Distance Geometry and Applications

Antonio Mucherino

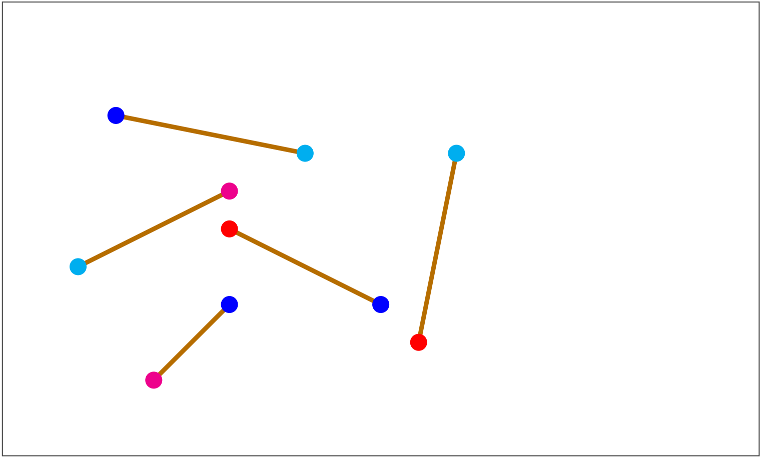
IRISA/INRIA, University of Rennes 1

antonio.mucherino@irisa.fr

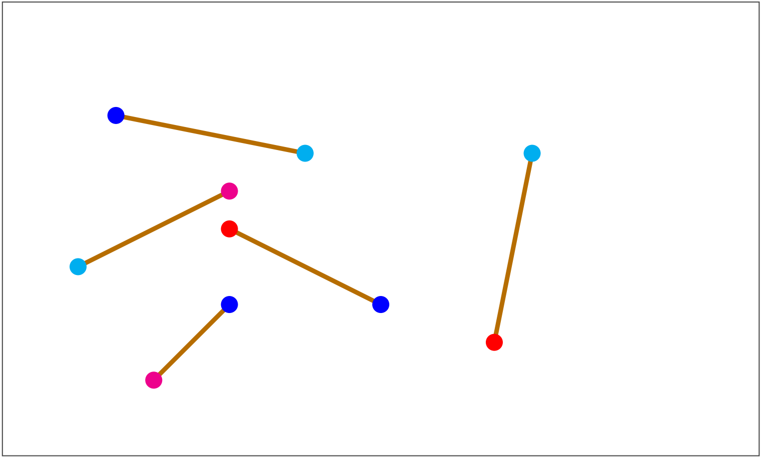
joint work with: many people . . .

UFSC, Math department,
Florianópolis (SC), Brazil
March 8th 2018

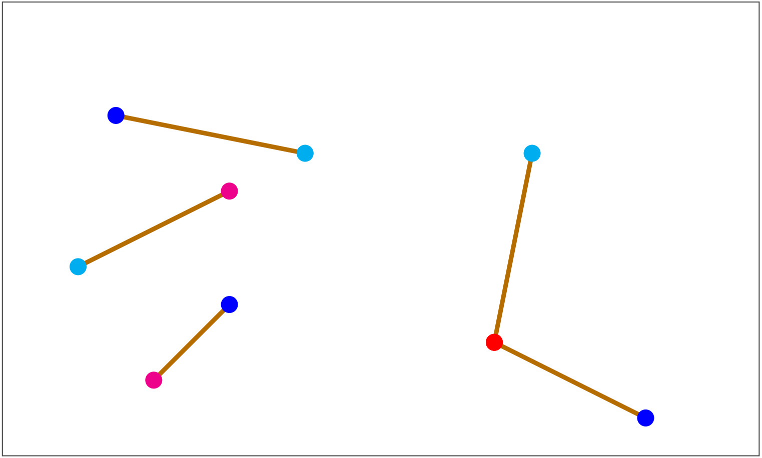
Distance Geometry



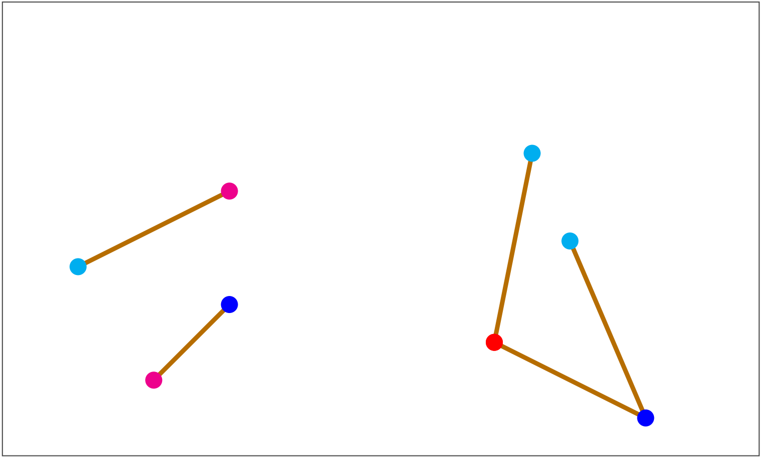
Distance Geometry



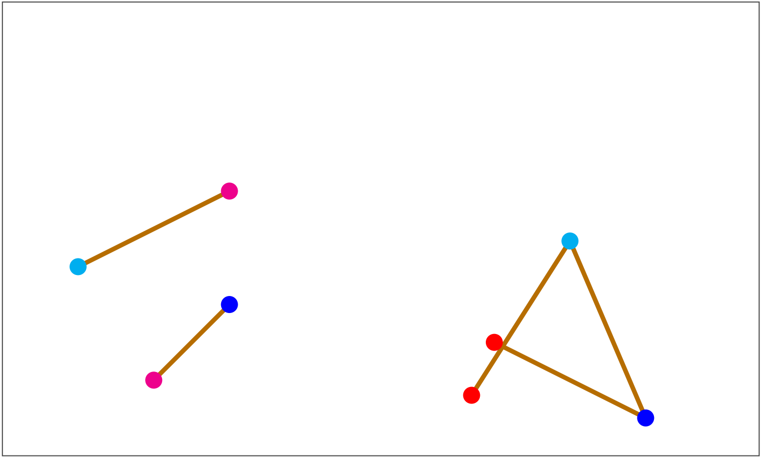
Distance Geometry



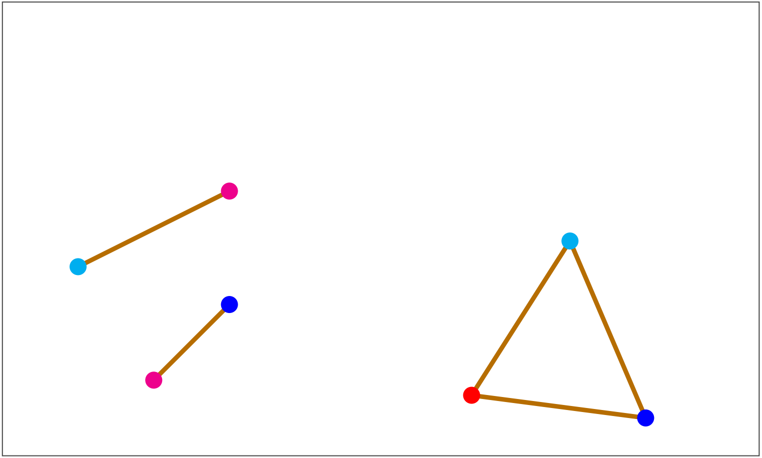
Distance Geometry



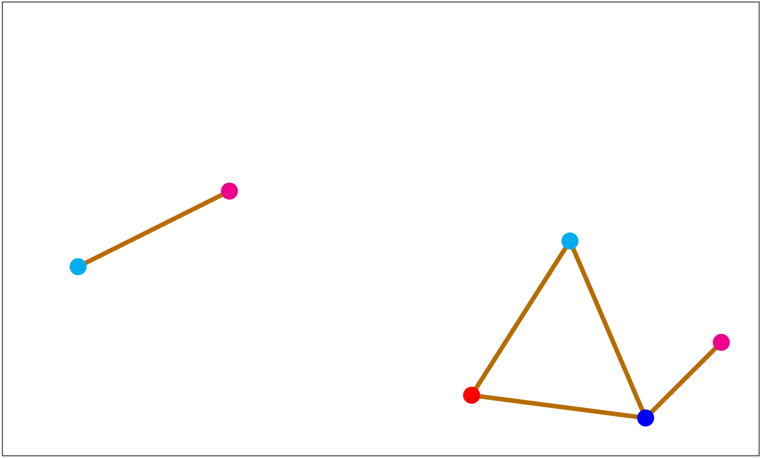
Distance Geometry



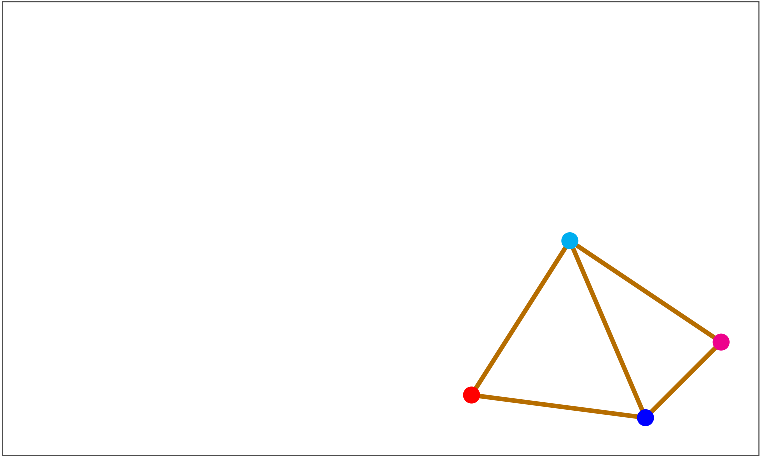
Distance Geometry



Distance Geometry



Distance Geometry



dynDGP

A. Mucherino

DGP

Applications

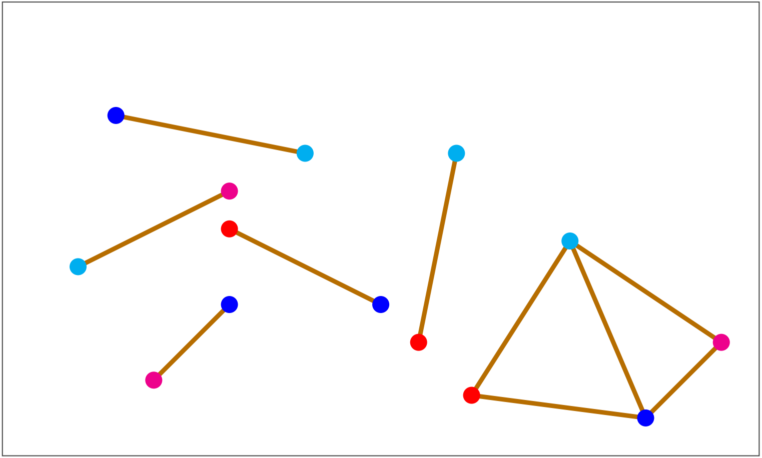
Motion adaptation

Retargeting

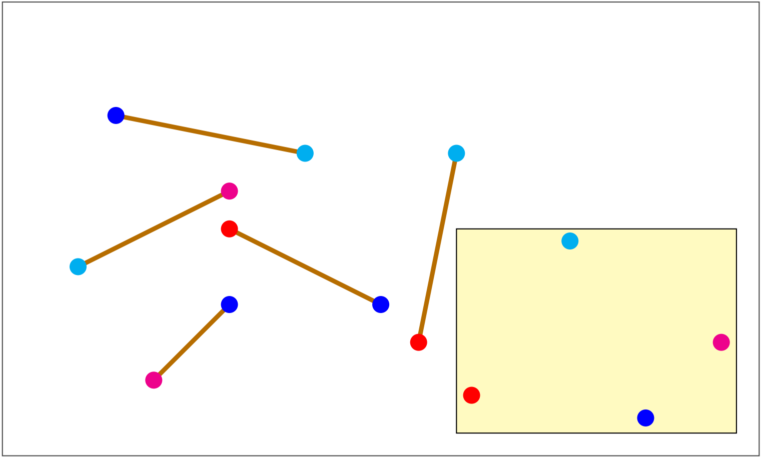
The DDGP

Challenges

Distance Geometry



Distance Geometry



The Distance Geometry Problem

Let $G = (V, E, (\delta, \pi))$ be a simple weighted undirected graph:

V	represents a set of objects
E	indicates whether distances are known
δ	provides the distance value
π	assigns a priority to the distance

Definition

The **DGP** in dimension K.
Determine whether there exists an realization

$$x : V \longrightarrow \mathbb{R}^K$$

of G in $\mathbb{R}^K$ such that, for all edges $(u, v) \in E$,

$$||x_u - x_v|| \approx \delta(u, v).$$

When not all distances can be satisfied, distances δ having higher priorities π are to be privileged.

DGP complexity

Embeddability of Weighted Graphs in k-Space is Strongly NP-Hard

(Extended Summary)[1]

JAMES B. SAXE
Computer Science Department
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

Abstract--In this paper we investigate the complexity of embedding edge-weighted graphs into Euclidean spaces: Given an (incomplete) edge-weighted graph, G, can the vertices of G be mapped to points in Euclidean k-space in such a way that any two vertices connected by an edge are mapped to points whose distance is equal to the weight of the edge? We prove that the preceding problem is NP-Hard (by reduction from 3-Satisfiability), even when k=1 and the edge weights are restricted to take on the values 1 and 2. Related results are shown for the problem of testing the uniqueness of a known embedding and for variations involving inexact edge weights.

1. Introduction

In many applications of distributed sensor networks[2] there arises the problem of determining the locations of sensors from incomplete (and possibly errorful) information about their distances from each other and from fixed landmarks. This prompts us to ask the following geometric questions:

this result to higher
questions concerning
inherently involves r
relevance to an "app
discuss versions of th
distance matrix is kno
We show that these
in the paper. Finally,

2. Fundamental Co

We begin by introd

Definitions:

A weighted grap
an unordered p
[0,∞). The elem
the edges of G.
of e in G (or sim

Definitions:

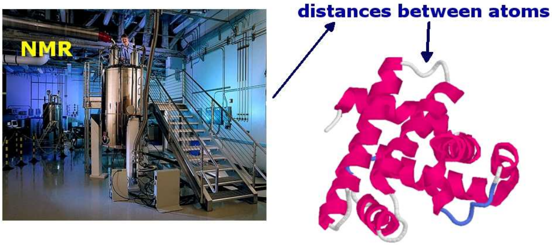
Let G = <V,E,W

Applications

- let's start with some *static* applications

Protein conformations

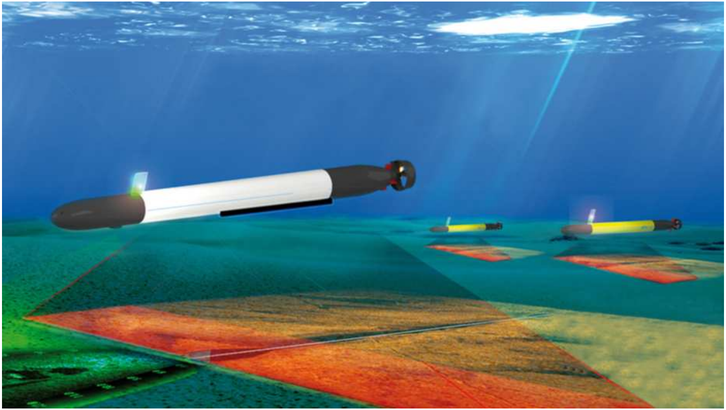
Nuclear Magnetic Resonance (NMR) is able to provide some of the distances between pairs of atoms of a molecule.



Main limitations:

- only short range distances are provided;
- no distances are *exact*, a certain interval is rather given;
- only distances between hydrogen atoms are generally available.

Sensor Network Localization



© projet Comet

Neotek (Lorient) et École Nationale supérieure des techniques avancés (ENSTA) de Brest

Dynamical Applications

- and if we need to deal with dynamical problems?

How to deal with the dynamics?

dynDGP

A. Mucherino

DGP

Applications

Motion adaptation

Retargeting

The DDGP

Challenges

Springer Link

International Conference on Geometric Science of Information

GSI 2017: Geometric Science of Information pp 821-829 | Cite as

An Approach to Dynamical Distance Geometry

Authors | Authors and affiliations

Antonio Mucherino, Douglas S. Gonçalves

Conference paper

First Online: 24 October 2017

344 Downloads

Part of the Lecture Notes in Computer Science book series (LNCS, volume 10589)

The dynamical Distance Geometry Problem

Let $G = (V \times T, E, (\delta, \pi))$ be a simple weighted undirected graph:

- V represents a set of objects u, v, etc.
- T represents a (discrete) set of temporal instants q, t, etc.
- E indicates the existence of distances between u_q and $v_t \in V \times T$
- δ distance value
- π priority of distance

Definition

The dynamical **DGP** (dynDGP) in dimension K.
Determine the realization

$$x : V \times T \longrightarrow \mathbb{R}^K$$

of G in $\mathbb{R}^K$ such that the overall error on the given distances is minimized. *When a realization with null error does not exist, distances δ having higher priorities π are to be privileged.*

Motion adaptation: retargeting

CGTrader.com

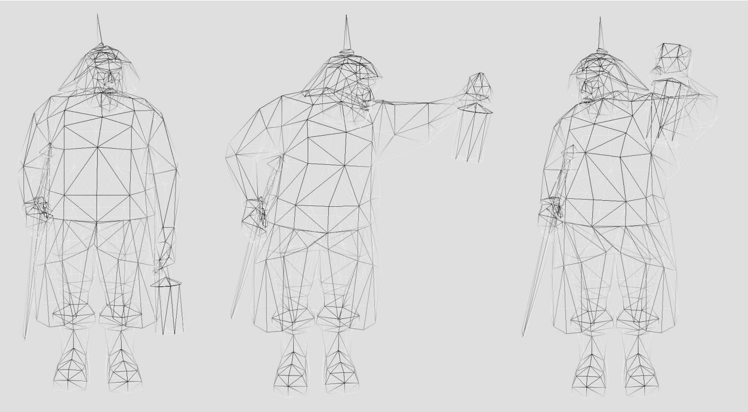
Motion adaptation: retargeting

CGTrader.com

Motion adaptation: retargeting

CGTrader.com

Motion adaptation: retargeting

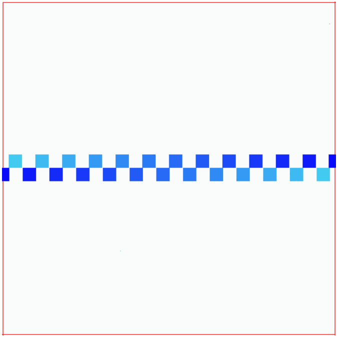


CGTrader.com

Motion Adaptation

- we represent a motion by distances
- new distance constraints can manipulate such a motion

Two crossing people



Two particles going to each other from opposite directions in a $1\text{cm} \times 1\text{cm}$ box.

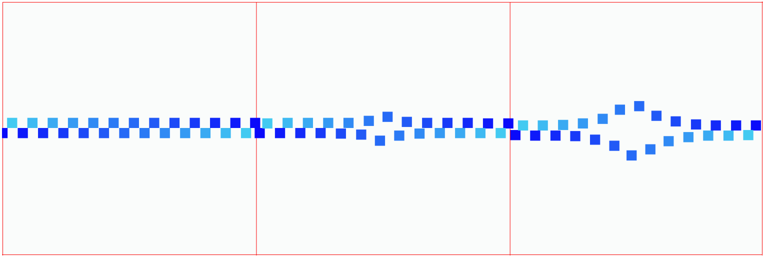
Motion manipulation

Let us create a dynDGP instance such that:

- the motion of every v is preserved by using the original distances
$$\delta(v_q, v_t) \quad \forall q \,:\, t-3 \leq q < t,$$
- collisions are avoided by including the constraint:
$$\delta(u_t, v_t) > \Delta \quad \forall t \in T, \forall u, v \in V \,:\, u \neq v,$$
where Δ is strictly positive.

The priority to the distances is assigned so that all newly introduced distances have maximal priority, and the distances between closer frames are more important.

Two crossing people avoiding collisions



original $\Delta > 0.1$ $\Delta > 0.2$

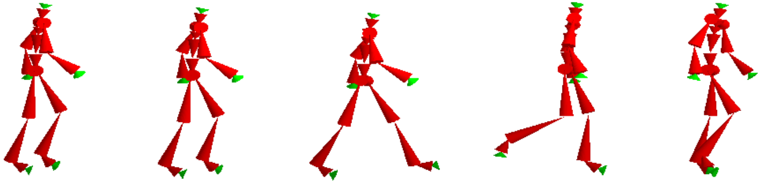
Frame by frame, we use here a spectral gradient approach to solve the DGP, where the original animation is given as a starting point.

Retargeting

- adapting human motions. . .

A little step for the human skeleton

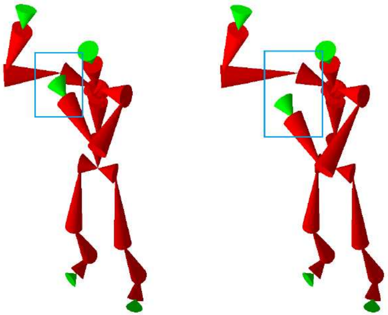
A human motion can be represented by the trajectories of the joints of a given skeletal structure.



⇒ *How to impose the same movement to a different skeleton?*

Retargeting: the classical approach

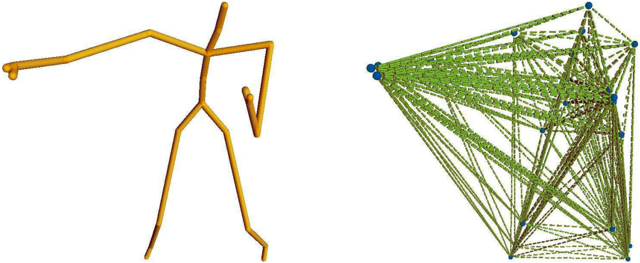
Classical approaches are based on bone angle-transfer.



Also, they cannot avoid undesired collisions.

Retargeting: a distance-based approach

We can represent human motions by distances.



They can represent either *bones*, or rather *relative movements*.

Retargeting: some preliminary solutions

video clip

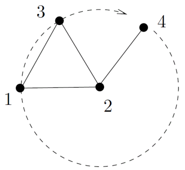
`https://dl.acm.org/authorize.cfm?key=N49551`

The Discretizable DGP

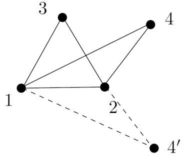
- reducing the search space of our DGPs to a discrete (and finite!) domain. . .

The discretization

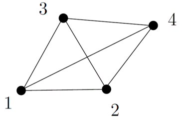
In dimension 2, how many positions for these points?



$\longleftarrow$ Infinite positions



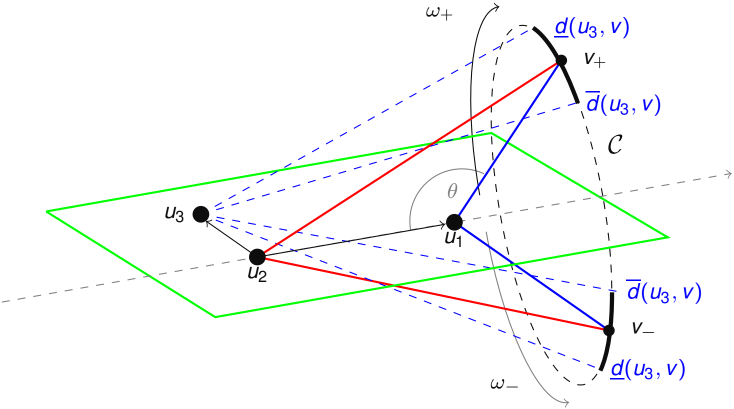
$\longleftarrow$ Two positions



$\longleftarrow$ One position

The discretization

And in dimension 3?



Thanks a lot Douglas for this picture!

The **Discretizable DGP** (DDGP)

Given a graph $G = (V, E, d)$, there must exist a vertex order on V such that

(A1) $G[\{1, 2, \ldots, K\}]$ is a clique consisting of exact distances;

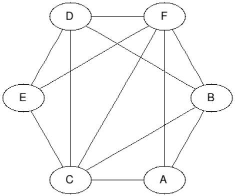
(A2) $\forall v \in V \,:\, v > K, \quad \exists\, u_1, u_2, \ldots, u_K \,:$

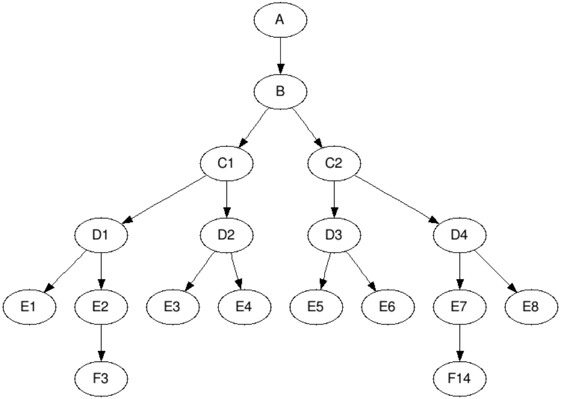
$$\begin{cases} u_1 < v,\ u_2 < v,\ \ldots\ u_K < v, \\ \{(u_1, v), (u_2, v), \ldots, (u_{K-1}, v)\} \subset E',\ (u_K, v) \in E, \\ \mathcal{V}(S[u_1, u_2, \ldots, u_K]) \neq 0, \end{cases}$$

where $S[\ldots]$ is the simplex defined by u_1, u_2, $\ldots$, u_K, and $\mathcal{V}$ is the volume of its argument.

$\Longrightarrow$ For all vertices $v > K$, candidate vertex positions can be generated by intersecting $K - 1$ spheres with 1 spherical shell;
$\Longrightarrow$ After the discretization, the problem remains **NP-hard**.

A Branch & Prune algorithm





Some computational experiments

Numerical results on artificially generated instances from the PDB.

	Instance			BP			
PDB ID	*aa*	$\|V\|$	$\|E\|$	D	Calls	Time	MDE
2JMY	15	120	660	5	2983	0.01	1e-16
2KXA	24	177	973	3	5064	0.01	6e-03
1DSK	28	222	1210	4	53890	0.14	1e-06
2PPZ	36	287	1522	3	442965	1.87	4e-08
1AQR	40	310	1596	4	114671	0.20	6e-03
2ERL	40	324	1792	3	10410	0.03	1e-03
2E2F	41	315	1716	3	19916	0.06	9e-03
1FJK	52	417	2306	4	925090	3.07	2e-06
2JWU	56	448	2416	4	226870	0.81	1e-02
2KIQ	57	455	2452	4	317136	1.12	7e-04
2LOW	64	497	2650	3	3738152	8.79	2e-07

This BP implementation integrates some additional features that allow for speeding up the algorithm. D is the discretization factor of the arcs.

The main challenge

How to perform the search **without** *discretizing the arcs?*

Thanks!

antonio.mucherino@irisa.fr