological ordering of each of these graphs defines the relative order between the cells. To partition $\mathcal{G}_{ij}$ into two acyclic subgraphs (this problem in general is NP-complete [17]), we investigate the specific properties of graphs in question, allowing us to define a 2-SAT formula Φ_{ij} which is satisfiable if and only if the required bipartition of $\mathcal{G}_{ij}$ exists. Finally, to locate in each hole H_i the cells coming from different subgraphs $\mathcal{G}^+_{j'i}, \mathcal{G}^-_{ij}$, and $\mathcal{G}^-_{ij''}$ with $j' < i < j < j''$, we use the following separation rule: the cells of $\mathcal{G}^+_{j'i}$ are located to the left of the cells of $\mathcal{G}^-_{ij}$ and the cells of $\mathcal{G}^-_{ij}$ are located to the right of the cells of $\mathcal{G}^-_{ij''}$. Due to space constraints, all missing proofs are given in the full version [9].

2. Preliminary results

The $\prec$-restricted problem is obtained from l_∞-FITTING-BY-ROBINSON by fixing the total order $\prec$ on X. Let $\check{d}_\prec$ be a dissimilarity defined by setting $\check{d}_\prec(x, y) = \max\{d(u, v) : x \prec u \prec v \prec y\}$ for all $x, y \in X$ with $x \prec y$ (we suppose here that $a \prec a$ for any $a \in X$).

Let $2\tilde{\epsilon}_{\prec} = ||d - \check{d}_{\prec}||_{\infty}$ and let $\tilde{d}_{\prec}$ be the (Robinsonian) dissimilarity obtained from $\check{d}_{\prec}$ by setting $\tilde{d}_{\prec}(x,y) = \max\{\check{d}_{\prec}(x,y) - \tilde{\epsilon}_{\prec}, 0\}$ for all $x, y \in X, x \neq y$. Then, the following holds:

Proposition 2.1. *For a total order $\prec$ on X and $d \in \mathcal{D}$, $\tilde{d}_{\prec}$ minimizes $||d - d'||_{\infty}$.*

Proposition 2.1 establishes that an optimal solution of the problem l_{∞}-FITTING-BY-ROBINSON can be selected among $n!$ Robinsonian dissimilarities of the form $\tilde{d}_{\prec}$. In the full version, we show that the natural heuristic similar to the factor 3 approximation algorithms of Håstad et al. [15] and Agarwala et al. [1] (which instead of $n!$ total orders considers only n orders) does not provide a constant-factor approximation algorithm for our problem. Proposition 2.1 also implies that the optimal error ϵ^* in l_{∞}-FITTING-BY-ROBINSON belongs to a well-defined list $\Delta = \{\frac{1}{2}|d(x,y) - d(x',y')| : x, y, x', y' \in X\}$ of size $O(n^4)$.

Given $d \in \mathcal{D}$ and $\epsilon \in \Delta$, we define a partial order $\preccurlyeq$ such that every ϵ-compatible total order $\prec$ refines either $\preccurlyeq$ or its dual. For this, we set $p \preccurlyeq q$ for two arbitrary elements $p, q \in X$, and close $\preccurlyeq$ using the properties of partial orders and the following observation: *if $d(x,y) > \max\{d(x,z), d(z,y)\} + 2\epsilon$, then in all ϵ-compatible with d orders z must be located between x and y.* In this case, if we know that two of the elements x, z, y are in relation $\preccurlyeq$ then we can extend this relation to the whole triplet. For example, if we know that $x \preccurlyeq z$, then we conclude that also $z \preccurlyeq y$ and $x \preccurlyeq z$. If the resulting $\preccurlyeq$ is not a partial order, then d does not admit an ϵ-compatible total order. So, further let $\preccurlyeq$ be a partial order. For two disjoint subsets A, B of X, set $A \preccurlyeq B$ if $a \preccurlyeq b$ for any $a \in A$ and $b \in B$. We write $x?y$ if neither $x \preccurlyeq y$ nor $y \preccurlyeq x$ hold. For two numbers α and β we will use the following notations (i) $\alpha \approx_c \beta$ if $|\alpha - \beta| \leq c\epsilon$, (ii) $\beta \gtrsim_c \alpha$ if $\beta \geq \alpha - c\epsilon$, and (iii) $\beta \gg_c \alpha$ if $\beta > \alpha + c\epsilon$. We continue with basic properties of the canonical partial order $\preccurlyeq$: *If $w \preccurlyeq \{v, z\}$, $v?z$, $u \preccurlyeq v$, $u?z$, and $w?u$, then:* (i) $d(v,w) \approx_2 d(z,w)$; (ii) $d(v,z) \lesssim_2 \min\{d(v,w), d(z,w)\}$*;* (iii) $d(w,z) \approx_4 \{d(u,v), d(u,z)\}$; (iv) $d(w,u) \lesssim_2 \min\{d(w,v), d(u,v)\}$.

3. Pairwise admissible holes

3.1. Admissible holes. Let $P = (a_1, a_2, \ldots, a_{p-1}, a_p)$ be a maximal chain of the partial order $\preccurlyeq$. For notational convenience, we assume that all elements of X° must be located between a_1 and a_p (a_1 and a_p can be artificially added); this way, every element of X° must be located in a hole. Let H_{ij} be the union of all holes comprised between a_i, a_j. For $x \in X^{\circ}$, denote by $H(x)$ the union of all holes H_i such that $x?a_i$ or $x?a_{i+1}$. If $H(x) = H_{ij}$, the holes H_i and H_{j-1} are called *bounding holes*; see Fig. 1 (note that $a_i = \max\{a_k \in P : a_k \preccurlyeq x\}$ and $a_j = \min\{a_k \in P : x \preccurlyeq a_k\}$ for $x \in X^{\circ}$). All other holes of $H(x)$ are called *inner holes.* Since $x \notin P$, $H(x)$ contains at least two holes. The hole H_k of $H(x)$ is *x-admissible,* if the total order on $P \cup \{x\}$ obtained from $\preccurlyeq$ by adding the relation $a_k \preccurlyeq x \preccurlyeq a_{k+1}$ is ϵ-compatible with d. It can be easily shown that the bounding holes of $H(x)$ must be x-admissible. Denote by d_x the mean value of $\min\{d(x, a_k) : i < k < j\}$ and $\max\{d(x, a_k) : i < k < j\}$. We call $\delta_k = d(a_k, a_{k+1})$ the *size* of the hole H_k. Then the following holds:

Lemma 3.1. *If an inner hole H_k of $H(x)$ is x-admissible, then $d_x \approx_1 \{d(x, a_k), d(x, a_{k+1})\}$ $\approx_2 \delta_k$. In particular, $\delta_k \approx_3 d_x$. More generally, for all $k, k' \in]i, j[$, we have $d_x \gtrsim_3 d(a_k, a_{k'})$.*

3.2. Pairwise admissible holes. A pair $\{H_k, H_{k'}\}$ of holes is called *(x, y, c)-admissible* if H_k is x-admissible, $H_{k'}$ is y-admissible, and the total order on $P \cup \{x, y\}$ obtained by adding to $\preccurlyeq$ the relations $a_k \preccurlyeq x \preccurlyeq a_{k+1}$ and $a_{k'} \preccurlyeq y \preccurlyeq a_{k'+1}$ is $c\epsilon$-compatible. Denote by $AH(x)$ the set of all x-admissible holes H_k so that for each $y \in X^{\circ}$, $y \neq x$, there exists an y-admissible hole $H_{k'}$ such that $\{H_k, H_{k'}\}$ is a $(x, y, 1)$-admissible pair. Further we can assume

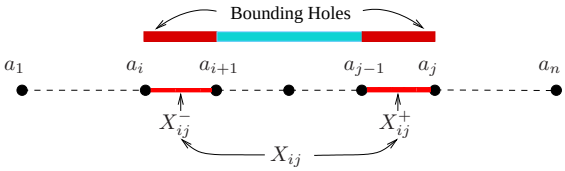

Figure 1: Bounding holes and the partition of X_{ij} into X_{ij}^- and X_{ij}^-

that for any $x \in X^\circ$ the bounding holes of $H(x) = H_{ij}$ belong to $AH(x)$. Otherwise, if say $H_i \notin AH(x)$, then $a_{i+1} \prec x$ in any ϵ-compatible total order $\prec$ extending $\preccurlyeq$, thus we can augment the canonical partial order $\preccurlyeq$ by setting $a_{i+1} \preccurlyeq x$ and by reducing the segments $H(x)$ accordingly. Next we investigate the pairwise admissible locations of x and y in function of the mutual geometric location of the segments $H(x)$ and $H(y)$ and of the values $d(x, y), d_x$, and d_y. We distinguish the following cases: **(H1)** $H(x) = H(y)$; **(H2)** $H(x)$ and $H(y)$ are disjoint; **(H3)** $H(x)$ and $H(y)$ overlap in at least 2 holes ($H(x) \circ H(y)$); **(H4)** $H(x)$ and $H(y)$ overlap in a single hole ($H(x) * H(y)$); **(H5)** $H(y)$ is a proper subinterval of $H(x)$ ($H(y) \Subset H(x)$). This classification of pairs $\{x, y\}$ of X° is used in the design of our approximation algorithm. Also the proofs of several results employ a case analysis based on (H1)-(H5). We continue with the following result. It specifies the constraints on pairs of elements, each element of X° can be located in one of its bounding holes.

Proposition 3.2. *For two elements $x, y \in X^\circ$, any location of x in a bounding hole of $H(x) = H_{ij}$ and any location of y in a bounding hole of $H(y) = H_{i'j'}$ is $(x, y, 12)$-admissible, unless $H(x) = H(y)$ and $d(x, y) \ll_3 \max\{d_x, d_y\}$ or $d(x, y) \gg_3 \max\{d_x, d_y\}$, subject to the following three constraints:* (i) *if $H(x) \Subset H(y)$, x and y are located in a common bounding hole, then x is between y and a_{i+1};* (ii) *if $H(x) * H(y)$, then $i < i'$ implies $x \prec y$;* (iii) *if $H(x) = H(y)$, x and y are located in the same bounding hole, and $d_y \ll_4 d_x$, then y is between x and a_{i+1}. If $H(x) = H(y)$ and $d(x, y) \gg_3 \max\{d_x, d_y\}$, then the only $(x, y, 1)$-admissible locations are the two locations of x and y in different bounding holes. If $H(x) = H(y)$ and $d(x, y) \ll_3 \max\{d_x, d_y\}$, then any $(x, y, 1)$-admissible location is in common x- and y-admissible holes.*

4. Distributing elements to holes

In this section, we describe how, for each hole H_i, to compute the set X_i of elements of X° which will be located in H_i. This set consists of some x such that H_i is a bounding hole of $H(x)$. Additionally, each X_i will be partitioned into an ordered list of cells, to which we perform recursive calls. Let X_{ij} consist of all $x \in X^\circ$ such that $H(x) = H_{ij}$. The sets X_{ij} form a partition of X°. In the next subsections, we will show how to partition each X_{ij} into two subsets X_{ij}^- and X_{ij}^+, so that X_{ij}^- will be located in H_i and X_{ij}^+ in H_{j-1}; see Fig. 1.

4.1. Blocks, cells, and clusters. Two elements $x, y \in X_{ij}$ are called *linked* (*separated*) if in all $(x, y, 1)$-admissible locations x and y must be placed in the same hole (in distinct bounding holes). Two subsets A and B of X_{ij} must be *separated* if all $x \in A$ and $y \in B$ are separated. Let S_{ij} and L_{ij} be the sets of all pairs $x, y \in X_{ij}$ such that $d(x, y) \gg_3 \max\{d_x, d_y\}$, resp., $d(x, y) \ll_3 \max\{d_x, d_y\}$. By Proposition 3.2, all pairs of S_{ij} are separated and all pairs of L_{ij} are linked. Since "be linked" is an equivalence relation, all vertices of the same connected component (called *block*) of the graph $\mathcal{L}_{ij} = (X_{ij}, L_{ij})$ are linked. We continue by investigating in which cases two blocks of $\mathcal{L}_{ij}$ are separated or linked. For $x, y \in X_{ij}$, set $x \rightarrowtail y$ iff **(A1)** $d_x \ll_4 d_y$ or **(A2)** $d_x \gtrsim_4 d_y$ and there exists $z \in X_{ij}$ such

that $xz, yz \notin L_{ij}$ and $d(x,z) \ll_{16} d(y,z)$. If $x,y,z \in X_{ij}$ satisfy **(A2)**, then it can be shown that y and z are *strongly separated*, i.e., $d(y,z) \gg_9 \max\{d_y, d_z\}$. Additionally, we show that if $x \rightarrowtail y$, then $x \prec y$ in all ϵ-compatible orders $\prec$ such that $a_{i+1} \prec \{x,y\}$ and $y \prec x$ in all ϵ-compatible orders $\prec$ such that $\{x,y\} \prec a_{j-1}$.

On X_{ij} we define a directed graph $\mathcal{L}_{ij}^{\rightarrow}$: we draw an arc $x \to y$ iff **(L1)** $x \rightarrowtail y$ and x,y belong to a common block of $\mathcal{L}_{ij}$ or **(L2)** $d(x,y) \ll_5 \max\{d_x, d_y\}$. If **(L2)** is satisfied, then $xy \in L_{ij}$ and $y \to x$ hold. The strongly connected components of $\mathcal{L}_{ij}^{\rightarrow}$ are called *cells*. Every block is a disjoint union of cells. Indeed, if x,y belong to a common cell, let R be a directed path of $\mathcal{L}_{ij}^{\rightarrow}$ from x to y. Pick any arc $u \to v$ of R. If it has type **(L2)**, then $uv \in \mathcal{L}_{ij}$. Otherwise, if $u \to v$ has type **(L1)**, then u and v belong to a common block. Thus the ends of all arcs of any path between x,y belong to a common block.

Lemma 4.1. *Let $x,x',y \in X_{ij}$. If x,x' belong to a common cell, but $\{x,x'\}$ and y belong to distinct blocks, then there does not exist an ϵ-compatible order such that $x \prec y \prec x'$.*

Lemma 4.2. *For cells C',C'', if $x,x' \in C'$, $y,y' \in C''$, and $x \rightarrowtail y$, $y' \rightarrowtail x'$, then C' and C'' must be separated.*

Proof. Let B',B'' be the blocks containing C',C''. If $B' = B''$, as $x \rightarrowtail y$ and $y' \rightarrowtail x'$, they are (L1)-arcs, hence $x \to y$ and $y' \to x'$. This is impossible since $\{x,x'\}$ and $\{y,y'\}$ belong to distinct cells. Thus $B' \neq B''$. By Lemma 4.1, if we locate x,x',y,y' in the same bounding hole H_j, either $\{x,x'\} \prec \{y,y'\}$ or $\{y,y'\} \prec \{x,x'\}$ holds. On the other hand, $x \rightarrowtail y$, $y' \rightarrowtail x'$ imply that $x \prec y$ and $y' \prec x'$. Thus C' and C'' must be separated. ∎

Now, let $\mathcal{S}_{ij}$ be a graph having cells as vertices and an edge between two cells C',C'' iff **(S1)** there exist $x,y \in X_{ij}$, x in the same block as C' and y in the same block as C'' such that $xy \in S_{ij}$ or **(S2)** there exist x,x' in the same block as C' and y,y' in the same block as C'' such that each pair xx' and yy' belong to a common cell, and $x \rightarrowtail y, y' \rightarrowtail x'$. By Proposition 3.2 and Lemma 4.2, in cases **(S1)** and **(S2)** the sets C' and C'' must be separated. The graph $\mathcal{S}_{ij}$ must be bipartite, otherwise no ϵ-compatible order exist. Now, for each connected component of $\mathcal{S}_{ij}$ consider its canonical bipartition $\{A',A''\}$, and draw an edge between any two cells, one from A' and another from A''. Denote the obtained graph also by $\mathcal{S}_{ij}$. Call the union of cells from A' (or from A'') a *cluster*. The clusters $\mathcal{K}'$ and $\mathcal{K}''$ of A' and A'' are called *twins*. From the construction, we immediately obtain that all elements of a cluster are linked and two twin clusters are separated. A connected bipartite component $\{\mathcal{K}',\mathcal{K}''\}$ of $\mathcal{S}_{ij}$ is called a *principal component* if there exists $x \in \mathcal{K}'$ and $y \in \mathcal{K}''$ such that x and y are strongly separated.

4.2. Partitioning X_{ij} into X_{ij}^- and X_{ij}^+. We describe how to partition X_{ij} into the subsets X_{ij}^- and X_{ij}^+. For this, we define a directed graph $\mathcal{G}_{ij}$ having cells as vertices, and an arc $C' \rightarrowtail C$ with tail C' and head C exists iff one of the following conditions is satisfied: **(G1)** C' and C belong to twin clusters of $\mathcal{S}_{ij}$; **(G2)** C' and C are not connected by (G1)-arcs and there exist $x \in C$ and $x' \in C'$ such that $d_{x'} \ll_4 d_x$; **(G3)** C' and C are not connected by (G1)- or (G2)-arcs and there exist $x \in C, x' \in C'$, and $z \in X_{ij}$ such that $xz, x'z \notin L_{ij}$ and $d(x',z) \ll_{16} d(x,z)$. A head of a (G3)-arc is called a (G3)-*cell*. A (Gi)-*cycle* is a directed cycle of $\mathcal{G}_{ij}$ with arcs of type **(Gi)**, $i = 1,2,3$. The (G1)-cycles are exactly the cycles of length 2. A *mixed cycle* is a directed cycle containing arcs of types **(G2)** and **(G3)**. Finally, an *induced cycle* is a directed cycle $\mathcal{C}$ such that for two cells $C,C' \in \mathcal{C}$ we have $C' \rightarrowtail C$ if and only if C is the successor of C' in $\mathcal{C}$. Our next goal is to establish that either the set

of cells can be partitioned into two subsets such that the subgraphs of $\mathcal{G}_{ij}$ induced by these subsets do not contain directed cycles or no ϵ-compatible order exist. Deciding if a directed graph can be partitioned into two acyclic subgraphs is NP-complete [17]. In our case, this can be done in polynomial time by exploiting the structure of $\mathcal{G}_{ij}$.

Lemma 4.3. *If $\mathcal{C} = (C_1, C_2, \ldots, C_k, C_1)$ is a directed cycle of $\mathcal{G}_{ij}$, then for any ϵ-compatible order, $\mathcal{C}$ has a cell located in the hole H_i and a cell located in the hole H_{j-1}.*

Proof. The assertion is obvious if $\mathcal{C}$ is a (G1)-cycle. So, suppose that all arcs of $\mathcal{C}$ have type **(G2)** or **(G3)**. The definition of cells implies that $\mathcal{C}$ contains two consecutive cells, say C_1 and C_k, which belong to different blocks. Suppose that there exists an ϵ-compatible order $\prec$ such that no element of $\cup_{l=1}^k C_l$ is located in the hole $H_i = [a_i, a_{i+1}]$, i.e., $a_{i+1} \prec \cup_{l=1}^k C_l$. In each C_l pick two elements x_l, y_l such that $x_l \rightarrowtail y_{l+1(modk)}$. Then $x_l \prec y_{l+1(modk)}$ for all $l = 1, \ldots, k$. We divide the cells of $\mathcal{C}$ into groups: a group consists of all consecutive cells of $\mathcal{C}$ belonging to one and the same block. The first group starts with C_1, while the last group ends with C_k. We assert that if $\{C_{l-q}, \ldots, C_l\}$ and $\{C_{l+1}, \ldots, C_{l+r}\}$ are two consecutive groups of $\mathcal{C}$, then $C_l \prec C_{l+1} \cup \cdots \cup C_{l+r}$ (all indices here are modulo k). Indeed, pick $u \in C_l$ and $v \in C_{l+1}$. Since $\{x_l, u\}$ and $\{y_{l+1}, v\}$ belong to different blocks while each of these pairs belong to a common cell, applying Lemma 4.1 to each of the triplets of the quadruplet x_l, u, y_{l+1}, v, we infer that in the total order $\prec$ none of y_{l+1}, v is located between x_l and u and none of x_l, u is located between y_{l+1} and v. Since $x_l \prec y_{l+1}$, we conclude that $\{x_l, u\} \prec \{y_{l+1}, v\}$, yielding $C_l \prec C_{l+1}$. Now, consider the cell C_{l+2}. The element y_{l+2} must be located to the right of x_{l+1}, therefore to the right of C_l. Since C_{l+2} and C_l belong to different blocks, we can show that $C_l \prec C_{l+2}$ by using exactly the same reasoning as for the cells C_l and C_{l+1}. Continuing this way, we obtain the required relationship $C_l \prec C_{l+1} \cup \cdots \cup C_{l+r}$. This establishes the assertion. Suppose that $[1, i_1], [i_1 + 1, i_2], \ldots, [i_r + 1, k]$ are the indices of cells defining the beginning and the end of each group. From our assertion we infer that $C_k \prec C_{i_1} \prec C_{i_2} \prec \ldots \prec C_{i_r} \prec C_k$, contrary that $\prec$ is a total order. ∎

Lemma 4.4. *If $C \rightarrowtail C'$ is a (G3)-arc and C belongs to a principal component, then C and C' belong to the same cluster. In particular, $\mathcal{G}_{ij}$ does not contain (G3)-cycles or no ϵ-compatible order exist. Moreover, $\mathcal{G}_{ij}$ does not contain (G2)-cycles.*

Proof. Let xy be a strongly separated pair with $x \in C$. Since $C \rightarrowtail C'$ is a (G3)-arc, there exist $y' \in C$ and $x' \in C'$ such that $y' \rightarrowtail x'$ is an (A2)-arc. Then there exists z' such that $x'z'$ is strongly separated. If xz and $x'y'$ belong to different principal components, then there exists a (G2)-arc from C' to C or from C to C'. In the first case, C and C' obey **(S2)**, thus we cannot have a (G3)-arc from C to C'. Analogously, in the second case, we deduce that we have at the same time a (G3)-arc and a (G2)-arc from C to C'. This is impossible, so C and C' belong to a common principal component. Now, if $\mathcal{G}_{ij}$ contains a (G3)-cycle, then the first assertion implies that all its cells belong to the same cluster, and Lemma 4.3 yields that no ϵ-compatible order exist. Finally, let $\mathcal{C} = (C_1, C_2, \ldots, C_k, C_1)$ be a (G2)-cycle. In each C_i, pick x_i, y_i so that $d_{x_i} \ll_4 d_{y_{i+1(modk)}}$. Since there is no **(G2)** or **(G3)** arc from $C_{i+1(modk)}$ to C_i, we get $d_{y_i} \lesssim_4 d_{x_{i+1(modk)}}$, yielding $d_{x_i} \ll_4 d_{y_{i+1(modk)}} \lesssim_4 d_{x_{i+2(modk)}}$. Thus $d_{x_i} < d_{x_{i+2(modk)}}$ for $i = 1, \ldots k$. Then $d_{x_1} < d_{x_3} < \cdots < d_{x_{k-1}} < d_{x_1}$ for even k and $d_{x_1} < d_{x_3} < \cdots < d_{x_k} < d_{x_2} < d_{x_4} < \cdots < d_{x_{k-1}} < d_{x_1}$ for odd k, a contradiction. ∎

To complete the bipartition of cells into two acyclic subgraphs of $\mathcal{G}_{ij}$, it remains to deal with induced mixed cycles. The following results precise their structure.

Lemma 4.5. *Any induced mixed cycle $\mathcal{C}$ of $\mathcal{G}_{ij}$ contains one or two (G2)-arcs, and if $\mathcal{C}$ contains two such arcs, then they are consecutive.*

Lemma 4.6. *Let $C' \rightarrowtail C$ be a (G3)-arc, $C \rightarrowtail C''$ be a (G2)-arc, and suppose that there is no (G2)-arc from C' to C''. If C, C' do not belong to distinct twin clusters and C, C'' do not belong to the same cluster, then C and C' must be separated.*

Thus a mixed cycle $\mathcal{C}$ contains either one (G2)-arc ($\mathcal{C}$ is a 1-*cycle*) or two consecutive (G2)-arcs ($\mathcal{C}$ is a 2-*cycle*), all other arcs of $\mathcal{C}$ being (G3)-arcs. By Lemma 4.4, the heads of all (G3)-arcs of $\mathcal{C}$ are (G3)-cells of the same cluster $\mathcal{K}$. Then we say that the cycle $\mathcal{C}$ *intersects* the cluster $\mathcal{K}$. For a (G2)-arc $C_0 \rightarrowtail C$ and a cluster $\mathcal{K}$, we show how to detect if there exists a 1- or 2-cycle $\mathcal{C}$ passing via $C_0 \rightarrowtail C$ and intersecting $\mathcal{K}$. We consider the case of 1-cycles. Then C_0 must be a (G3)-cell of $\mathcal{K}$. Note that an induced 1-cycle cannot contain cells C' such that $C_0 \rightarrowtail C'$ is a **(G2)** or **(G3)**-arc. Hence, we can remove all such cells of $\mathcal{K}$. Analogously, we remove all cells C' so that $C' \rightarrowtail C$ is an arc. In the subgraph induced by the remaining cells of $\mathcal{K}$ we search for a shortest directed path $\mathcal{Q} = C \rightarrowtail C_1 \rightarrowtail \cdots \rightarrowtail C_k \rightarrowtail C_0$ so that the first arc $C \rightarrowtail C_1$ and the last arc $C_k \rightarrowtail C_0$ of this path are (G3)-arcs. This can be done in polynomial time by testing all possible choices for C_1 and C_k and applying for each pair a shortest path finding algorithm in an acyclic graph. If such a path $\mathcal{Q}$ does not exist, then no required induced cycle $\mathcal{C}$ exist. Otherwise, the path $\mathcal{Q}$ together with the arc $C_0 \rightarrowtail C$ define an induced cycle $\mathcal{C}$ having exactly one (G2)-arc. Indeed, if $C_i \rightarrowtail C_j$ is a **(G2)** or **(G3)**-arc and $|i-j| > 2$, since the subgraph induced by $\mathcal{K}$ is acyclic, we must have $i < j$. This contradicts the minimality of the path $\mathcal{Q}$. So, the resulting cycle is indeed induced. It remains to note that $\mathcal{C}$ does not contain other (G2)-arcs, because by Lemma 4.5 in an induced cycle the (G2)-arcs are consecutive. Analogously, we can decide if there exists a 2-cycle passing via $C_0 \rightarrowtail C$ and intersecting $\mathcal{K}$, and having a second (G2)-arc of the form $C \rightarrowtail C_0'$ or $C_0' \rightarrowtail C_0$. Therefore, we have the following result:

Lemma 4.7. *For a (G2)-arc $C_0 \rightarrowtail C$ and a cluster $\mathcal{K}$, one can decide in polynomial time if there exists an induced 1- or 2-cycle $\mathcal{C}$ passing via $C' \rightarrowtail C$ and intersecting $\mathcal{K}$.*

For a cell C, let $\Omega_1(C)$ be the set of (G2)-arcs $C_0 \rightarrowtail C$ belonging to a 1-cycle intersecting a cluster $\mathcal{K}$ not containing C. Let $\Omega_2(C)$ be the set of (G2)-arcs $C_0 \rightarrowtail C$ belonging to a 2-cycle $\mathcal{C}$ intersecting a cluster $\mathcal{K}$ not containing C and passing via $C_0 \rightarrowtail C$ so that the arc of $\mathcal{C}$ entering C_0 is a (G3)-arc. In both cases C_0 belongs to $\mathcal{K}$: C_0 is a head of a (G3)-arc of $\mathcal{C}$, and all such heads belong to $\mathcal{K}$. Finally, let $\Omega_3(C)$ be the set of (G2)-arcs $C \rightarrowtail C_0$ belonging to a 2-cycle $\mathcal{C}$ intersecting a cluster $\mathcal{K}$, so that C belongs to $\mathcal{K}$ and the arc of $\mathcal{C}$ entering C has type **(G2)**. Fig. 2 illustrates this classification. For each cell C of $\mathcal{G}_{ij}$ we introduce a binary variable x_C satisfying the following constraints: **(F1)** $x_{C'} = x_{C''}$, if C', C'' belongs to the same cluster; **(F2)** $x_{C'} \neq x_{C''}$, if C', C'' belong to twin clusters; **(F3)** $x_C \neq x_{C_0}$, if the arc $C_0 \rightarrowtail C$ belongs to $\Omega_1(C) \cup \Omega_2(C)$; **(F4)** $x_C \neq x_{C_0}$, if the arc $C \rightarrowtail C_0$ belongs to $\Omega_3(C)$. Define a 2-SAT formula Φ_{ij} by replacing every constraint $a = b$ by two clauses $(a \vee \bar{b})$ and $(\bar{a} \vee b)$ and every constraint $a \neq b$ by two clauses $(a \vee b)$ and $(\bar{a} \vee \bar{b})$.

Proposition 4.8. *If the 2-SAT formula Φ_{ij} admits a satisfying assignment A, then the sets $X_{ij}^- = \{C : A(x_C) = 0\}$ and $X_{ij}^+ = \{C : A(x_C) = 1\}$ define a partition of $\mathcal{G}_{ij}$ into two acyclic subgraphs. Conversely, given an ϵ-compatible order on X, the assignment A defined by setting $A(x_C) = 0$ if C is located in H_i, $A(x_C) = 1$ if C is located in H_{j-1}, and $A(x_{C'}) = A(x_{C''})$ if C' and C'' are located in a common inner hole, is a true assignment for Φ_{ij}. In particular, if Φ_{ij} is not satisfiable, then no ϵ-compatible order exist.*

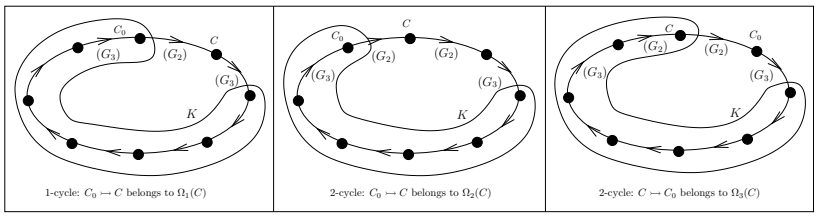

Figure 2: To the classification of the arcs incident to a cell C

Proof. Let A be a true assignment of Φ_{ij} and the partition X_{ij}^-, X_{ij}^+ of X_{ij} be defined as above. Denote by $\mathcal{G}_{ij}^-$ and $\mathcal{G}_{ij}^+$ the subgraphs induced by X_{ij}^- and X_{ij}^+. **(F1)** forces every cluster to be included in one set. **(F2)** implies that the twin clusters are separated. Hence $\mathcal{G}_{ij}^-$ and $\mathcal{G}_{ij}^+$ do not contain (G1)-cycles: if C and C' are the two cells of a (G1)-cycle, then $(x_C \vee x_{C'}) \wedge (\bar{x}_C \vee \bar{x}_{C'})$ yields $A(x_C) \neq A(x_{C'})$. By Lemma 4.4, $\mathcal{G}_{ij}$ does not contain (G2)-cycles. Since the cells of a (G3)-cycle are contained in the same cluster and each cluster induces an acyclic subgraph, $\mathcal{G}_{ij}^-$ and $\mathcal{G}_{ij}^+$ do not contain (G3)-cycles as well. Now, let $\mathcal{G}_{ij}^+$ contain a mixed cycle. Then it also contains an induced mixed cycle $\mathcal{C}$. From Lemma 4.5 we infer that $\mathcal{C}$ has either one (G2)-arc $C_0 \rightarrowtail C$ or exactly two consecutive (G2)-arcs $C_0 \rightarrowtail C \rightarrowtail C''$. In the first case, we conclude that $C_0 \rightarrowtail C$ belongs to $\Omega_1(C)$, thus **(F3)** yields $x_C \neq x_{C_0}$, contrary to the fact that $A(x_C) = A(x_{C'}) = 1$. Analogously, in the second case, we deduce that either $x_C \neq x_{C_0}$ and the arc $C_0 \rightarrowtail C$ belongs to $\Omega_2(C)$ or $x_C = x_{C_0}$ and the arc $C \rightarrowtail C''$ belongs to $\Omega_3(C)$, whence $x_C \neq x_{C''}$. Then we obtain a contradiction with the assumption that $A(x_{C_0}) = A(x_C) = A(x_{C''}) = 1$. This shows that the subgraphs $\mathcal{G}_{ij}^-$ and $\mathcal{G}_{ij}^+$ obtained from the true assignment A of Φ_{ij} are acyclic.

Conversely, let A be an assignment obtained from an ϵ-compatible order as defined in the proposition. We assert that A is a true assignment for Φ_{ij}, i.e., it satisfies the constraints **(F1)-(F4)**. This is obvious for constraints **(F1)** and **(F2)**, because if two cells C', C'' belong to the same cluster, then they will be located in the same hole and we must have $A(x_{C'}) = A(x_{C''})$. If C' and C'' belong to distinct twin clusters, then they must be separated, therefore the unique ϵ-admissible location of C' and C'' will be in different bounding holes, thus $A(x_{C'}) \neq A(x_{C''})$. Now, pick an arc $C_0 \rightarrowtail C$ which belongs to $\Omega_1(C) \cup \Omega_2(C)$. If $C_0 \rightarrowtail C$ belongs to $\Omega_1(C)$, then there exists a 1-cycle $\mathcal{C}$ passing via $C_0 \rightarrowtail C$ and intersecting a cluster $\mathcal{K}$. Since all cells of $\mathcal{C}$, except C, are heads of (G3)-arcs, they all belong to $\mathcal{K}$, i.e., they have the same value in the assignment. By Lemma 4.3, C must be separated from C_0 (namely C and C' must be located in different bounding holes), showing that $A(x_C) \neq A(x_{C_0})$. If $C_0 \rightarrowtail C$ belongs to $\Omega_2(C)$, then let $\mathcal{C}$ be a 2-cycle passing via $C_0 \rightarrowtail C$ and intersecting the cluster $\mathcal{K}$ not containing C. Additionally, we know that the arc $C' \rightarrowtail C_0$ of $\mathcal{C}$ entering C_0 is a (G3)-arc, thus C_0 belongs to $\mathcal{K}$. Since C' cannot belong to the twin cluster of $\mathcal{K}$ (this will contradicts that $C' \rightarrowtail C_0$ is a (G3)-arc) and since C does not belong to $\mathcal{K}$, from Lemma 4.6 we infer that C_0 and C are separated, thus $A(x_C) \neq A(x_{C_0})$. Finally, let $C \rightarrowtail C_0$ belong to $\Omega_3(C)$. Then there exists a 2-cycle $\mathcal{C}$ passing via $C \rightarrowtail C_0$ and intersecting the cluster $\mathcal{K}$, such that C belongs to $\mathcal{K}$ and the arc of $\mathcal{C}$ entering C has type **(G2)**. Since all cells of $\mathcal{C}$ except C and C_0 are heads of (G3)-arcs, they all belong to $\mathcal{K}$. Since C also belongs to this cluster, by Lemma 4.3, C_0 must be separated from the remaining cells of $\mathcal{C}$, yielding $x_C \neq x_{C_0}$. Hence A satisfies the constraints **(F1)-(F4)**. This shows, in particular, that if Φ_{ij} is not satisfiable, then no ϵ-compatible order exist. ∎

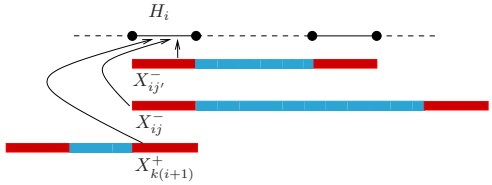

Figure 3: Relative location of the cells of $X^+_{k(i+1)}, X^-_{ij'}$, and X^-_{ij} $(k < i, j' < j)$ in H_i

4.3. Sorting the cells of X^-_{ij} and X^+_{ij}. Let $\mathcal{G}^-_{ij}$ and $\mathcal{G}^+_{ij}$ be the subgraphs of $\mathcal{G}_{ij}$ induced by the sets X^-_{ij} and X^+_{ij} obtained from the true assignment of the 2-SAT formula Φ_{ij}. We will locate all cells of X^-_{ij} in the hole H_i and all cells of X^+_{ij} in the hole H_{j-1} of H_{ij}. The elements from two cells C', C'' located in the same hole will not be mixed, i.e., C' will be placed to the right of C'', or vice versa. To specify the total order among cells, we use that $\mathcal{G}^-_{ij}$ and $\mathcal{G}^+_{ij}$ are acyclic, therefore each of them admit a topological order. We compute a topological order $C_{j_1} \prec C_{j_2} \prec \ldots \prec C_{j_p}$ on the cells of X^+_{ij} and a dual topological order $C_{i_q} \prec C_{i_{q-1}} \prec \ldots \prec C_{i_1}$ on the cells of X^-_{ij}. We locate the cells of X^+_{ij} in H_{j-1} and the cells of X^-_{ij} in H_i according to these orders. The following two results relay the topological orders on the cells with the order on the distances between elements from such cells.

Lemma 4.9. *Let C', C'' be two cells of X^+_{ij}. If $C' \prec C''$ in the topological order, then for any $y \in C'$, $z \in C''$ and $x \in X^-_{ij}$, we have $d_y \lesssim_4 d_z$ and $d(x,y) \lesssim_{16} d(x,z)$.*

Proof. Since C', C'' belong to X^+_{ij}, they are not connected by (G1)-arcs. Since $C' \prec C''$ in the topological order, there is no arc from C'' to C'. As $C'' \rightarrowtail C'$ is not a (G2)-arc, we must have $d_z \gtrsim_4 d_y$. As $C'' \rightarrowtail C'$ is not a (G3)-arc, we obtain $d(x,y) \lesssim_{16} d(x,z)$. ∎

Lemma 4.10. *Let C, C', C'' be three distinct cells of the graph $\mathcal{G}_{ij}$. If the algorithm returns the total order $\prec$ and $C \prec C' \prec C''$, then for any $x \in C, y \in C', z \in C''$ or $x, y, z \in C \cup C'$ and $x \prec y \prec z$, we have $d(x,z) \gtrsim_{16} \max\{d(x,y), d(y,z)\}$.*

After fixing the relative position of each cell C of X_{ij}, we make a recursive call to C. For this, we update the canonical order $\preccurlyeq$ in the following way: if C is located in X^+_{ij}, we set $x \preccurlyeq^+ y$ if $x \rightarrowtail y$, otherwise, if C is located in X^-_{ij}, we set $x \preccurlyeq^- y$ if $y \rightarrowtail x$. Since $\preccurlyeq^+$ and $\preccurlyeq^-$ are dual, if we apply to them the "closing" rules, we will obtain two dual partial orders, denoted also by $\preccurlyeq^+$ and $\preccurlyeq^-$. The restriction on C of every ϵ-compatible order $\prec$ on X is an extension of $\preccurlyeq^+$ or $\preccurlyeq^-$: since all elements of C will be placed in the same hole, either $a_{i+1} \prec C$ or $C \prec a_j$. If $a_{i+1} \prec C$, then $x \prec y$ for all $x, y \in C$ such that $x \rightarrowtail y$. Hence $\prec$ is a linear extension of $\preccurlyeq^+$. Therefore, if the recursive call to a cell C returns the answer "not", then no ϵ-compatible total order on X exist. Else, it returns a total order on C, which is 16ϵ-compatible by induction hypothesis. Then, the total order between the cells of $\mathcal{G}_{ij}$ and the total orders on cells are concatenated to give a single total order $\prec$ on X_{ij}.

4.4. Defining the total order on X_i. Recall that X_i is the set of all elements of X° located in the hole H_i. According to our algorithm, X_i is the disjoint union of all sets X^-_{ij} $(j > i+1)$ and $X^+_{k(i+1)}$ $(k < i)$. We just defined a total order between the cells of each of the sets $X^-_{ij}, X^+_{k(i+1)}$, and applying recursion we defined a total order on the elements of each cell. To obtain a total order on the whole set X_i it remains to define a total order between the sets X^-_{ij} $(j > i+1)$ and $X^+_{k(i+1)}$ $(k < i)$. For this, we locate each $X^+_{k(i+1)}$ $(k < i)$ to the

left of each X_{ij}^- $(j > i)$. Given two sets $X_{k(i+1)}^+, X_{k'(i+1)}^+$ $(k, k' < i)$, we locate $X_{k(i+1)}^+$ to the left of $X_{k'(i+1)}^+$ if and only if $k < k'$, i.e., iff $H_{k(i+1)} \Subset H_{k'(i+1)}$. Analogously, given $X_{ij}^-, X_{ij'}^-$ $(j, j' > i+1)$, we locate $X_{ij'}^-$ to the right of X_{ij}^- if and only if $j' < j$, i.e., iff $H_{ij'} \Subset H_{ij}$. This location is justified by the Proposition 3.2 and is illustrated in Fig. 3.

5. The algorithm and its performance guarantee

We have collected all necessary tools to describe the algorithm. It consists of three procedures l_∞-Fitting_by_Robinson, Refine, and Partition_and_Sort. The main procedure l_∞-Fitting_by_Robinson constructs the sorted list Δ of feasible values for the optimal error ϵ^*. Its entries are considered in a binary search fashion and the algorithm returns the smallest value $\epsilon \in \Delta$ occurring in this search for which the answer "not" is not returned (i.e., the least ϵ for which a 16ϵ-compatible total order on X exists). To decide, if, for a given ϵ, such an order exists, the procedure Refine$(X, \preccurlyeq, \epsilon)$ constructs (and/or updates) the canonical partial order $\preccurlyeq$ and computes a maximal chain P of $(X, \preccurlyeq)$. For each element $x \in X^\circ := X \setminus P$, Refine computes the set $AH(x)$ of all x-holes which participate in $(x, y, 1)$-admissible locations for all $y \in X^\circ$ and defines the segment $H(x)$. For each pair $i < j-1$, Refine constructs the set X_{ij} and makes a call of the procedure Partition_and_Sort(X_{ij}), which returns the bipartition $\{X_{ij}^-, X_{ij}^+\}$ of X_{ij} and a total order on the cells of X_{ij}^- and X_{ij}^+. Then Refine concatenates in a single total order on cells the total orders on cells coming from different sets assigned to the same hole. After this, Refine is recursively applied to each cell occurring in some graph $\mathcal{G}_{ij}$. The returned total orders on cells are concatenated into a single total order $\prec$ on X according to the total orders between cells and between holes; then $\prec$ is returned by the algorithm l_∞-Fitting_by_Robinson. The procedure Partition_and_Sort constructs the graphs $\mathcal{L}_{ij}$ and $\mathcal{L}_{ij}^{\rightarrow}$. Using these graphs, X_{ij} is partitioned into blocks and cells, then graph $\mathcal{S}_{ij}$ and its clusters are constructed. Using the cells, the directed graph $\mathcal{G}_{ij}$ is constructed. If $\mathcal{S}_{ij}$ is not bipartite or $\mathcal{G}_{ij}$ contains (G3)-cycles, then Partition_and_Sort returns the answer "not". Otherwise, for each cell C and each cluster $\mathcal{K}$, it tests if there exists a 1-cycle and/or a 2-cycle passing via C and intersecting $\mathcal{K}$. Consequently, for each cell C, the lists $\Omega_1(C), \Omega_2(C)$, and $\Omega_3(C)$ of (G2)-arcs are computed. These lists are used to construct the 2-SAT formula Φ_{ij}, which is solved by the algorithm of [2]. If Φ_{ij} admits a true assignment A, then $X_{ij}^- = \{C : A(x_C) = 0\}$ and $X_{ij}^+ = \{C : A(x_C) = 1\}$ define a bipartition of X_{ij} into two acyclic subgraphs $\mathcal{G}_{ij}^-, \mathcal{G}_{ij}^+$ of $\mathcal{G}_{ij}$. Then Partition_and_Sort locates the cells from X_{ij}^+ in the hole H_{j-1} according to the topological order of the acyclic graph $\mathcal{G}_{ij}^+$ and it locates the cells from X_{ij}^- in the hole H_i according to the dual topological order of $\mathcal{G}_{ij}^-$. Note that if at some stage Refine or Partition_and_Sort returns the answer "not", then there does not exists any ϵ-compatible total order on X and the current value of ϵ is too small. The total complexity of the algorithm is $O(n^6 \log n)$. We formulate now the main result of our paper:

Theorem 5.1. *For $\epsilon \in \Delta$, if the algorithm returns the answer "not", then the dissimilarity d is not ϵ-Robinson, else, it returns a 16ϵ-compatible total order $\prec$ on X. In particular, the algorithm is a factor 16 approximation algorithm for* l_∞-FITTING-BY-ROBINSON.

Proof. First, note that no ϵ-compatible order exist in all cases when the algorithm returns the answer "not". Indeed, Lemma 4.4, Propositions 3.2 and 4.10 cover all such cases except the case when this answer is returned by a recursive call. In this case, the induction

assumption implies that no ϵ-compatible total order on C extending $\preccurlyeq^+$ (and therefore its dual $\preccurlyeq^-$) exist. Then we infer that no ϵ-compatible order on X exist as well.

Now, let the algorithm return a total order $\prec$. Suppose by induction assumption that $\prec$ is 16ϵ-compatible on each cell to which a recursive call is applied. On the chain P, the total order $\prec$ coincides with $\preccurlyeq$, therefore $\prec$ is ϵ-compatible on P. Moreover, $\prec$ is ϵ-compatible on $P \cup \{x\}$ for any $x \in X^\circ$, because every element x is located in a bounding hole of $H(x)$ which is x-admissible. Finally notice that $\prec$ is 12ϵ-compatible on $P \cup \{x, y\}$ for any $x, y \in X^\circ$ because by Proposition 3.2 the bounding hole of $H(x)$ and the bounding hole of $H(y)$ into which x and y are located define a $(x, y, 12)$-admissible pair. To prove that $\prec$ is 16ϵ-compatible on the whole set X, it suffices to show that $d(x, z) \gtrsim_{16} \max\{d(x, y), d(y, z)\}$ for any three elements $x, y, z \in X$ such that $x \prec y \prec z$. From previous discussion, we can suppose that $x, y, z \in X^\circ$. For this, we distinguish the Cases **(H1)-(H5)** in function of the mutual location of segments $H(x)$ and $H(z)$ and in each case we show the required inequality. The respective case analysis is given in [9].

■

References

[1] R. Agarwala, V. Bafna, M. Farach, B. Narayanan, M. Paterson, and M. Thorup, On the approximability of numerical taxonomy (fitting distances by tree metrics), *SIAM J. Comput.* **17** (1999), 1073-1085.

[2] B. Aspvall, M.F. Plass, and R.E. Tarjan, A linear time algorithm for testing the truth of certain quantified boolean formulas, *Inf. Proc. Lett.*, **8** (1979), 121-123.

[3] J.E. Atkins, E.G. Boman, and B. Hendrickson, A spectral algorithm for seriation and the consecutive ones problem, *SIAM J. Comput.* **28** (1998), 297–310.

[4] M. Bădoiu, Approximation algorithm for embedding metrics into a two-dimensional space, *SODA 2003.*

[5] M. Bădoiu, A. Gupta, K. Dhamdhere, Y. Rabinovich, H. Räcke, and R. Ravi, A. Sidiropoulos, Approximation algorithms for low-distortion embeddings into low-dimensional spaces, *SODA, 2005.*

[6] G. Caraux and S. Pinloche, PermutMatrix: a graphical environment to arrange gene expression profiles in optimal linear order, *Bioinformatics* **21**(2005), 1280-1281.

[7] V. Chepoi and B. Fichet, l_∞-Approximation via subdominants, *J. Math. Psych.,* **44** (2000), 600-616.

[8] V. Chepoi, B. Fichet, and M. Seston, Seriation in the presence of errors: NP-hardness of l_∞-fitting Robinson structures to dissimilarity matrices, *J. Classification* (to appear).

[9] V. Chepoi and M. Seston, Seriation in the presence of errors: an approximation algorithm for fitting Robinson structures to dissimilarity matrices (submitted).

[10] B. Chor and M. Sudan, A geometric approach to betweenness, *SIAM J. Discr. Math.* **11** (1998), 511–523.

[11] E. Diday, Orders and overlapping clusters by pyramids, In *Multidim. Data Analysis,* pp. 201-234, 1986.

[12] C. Durand and B. Fichet, One-to-one correspondences in pyramidal representation: a unified approach, In *Classification and Related Methods of Data Analysis,* pp. 85-90, North–Holland, 1988.

[13] M. Farach, S. Kannan, et T. Warnow, A robust model for finding optimal evolutionary trees, *Algorithmica,* **13** (1995), 155-179.

[14] M. Hahsler, K. Hornik, and C. Buchta, Getting things in order: an introduction to the R package seriation, *J. Statistical Software,* **25** (2008), 1-34.

[15] J. Hstad, L. Ivansson, and J. Lagergren, Fitting points on the real line and its application to RH mapping, In *ESA 1998* and *J. Algorithms,* **49** (2003), 42-62.

[16] L.J. Hubert, Some applications of graph theory and related nonmetric techniques to problems of approximate seriation, *British J. Math. Stat. Psych.*, **27** (1974), 133-153.

[17] D.S. Johnson, The NP-completeness column: an outgoing guide, *J. Algorithms,* **3**(1982), 182-195.

[18] D.G. Kendall, Seriation from abundance matrices, *Mathematics in the Archaeological and Historical Sciences,* Eds., F. R. Hodson, D. G. Kendall, and P. Tăutu, pp. 215-252, 1971.

[19] B. Mirkin and S. Rodin, *Graphs and Genes*, Springer, 1984.

[20] W. S. Robinson, A method for chronologically ordering archaeological deposits, *American Antiquity,* **16** (1951), 293–301.

This work is licensed under the Creative Commons Attribution-NoDerivs License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nd/3.0/.