Rigid and Globally Rigid Graphs

Tibor Jordán

BSM Research opportunities project

Introduction

Let us start with a slightly informal description. More precise definitions will be given below. We shall work with d-dimensional *bar-and-joint frameworks* (also called geometric graphs) and analyse their rigidity properties. Such a framework consists of rigid bars (line segments) that meet at universal joints. The bars can move continuously in d-space so that the bar lengths and bar-joint incidences must be preserved. The framework is said to be *rigid* (in dimension d) if every such continuous motion of the framework results in a congruent framework, that is, the distance between each pair of joints is unchanged.

In the *graph* of the framework vertices correspond to the joints and edges correspond to the bars. It is known that if the framework is in sufficiently general position then rigidity depends only on its graph. For example, consider a framework in the plane consisting of four joints a, b, u, v and five bars, where bar uv is the missing bar (so its graph G consists of two triangles sharing an edge). It is rigid in the plane but is not rigid in three-space. Note that if we remove an arbitrary bar, the framework is no longer rigid in the plane.

The framework is said to be *globally rigid* if every other framework with the same graph and same bar lengths is congruent with it. Clearly, global rigidity implies rigidity. For example, the four-vertex framework mentioned above is not globally rigid in two dimensions (consider the framework obtained by reflecting joint u about the line of the bar which does not contain v).

The theory of rigid and globally rigid frameworks has surprisingly many applications in statics, structural analysis of molecules, sensor network localization, formation control, and elsewhere. There are lots of nice results as well as many open problems in this field.

Open problems

Rigidity theory is in the intersection of geometry, algebra, and combinatorics. It also includes quite a few questions concerning efficient algorithms. We shall chose an open problem suitable for the interested students. Two candidates are given below.

The first one is motivated by applications of rigidity in which redundancy is important (say, statics or formation control).

Problem 1 *Construct graphs (or frameworks) which are highly vertex- or edge-redundantly rigid, that is, which remain rigid (in two or three dimensions) after the loss of a prescribed number of joints or bars. Can you find examples with the minimum number of necessary bars (on the given number of vertices)? What is this minimum?*

The second one builds on the fact that if we consider two non-congruent frameworks on the same graph and with the same bar lengths in the plane, then (even if they are both rigid frameworks in two dimensions), there exists a continuous motion which takes one of them to the other in four dimensions. Sometimes such a motion exists even in three-space.

Problem 2 *Find interesting families of frameworks where the motion described above must be four-dimensional. What is the reason? What if the joints are in general position, without any algebraic relation between the coordinates. Is there an example of this kind?*

Basic definitions

A d-dimensional *framework* is a pair (G, p), where $G = (V, E)$ is a graph and p is a map from V to the d-dimensional Euclidean space R^d. We consider the framework to be a straight line *realization* of G in R^d. Intuitively, we can think of a framework (G, p) as a collection of bars and joints where each vertex v of G corresponds to a joint located at $p(v)$ and each edge to a rigid (that is, fixed length) bar joining its end-points. Two frameworks (G, p) and (G, q) are *equivalent* if $dist(p(u), p(v)) = dist(q(u), q(v))$ holds for all pairs u, v with $uv \in E$, where $dist(x, y)$ denotes the Euclidean distance between points x and y in R^d. Frameworks (G, p), (G, q) are *congruent* if $dist(p(u), p(v)) = dist(q(u), q(v))$ holds for all pairs u, v with $u, v \in V$. This is the same as saying that (G, q) can be obtained from (G, p) by an isometry of R^d. We say that (G, p) is *globally rigid* if every framework which is equivalent to (G, p) is congruent to (G, p).

A *motion* (or *flex*) of (G,p) to (G,q) is a collection of continuous functions $M_v : [0,1] \to R^d$, one for each vertex $v \in V$, that satisfy

$$M_v(0) = p(v) \text{ and } M_v(1) = q(v)$$

for all $v \in V$, and

$$dist(M_u(t), M_v(t)) = dist(p(u), p(v))$$

for all edges uv and for all $t \in [0,1]$. The framework (G,p) is *rigid* if every motion takes it to a congruent framework (G,q).

For more details see the survey paper *Combinatorial rigidity: graphs and matroids in the theory of rigid frameworks* by Tibor Jordán, Memoirs of the Mathematical Society of Japan, 2016, available also as Technical report TR-2014-12, Egerváry Research Group, Budapest, or the book *Counting on frameworks* by Jack E. Graver, Dolciani Mathematical Expositions, No. 25, The Mathematical Association of America.

Warm up exercises

Solve the first exercise and think about and make progress on the other two exercises before starting the research project.

Exercise 1. Characterize the rigid bar-and-joint frameworks in R^1.

Exercise 2. Construct rigid and globally rigid frameworks on n vertices in R^d, for all $n \geq 2$ and $d \geq 1$. Try to use as few edges as possible. Formulate a conjecture for the minimum number of edges needed (in terms of n and d).

Exercise 3. Consider a rectangular part of a square grid in the plane, say, with n rows and m columns of squares. Imagine that the points correspond to joints and the sides of the squares are all unit length bars. Such a framework is never rigid (why?). Try to make it rigid by adding a set of diagonal bars to the framework. (Diagonal bars are longer: each of them connects opposite corners of some square.) Characterize those sets of diagonal bars which make such a square grid framework rigid! What is the minimum number (in terms of n and m) of diagonal bars you need to rigidify the framework?

Tibor Jordán

`jordan@cs.elte.hu` April 12, 2020.