Rendiconti Sem. Mat. Univ. Pol. Torino
Vol. 79, 2021 (2021), 55 – 76

C. Bocci and C. Capresi

REALIZATION OF DISTANCE MATRICES BY UNICYCLIC GRAPHS

Abstract. Given a distance matrix D, we study the behavior of its compaction vector and reduction matrix with respect to the problem of the realization of D by a weighted graph. To this end, we first give a general result on realization by $n-$cycles and successively we mainly focus on unicyclic graphs, presenting an algorithm which determines when a distance matrix is realizable by such a kind of graph, and then, shows how to construct it.

1. Introduction

Let D be a matrix whose rows and columns are indexed by a set X. We assume that D is symmetric and has zero entries on the main diagonal. In phylogenetics, this kind of matrices are called *dissimilarity matrices*. Usually, we take $X = [n] := \{1,2,\ldots,n\}$. Hence a dissimilarity matrix D can also be seen as a map $D : [n]^2 \to \mathbb{R}$, with $D(i, j) = D(j,i)$ and $D(i,i) = 0$ for each $i, j \in [n]$.

A *distance matrix* (sometimes also called *metric*) is a non-negative dissimilarity matrix which satisfies the triangle inequality $D(i, j) \leq D(i,k)+D(k, j)$ for all $i, j, k \in X$. In [7, 10], one can find some results about this kind of matrices.

We say that D has a *realization* if there is a weighted graph (so one where a non-negative weight is assigned to each edge) whose nodes set contains X and such that the distance (i.e. the length of the shortest path) between nodes $i, j \in X$ is exactly $D(i, j)$. The problem of realizing a distance matrix by a graph of minimal total weight is a difficult problem. A lot of authors have considered the special case where the graph is a tree and the distance matrix is a tree metric, also called an additive metric. This case has been studied intensively and is well understood. The main result is the so-called Tree Metric Theorem, (see [4] or [16, Theorem 2.36]), which is based on the *four-point condition*, which is a necessary and sufficient condition for a matrix to be realized by a tree.

Efficient algorithms constructing such trees were published in [8] or [18]. But real data describe merely a tree metric because they arise from a similarity measure that includes errors. Such a measure appears in various fields such as the study of evolution ([1, 14]), the synthesis of certain electrical circuits ([9]) or the traffic modelling ([6]). Thus, many authors address the problem of realization by a graph which is not a tree. For example in [17] and [18] the author characterizes, respectively, distance matrices which have a unicyclic graph as unique optimal realization and introduces an algorithm for finding optimal graph of non-tree realizable distance matrices. In [5] can be found results of realization for some particular classes of graphs: paths, caterpillar, cycles, bipartite graphs, complete graphs and planar graphs. The particular class of cactus metric is studied deeply in [11].

In this paper we consider the problem of realizing a distance matrix by a unicyclic graph, i.e. a graph containing only one cycle. This kind of graphs represents a first non-trivial case in the study of phylogenetic networks. See, for example, Figure 4 in [3], Figures 2 and 5 in [12], Figures 1 and 4 in [19].

Using and generalizing the definitions of compaction and reduction processes contained in [20] (and basing, especially compaction, on the results in [10]), we establish a recursive method to "contract" a distance matrix D, until we end with a matrix where contractions are no more possible. The analysis of this final matrix by Theorem 2, and Propositions 1 and 2, permits to establish if it is realizable, or not, by an $n-$cycle or a tree. Then, in case of positive answer, moving backward we are able to reconstruct the tree or the unicyclic graph which realizes D. Moreover, Theorem 2, combined with a condition on compaction values, furnishes a result on optimality of the realization; this is stated in Proposition 3. The case of unicyclic graphs can be seen as a particular case of cactus graphs, however we would like to point out that the approach of this paper is different with respect to [11]: the attempt, here, is to give an algorithm that gives an optimal realization of a distance matrix M, just working with repeated processes of compaction and reduction on it. Our interest in such approach is due to the fact that the formula in Theorem 2 can be written as a tropical polynomials and, considering also the compaction vector, we get another tropical polynomial (formula (15) in Section 6) which is the corner locus of a dissimilarity matrix which realizes a unicyclic graph with pendant edges. As explained in Section 6, we hope that this approach leads to the description of the moduli space of tropical curves of genus 1.

2. Preliminaries

We initially recall, from [20], the definitions of compaction and reduction processes.

Let D be a distance matrix of order $n \times n$ and consider the $n \times n-$matrices E_i where

$$(E_i)(j,k) = \begin{cases} 1 & \text{if } j = i \neq k \\ 1 & \text{if } j \neq i = k \\ 0 & \text{elsewhere} \end{cases}$$

Let $D_i(\alpha) = D - \alpha \cdot E_i$. The following result establishes for which values of α, $D_i(\alpha)$ is still a distance matrix.

THEOREM 1 ([10], Lemma 1). *$D_i(\alpha)$ is a distance matrix if and only if $\alpha \leq \frac{1}{2} \cdot (D(p,i) + D(i,r) - D(p,r))$, for all $p, r \neq i$.*

The new matrix $D_i(\alpha)$, obtained from D, is called the $i-$**th compaction matrix of** D **with respect to** α. The compaction of an index i of D leads to a new matrix with, possibly, equal rows and, by symmetry, equal columns (see [20]). By deleting all, but one, repeated rows and columns we obtain a new matrix which is called $i-$**th reduction matrix of** D **with respect to** α.

Here, we are interested in considering the maximal value α for which $D_i(\alpha)$ is still a distance matrix. Moreover, we want to work for each index i of compaction, with $i = 1,\dots,n$, and collect these data into a vector. This leads to the following

DEFINITION 1. *Given a distance matrix D of order $n \times n$, the* **compaction vector** *of D, $\mathbf{a}_D = (a_1,\dots,a_n)$, is defined as*

$$a_i = \frac{1}{2} \cdot \min_{p \neq i, r \neq i} \{D(p,i) + D(i,r) - D(p,r)\}.$$

Consider the tree in Figure 1. Its distance matrix is

$$D = \begin{pmatrix} 0 & 3 & 5 & 6 \\ 3 & 0 & 6 & 7 \\ 5 & 6 & 0 & 7 \\ 6 & 7 & 7 & 0 \end{pmatrix}. \tag{1}$$

The compaction vector of D is $\mathbf{a}_D = (1,2,3,4)$.

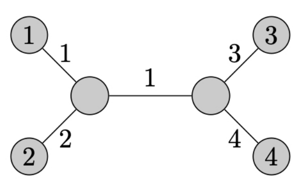

Figure 1: The tree of matrix in (1).

It is a well known fact that, in case D is obtained from a weighted graph and if i is a leaf, then there is a realization of D in which the entry a_i is the weight of the edge connecting the leaf i to its internal node ([20]). As a matter of fact, consider the graph in Figure 2(a). Its distance matrix is

$$D = \begin{pmatrix} 0 & 3 & 5 & 4 \\ 3 & 0 & 5 & 5 \\ 5 & 5 & 0 & 5 \\ 4 & 5 & 5 & 0 \end{pmatrix} \tag{2}$$

and its compaction vector is $\mathbf{a}_D = (1, \frac{3}{2}, \frac{5}{2}, 2)$. In this case, the entries of $\mathbf{a}_D$ do not correspond to the weights of the edges incident to the leaves. However, we can notice that the graph in Figure 2(b) has the same distance matrix, and hence the same compaction vector, of the graph in Figure 2(a) and, for this graph, the weights of the edges connecting the leaves to the internal nodes correspond to the entries of the compaction vector. Both these graphs are realizations of the matrix in (2).

Once the compaction vector $\mathbf{a}_D$ of D is computed, we consider the following matrix

$$D(\mathbf{a}_D) = D - a_1 \cdot E_1 - \cdots - a_n \cdot E_n \tag{3}$$

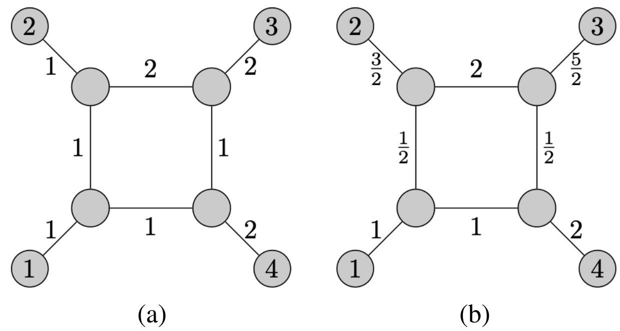

Figure 2: The graphs of matrix in (2).

DEFINITION 2. *The matrix* $D(\mathbf{a}_D)$ *in* (3) *is called the* **compaction matrix of** D.

DEFINITION 3. *Let* $D(\mathbf{a}_D)$ *be the compaction matrix of* D *as defined in* (3)*. The matrix* $\hat{D}_{\mathbf{a}_D}$ *obtained removing from* $D(\mathbf{a}_D)$ *all but one repeated rows and columns is called* **reduction matrix of** D*. If there are not equal rows (and columns) in* $D(\mathbf{a}_D)$*, then* $\hat{D}_{\mathbf{a}_D} = D(\mathbf{a}_D)$.

Notice that now we omit the expression "with respect to $\mathbf{a}_D$" and we do not refer to a specific index i, since we are considering the compaction process along all the indices and only with respect to the corresponding values of the compaction vector; while in the standard definition of compaction matrix the computation is taken only with respect to an index and a given value α ([20]). Thus process of reduction is different with respect to the one in [20], where only one of the equal rows (and columns) is removed.

EXAMPLE 1. Let D be the following distance matrix

$$D = \begin{pmatrix} 0 & 4 & 6 & 6 & 3 \\ 4 & 0 & 5 & 5 & 5 \\ 6 & 5 & 0 & 2 & 5 \\ 6 & 5 & 2 & 0 & 5 \\ 3 & 5 & 5 & 5 & 0 \end{pmatrix}$$

Its compaction vector is

$$\mathbf{a}_D = \left(1, \frac{3}{2}, 1, 1, 1\right)$$

from which we get

$$D(\mathbf{a}_D) = D - E_1 - \frac{3}{2}E_2 - E_3 - E_4 - E_5 = \begin{pmatrix} 0 & \frac{3}{2} & 4 & 4 & 1 \\ \frac{3}{2} & 0 & \frac{5}{2} & \frac{5}{2} & \frac{5}{2} \\ 4 & \frac{5}{2} & 0 & 0 & 3 \\ 4 & \frac{5}{2} & 0 & 0 & 3 \\ 1 & \frac{5}{2} & 3 & 3 & 0 \end{pmatrix}.$$

Notice that the third and fourth rows of $D(\mathbf{a}_D)$ are equal, so the reduction matrix is

$$\hat{D}_{\mathbf{a}_D} = \begin{pmatrix} 0 & \frac{3}{2} & 4 & 1 \\ \frac{3}{2} & 0 & \frac{5}{2} & \frac{5}{2} \\ 4 & \frac{5}{2} & 0 & 3 \\ 1 & \frac{5}{2} & 3 & 0 \end{pmatrix}.$$

We prove now two results concerning particular behavior of compaction and reduction matrices.

PROPOSITION 1. *Let D be a distance matrix of order $n \times n$, if its reduction matrix $\hat{D}_{\mathbf{a}_D}$ has order 2, then D is realizable by a tree.*

Proof. If $\hat{D}_{\mathbf{a}_D}$ has order 2, then

$$\hat{D}_{\mathbf{a}_D} = \begin{pmatrix} 0 & \delta \\ \delta & 0 \end{pmatrix} \tag{4}$$

for some $\delta \in \mathbb{R}_{>0}$.

Moreover, for reduction process, there exist positive integers n_1 and n_2, with $n_1 + n_2 = n$, such that $D(\mathbf{a}_D)$ has a set of n_1 equal rows (and correspondent columns) and a set of n_2 equal rows (and correspondent columns). For simplicity we assume that the first n_1 rows are equal, and hence the remaining n_2 rows are equal.

Then by (4) we get that $D(\mathbf{a}_D)$ has the form

$$\left(\begin{array}{c|c} \mathbf{0}_{n_1 \times n_1} & \delta \mathbf{1}_{n_1 \times n_2} \\ \hline \delta \mathbf{1}_{n_2 \times n_1} & \mathbf{0}_{n_2 \times n_2} \end{array}\right) \tag{5}$$

where $\mathbf{0}_{k \times l}$ denotes the null matrix of order $k \times l$ and $\delta \mathbf{1}_{k \times l}$ denote the matrix of order $k \times l$ whose entries are all equals to δ.

Let $\mathbf{a}_D = (a_1, \ldots, a_n)$ be the compaction vector of D, by definition of compaction matrix we know that

$$D(\mathbf{a}_D) = \begin{pmatrix} 0 & D(1,2) - a_1 - a_2 & \cdots & D(1,n) - a_1 - a_n \\ D(2,1) - a_1 - a_2 & 0 & \cdots & D(2,n) - a_2 - a_n \\ \vdots & \vdots & \ddots & \vdots \\ D(n,1) - a_1 - a_n & D(n,2) - a_2 - a_n & \cdots & 0 \end{pmatrix} \tag{6}$$

Comparing the entries of the matrices in (5) and (6) one has

$$D(i,j) = \begin{cases} 0 & \text{if } i = j \\ a_i + a_j & \text{if } 1 \le i < j \le n_1 \text{ or } n_1 + 1 \le i < j \le n_1 + n_2 \\ \delta + a_i + a_j & \text{otherwise} \end{cases}$$

from which follows that D has a realization by the tree in Figure 3. □

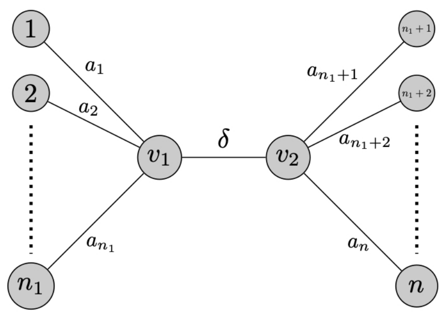

Figure 3: Realization for Proposition 1.

EXAMPLE 2. Let D be the distance matrix

$$D = \begin{pmatrix} 0 & 2 & 3 & 3 \\ 2 & 0 & 3 & 3 \\ 3 & 3 & 0 & 2 \\ 3 & 3 & 2 & 0 \end{pmatrix}$$

From its compaction vector

$$\mathbf{a}_D = (1,1,1,1)$$

we get

$$D(\mathbf{a}_D) = D - E_1 - E_2 - 3E_3 - E_4 = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}.$$

In this case $D(\mathbf{a}_D)$ has two pairs of equal rows and, after the reduction process, we get

$$\hat{D}_{\mathbf{a}_D} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Then, by Proposition 1, D has a realization by the tree in Figure 4.

The second result concerns the compaction matrix.

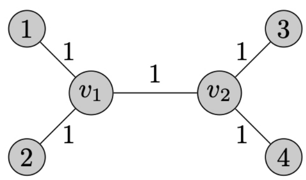

Figure 4: Tree which realizes the matrix of Example 2.

PROPOSITION 2. *Let D be a distance matrix of order $n \times n$, if its compaction matrix $D(\mathbf{a}_D)$ is the null matrix, then D is realizable by a star tree with n leaves (and a unique internal node of valency n).*

Proof. Let $\mathbf{a}_D = (a_1, \dots, a_n)$ be the compaction vector of D. By definition of compaction matrix we know that

$$D(\mathbf{a}_D) = \begin{pmatrix} 0 & D(1,2) - a_1 - a_2 & \cdots & D(1,n) - a_1 - a_n \\ D(2,1) - a_1 - a_2 & 0 & \cdots & D(2,n) - a_2 - a_n \\ \vdots & \vdots & \ddots & \vdots \\ D(n,1) - a_1 - a_n & D(n,2) - a_2 - a_n & \cdots & 0 \end{pmatrix}$$

Since by hypothesis, $D(\mathbf{a}_D)$ is the null matrix, one has

$$D(i,j) = \begin{cases} a_i + a_j & \text{if } i \neq j \\ 0 & \text{otherwise} \end{cases}$$

from which easily follows that D has a realization by the star tree in Figure 5. $\square$

EXAMPLE 3. Let D be the distance matrix

$$D = \begin{pmatrix} 0 & 3 & 4 & 5 \\ 3 & 0 & 5 & 6 \\ 4 & 5 & 0 & 7 \\ 5 & 6 & 6 & 0 \end{pmatrix}$$

From its compaction vector

$$\mathbf{a}_D = (1,2,3,4)$$

we get

$$D(\mathbf{a}_D) = D - E_1 - 2E_2 - 3E_3 - 4E_4 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Then, by Proposition 2, D has a realization by the tree in Figure 6.

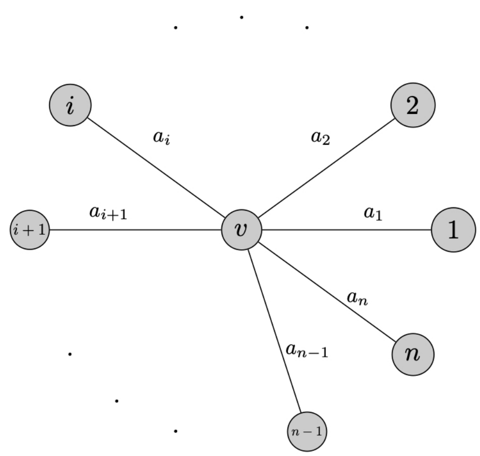

Figure 5: Realization for Proposition 2.

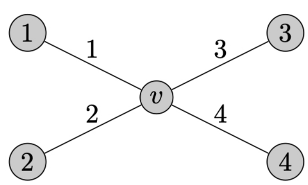

Figure 6: Star tree which realizes the matrix of Example 3.

3. Realizations by cycles

In this section we give a useful result for matrices which are realizable by cycles. While in other research papers the order of the vertices is given a priori, or the vertices are eventually re-labelled after some algorithm (see, for example, Definition 4.2 in [5]), the following theorem gives a more general characterization of such matrices also if the order of vertices is not known.

Let $\pi = (i_1, i_2, \dots, i_n)$ be a permutation on $[n]$. We denote by $\pi^s(i)$ the image of i after applying s time the permutation π.

We recall that a permutation π of $[n]$ is **real** if it has only one term in the cycle-notation. For example $\pi_1 = (2,3,4,5,1)$ is real, while $\pi_2 = (2,1,4,5,3)$ is not since it can be written as $(2,1)(4,5,3)$.

THEOREM 2. *A distance matrix D, of order n, has a realization by an $n-$cycle*

C if and only if there exists a real permutation π of $[n]$, such that

$$D(i,\pi^s(i)) = \min\left\{\sum_{t=0}^{s-1} D(\pi^t(i),\pi^{t+1}(i)), \sum_{t=s}^{n-1} D(\pi^t(i),\pi^{t+1}(i))\right\} \tag{7}$$

for all $i = 1,...,n$ and $s = 1,...,n-1$.

Proof. If D has a realization by an $n-$cycle C, let $i_1, i_2, \dots, i_n$, with $i_1 = 1$, be the labels of the nodes taken in clockwise order. Then $\pi = (i_2, \dots, i_n, 1)$ is a real permutation and since, on an $n-$cycle, the distance between nodes i and $\pi^s(i)$ is the minimum of the two paths from i to $\pi^s(i)$, equations (7) are satisfied, for all $i = 1,...,n$ and $s = 1,...,n-1$.

Vice versa, suppose there exists a real permutation π for which D satisfies equation (7), for all $i = 1,...,n$ and $s = 1,...,n-1$. Then let C be the graph of vertices $V = \{1,2,\dots,n\}$ and edges

$$E = \{(\pi^i(1),\pi^{i+1}(1)),\, i = 0,\dots,n-1\}.$$

It is easy to check that C is an $n-$cycle. Then, putting the weight $D(\pi^i(1),\pi^{i+1}(1))$ on the edge $(\pi^i(1),\pi^{i+1}(1))$, for all $i = 0,\dots,n-1$, we get a weighted $n-$cycle that, by equation (7), is a realization of D. □

According to the fact that, on an $n-$cycle, the distance between nodes i and j is the minimum of the two paths from i to j, the reader could be surprised of formula (7) instead of the following formula

$$\begin{aligned} D(i,j) = \min\{&D(i,i+1)+D(i+1,i+2)+\cdots+D(j-1,j),\\ &D(i,i-1)+D(i-1,i-2)+\cdots+D(j+1,j)\}\end{aligned} \tag{8}$$

where the indices are taken modulo n. Notice that (8) expresses the condition of a path for an $n-$cycle where the indices are labeled in clockwise order.

However in general, it is not true that if a distance matrix D, of order $n \times n$, has a realization by an $n-$cycle C, then the adjacent vertices of C correspond to adjacent rows (or columns) of D.

For example, if we consider the distance matrix

$$D = \begin{pmatrix} 0 & 3 & 4 & 1 & 3 & 2\\ 3 & 0 & 2 & 4 & 3 & 1\\ 4 & 2 & 0 & 3 & 1 & 3\\ 1 & 4 & 3 & 0 & 2 & 3\\ 3 & 3 & 1 & 2 & 0 & 4\\ 2 & 1 & 3 & 3 & 4 & 0 \end{pmatrix} \tag{9}$$

we can notice that (8) does not work for some choice of i and j. For example

$$\begin{aligned} 3 = D(1,5) &\neq \min\{D(1,2)+D(2,3)+D(3,4)+D(4,5), D(1,6)+D(6,5)\}\\ &= \min\{3+2+3+2, 2+4\}\\ &= \min\{10,6\} = 6 \end{aligned} \tag{10}$$

However, D has a realization by the $6-$cycle in Figure 7. Here we can notice that the vertices are ordered following the permutation $\pi = (4,5,3,2,6,1)$. Since $\pi^2(1) = 5$, we can compute again (10) using (7), obtaining

$$\begin{aligned}
3 = D(1,5) = D(1,\pi^2(1)) = \\
&= \min\left\{\sum_{t=0}^{1} D(\pi^t(i),\pi^{t+1}(i)), \sum_{t=2}^{5} D(\pi^t(i),\pi^{t+1}(i))\right\} \\
&= \min\{D(1,4)+D(4,5), D(5,3)+D(3,2)+D(2,6)+D(6,1)\} \\
&= \min\{1+2, 1+2+1+2\} \\
&= \min\{3,6\} = 3.
\end{aligned}$$

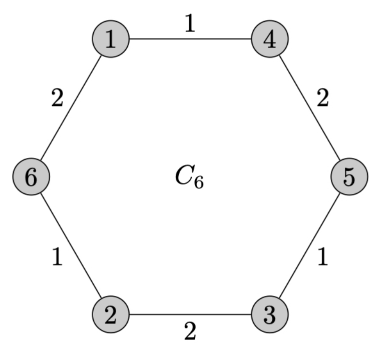

Figure 7: Realization of the matrix in (9).

REMARK 1. If $s = 1$, then formula (7) become

$$D(i,\pi(i)) = \min\left\{D(i,\pi(i)), \sum_{t=1}^{n-1} D(\pi^t(i),\pi^{t+1}(i))\right\}$$

which is trivially satisfied. The same happens when $s = n-1$. Hence these two cases can be omitted in formula (7).

A realization of a metric D is minimal if the removal of an arbitrary edge yields a graph that does not realize D. Given a metric D on X with $|X| \geq 4$, we say that D is *cyclelike* if there is a minimal realization for D that is a cycle. This type of metric was discussed in, e.g., [5] , [13], [17]. In [5], the authors characterize cyclelike metrics according to equation (8), after performing an algorithm that re-enumerate rows and columns of the metric.

We want to mention a useful result from [13].

THEOREM 3 ([13], Theorem 4.4). *Suppose D is a cyclelike metric on a finite set X and a cycle C is a minimal realization of D with $V(C) = X = \{v_1, v_2, \ldots, v_m\}$, $m \geq 4$ and $E(C) = \{\{v_i, v_{i+1}\} : 1 \leq i \leq m\}$, where the indices are taken modulo m. Then, C is an optimal realization of D if and only if*

$$D(v_{i-1}, v_i) + D(v_i, v_{i+1}) = D(v_{i-1}, v_{i+1}) \tag{11}$$

holds for all i. In this case, C is the unique optimal realization of D.

It is not true, in general, that a metric D satisfying the conditions of Theorem 2, must satisfy also equations (11), where the order of the indices is given by the permutation π. However, this becomes true when we add an extra condition on D, thus we have the following result.

PROPOSITION 3. *Let D be a distance matrix such that $\mathbf{a}_D$ is the null vector. If D satisfies the conditions of Theorem 2, then the realization of D by a cycle C is optimal.*

Proof. For simplicity of notation, we assume that D satisfies equations (7) with respect to the permutation $\pi = (2, 3, \ldots, n-1, n, 1)$.

Suppose that the realization C is not optimal, then, according to Theorem 3 there are vertices v_{i-1}, v_i, v_{i+1} such that

$$D(v_{i-1}, v_{i+1}) < D(v_{i-1}, v_i) + D(v_i, v_{i+1}). \tag{12}$$

Since D satisfies the conditions of Theorem 2 we must have

$$D(v_{i-1}, v_{i+1}) = D(v_{i+1}, v_{i+2}) + D(v_{i+2}, v_{i+3}) + \cdots + D(v_{i-2}, v_{i-1}),$$

where the indices are taken modulo n. Let us add an edge between nodes v_{i-1} and v_{i+1} of weight $W := D(v_{i+1}, v_{i+2}) + D(v_{i+2}, v_{i+3}) + \cdots + D(v_{i-2}, v_{i-1})$. We can then apply an elementary cycle reduction (see [10]) obtaining a new node w and the following set of weighted edges

- the edge (v_{i-1}, w) of weight $\frac{1}{2}\left(D(v_{i-1}, v_i) + W - D(v_i, v_{i+1})\right)$
- the edge (v_i, w) of weight $\frac{1}{2}\left(D(v_{i-1}, v_i) + D(v_i, v_{i+1}) - W\right)$
- the edge (v_{i+1}, w) of weight $\frac{1}{2}\left(D(v_{i+1}, v_i) + W - D(v_{i-1}, v_i)\right)$

By (12), the edge (v_i, w) has strictly positive weight and this number is exactly the $i-$th entry of the compaction vector $\mathbf{a}_D$, which is a contradiction. □

The application of Theorem 2 requires, in principle, to check all possible $n!$ permutations, but this is intractable also for n of moderate size. However, the following greedy algorithm permits to find an expected permutation π in $O(n^2)$ time.

INPUT: a distance matrix D of order n

(1) set $\pi_n = 1$ and $L = \{1\}$

(2) choose the two minimal entries $D(1,i)$ and $D(1,j)$ in the first row (if more, choose two of them);

(3) set $\pi_2 = i$, $\pi_{n-1} = j$ and $L = \{1,i,j\}$

(4) for s from 2 to $n-2$ do

(4.1) choose minimal entry $D(\pi_s,k)$ in π_s-th row with $k \notin L$ (if more, choose one of them);

(4.2) set $\pi_{s+1} = k$ and $L := L \cup \{k\}$

(5) if $D(\pi_{n-1}, j)$ is one of the two minimal entries in π_{n-1}-th row, then π is the expected permutation

4. Algorithm of realization of a matrix by a graph of genus 1

The idea of the algorithm is to iterate the processes of compaction and reduction, obtaining at every step i a new matrix $D(i)$. The algorithm will stop when, at a certain step t, we will be in one of the following cases

(1) the matrix $D(t)$ has order 2;

(2) the compaction matrix $D(\mathbf{a}_{D(t)})$ is the null matrix;

(3) the compaction vector $\mathbf{a}_{D(t)}$ is the null vector and neither cases (1) or (2) are verified.

If we have kept track of all compaction vectors $\mathbf{a}_{D(i)}$ and of all the rows and columns eliminated in the matrices $D(\mathbf{a}_{D(i)})$, by going backwards, we can construct the graph realizing the starting matrix D, from the graph $G(t)$ which realizes the last matrix $D(t)$, adding at every step $t-i$, with $1 \le i \le t$, new distinct nodes with edges adjacent to the graph G_{t-i+1}.

From Propositions 1 and 2 we know that in cases (1) and (2), the matrix D_t is realizable by a tree and hence, as we have just said, we will get that D is realizable by a tree. Finally, if we are in case (3) and we verify, by Theorem 2, that $D(t)$ is realizable by an $m-$cycle, with $m \le n$ then the matrix D can be realized by a unicyclic graph.

We divide the algorithm into two parts: a first part of analysis, which determines if or not a distance matrix can be realized by a unicyclic graph and then, a part of reconstruction which, using data from the analysis part, constructs the desired unicyclic graph.

Algorithm

INPUT: a distance matrix D of order n.

Analysis:

Step 0: Let $t=0$, $D(t)=D$, $\Theta=\{1,2,\dots,n\}$ and $\rho=n$.

Step 1: Compute the compaction vector $\mathbf{a}_{D(t)}$ of $D(t)$.
If $\mathbf{a}_{D(t)}$ is the null vector go to **Step 4**.

Step 2: Compute the compaction matrix $D(\mathbf{a}_{D(t)})$ of $D(t)$. If $D(\mathbf{a}_D(t))$ is the zero matrix, then, by Proposition 2, $D(t)$ is realizable by a tree $G(V,E)$ and go to **Step 5**.

Step 3: Let

$$S(t)=\Big\{\{i_{1,1}(t),i_{1,2}(t),\dots,i_{1,s_1(t)}(t)\},\dots,\{i_{\theta(t),1}(t),i_{\theta(t),2}(t),\dots,i_{\theta(t),s_\theta(t)}(t)\}\Big\}$$

be the collection of all distinct (ordered) subset of Θ of indices of equal rows of $D(\mathbf{a}_{D(t)})$, with $|S(t)|=\theta(t)$, and let

$$S'(t)=\{j_1(t),j_2(t),\dots,j_{\sigma(t)}(t)\}$$

be the set of (ordered) indices of Θ not in any subset of $S(t)$, with $|S'(t)|=\sigma(t)$.

For k from 1 to $\theta(t)$ do

Step 3.1: Remove from $D(t)$ all rows and columns indexed by

$$i_{k,2}(t),i_{k,3}(t),\dots i_{k,s_k(t)}(t)$$

Step 3.2: Relabel $i_{k,1}(t)$-th row and column of $D(t)$ by $\rho+k$

Step 3.3: Set

$$\Theta:=\Big(\Theta\cup\{\rho+k\}\Big)\setminus\{i_{k,1}(t),i_{k,2}(t),\dots,i_{k,s_k}(t)\}.$$

For k from 1 to $\sigma(t)$ do

Step 3.4: Relabel $j_k(t)$-th row and column of $D(t)$ by $\rho+\theta(t)+k$

Step 3.5: Set

$$\Theta:=\Big(\Theta\cup\{\rho+\theta(t)+k\}\Big)\setminus\{j_k(t)\}.$$

We get the reduction matrix $\hat{D}(\mathbf{a}_{D(t)})$ with a new labelling of rows and columns.

If $\hat{D}(\mathbf{a}_{D(t)})$ has order 2 then, by Proposition 1, $D(t)$ is realizable by a tree $G(V,E)$ and go to **Step 5**.

Otherwise set

$$\rho := \rho + \theta(t) + \sigma(t),$$

$$t := t + 1,$$

$$D(t) := \hat{D}(\mathbf{a}_{D(t-1)})$$

and return to **Step 1**.

Step 4: Check if $D(t)$ is realizable by an $m-$cycle $G(V,E)$ with $m = |\Theta|$. If so, then D can be realized by a unicyclic graph oand go to **Step 5**.

Otherwise, D has not a realization by a tree or a unicyclic graph then EXIT.

Reconstruction:

Step 5 For τ from $t-1$ to 0 do

Set $\rho := \rho - \theta(\tau) - \sigma(\tau)$

Step 5.1: for κ from 1 to $\sigma(\tau)$ do

If $(\mathbf{a}_{D(\tau)})_{j_\kappa(\tau)} \neq 0$ then

add node: $V := V \cup \{j_\kappa(\tau)\}$

add weighted edge: $E := E \cup \{(\rho + \theta(\tau) + \kappa, j_\kappa(\tau), (\mathbf{a}_{D(\tau)})_{j_\kappa(\tau)})\}$

Else

replace node: $V := (V \setminus \{\rho + \theta(\tau) + k\}) \cup \{j_\kappa(\tau)\}$

Step 5.2: for κ from 1 to $\theta(\tau)$ do

Add nodes: $V := V \cup \{i_{\kappa,1}(\tau), i_{\kappa,2}(\tau), \ldots, i_{\kappa,s_\kappa(\tau)}(\tau)\}$

Add weighted edges:

$$E := E \cup \{(\rho + \kappa, i_{\kappa,1}(\tau), (\mathbf{a}_{D(\tau)})_{i_{\kappa,1}(\tau)}), \ldots, (\rho + \kappa, i_{\kappa,s_\kappa}(\tau), (\mathbf{a}_{D(\tau)})_{i_{\kappa,s_\kappa(\tau)}(\tau)})\}$$

OUTPUT: the graph G realizing D.

It is easy to observe that, in the algorithm, the equal rows in a compaction matrix will correspond to nodes which are adjacent to the same interior node.

The algorithm has been implemented in `Maple`™. Here, the stored data of compaction vectors and reduction matrices are used to build the weighted adjacency matrix of the graph which realizes the distance matrix D. The algorithm can be found as ancillary file `algorithm.mw`, in the arXiv version of this paper ([2]).

The study of recursive compaction vectors is equivalent to removing all pendant trees in the optimal realization. In such a procedure, the number of operations grows in a time complexity of $O(n^4)$, where n is the order of the distance matrix D. As a matter of fact, computing an entry of the compaction vector is done in a time complexity of $O(n^2)$. This must be done for the n entries of the compaction vector. Moreover, we need to compute the compaction vector for the successive steps of the algorithm. The number of steps is at most n.

We conclude this section with a result about optimality of this realization.

PROPOSITION 4. *If the algorithm outputs a unicyclic graph realizing D, then this realization is optimal.*

Proof. In the analysis part, the algorithm checks if the matrix $D(t)$ satisfies equations (7) at Step 4, which is reached if the compaction vector of $D(t)$ is the null vector. Hence, by Proposition 3, if $D(t)$ has a realization by a cycle C, this realization is optimal.

Successively, the reconstruction part of the algorithm reconstructs the graph G realizing D, starting from C and adding, recursively, pendant edges. Hence, by the following theorem, the realization of D is optimal. $\square$

THEOREM 4 ([10], Theorem 5). *If $0 \leq a \leq a_i$ and if $G_i(a)$ is an optimal realization of $D_i(a)$, then G obtained from $G_i(a)$ adding the vertex v_i' to $G_i(a)$ and the edge (v_i', v_i) of weight a, is an optimal realization of D.*

5. A computational example

The following example shows how the algorithm works.

Consider the following matrix

$$D = \begin{array}{c} \\ \mathbf{1}\\ \mathbf{2}\\ \mathbf{3}\\ \mathbf{4}\\ \mathbf{5}\\ \mathbf{6}\\ \mathbf{7}\\ \mathbf{8}\\ \mathbf{9} \end{array} \begin{array}{c} \begin{array}{ccccccccc} \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} & \mathbf{7} & \mathbf{8} & \mathbf{9} \end{array} \\ \begin{pmatrix} 0 & 4 & 2 & 6 & 4 & 6 & 2 & 2 & 5 \\ 4 & 0 & 4 & 4 & 4 & 4 & 4 & 4 & 4 \\ 2 & 4 & 0 & 6 & 4 & 6 & 2 & 2 & 5 \\ 6 & 4 & 6 & 0 & 5 & 2 & 6 & 6 & 4 \\ 4 & 4 & 4 & 5 & 0 & 5 & 4 & 4 & 3 \\ 6 & 4 & 6 & 2 & 5 & 0 & 6 & 6 & 4 \\ 2 & 4 & 2 & 6 & 4 & 6 & 0 & 2 & 5 \\ 2 & 4 & 2 & 6 & 4 & 6 & 2 & 0 & 5 \\ 5 & 4 & 5 & 4 & 3 & 4 & 5 & 5 & 0 \end{pmatrix} \end{array} \tag{13}$$

We set $t = 0$, $D(0) = D$, $\Theta = \{1,2,3,4,5,6,7,8,9\}$ and $\rho = 9$. The compaction vector of $D(0)$ is $\mathbf{a}_{D(0)} = (1,1,1,1,1,1,1,1,1)$. Since $\mathbf{a}_{D(0)}$ is not the zero vector we pass to

Step 2 computing the compaction matrix

$$D(\mathbf{a}_{D(0)}) = \begin{array}{c} \\ \mathbf{1} \\ \mathbf{2} \\ \mathbf{3} \\ \mathbf{4} \\ \mathbf{5} \\ \mathbf{6} \\ \mathbf{7} \\ \mathbf{8} \\ \mathbf{9} \end{array} \begin{array}{c} \begin{array}{ccccccccc} \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} & \mathbf{7} & \mathbf{8} & \mathbf{9} \end{array} \\ \begin{pmatrix} 0 & 2 & 0 & 4 & 2 & 4 & 0 & 0 & 3 \\ 2 & 0 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 0 & 2 & 0 & 4 & 2 & 4 & 0 & 0 & 3 \\ 4 & 2 & 4 & 0 & 3 & 0 & 4 & 4 & 2 \\ 2 & 2 & 2 & 3 & 0 & 3 & 2 & 2 & 1 \\ 4 & 2 & 4 & 0 & 3 & 0 & 4 & 4 & 2 \\ 0 & 2 & 0 & 4 & 2 & 4 & 0 & 0 & 3 \\ 0 & 2 & 0 & 4 & 2 & 4 & 0 & 0 & 3 \\ 3 & 2 & 3 & 2 & 1 & 2 & 3 & 3 & 0 \end{pmatrix} \end{array}$$

Since $D(\mathbf{a}_{D(0)})$ is not the zero matrix we pass to **Step 3**. Here we have

$$S(0) = \Big\{\{1,3,7,8\},\{4,6\}\Big\} \quad S'(0) = \{2,5,9\},$$
$$\theta(0) = 2 \qquad \sigma(0) = 3.$$

Hence, we remove rows (and columns) labeled by 3, 7, 8 and 6 and relabel

- rows (and columns) 1 and 4 respectively by $10 (= \rho+1)$ and $11 (= \rho+2 = \rho+\theta(0))$,
- rows (and columns) 2,5 and 9 respectively by $12 (= \rho+\theta(0)+1)$, 13 and $14 (= \rho+\theta(0)+3 = \rho+\theta(0)+\sigma(0))$.

We get

$$\Theta(t) = \{10,11,12,13,14\}$$

and

$$\hat{D}(\mathbf{a}_{D(0)}) = \begin{array}{c} \\ \mathbf{10} \\ \mathbf{12} \\ \mathbf{11} \\ \mathbf{13} \\ \mathbf{14} \end{array} \begin{array}{c} \begin{array}{ccccc} \mathbf{10} & \mathbf{12} & \mathbf{11} & \mathbf{13} & \mathbf{14} \end{array} \\ \begin{pmatrix} 0 & 2 & 4 & 2 & 3 \\ 2 & 0 & 2 & 2 & 2 \\ 4 & 2 & 0 & 3 & 2 \\ 2 & 2 & 3 & 0 & 1 \\ 3 & 2 & 2 & 1 & 0 \end{pmatrix} \end{array}$$

Since $\hat{D}(\mathbf{a}_{D(0)})$ has not order 2, we come back to **Step 1** after setting $t = 1$, $\rho = 14$ and $D(1) = \hat{D}(\mathbf{a}_{D(0)})$.

The compaction vector of $D(1)$ is $\mathbf{a}_{D(1)} = (1,0,1,0,0)$ and again we pass to **Step 2** computing the compaction matrix

$$D(\mathbf{a}_{D(1)}) = \begin{array}{c} \\ \mathbf{10} \\ \mathbf{12} \\ \mathbf{11} \\ \mathbf{13} \\ \mathbf{14} \end{array} \begin{array}{c} \begin{array}{ccccc} \mathbf{10} & \mathbf{12} & \mathbf{11} & \mathbf{13} & \mathbf{14} \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 1 & 2 \\ 1 & 0 & 1 & 2 & 2 \\ 2 & 1 & 0 & 2 & 1 \\ 1 & 2 & 2 & 0 & 1 \\ 2 & 2 & 1 & 1 & 0 \end{pmatrix} \end{array}$$

This matrix has not equal rows so $\theta(1) = 0$ and $\sigma(1) = 5$ with

$$S'(1) = \{10, 12, 11, 13, 14\}.$$

After **Step 3.4** and **Step 3.5** we get $\Theta = \{15, 16, 17, 18, 19\}$ and

$$\hat{D}(\mathbf{a}_{D(1)}) = \begin{array}{c} \\ \mathbf{15} \\ \mathbf{16} \\ \mathbf{17} \\ \mathbf{18} \\ \mathbf{19} \end{array} \begin{array}{c} \begin{array}{ccccc} \mathbf{15} & \mathbf{16} & \mathbf{17} & \mathbf{18} & \mathbf{19} \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 1 & 2 \\ 1 & 0 & 1 & 2 & 2 \\ 2 & 1 & 0 & 2 & 1 \\ 1 & 2 & 2 & 0 & 1 \\ 2 & 2 & 1 & 1 & 0 \end{pmatrix} \end{array}$$

Since it has not order 2, we set $t = 2$, $\rho = 19$ and $D(2) = \hat{D}(\mathbf{a}_{D(1)})$.

Computing the compaction vector of $D(2)$ we get the null vector. Hence, in the analysis part of the algorithm we move to **Step 4**.

Notice that $D(2)$ satisfies Formula (7) of Theorem 2 for $\pi = (4, 5, 3, 2, 1)$ hence it is realizable by the 5-cycle G in Figure 8(a) and, moreover, D will be realizable by a unicyclic graph.

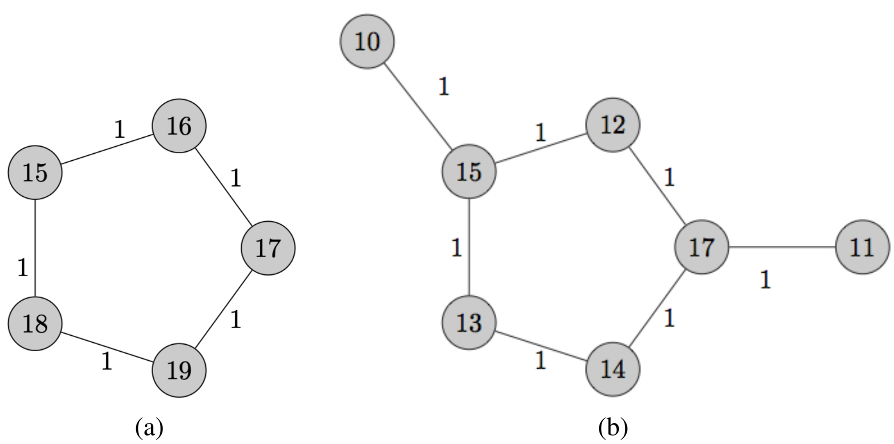

Figure 8: Intermediate steps for the realization of D in (13).

We start now the reconstruction part of the algorithm to determine such a graph. Since for $\tau = 1 (= t - 1)$ one has $\theta(1) = 0$ and $\sigma(1) = 5$ we set $\rho = 14$ and we perform only **Step 5.1**. More in details, one has that

- since $(\mathbf{a}_{D(1)})_{\mathbf{10}} = 1$ we add node $\{10\}$ and edge $(15, 10, 1)$.
- since $(\mathbf{a}_{D(1)})_{\mathbf{12}} = 0$ we replace node $\{16\}$ by node $\{12\}$.
- since $(\mathbf{a}_{D(1)})_{\mathbf{11}} = 1$ we add node $\{11\}$ and edge $(17, 11, 1)$.

- since $(\mathbf{a}_{D(1)})_{\mathbf{13}} = 0$ we replace node $\{18\}$ by node $\{13\}$.
- since $(\mathbf{a}_{D(1)})_{\mathbf{14}} = 0$ we replace node $\{19\}$ by node $\{14\}$.

Thus, we get the graph in Figure 8(b).

Now, we perform one more time **Step 5** for $\tau = 0$ (hence $\rho = 14 - \theta(0) - \sigma(0) = 9$). Performing **Step 5.1** one has that

- since $(\mathbf{a}_{D(0)})_{\mathbf{2}} = 1$ we add node $\{2\}$ and edge $(12,2,1)$.
- since $(\mathbf{a}_{D(0)})_{\mathbf{5}} = 1$ we add node $\{5\}$ and edge $(13,5,1)$.
- since $(\mathbf{a}_{D(0)})_{\mathbf{9}} = 1$ we add node $\{9\}$ and edge $(14,9,1)$.

Performing **Step 5.2** we add

- nodes $\{1\}$, $\{3\}$,$\{7\}$ and $\{8\}$ and edges $(10,1,1)$, $(10,3,1)$ $(10,7,1)$ and $(10,8,1)$
- nodes $\{4\}$ and $\{6\}$ and edges $(11,4,1)$ and $(11,6,1)$

This concludes the algorithm giving the unicyclic graph in Figure 9, which realizes D.

We end this section with some examples performed by our algorithm in the software `Maple`$^{\text{TM}}$. Here, starting from some weighted unicyclic graphs, we computed their distance matrices and we used them, as input, for the algorithm. The Figure 10 show the original graphs (on the left) and the outputs of the algorithm in `Maple`$^{\text{TM}}$ (on the right).

6. Future directions

In this paper we present an algorithm, running in time $O(n^4)$, such that, taken as input a distance matrix D, firstly returns if D is or not realizable by a tree or a unicyclic graph and, in case of affirmative answer, it reconstruct the graph itself. This algorithm concerns the use of the well-known processes of compaction and reduction, even though that of compaction has been slightly modified in such a way to be performed along all the indices of a given distance matrix D; this approach leads to the definition of the compaction vector of the matrix D itself.

The conditions given by Theorem 2, to check if a distance matrix has a realization by a $n-$cycle, suggests us an important direction of research. As a matter of fact, formula (7 of Theorem 2 means that $D(i,\pi^s(i))$ is equal to the minimum between $D(i,\pi(i))+D(\pi(i),\pi^2(i))+\cdots+D(\pi^{s-1}(i),\pi^s(i))$ and $D(\pi^s(i),\pi^{s+1}(i))+\cdots+D(\pi^{n-1}(i),i)$. Then the minimum is attained at least twice among the terms

$$\begin{gathered} D(i,\pi^s(i)), \\ D(i,\pi(i))+D(\pi(i),\pi^2(i))+\cdots+D(\pi^{s-1}(i),\pi^s(i)) \\ D(\pi^s(i),\pi^{s+1}(i))+\cdots+D(\pi^{n-1}(i),i) \end{gathered}$$

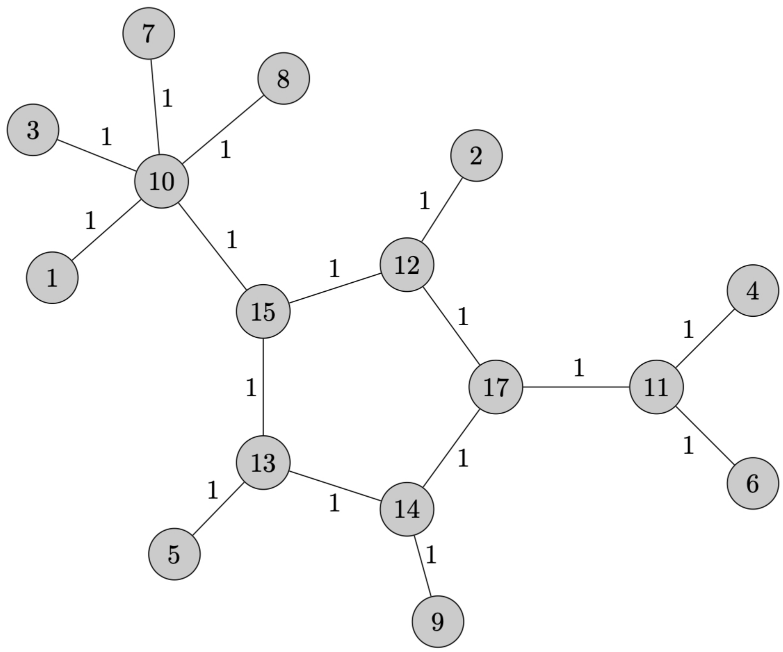

Figure 9: Unicyclic graph realizing the matrix D in (13).

In terms of Tropical Mathematics ([15]) this means that the entries of the distance matrix are a zero of the tropical polynomial

$$
\begin{aligned}
p_{is} := {} & D(i,\pi^s(i)) \oplus \\
& \oplus \big(D(i,\pi(i)) \otimes \cdots \otimes D(pi^{s-1}(i),\pi^s(i))\big) \oplus \\
& \oplus \big(D(\pi^s(i),\pi^{s+1}(i)) \otimes \cdots \otimes D(\pi^{n-1}(i),i)\big)\,.
\end{aligned} \tag{14}
$$

Hence the p_{is}'s are exactly the tropical equations for the space of weighted $n-$cycles.

The next step is that of understanding how to modify these equations to be satisfied by a distance matrix which has a realization by a unicyclic graph. Reaching this goal, will give us, not only a complete characterization, tropically speaking, of such distance matrices, but also the tropical equations for the moduli space $\mathscr{M}_1$ of elliptic tropical curves.

Some preliminary results, in this direction, show, for example that if D has a realization by an $n-$cycle C with only one pendant edge for each node of C, then its

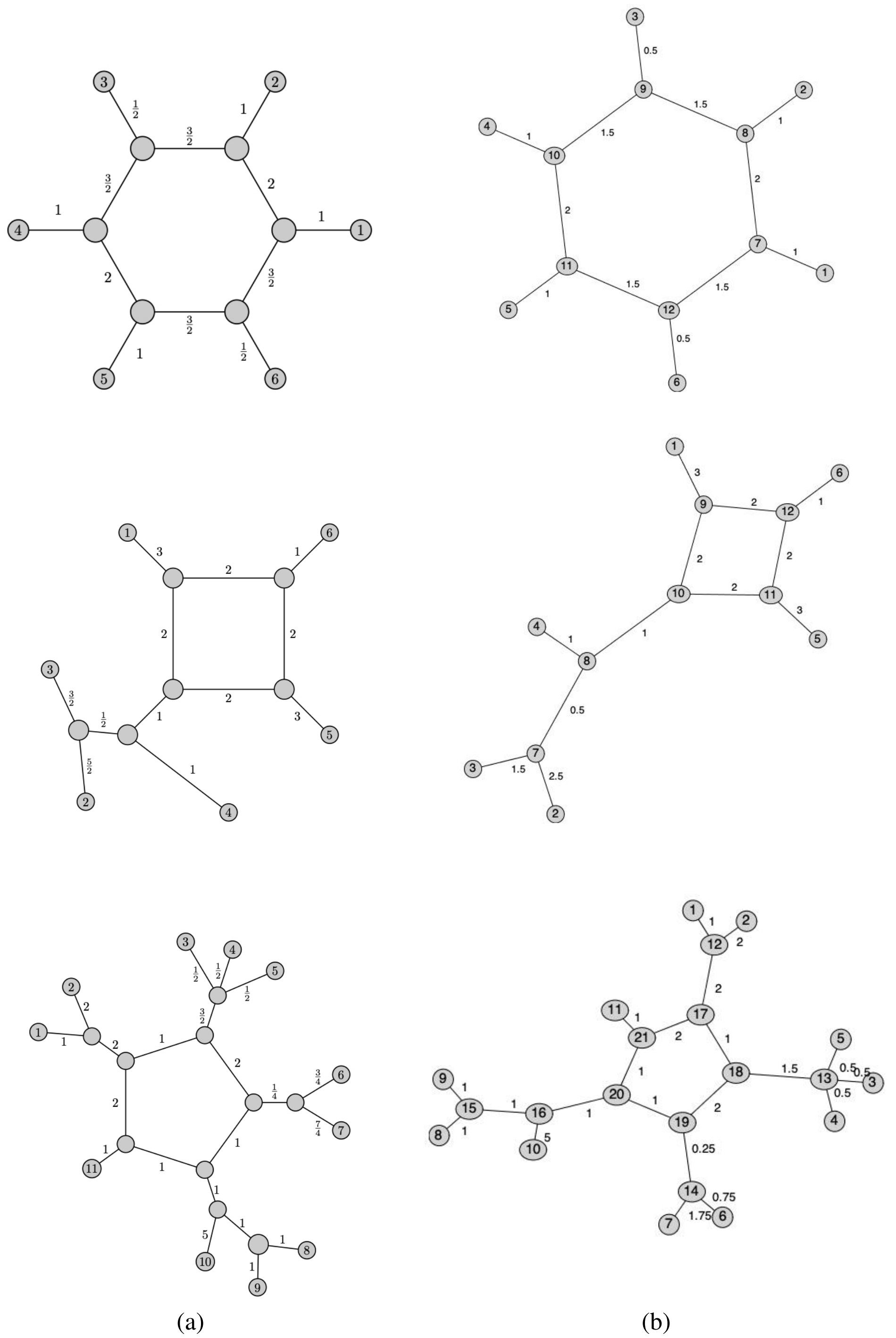

Figure 10: Examples in `Maple`™.

entries are a zero of the following kind of polynomials

$$
\begin{aligned}
\tilde{p}_{is} := & \left[\left(\bigotimes_{j \neq i, \pi^s(i)} 2a_j \right) \otimes D(i, \pi^s(i)) \right] \oplus \\
& \oplus \left[\left(\bigotimes_{j=\pi^{s+1}(i)}^{\pi^{n-1}(i)} 2a_j \right) \otimes D(i, \pi(i)) \otimes \cdots \otimes D(\pi^{s-1}(i), \pi^s(i)) \right] \oplus \\
& \oplus \left[\left(\bigotimes_{j=\pi^1(i)}^{\pi^{s-1}(i)} 2a_j \right) \otimes D(\pi^s(i), \pi^{s+1}(i)) \otimes \cdots \otimes D(\pi^{n-1}(i), i) \right]
\end{aligned} \tag{15}
$$

where the a_i are the entries of the compaction vector of D. Since these entries can be expressed in terms of the entries of D, it is easy to check that the polynomials $\tilde{p}_{is}$ are homogeneous, while the ones in (14) are not. Moreover, if D has a realization by an $n-$cycle then its entries are a zero also of this new set of polynomials, giving the hope to use them for a kind of "four point condition" for unicyclic graphs and then characterize $\mathscr{M}_1$.

References

[1] ATTESON K., *The performance of neighbor-joining algorithms of phylogeny reconstruction*, in: COCOON'97: Computing and Combinatorics, in Lectures Notes in Computer Science **1276** (1997), 101–110.

[2] BOCCI C. AND CAPRESI C., *Realization of distance matrices by graphs of genus 1*, arXiv:2012.07698

[3] BORDEWICH M., HUBER K.T., MOULTON V. AND SEMPLE C., *Recovering normal networks from shortest inter-taxa distance information*, Journal of Mathematical Biology **77** 3 (2018), 571–594.

[4] BUNEMAN P., *Note on the Metric Properties of Trees*, J. Combinatorial Theory **17** (1974), 48–50.

[5] BALDISSERRI A. AND RUBEI E., *Distance matrices of some positive-weighted graphs*, Australasian Journal of Combinatorics **70** 2 (2018), 195–201.

[6] BURTON D. AND TOINT P.L., *On the use of an inverse shortest paths algorithm for recovering linearly correlated costs*, Tech. rep., Facultés Universitaires ND de la Paix, 1997.

[7] CHUNG F., GARRETT M., GRAHAM R.L. AND SHALCROSS D., *Distance realization problems with applications to Internet tomography*, J. Computer Systems and Sciences **63** (2001), 432–448.

[8] Culberson J. C. and Rudnicki P., *A fast algorithm for constructing tree from distance matrices*, Inf. Process. Lett. **30** (1989), 215–220.

[9] Hakimi S.L. and Yau S.S., *Distance matrix and the synthesis of n-port resistance network*, Technical Report **5** (1963), Network Theory Group.

[10] Hakimi S.L. and Yau S.S., *Distance matrix of a graph and its realizability*, Quart. Appl. Math. **22** (1964), 305–317.

[11] Hayamizu M., Huber K.T., Moulton V. and Murakami Y., *Recognizing and realizing cactus metric*, Information Processing Letter **157**, 2020.

[12] Huber K.T., Moulton V., Semple C. and Wu T., *Quarnet inference rules for level-1 networks*, Bulletin of Mathematical Biology **80** 8 (2017), 2137–2153.

[13] Imrich W., Simões-Pereira J.M.S. and Zamfirescu C.M., *On Optimal embedding of metrics in graphs*, J. of Combinatorial Theory, Series B **36** 1 (1984), 1–15.

[14] Mau B. and Newton M.A., *Phylogenetic inference for binary data on dendograms using markov chain monte carlo*, Journal of computational and graphical statistics **6** 1 (1997), 122–131.

[15] Maclagan D. and Sturmfels B., *Introduction to tropical geometry*, Graduate Studies in Mathematics vol. 161, AMS, Providence, RI, 2015.

[16] L. Pachter and Sturmfels B., *Algebraic statistics for computational biology*, Cambridge University Press, New York, 2005.

[17] Simões-Pereira J.M.S., *A note on distance matrices with unicyclic graph realizations*, Discrete Mathematics **65** 3 (1987), 277–287.

[18] Simões-Pereira J.M.S., *An algorithm and its role in the study of optimal graph realizations of distance matrices*, Discrete Mathematics **79** (1989/90), 299–312.

[19] Van Iersel L. and Moulton V., *Encode tree-child and level-2 phylogenetic networks*, J. Math. Biol. **68** 7 (2014), 1707–1729.

[20] Varone S., *A constructive algorithm for realizing a distance matrix*, European Journal of Operational Research **174** 1 (2006), 102–111.

AMS Subject Classification: 05C12, 92D15, 68R05

Cristiano BOCCI, Chiara CAPRESI,
Department of Information Engineering and Mathematics, University of Siena
Via Roma 56, 53100, Siena, ITALY
e-mail: `cristiano.bocci@unisi.it`, `capresi3@student.unisi.it`

Lavoro pervenuto in redazione il 07.10.2021.