Totally Balanced Dissimilarities

François Brucker, Pascal Prea, Célia Châtel

Totally balanced dissimilarities

François Brucker, Pascal Préa and Célia Châtel

École Centrale Marseille
LIF, CNRS UMR 7279
38 rue Joliot-Curie - F-13451 Marseille Cedex 20, France.
email: francois.brucker@centrale-marseille.fr
email: pascal.prea@centrale-marseille.fr
email: celia.chatel@centrale-marseille.fr

Abstract. We show in this paper a bijection between totally balanced hypergraphs and so called totally balanced dissimilarities. We give an efficient way ($O(n^3)$ where n is the number of elements) to (i) recognize if a given dissimilarity is totally balanced and (ii) approximate it if it is not the case. We also introduce a new kind of dissimilarity which generalize chordal graphs and allows a polynomial number of cluster that can be easily computed and interpreted.

June 26, 2017 – 16:26

1 Introduction

Totally balanced hypergraphs, initially defined by Lovasz (1968), is a hypergraph structure which corresponds to the notion of tree for graphs (see Lehel, 1985) thus occurring in various applications like linear programming, databases structures of phylogenetic problems.

Totally balanced hypergraphs are equivalent to Γ-free 0/1-matrices (Antsee and Farber, 1984), strongly chordal graphs (Farber, 1983) or dismantlable lattices (Brucker and Gély, 2010). These alternative characterizations make them a good classification model because their clusters can be interpreted either as a sup of two elements (dismantlable lattice) or as a maximal clique of some graph (strongly chordal graph), and the whole structure is linked by an underlying family of trees. Moreover, they admit a convenient graphical representation (Brucker and Préa, 2015). Finally totally balanced hypergraphs are weak-hierarchies (Brucker and Gély, 2010 — see Bandelt and Dress (1989) for a definition of weak-hierarchies), they have a lot of good practical properties (see Bertrand and Diatta, 2014 for instance) like admitting a relatively small number of overlapping clusters (at most the square of the number of elements) or being equivalent to an underlying metric model.

We will characterize in this paper the set of dissimilarities, named totally balanced dissimilarities, whose cluster set is a totally balanced hypergraph.

This characterization will also allow us to give new demonstrations of some well known theorems for graphs and to generalize them to dissimilarities.

We will also give three polynomial algorithms (in $O(n^3)$ operations where n is the number of elements) for dealing with those dissimilarities: one to check if a given dissimilarity is totally balanced, one to approximate (if necessary) a given dissimilarity into a totally balanced dissimilarity and the last one to give all the clusters associated with a totally balanced dissimilarity.

This paper is organized as follows. Section 2 gives a definition of totally balanced dissimilarities and shows that they are in bijection with totally balanced hypergraphs. Section 3 gives an efficient way of determining the clusters of those dissimilarities by using a new kind of dissimilarities (chordal dissimilarities). We then show an algorithm for determining if a given dissimilarity is totally balanced (Section 4) and a way to approximate it if it is not the case (Section 5). We finally give an example of the use of these algorithms on a real dataset (Section 6).

2 Totally Balanced Structure correspondances

We will define in this section totally balanced hypergraphs through a linear order, named totally balanced order. Moreover, this order will be used to link the different totally balanced structures (hypergraphs: Proposition 3, dissimilarities: 5 and binary matrices: Proposition 13). Finally, we define totally balanced dissimilarities and prove that there is a bijection between totally balanced hypergraphs and the cluster hypergraphs of those dissimilarities (Proposition 6).

2.1 Hypergraphs

Recall that an *hypergraph* is a couple $H = (V, E)$ where $E \subseteq 2^V$. As for a graph, elements of V and E are named *vertices* and *edges* respectively. We will also denote by $\overline{H}$ the *closure* of H: $\overline{H} = (V, \overline{E})$ where $\overline{E}$ is the closure by intersection of E ($\overline{E}$ is the smallest subset of 2^V such that $E \subseteq \overline{E}$ and $X, Y \in \overline{E} \implies X \cap Y \in \overline{E} \vee X \cap Y = \emptyset$) and by $H|_W$ the restriction of H to $W \subset V$: $H|_W = (W, E|_W)$, where $E|_W = \{X \cap W \mid X \in E, X \cap W \neq \emptyset\}$.

A sequence $(v_1, e_1, \ldots, v_k, e_k)$ where $v_i \in V$ and $e_i \in E$ $(1 \leq i \leq k)$ is a *path* in H if $v_i, v_{i+1} \in e_i$. A *cycle* is a path $(v_1, e_1, \ldots, v_k, e_k)$ with $v_1 \in e_k$. Moreover, a cycle $(v_1, e_1, \ldots, v_k, e_k)$ is a *special cycle* if $k \geq 3$ and $v_i \in e_j$ if and only if $j = i$, $j = i - 1$ or $(i, j) = (1, k)$.

An hypergraph is *totally balanced* if it has no special cycle. This can be seen as the hypergraph version of trees (a graph with no cycle). We have the two elementary properties:

Proposition 1. *An hypergraph $H = (V, E)$ is totally balanced if and only if its closure is totally balanced.*

Proposition 2. *If an hypergraph $H = (V, E)$ is totally balanced then for all $X, Y, Z \in E$: $X \cap Y \cap Z \in \{X \cap Y, Y \cap Z, X \cap Z\}$*

Let $H = (V, E)$ a hypergraph. A linear order $v_1 < v_2 < \cdots < v_n$ on V is *totally balanced* (with H) if the sets $\{X \mid v_i \in X, X \in E|_{v_i,\ldots,v_n}\}$ are chains for all $1 \leq i \leq n$. A set $C \subset 2^V$ is a *chain* if for all $X, Y \in C$, either $X \subseteq Y$ or $Y \subseteq X$.

Proposition 3 (Brucker and Gely, 2010). *An hypergraph H is totally balanced if and only if it admits a totally balanced ordering.*

2.2 Dissimilarities

A *dissimilarity* on a set V is a function d from $V \times V$ to the non-negative real numbers which is symmetrical ($d(x, y) = d(y, x)$) and admits a zero-diagonal ($d(x, x) = 0$). All the dissimilarities occurring in this paper will be assumed to be *proper*: $d(x, y) = 0 \iff x = y$.

For $\sigma \geq 0$, the *threshold graph* G_σ of d admits V as a vertex set and the pairs xy such that $d(x, y) \leq \sigma$ as edge set. The maximal cliques of the threshold graphs are called *clusters*. For $\sigma \geq 0$ and $x \in V$, the *ball* $B(x, \sigma)$ is the set $\{y \in V : d(x, y) \leq \sigma\}$. Given a dissimilarity d on V one can associate two hypergraphs:

- its *cluster hypergraph* $\mathcal{C}(d) = (V, E)$ where E is the set of all its clusters (the maximal cliques for all its threshold graphs),
- its *ball hypergraph* $\mathcal{B}(d) = (V, E)$ where E is the set of all the balls of d.

The *diameter* of a set $X \subseteq V$ is equal to $\mathrm{diam}(X) = \max\{d(x, y) \mid x, y \in X\}$. The diameter of a set X is the smallest α for which X is a clique of the threshold graph G_α.

We define then the *totally balanced dissimilarities* as the dissimilarities for which $\mathcal{C}(d)$ is totally balanced. An order $v_1, \ldots v_n$ is said to be *totally balanced* with d if it is totally balanced with $\mathcal{C}(d)$.

Given dissimilarity d on V, we say that $v_1, \ldots, v_n$ is a *chordal order* for d if for all $i \leq j, k$, $d(v_j, v_k) \leq \max\{d(v_i, v_j), d(v_i, v_k)\}$. A dissimilarity is said to be *chordal* if it admits a chordal order.

Note that for a given totally balanced dissimilarity d, the totally balanced orders of d are chordal orders. Proposition 4 characterize the chordal orders that are also totally balanced.

Proposition 4. *An order $v_1 < \cdots < v_n$ on V is totally balanced for a dissimilarity d on V if and only if:*

1. *it is a chordal order,*
2. *for all (g, h, i, j, k) with $g, h \leq i < j, k$: either $d(v_g, v_k) \leq \max\{d(v_g, v_i), d(v_g, v_j)\}$, or $d(v_h, v_j) \leq \max\{d(v_h, v_i), d(v_h, v_k)\}$.*

Proof. First suppose that the order is totally balanced for $\mathcal{C}(d)$ but not chordal for d. Then there exist $i < j, k$ such that $d(v_j, v_k) > \max\{d(v_i, v_j), d(v_i, v_k)\}$; so there exists two maximal cliques X and Y of the threshold graph $G_{\max\{d(v_i,v_j),d(v_i,v_k)\}}$ such that X contains v_i and v_j but not v_k and Y contains v_i and v_k but not v_j. Thus the order cannot be totally balanced for $\mathcal{C}(d)$.

Second suppose that the order it totally balanced and chordal but there exist $g, h \leq i \leq j, k$ such that $d(v_g, v_k) > \max\{d(v_g, v_i), d(v_g, v_j)\}$, and $d(v_h, v_j) > \max\{d(v_h, v_i), d(v_h, v_k)\}$. So there exists a maximal clique X of the threshold graph $G_{\max\{d(v_g,v_i),d(v_g,v_j)\}}$ containing g, i and j (because the order is chordal) but not k and a maximal clique Y of the threshold graph $G_{\max\{d(v_h,v_i),d(v_g,v_k)\}}$ containing g, i and k (because the order is chordal) but not j. So the order cannot be totally balanced for $\mathcal{C}(d)$.

Consequently, if $v_1 < \cdots < v_n$ on V is totally balanced for $\mathcal{C}(d)$ then the two conditions are satisfied.

Conversely, we suppose that the order is not totally balanced for $\mathcal{C}(d)$. Let i be an index such that $\{Z \mid v_i \in Z, Z \in \mathcal{C}(d)|_{v_i,\ldots,v_n}\}$ is not a chain. There exist $j, k > i$ and X, Y in $\mathcal{C}(d)$ such that $v_j \in X \setminus Y$ and $v_k \in Y \setminus X$. We suppose with no loss of generality that the diameter of X is not smaller than the diameter of Y.

Since X and Y are maximal cliques of some threshold graphs of d, there exists $v_g \in X$ and $v_h \in Y$ such that $d(v_g, v_k) > \mathrm{diam}(X)$ and $d(v_h, v_j) > \mathrm{diam}(Y)$. In addition, $v_g \in X \setminus Y$ and $v_h \in Y \setminus X$ (otherwise, we would have $X \subset Y$ or $Y \subset X$).

If the order is not chordal the property is verified. We suppose now that the order is chordal. We will prove that the second condition of the property is not satisfied.

Since the diameter of X is not smaller than the diameter of Y we have that $g < i$; otherwise, we would have $d(v_g, v_k) \leq \max\{d(v_i, v_g), d(v_i, v_k)\} \leq \max\{\mathrm{diam}(X), \mathrm{diam}(Y)\} \leq \mathrm{diam}(X)$.

If $h < i$, then the second condition is not satisfied for the 5-tuple (g, h, i, j, k).

If $i < h$, $d(v_h, v_j) \leq \max\{d(v_i, v_h), d(v_i, v_j)\}$. As $d(v_i, v_h) \leq \mathrm{diam}(Y) < d(v_h, v_j)$, we have $d(v_h, v_j) \leq d(v_i, v_j)$ and $d(v_i, v_j) > \mathrm{diam}(Y)$. So, there exist $v_{g'} \in X$ such that $d(v_{g'}, v_h) > \mathrm{diam}(X)\}$. In addition, $g' < i$ (otherwise, we would $i < g', h$ and $d(v_{g'|,v_h} > \max\{d(v_i, v_h), d(v_i, v_{g'}\})$. Thus the second property is not verified for the 5-tuple (g', i, i, j, h).
□

Graphs whose maximal cliques hypergraph $\mathcal{C}(G)$ is totally balanced have been characterized by Farber (1983). They are known as the *strongly chordal graphs*, which are a subclass of *chordal graphs*.

A graph is said to be chordal if every cycle $x_1x_2\dots x_nx_1$ with $n \geq 3$ admits a chord (an edge x_ix_j with $i < j+1$). They can be characterized by the existence of *simplicial ordering* $v_1, \dots, v_n$ of the vertices, i.e. an ordering such that for all $i < j, k$, if v_iv_j and v_iv_k are edges then v_jv_k is also an edge (Dirac, 1961).

Since a graph $G = (V, E)$ can be seen as a dissimilarity d_G on V ($d_G(x, y) = 1$ if $xy \in E$ and $d_G(x, y) = 2$ otherwise), chordal orders for dissimilarities are a direct generalization of simplicial orderings, and dissimilarities admitting a chordal order a direct generalization of chordal graphs.

It is known that $\mathcal{B}(d_G)$ is totally balanced if and only if $\mathcal{C}(d_G)$ is totally balanced (Farber, 1983). This is no more true for (general) dissimilarities for which we only have an implication as proved by Proposition 5 and the counter example of Table 1.

Proposition 5. *For a given dissimilarity d on V, if $\mathcal{B}(d)$ admits $v_1 < \dots < v_n$ as a totally balanced order then it is also a totally balanced order for $\mathcal{C}(d)$.*

Proof. Let $v_1 < \dots < v_n$ be a totally balanced order for $\mathcal{B}(d)$. Let v_i, v_j, v_k be three elements of V such that $v_i < v_j, v_k$. If $d(v_j, v_k) > \max\{d(v_i, v_j), d(v_i, v_k)\}$, we would have $v_k \notin B(v_j, d(v_i, v_j))$ and $v_j \notin B(v_i, d(v_i, v_k))$. As v_i is in these two balls, the order would not be totally balanced for $\mathcal{B}(d)$. So Condition 1 of Property 4 is verified.

Let (g, h, i, j, k) be a 5-tuple not verifying Condition 2 of Property 4. We would have $v_k \notin B(v_g, \max\{d(v_g, v_i), d(v_g, v_j)\})$ and $v_j \notin B(v_h, \max\{d(v_h, v_i), d(v_h, v_k)\})$. As v_i is in these two balls, the order would not be totally balanced order for $\mathcal{B}(d)$.
□

Table 1: A dissimilarity d for which $\mathcal{C}(d)$ is totally balanced (admits $x < y < z < t$ as a totally balanced order) but $\mathcal{B}(d)$ is not $((z, B(z, 2), x, B(y, 2), y, B(t, 1))$ is a special cycle).

x	0
y	2 0
z	2 3 0
t	2 1 1 0
	x y z t

2.3 Bijections

The bijection between totally balanced hypergraphs and totally balanced dissimilarities is a special case of one of the general bijections of Bertrand (2000). One can associate to a given a hypergraph $H = (V, E)$ a valuation f from E to $\mathbb{R}^+$. The couple (H, f) is said to be a *valued hypergraph*. The valuation f is said to be a *strict index* if if $X \subsetneq Y$ implies $f(X) < f(Y)$. Note that the cardinal of a cluster is a strict index for any hypergraph and the diameter is a strict index of $\mathcal{C}(d)$ for any dissimilarity d.

Theorem 1 (Bertrand, 2000). *Let $H = (V, E)$ be an hypergraph containing the singletons and V as edges and such that for any edges X, Y, Z, we have $X \cap Y \cap Z \in \{X \cap Y, Y \cap Z, X \cap Z\}$ and let f be a strict index of H such that $f(\{x\}) = 0$ for all $x \in V$. Then the dissimilarity d on V defined by $d(x, y) = \min\{f(X) | X \in E, x, y \in X\}$ is such that:*

- *$\mathcal{C}(d) = E$*
- *for all $X \in E$, $f(X)$ is the diameter of X for d*

As a direct consequence, we have:

Proposition 6. *Let $H = (V, E)$ be a totally balanced hypergraph containing the singletons and the whole set as edges, and let f be a strict index of H. Then the dissimilarity d defined by $d(x, y) = \min\{f(X) | X \in E, x, y \in X\}$ is such that $\mathcal{C}(d) = E$.*
Conversely, for any totally balanced dissimilarity d on V:

- *$(V, \mathcal{C}(d))$ is a totally balanced hypergraph containing the singletons and the whole set as edges.*
- *The diameter is a strict index on $(V, \mathcal{C}(d))$.*

3 Clusters

In the general case, a dissimilarity on a n-set can have an exponential number of clusters. We will show in this section that a chordal dissimilarity have at most n^2 clusters. Moreover, we will give an $O(n^3)$ algorithm to compute them. This algorithm is linear in the size of the output in the worst case ($O(n^2)$ clusters, each of size $O(n)$). Since totally balanced dissimilarities are a subset of chordal dissimilarities, this also gives a linear algorithm (in the worst case) for finding the clusters of totally balanced dissimilarities.

Proposition 7. *Let d be a dissimilarity on V admitting $v_1 < \cdots < v_n$ as a chordal order, then for every $X \in \mathcal{C}(d)$, $X = B(v^*, \alpha)|_{\{v \geq v^*\}}$ where:*

- *v^* is the smallest element of X (according to the chordal order).*
- *α is the diameter of X (according to d).*

Proof. If $v, w \in B(v^*, \alpha)|_{\{v \geq v^*\}}$ then $d(v, w) \leq \max\{d(v^*, v), d(v^*, w)\} \leq \alpha$ (the order is chordal). So the truncated ball $B(v^*, \alpha)|_{\{v \geq v^*\}}$ is a clique of diameter α.

Moreover, by construction, the maximal clique X is included in $B(v^*, \alpha)|_{\{v \geq v^*\}}$. Thus $X = B(v^*, \alpha)|_{\{v \geq v^*\}}$.
□

Proposition 7 shows that the maximal cliques of d are *truncated balls*, i.e. $\mathcal{C}(d) \subseteq \{B(v, d(v, w))|_{\{x \geq v\}} \mid v \leq w\}$. The number of clusters is then bounded by $O(|V|^2)$. Table 2 shows a dissimilarity with its truncated balls and maximal cliques.

Table 2: A totally balanced dissimilarity d admitting $x < z < t < y < u$ as a chordal order.

d

x	0					
y	3	0				
z	4	4	0			
t	5	2	5	0		
u	3	3	4	5	0	
	x	y	z	t	u	

$B(a, d(a, b))|_{v \geq a}$

a	x	x	x	x	z	z	z	t	t	y
b	z	t	y	u	t	y	u	y	u	u
x	×	×	×	×						
z	×	×			×	×	×			
t		×			×			×	×	
y	×	×	×	×	×	×	×	×	×	×
u	×	×	×	×	×	×	×		×	×
diameter	4	5	3	3	5	4	4	2	5	3

$\mathcal{C}(d)$

a	x	x	x	t
b	z	t	y	y
x	×	×	×	
z	×	×		
t		×		×
y	×	×	×	×
u	×	×	×	
diameter	4	5	3	2

Proposition 8. *Let d be a dissimilarity on V admitting $v_1 < \cdots < v_n$ as a chordal order. The truncated ball $B(v_i, d(v_i, v_j))|_{\{v \geq v_i\}}$ with $i \leq j$ is a maximal clique with diameter $d(v_i, v_j)$ if for all $v_k < v_i$:*

$$\max\{d(v_k, v_i), d(v_k, v_j)\} = d(v_i, v_j) \implies$$
$$\mathit{Card}(B(v_k, d(v_i, v_j))|_{\{v \geq v_i\}}) < \mathit{Card}(B(v_i, d(v_i, v_j))|_{\{v \geq v_i\}})$$

Proof. Note $\alpha = d(v_i, v_j)$. According to Proposition 7, the truncated ball $B_i = B(v_i, \alpha)|_{\{v \geq v_i\}}$ is a clique. It is maximal if it is not included in another clique of same diameter, i.e. if there does not exist a truncated ball $B_k = B(v_k, \alpha)|_{\{v > v_k\}}$ with $v_k < v_i$ such that $B_i \subset B_k$.

If it is the case, $d(v_k, v_i) \leq \alpha$ and $d(v_k, v_j) \leq \alpha$. As the orded is chordal, $d(v_i, v_j) \leq \max\{d(v_k, v_i), d(v_k, v_j)\}$. So $\alpha = \max\{d(v_k, v_i), d(v_k, v_j)\}$.

Let $v_l > v_i$ be an element of B_k, $d(v_i, v_l) \leq \max\{d(v_k, v_i), d(v_k, v_l\} \leq \alpha$. So $B_k|_{\{v \geq v_i\}} \subseteq B_i|_{\{v \geq v_i\}}$. Clearly, $B_i|_{\{v \leq v_i\}} \subseteq B_k|_{\{v \leq v_i\}}$. So, $B_i \subseteq B_k$ if and only if $\mathrm{Card}(B_i) < \mathrm{Card}(B_k|_{\{v \geq v_i\}})$
□

Algorithm 1 uses propositions 7 and 8 to compute the clusters of a dissimilarity, one of its chordal orders being given. It iterates over all the possible balls according to the chordal order and keeps those which are clusters. To do that task efficiently, the algorithm maintains numbers $S_k(t)$ for $1 \leq k, t \leq n$ which are defined by:

- At step i, if there exist a truncated ball $B_k(\alpha) = B(v_k, \alpha)|_{v>v_k}$ such that $\mathrm{Card}(B_k(\alpha)|_{v>v_i}) = t$, then $S_k(t) = \alpha$. Notice that, since $B_k(\alpha)$ is a truncated ball, there is at most one $B_k(\alpha)$ with t elements.
- At Step i, if no such truncated ball exists, $S_k(t) = -1$.

By proposition 8, a given truncated ball $C_{ij} = B(v_i, d(v_i, v_j))|_{\{v \geq v_i\}}$ with $i \leq j$ is a maximal clique if it does not exist $k < i$ such that $d(v_k, v_i) \leq d(v_i, v_j)$ and $S_k(|C_{ij}|) = d(v_i, v_j)$.

The numbers $S_k(t)$ allow us to check efficiently (in $O(|V|)$) the conditions of Proposition 8. So we have the following proposition 9.

Proposition 9. *For a given chordal dissimilarity d and a chordal order $v_1 < \cdots < v_n$, Algorithm 1 returns the set $\mathcal{C}(d)$ in $O(|V|^3)$ operations and size.*

4 Recognition

We will give in this section an $O(|V|^3)$ algorithm to determine if a given dissimilarity d on V is chordal or totally balanced. This algorithm first check whether a given dissimilarity is chordal or not by finding one of its chordal orders (Algorithm 2). We then compute its cluster hypergraph and its associated binary matrix. We linearly check (Spinrad, 2003) that its doubly lexical ordering is Γ-free.

4.1 Chordal orders

Finding a chordal order for a given dissimilarity d can be iteratively done using Proposition 10.

Proposition 10. *If a dissimilarity d on a set V admits a chordal order then for all $x \in V$ such that $\forall y, z \in V$ $d(y, z) \leq \max\{d(x, y), d(x, z)\}$, the restriction of d on $V \backslash \{x\}$ admits a chordal order.*

Proof. If d admits a chordal order $v_1 < v_2 < \cdots < v_n$, then $v_2 < \ldots < v_n$ is a chordal order on $\{v_2, \ldots v_n\}$. If there exits v_i such that or all $y, z \in V$, $d(y, z) \leq \max\{d(v_i, y), d(v_i, z)\}$, then $v_i < v_1 < \cdots < v_{i-1} < v_{i+1} < \cdots < v_n$ is also a chordal order for d.
□

Algorithm 2 uses then Proposition 10 iteratively to find a chordal order for a given dissimilarity d.

Algorithm 1: CLUSTER-SET-CONSTRUCTION

Data: A dissimilarity d on a n-set V and one of its chordal order $v_1 < \cdots < v_n$

Result: The cluster set $\mathcal{C}(d)$

```
// Initialization
```
$\mathcal{C} \leftarrow \emptyset$

$S_1(t) \leftarrow -1$ **for** $1 \leq t \leq n$

```
// All the balls with center v_1 are clusters
```
for $j \leftarrow 1$ **to** n **do**

  add $B(v_1, d(v_1, v_j))$ to $\mathcal{C}$

  $S_1(|B(v_1, d(v_1, v_j))| - 1) \leftarrow d(v_1, v_j)$

```
// Main Loop
```
for $i \leftarrow 2$ **to** n **do**

  $V_i \leftarrow \{v_i, \ldots, v_n\}$

  $S_i(t) \leftarrow -1$ **for** all $1 \leq t \leq n$

  for $j \leftarrow i$ **to** n **do**

    $C_{ij} \leftarrow B(v_i, d(v_i, v_j)) \cap V_i$

    if $\nexists k < i$ **such that** $(d(v_k, v_i) \leq d(v_i, v_j)$ **and** $S_k(|C_{ij}|) = d(v_i, v_j))$ **then**

      ```
      // C_ij is not included in a cluster with same
         diameter: its a cluster
      ```

      add C_{ij} to $\mathcal{C}$

      $S_i(|C_{ij}| - 1) \leftarrow d(v_i, v_j)$

  ```
  // Update Sizes
  ```

  for $j \leftarrow 1$ **to** $i - 1$ **do**

    for $t \leftarrow 1$ **to** n **do**

      if $S_j(t) \geq d(v_j, v_i)$ **then**

        if $t > 1$ **and** $S_j(t-1) = -1$ **then**

          $S_j(t-1) \leftarrow S_j(t)$

        $S_j(t) \leftarrow -1$

return $\mathcal{C}$

Proposition 11. *Given a dissimilarity d on V, Algorithm 2 returns a chordal order (if there exists one) in $O(|V|^3)$ operations.*

Proof. At each step, $N_i(v)$ is the number of pairs $\{x, y\} \subset V_i$ for which $d(x, y) > \max\{d(v, x), d(v, y)\}$. By Proposition 10, Algorithm 2 return a chordal order. Clearly, it runs in $O(|V|^3)$ operations.
□

Algorithm 2: CHORDAL-ORDER-CONSTRUCTION

Data: A dissimilarity d on a set V
Result: A chordal order for d, if d is chordal

// Initialization
$V_1 \leftarrow V$
for $v \in V$ **do**
  $N_1(v) \leftarrow 0$
  for $x \neq y \in V$ **do**
    if $d(x,y) > \max\{d(v,x), d(v,y)\}$ **then** $N_1(v) \leftarrow N_1(v) + 1$
// Main Loop
$i \leftarrow 1$
while $V_i \neq \emptyset$ **do**
  Let $v^* \in V_i$ be such that $N_i(v^*) = \min_{v \in V_i} N_i(v)$
  if $N_i(v^*) > 0$ **then**
    // There is no chordal order
    FAIL
  $v_i \leftarrow v^*$
  $V_{i+1} \leftarrow V_i \backslash \{v^*\}$
  for $v \in V_{i+1}$ **do**
    $N^* \leftarrow 0$
    for $w \in V_{i+1}$ **do**
      if $d(v^*, w) > \max\{d(v, v^*), d(v, w)\}$ **then** $N^* \leftarrow N^* + 1$
    $N_{i+1}(v) \leftarrow N_i(v) - 2 * N^*$
  $i \leftarrow i + 1$
return $v_1 < \cdots < v_n$

Table 3 shows the progression of the different N_i when computing the chordal order $x < z < t < y < u$ for the dissimilarity d of Table 2.

Table 3: Values for N_i when computing the chordal order $x < z < t < y < u$ for the dissimilarity d of Table 2.

	N_1	N_2	N_3	N_4	N_5
x	0				
z	0	0			
t	0	0	0		
y	6	4	2	0	
u	0	0	0	0	0

4.2 Totally balanced orders and Γ-free matrices

Finding a totally balanced order can be done using the equivalence between totally balanced hypergraphs and Γ-free matrices.

A $n \times m$ binary matrix M is equivalent to a hypergraph $\mathcal{H}(M) = (\{1, \ldots, n\}, \{C_1, \ldots, C_m\})$ where $C_j = \{i \mid M_{i,j} = 1\}$ $(1 \leq j \leq m)$. Conversely, given a hypergraph $H = (V, E)$, if we label V as $\{v_1, \ldots, v_n\}$ and E as $\{e_1, \ldots, e_m\}$, H is equivalent to a $n \times m$ binary matrix $\mathcal{M}(H)$ where $\mathcal{M}(H)_{i,j} = 1$ whenever $v_i \in e_j$.

We will say that a binary matrix M is *totally balanced* if $\mathcal{H}(M)$ is totally balanced. A $n \times m$ binary matrix M is said to be *Γ-free* whenever for any $1 \leq i < i' \leq n$ and any $1 \leq j < j' \leq m$: $M_{i,j} = M_{i,j'} = M_{i',j} = 1$ implies $M_{i',j'} = 1$. A *Γ-free ordering* of a matrix M is an ordering of its lines and columns such that M (when re-ordered) is Γ-free.

A *doubly lexical ordering* of an $n \times m$ binary matrix is an ordering of its lines and columns such that if the rows and columns are viewed as n or m digit numbers read from right to left for lines and from bottom to top for columns, both rows and columns occur in increasing order. Every binary matrix admits a doubly lexical ordering. Doubly lexical orderings, Γ-free matrices and totally balanced matrices are linked by Theorem 2.

Theorem 2 (Antsee and Farber, 1984). *Given a binary matrix M, the three following assertions are equivalent:*

1. *M is a totally balanced binary matrix,*
2. *There is a doubly lexical ordering of M which is Γ-free,*
3. *Every doubly lexical ordering of M is Γ-free.*

By Theorem 2, determining if a hypergraph H is totally balanced can be done by doubly lexical order $\mathcal{M}(H)$ and determine if this ordering is Γ-free. Finding a doubly lexical ordering of a binary matrix and determining if this ordering is Γ-free can be done by two algorithms of Spinrad. Given a $n \times m$ binary matrix M, Spinrad (1993) gives a linear algorithm (in $O(nm)$ operations) which returns a doubly lexical ordering of M. The matrix M is then totally balanced if the ordering is also Γ-free, which can also be checked (and approximated by adding 1 to avoid Γ's if necessary) in $O(nm)$ operations. See for instance Spinrad (2003) for a description of these three algorithms (the doubly lexical ordering, the check if a given matrix admits a Γ and the approximation into a Γ-free matrix).

The above algorithms give a $O(|V| \cdot |E|)$ operations and space procedure to determine whether a given hypergraph $H = (V, E)$ is totally balanced by checking if a doubly lexical ordering of $\mathcal{M}(H)$ is Γ-free. Since, by Proposition 2, a totally balanced hypergraph cannot have more than $(|V|^2 + |V|)/2$ clusters, this is also a $O(|V|^3)$ time and space algorithm.

Finally, Propositions 12 and 13 show that this will also lead to find the totally balanced orders of a totally balanced hypergraph. These two propositions link Γ-free and totally balanced orderings.

Proposition 12. *If a $n \times m$ binary matrix M is Γ-free, the natural order of its lines $(1, 2, \ldots, n)$ is totally balanced for $\mathcal{H}(M)$.*

Proof. Let M be a Γ-free matrix, and i, j, j' be such that $j < j'$ and $M_{i,j} = M_{i,j'} = 1$. If $M_{i',j} = 1$ with $i' \geq i$, then $M_{i',j'} = 1$. Thus $C_j \cap \{i, \ldots, n\} \subseteq C_{j'} \cap \{i, \ldots, n\}$ if $i \in C_j \cap C_{j'}$ and $j \leq j'$, i.e. the order $1, 2, \ldots, n$ is totally balanced for $\mathcal{H}(M)$.
□

Proposition 13. *If the order $v_1 < \cdots < v_n$ is totally balanced for $H = (V, E)$, then there exists a Γ-free column ordering for $\mathcal{M}(H)$ where line i corresponds to v_i.*

Proof. For X and Y in E, we denote by V_{XY} the set $\{v_{i_{XY}}, v_{i_{XY}+1}, \ldots, v_n\}$ where $i_{XY} = \min\{i \mid v_i \in X \cap Y\}$. If $X \cap Y = \emptyset$, i_{XY} does not exist and we define $V_{XY} = \emptyset$. We consider the directed graph G with E as vertex set and such that (X, Y) is an arc (we denote it by $X \to Y$) if $X \cap V_{XY} \subsetneq Y \cap V_{XY}$. Notice that, if $X \to Y$, then $V_{XY} \neq \emptyset$ (the converse is false).

We now proof that if $X \to Y$ and $Y \to Z$, then $X \to Z$ or $i_{XY} < i_{YZ}$. Suppose that $i_{XY} \geq i_{YZ}$. In this case, $v_{i_{XY}} \in Z \cap X$, thus i_{XZ} exists; moreover $i_{XZ} \leq i_{XY}$. As $X \to Y$, there exists $j > i_{XY}$ such that $v_j \in Y \setminus X$. As $Y \to Z$ and $i_{YZ} \leq i_{XY}$, $v_j \in Z \setminus X$. As the order is totally balanced, $X \cap V_{XZ}$ is included in, equal to or contains $Z \cap V_{XZ}$. As $v_j \in V_{XZ}$, $X \cap V_{XZ} \subsetneq Z \cap V_{XZ}$ and $X \to Z$.

So the directed graph G is acyclic and its vertex set admits a topological order $<_T$ (if $X_i \to X_j$ then $i <_T j$). Ordering the columns of $\mathcal{M}(H)$ along $<_T$ and the lines by the totally balanced order of H makes the matrix $\mathcal{M}(H)$ Γ-free.
□

4.3 Recognition procedure

Putting together the preceding parts, we get a simple algorithm which recognize in time $O(|V|^3)$ if a given dissimilarity d on V is totally balanced or not:

Step 1 of Algorithm 3 is Algorithm 2. Step 4 is Algorithm 1. Both steps run in $O(n^3)$. Since the matrix $\mathcal{M}(\mathcal{C}(d))$ has n lines and n^2 columns, the test on line 6 also runs in $O(n^3)$.

A doubly lexical order of the dissimilarity in Table 2 is presented in Table 4. The matrix is Γ-free: the dissimilarity of Table 2 is totally balanced.

Algorithm 3: CHECK-TOTALLY-BALANCED-DISSIMILARITY

Data: A dissimilarity d on a n-set V

```
Try to find a chordal order
if ∄ chordal order then
    return "d is not chordal thus not totally balanced"
Compute M(C(d))
Order M(C(d)) along a doubly lexical order
if M(C(d)) is Γ-free then
    return "d is totally balanced"
else
    return "d is chordal but not totally balanced"
```

Table 4: Doubly lexical ordering of the binary matrix of Table 2.

a	t	x	x	x
b	y	y	z	y
t	×			×
z			×	×
x		×	×	×
u		×	×	×
y	×	×	×	×
diameter	2	3	4	5

5 Approximation

We present in this section a $O(|V|^3)$ time and space procedure to approximate a dissimilarity d on V by a totally balanced one. If the initial dissimilarity is already totally balanced, the resulting dissimilarity will be the same. This procedure can be sketched as follows:

1. Approximate d into a chordal dissimilarity d' (See Section 5.1).
2. Compute $\mathcal{M}(d')$ and approximate it into a Γ-free matrix M' (See Section 5.2).
3. Associate a totally balanced dissimilarity to the valued $n \times m$ matrix $(M', (\alpha_i)_{1 \leq i \leq m})$ where the valuations $(\alpha_i)_{1 \leq i \leq m}$ come from d' (See Section 5.3).

5.1 Approximation by a chordal dissimilarity

In order to approximate a given dissimilarity d on V, one can proceed in two steps:

1. Find a possible linear order on V.
2. Approximate d into a dissimilarity chordally compatible with the chosen order.

Finding a potential order can be done using Section 4.1. Since the original dissimilarity may not be chordal, we have to remove in Algorithm 2 the check whether $N_d(v^*)$ equals 0 or not. If we keep at each step an element v^* which realizes the minimum of $N_d()$, we are assured that the algorithm will always return a linear order on V and that this order is chordal if d is chordal.

Let then d be a dissimilarity on V and $v_1 < \cdots < v_n$ an order on V. One can then use Algorithm 4 which approximate the dissimilarity d into a dissimilarity d' which admits this order as a chordal one.

Algorithm 4: CHORDAL-APPROXIMATION-ACCORDING-TO-ORDER

Data: A dissimilarity d on a n-set V and a linear order $v_1 < \cdots < v_n$ on V

Result: A dissimilarity d' admitting $v_1 < \cdots < v_n$ as chordal order.

```
d' ← d
for i ← 1 to n do
    for j ← i + 1 to n do
        for k ← j + 1 to n do
            if d'(v_j, v_k) > max(d'(v_i, v_j), d'(v_i, v_k)) then
                d'(v_j, v_k) ← max(d'(v_i, v_j), d'(v_i, v_k))
return d'
```

Proposition 14. *Given a dissimilarity d and an order $v_1 < \cdots < v_n$ on V, Algorithm 4 turns in $O(|V|^3)$ and returns a dissimilarity d' admitting $v_1 < \ldots v_n$ as chordal order.*

Moreover, if d admits $v_1 < \cdots < v_n$ as chordal order, $d' = d$.

Proof. Clearly, Algorithm 4 runs in time $O(|V|^3)$.

Let $i^* < j^* < k^*$ be three indices; we show that, when the algorithm stops, $d'(v_{j^*}, v_{k^*}) \leq \max(d'(v_{i^*}, v_{j^*}), d'(v_{i^*}, v_{k^*}))$. This is (or becomes) true when $i = i^*$, $j = j^*$ and $k = k^*$. After that step, the only distance (among $d'(v_{j^*}, v_{k^*})$, ($d'(v_{i^*}, v_{j^*})$ or $d'(v_{i^*}, v_{k^*})$)) that can be modified is $d'(v_{j^*}, v_{k^*})$, and it can only be lowered. So the condition remains true.

If d admits $v_1 < \cdots < v_n$ as chordal order, the condition "**if** $d'(v_j, v_k) > \max(d'(v_i, v_j), d'(v_i, v_k))$" is never satisfied, so d' remains equal to d.
□

5.2 Γ-free approximationn of binary matrices

Given an $n \times m$ binary matrix M, it is possible to approximate it by a totally balanced one with the following algorithm:

1. Reorder the lines and the columns of M such that M is doubly lexically ordered,
2. Check and approximate if necessary the doubly lexically ordered matrix M into a Γ-free one.

Each step can be done in $O(nm)$ operations: Step 1 is an algorithm of Spinrad (1993), Step 2 is one of Lubiw (1987). See Chapter 9 of Spinrad (2003) for a detailled exposition of these algorithms.

5.3 Dissimilarity from a Γ-free valued matrix

Given an $n \times m$ Γ-free binary matrix M and a vector $(\alpha_j)_{1 \leq j \leq m}$ of real positive numbers, our aim is to build a dissimilarity d such that:

$$d(x, y) = \min\{\alpha_j \mid M_{x,j} = M_{y,j} = 1\}$$

In addition, since M is Γ-free, d is totally balanced.

We suppose without loss of generality that the last column of M is full of 1's. Computing this dissimilarity can be done in $O(n^3)$ operations using Algorithm 5, as shown in Proposition 15

Proposition 15. *Given an $n \times m$ Γ-free binary matrix M with last column full of 1's and m real positive numbers, Algorithm 5 computes a dissimilarity d such that $d(x, y) = \min\{\alpha_j \mid M_{x,j} = M_{y,j} = 1\}$ in $O(nm + n^3)$ operations.*

Proof. Notice that for any i, j, $N_i[j] = \sum_{k \geq i} M_{i,j}$ and that there is no repetition in P_i. Thus, $|P_i| \leq n$ for any $i \leq n$. So the loop of Lines 15-16 runs in $O(n^2)$ and the algorithm performs in $O(nm + n^3)$ operations.

Since M is Γ-free, for every i, the sets $C_{ij} = \{i' \geq i : M_{i',j} = 1\}$, the lists N_i and the lists P_i are such that:

- If $M_{i,j} = M_{i,j'} = 1$, $N_i[j] < N_i[j'] \iff \{i\} \subseteq C_{ij} \subsetneq C_{ij'}$ (for all i, $\{C_{i,j} : 1 \leq j \leq m, M_{i,j} = 1\}$ is a chain); this allows to check wether $C_{ij} \subset C_{ij'}$ by comparing $N_i[j]$ and $N_i[j']$ (Line 10).
- If $j < j'$ and $M_{i,j} = M_{i,j'} = 1$, then $C_{ij} \subseteq C_{ij'}$. So after Line 14, P_i contains all indices of the interesting columns (and only them):
  - Indices not in P_i are useless: if, for $j \notin P_i$, $M_{i,j} = M_{i',j} = 1$, then there exists $j' \in P_i$ with $M_{i,j'} = M_{i',j'} = 1$ and $\alpha_{j'} < \alpha_j$.
  - All indices in P_i are useful: $\forall j < j' \in P_i$, $\alpha_j > \alpha_{j'}$ and $C_{ij} \subsetneq C_{ij'}$.

So d is such that $d(x, y) = \min\{\alpha_j \mid M_{x,j} = M_{y,j} = 1\}$.
□

Algorithm 5: DISSIMILARITY-COMPUTATION-FROM-MATRIX-AND-INDEX

Data: A $n \times m$ Γ-free binary matrix M with last column full of 1's and m real positive numbers α_j

Result: The dissimilarity d defined by $d(x, y) = \min\{\alpha_j \mid M_{x,j} = M_{y,j} = 1\}$

```
// Initialization
for j ← 1 to m do N_{n+1}[j] ← 0
for i ← 1 to n do d(i, i) ← 0
// Main Loop
for i ← n downto 1 do
    // Inner Loop Initialization
    for j ← 1 to m do
        N_i[j] ← N_{i+1}[j] + M_{i,j}
    P_i ← [m]
    c ← m
    // Place the column separations
    for j ← m downto 1 do
        if M_{i,j} = 1 and α_j < α_c then
            if N_i[c] > N_i[j] then
                Append j at the end of P_i
            else
                Replace the last element of P_i by j
            c ← j
    // Dissimilarity Update
    for j ← i to n do
        d(i, j) ← min{α_k | k ∈ P_i and M_{j,k} = 1}
return d
```

Actually, if we keep in M only the columns C_j such that $\nexists j' \neq j : C_j \subsetneq C'_j$ and $\alpha'_j \leq \alpha_j$, we get a matrix M' whose columns are strictly indexed by the α's. Moreover, Algorithm 5 returns the same dissimilarity. Thus, if M is Γ-free, the dissimilarity d, returned by Algorithm 5 is totally balanced, its clusters are the columns of M' and its diameters are the α's.

5.4 Totally balanced dissimilarity

It is possible to approximate a given dissimilarity d on V by a totally balanced one with Algorithm 6

All the steps of this algorithm come from previous sections:

Algorithm 6: TOTALLY-BALANCED-DISSIMILARITY-APPROXIMATION

Data: A dissimilarity d on a n-set V
Result: A totally balanced dissimilarity d' on V

1 Find a possible chordal order $v_1 < \dots v_n$ of V
2 Approximate d into a chordal dissimilarity d' admitting $v_1 < \dots v_n$ as a chordal order
3 Compute $M' = \mathcal{M}(\mathcal{C}(d'))$
4 Reorder M' along a doubly lexical order
5 **for** all columns C_j of M' **do** Compute $\alpha_j = \text{diam}(C_j)$
6 Approximate M' into a Γ-free matrix M''
7 Compute the dissimilarity d'' defined by $d''(x, y) = \min\{\alpha_j \mid {M''}_{x,j} = {M''}_{y,j} = 1\}$
8 **return** d''

- Step 1 of Algorithm 6 is the variant of Algorithm 2 defined in Section 5.1.
- Step 2 is Algorithm 4.
- Step 3 is Algorithm 1.
- Step 4 is an algorithm of Spinrad (1993).
- Given a cluster C_j, Step 5 can be done in $O(n)$ since $v_1 < \dots v_n$ is a chordal order: the diameter of C_j is $\max\{d(v*, w) : x \in C_j\}$, where $v*$ is the lowest element in C_j according to the chordal order. So, for the whole matrix, Step 5 is in $O(nm)$.
- Step 6 is an algorithm of Lubiw (1987)
- Step 7 is Algorithm 5

All these steps run in $O(n^3)$ (there are at most $O(n^2)$ clusters for a chordal dissimilarity on a n-set), so we have the hereafter proposition 16.

Proposition 16. *Algorithm 6 approximates a dissimilarity d on a n-set into a totally balanced one d' in $O(n^3)$. Moreover, if d is totally balanced, $d' = d$.*

6 Example and representation

As an example, we will use an archeological data set from Alberti (2013). The original data set consists in a matrix whose lines are objects and columns some archeological site of the Punta Milazzese (Aeolian Archipelago, Italy) settlement. An element of the matrix is the number of time a given object (line) has been found in a given archeological site (column). The original dissimilarity is the L_2 distance of the normalized data set: the smaller the dissimilarity between two objects, the more they are found together in a site.

We applied Algorithm 6 on these data. We got the results depicted in Figure 1. The color are associated with a hierarchical decomposition of the hypergraph (Brucker and Gély, 2009) and the cluster representation is an up

facing triangle if it has only one predecessor, a down facing triangle if it has only one successor (hence a losange if it is both) and a circle otherwise.

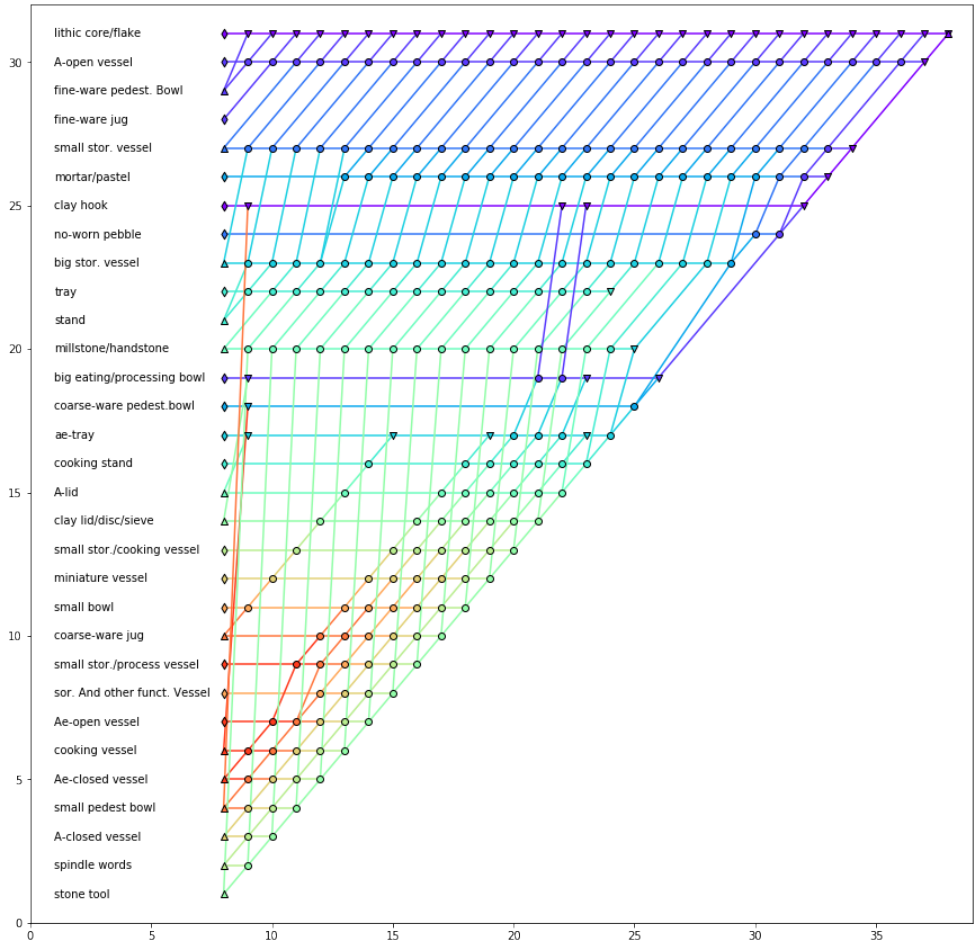

Figure 1: Clusters of the totally balanced approximation of the original L_2 dissimilarity. Element order is a totally balanced one. Height is equal to the cardinal of the cluster

By drawing the clusters using the diameter as height (Figure 2), we see that a lot of clusters are really close: they are either on the same horizontal line (thus a truncated balls for the same origin. See for intance the horizontal line for the first element) and have very close diameter, or on the same vertical line and form a close cluster succession (see for instance the lower part

of Figure 2 and the 4 vertical cluster lines). Finally using totally balanced

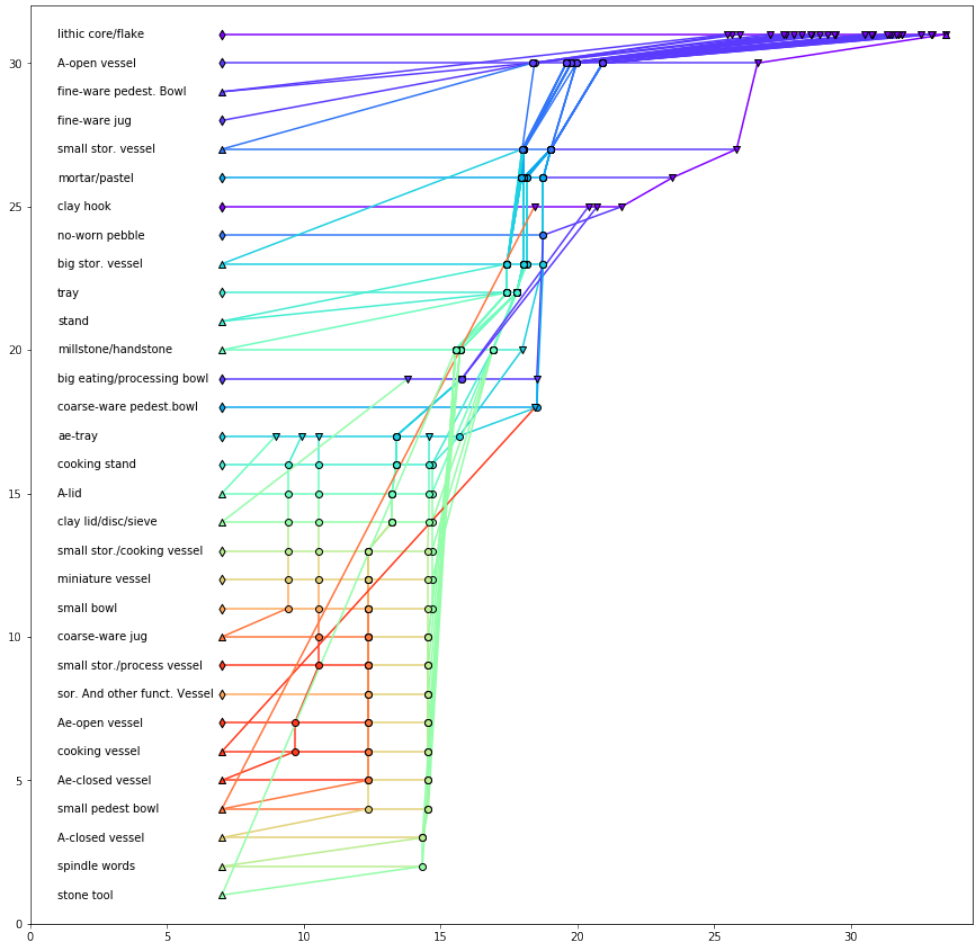

Figure 2: Same as Figure 1 with height equal to the diameter of the cluster

structures allows us to:

- Find representatives of the clusters according to the center of the truncated balls of the clusters (graphically represented by the top edge element).
- Find an order between the elements.
- Organize the data along horizontal (succession for a same element) or vertical (cluster chain) lines.

7 Conclusion

We have shown in this paper that totally balanced structures can be used in clustering through a special kind of dissimilarities. These structures allow an efficient approximation scheme and a convenient graphical representation.

Finally, this work shows a new class of dissimilarities, chordal dissimilarities, which is a good model for classification as they admit only few clusters that can be easily computed and interpreted. We will further study these dissimilarities as they in order to determine more of their properties (structural bijections or graphical representations for instance).

8 References

Alberti G. (2013) Making Sense of Contingency Tables in Archaeology: the Aid of Correspondence Analysis to Intra-Site Activity Areas Research, Journal of Data Science, 11, 501-536.

Anstee R.P. (1983) Hypergraphs with no special cycles, Combinatorica, 3, 141-146.

Antsee R.P. and Farber M (1984), Characterizations of totally balanced matrices. Journal of Algorithms, 5, 215-230 (1984).

Bandelt, H.-J., Dress, A. W. M. (1989), "Weak Hierarchies Associated with Similarity Measures – an Additive Clustering Technique", Bulletin of Mathematical Biology, *51*, 133–166.

Barthélemy and Brucker (2008), Binary Clustering. Journal of Discrete Applied Mathematics, 156, 1237-1250.

Bertrand, P. (2000), "Set Systems and Dissimilarities", European Journal of Combinatorics, 21, 727 – 743.

Bertrand, P. and Diatta, J. (2014), Weak Hierarchies: A Central Clustering Structure Clusters, Orders, And Trees: Methods and Applications, eds, F. Aleskerov, B. Goldengorin and P. M. Pardalo , Berlin: Springer-Verlag, chapter 14.

Brucker, F. (2005), From hypertrees to Arboreal Quasi-ultrametrics, Discrete Applied Mathematics, 147, 3–26.

Brucker, F. and Gély, A. (2009), Parsimonious cluster systems, Advances in Data Analysis and Classification, 3, 189–204.

Brucker, F. and Gély, A. (2010), Crown-free Lattices and Their Related Graphs" Order, 28:443–454.

Brucker, F. and Préa. P. (2015), Totally Balanced Formal Concept Representations, Proceedings of ICFCA 215, 169-182.

Diatta, J., and Fichet, B. (1994), From Asprejan Hierarchies and Bandelt-Dress Weak-hierarchies to Quasi-hierarchies, *New Approaches in Classification and Data Analysis*, Eds., E. Diday,Y. Lechevallier, M. Schader, P. Bertrand, and B. Burtschy, Berlin: Springer-Verlag, 111–118.

Dirac, G.A. (1961), On rigid circuit graphs, Abh. Math. Sem. Univ. Hamburg, 25 (1961) 71-76.

Farber, M. (1983), Characterizations of strongly chordal graphs, Discrete Mathematics, 43, 173-189.
Lehel, J. (1985), A Characterization of Totally Balanced Hypergraphs, Discrete Mathematics, 57, 59-65.
Lovasz L. (1968), Graphs and set systems, *Beiträge zur Graphentheorie*, Ed. H. Sachs et al. Teubner, Leipzig, 99-106.
Lubiw, A. (1987), Doubly lexical Orderings of Matrices, SIAM J. on Computing, 16, 854-879.
Spinrad, J. (1993), Doubly lexical Ordering of Dense 0-1 Matrices, Information Processing Letters, 45, 229-235.
Spinrad, J. (2003) *Efficient Graph representations* American Mathematical Society, 2003.