Discrete Mathematics 339 (2016) 2605–2612

Contents lists available at ScienceDirect

Discrete Mathematics

journal homepage: www.elsevier.com/locate/disc

On computational complexity of length embeddability of graphs

Mikhail Tikhomirov

Moscow Institute of Physics and Technology, 9 Institutskiy per., Dolgoprudny, Moscow Region, 141700, Russian Federation

ARTICLE INFO

Article history:
Received 21 October 2014
Received in revised form 11 May 2016
Accepted 11 May 2016
Available online 7 June 2016

Keywords:
Computational complexity
Graph embedding
Distance graph

ABSTRACT

A graph G is embeddable in $\mathbb{R}^d$ if the vertices of G can be assigned to points of $\mathbb{R}^d$ in such a way that all pairs of adjacent vertices are at distance 1. We show that verifying embeddability of a given graph in $\mathbb{R}^d$ is NP-hard in the case $d > 2$ for all reasonable notions of embeddability.

1. Introduction

The *unit-distance graph of* $S \subset \mathbb{R}^d$ is defined as the graph $G = (V, E)$, where $V = S$ and E is the set of all pairs of points $x, y \in S$ such that x and y are at distance 1. A graph is a *unit-distance graph in* $\mathbb{R}^d$ if it is isomorphic to the unit-distance graph of some set $S \subset \mathbb{R}^d$. Some famous problems concerning unit-distance graphs are the Erdős' unit distance problem on the maximal number of unit distances between n points in $\mathbb{R}^2$ (see [1,4,2]), the Hadwiger–Nelson problem on the chromatic number of $\mathbb{R}^2$ (see [1,3,9,15]), etc.; surveys of various results about unit-distance graphs can be found at [10,11].

We also consider a similar notion of *embeddability in* $\mathbb{R}^d$ (see, e.g., [14]). A graph $G = (V, E)$ is *embeddable in* $\mathbb{R}^d$ if there exists a mapping $\varphi : V \to \mathbb{R}^d$ such that $\|\varphi(u) - \varphi(v)\|_{\mathbb{R}^d} = 1$ for all $uv \in E$. It is clear that any unit-distance graph in $\mathbb{R}^d$ is embeddable in $\mathbb{R}^d$ but the converse does not always hold. These two notions differ in the following:

- Different vertices of an embeddable graph may be assigned to the same point in $\mathbb{R}^d$ while all vertices of a unit-distance graph should be assigned to pairwise distinct points.
- Non-adjacent vertices of an embeddable graph can be located at distance 1 while non-adjacent vertices of a unit-distance graph are forbidden to be placed at distance 1.

We will say that an embedding $\varphi : V \to \mathbb{R}^d$ is *strict* if all non-adjacent pairs of vertices are not placed at distance 1; we will say that an embedding $\varphi : V \to \mathbb{R}^d$ is *injective* if all vertices are mapped to pairwise distinct points of $\mathbb{R}^d$. It is clear that a graph G is a unit-distance graph in $\mathbb{R}^d$ iff there exists a strict and injective embedding of G in $\mathbb{R}^d$. Thus we obtain four different notions of embeddability (strict/non-strict, injective/non-injective) which include the two notions described above.

For each of the four notions of embeddability in $\mathbb{R}^d$ we can pose the computational decision problem of determining embeddability of chosen type for the given graph; we shall call this problem $\mathbb{R}^d$-**UNIT-DISTANCE-(STRICT)-(INJECTIVE)-EMBEDDABILITY** depending on the embeddability type. The computational complexity of these problems is studied in [14,7]. In [7] it is shown that $\mathbb{R}^d$-**UNIT-DISTANCE-(STRICT)-(INJECTIVE)-EMBEDDABILITY** is NP-hard for each type of

E-mail address: endagorion@gmail.com.

http://dx.doi.org/10.1016/j.disc.2016.05.011
0012-365X/

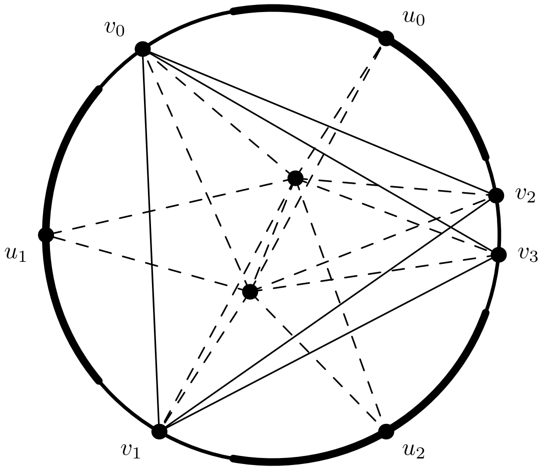

Fig. 1. An example of a reduction graph embedded in $\mathbb{R}^3$. Vertices of the auxiliary 1-simplex are in the center (not labeled), vertices of U and V_G are constrained to a circle. Bold parts of the circle are "forbidden" for vertices of V_G, three small non-bold arcs correspond to different colors of vertices of G. Chains constraining mutual position of vertices of V_G and U are not shown.

embeddability and each value of $d \geqslant 2$. Unfortunately, the proof in [7] for the case $d > 2$ is false as it is based on the result [6], which was refuted by Raigorodskii in [12,13]. We provide more details on the previous proof in Section 5.

In the present paper we provide a new proof for the case $d > 2$. The main result is

Theorem 1. *The computational problems $\mathbb{R}^d$-UNIT-DISTANCE-EMBEDDABILITY, $\mathbb{R}^d$-UNIT-DISTANCE-STRICT-EMBEDDABILITY, $\mathbb{R}^d$-UNIT-DISTANCE-INJECTIVE-EMBEDDABILITY, $\mathbb{R}^d$-UNIT-DISTANCE-STRICT-INJECTIVE-EMBEDDABILITY are NP-hard for each $d > 2$.*

To prove this result we construct a reduction of the classic NP-complete problem of graph 3-coloring (**3-COLORING**) (see [5]) to each of the four embeddability problems, which implies NP-hardness of all four mentioned problems. It should be mentioned that the question whether the described problems lie in NP is open.

Outline of the reduction

Here we briefly describe the following construction.

We define a *rod* as a unit-distance graph which is "rigid" enough to sustain the same distance between a pair of its vertices in any embedding. We formalize the notion in Section 2. Lemma 1 states that, under certain requirements, a rod subgraph is interchangeable with a weighted edge of corresponding length in terms of embeddability of the whole graph. Consequently, if we can build a rod which sustains distance l between a pair of vertices, we can just as well use a weighted edge of length l in its place.

It is clear that not all distances are realizable as the length of a rod (since there are only countably many rods). However, in Section 3 we show that the set of realizable distances is dense, and present a constructive way to build a rod such that its length is constrained to a chosen interval of non-zero length. Further in the construction, we use rods obtained this way as a kind of "wobbly" weighted edges since we have no way of setting the length precisely. We also show how to constrain distance between two vertices to a certain interval using a chain of two rods of suitable length.

The main part of the reduction from **3-COLORING** presented in Section 4 proceeds as follows. We constrain all vertices of the input graph G to a circle by connecting them to vertices of an auxiliary $(d-2)$-simplex. Then, we introduce three special vertices u_0, u_1, u_2 constrained to the circle along with chains between them so that they lie roughly at the vertices of an equilateral triangle inscribed in the circle. Further, we constrain all vertices of V so that they lie close to the middle of an arc formed by u_0, u_1, u_2 (thus assigning a definite "color" to each vertex), and forbid for adjacent vertices to have the same "color" by subdividing edges of G into chains. To finally obtain a unit-distance graph, we replace the weighted edges by rods via Lemma 1 (for a brief illustration see Fig. 1).

2. The notion of a rod

Let us introduce some necessary definitions.

A *weighted graph* $G = (V, E, w)$ is an ordered triple such that (V, E) is a graph and $w : E \to \mathbb{R}_+$ is a function that assigns a positive number to each element of E; for every edge $e \in E$ we will say that $w(e)$ is the *length* of the edge e. If $w \equiv 1$, the weighted graph G is called a *unit-distance graph*. A *length embedding* (or, more simply, an *embedding*) of the weighted graph $G = (V, E, w)$ in $\mathbb{R}^d$ is a map $\varphi : V \to \mathbb{R}^d$ such that for any edge $e = uv \in E$ distance between $\varphi(u)$ and $\varphi(v)$ is equal to $w(e)$.