ЧЕБЫШЕВСКИЙ СБОРНИК

Том 20. Выпуск 2.

Введение расстояния взаимодействия в контексте геометрии расстояния для движений человека

A. Mucherino (Rennes, France)

Mucherino Antonio — ИРИСА, университет Ренна 1, Ренн, Франция
e-mail: antonio.mucherino@irisa.fr

Аннотация

Задача динамической геометрии расстояния (dynDGP) – это недавно введенный подкласс геометрии расстояния, где проблемы имеют динамическую составляющую. Графы

$$G = (V \times T, E, \{\delta, \pi\})$$

dynDGP имеют множество вершин, которое является произведением множества двух множеств: множества V, содержащего объекты для анимации, и множества T, представляющего время.

В этой статье основное внимание уделяется специальным экземплярам dynDGP, которые используются для представления движения человека в проблеме адаптации, где множество V допускает скелетную структуру (S, χ).

"Расстояние взаимодействия" — это понятие введено в качестве возможной замены евклидова расстояния, которое способно улавливать информацию о представлении динамики задачи и некоторые начальные свойства этого нового расстояния.

Ключевые слова: динамическая дистанционная геометрия, дистанция взаимодействия, адаптация человеческого движения, ретаргетинг, анимированные скелетные структуры, симметричное квази-расстояние.

Библиография: 22 titles.

CHEBYSHEVSKII SBORNIK

Vol. 20. No. 2.

Introducing the Interaction Distance in the context of Distance Geometry for Human Motions

A. Mucherino (Rennes, France)

Mucherino Antonio — IRISA, University of Rennes 1, Rennes, France
e-mail: antonio.mucherino@irisa.fr

Abstract

The dynamical Distance Geometry Problem (dynDGP) is a recently introduced subclass of the distance geometry where problems have a dynamical component. The graphs

$$G = (V \times T, E, \{\delta, \pi\})$$

of dynDGPs have a vertex set that is the set product of two sets: the set V, containing the objects to animate, and the set T, representing the time. In this article, the focus is given to special instances of the dynDGP that are used to represent human motion adaptation problems, where the set V admits a skeletal structure (S, χ).

The "interaction distance" is introduced as a possible replacement of the Euclidean distance which is able to capture the information about the dynamics of the problem, and some initial properties of this new distance are presented.

Keywords: dynamical distance geometry, interaction distance, human motion adaptation, retargeting, animated skeletal structures, symmetric quasi-distance.

Bibliography: 22 titles.

1. Introduction

Given a weighted undirected graph $G = (V \times T, E, \{\delta, \pi\})$, the dynamical Distance Geometry Problem (dynDGP) [16, 20] in dimension 3 consists in finding a realization

$$x : (v, t) \in V \times T \longrightarrow (x_v^t, y_v^t, z_v^t) \in \mathbb{R}^3 \tag{1}$$

of G in the Euclidean space $\mathbb{R}^3$ such that the following objective is minimized:

$$\sigma(x) = \sum_{(u_q, v_t) \in E} \pi(u_q, v_t) \frac{|\, d((x_u^q, y_u^q, z_u^q), (x_v^t, y_v^t, z_v^t)) - \delta(u_q, v_t)\,|}{\delta(u_q, v_t)}, \tag{2}$$

where d is a distance function, and $|\cdot|$ indicates the absolute value of a real number.

It is important to point out the two different roles of δ and d in equ. (2). The former is one of the weights associated to the edges of the graph G, which corresponds to the available distance value between a vertex pair. The latter is instead a distance function: d associates, to every pair of positions, their corresponding distance. In the majority of the applications of distance geometry [19], the distance function d is the Euclidean distance:

$$d((x_u^q, y_u^q, z_u^q), (x_v^t, y_v^t, z_v^t)) = \sqrt[2]{(x_v^t - x_u^q)^2 + (y_v^t - y_u^q)^2 + (z_v^t - z_u^q)^2}.$$

The value of π associated to every distance represents the "importance" of such a distance, and it is generally referred to as the *priority* of the distance δ [9, 16].

The vertex (v, t) of the graph G is an ordered pair that represents a given object v at a certain instant t, and a realization x corresponds to an animation of the objects in V over the time steps in $T = \{1, 2, \ldots, n\}$. As in the equations already introduced above, the compact notations v_t and $\{u_q, v_t\}$ are employed for indicating a vertex and an edge of G, respectively. The coordinates of the object v at the time t obtained through the realization function x are indicated with compact notations x_v^t, y_v^t and z_v^t. Notice that, from a formal point of view, the only particularity of the dynDGP (w.r.t a standard distance geometry problem) is the fact that the vertex set of G is the

set product of two sets: V (the objects) and T (the time). In other words, a dynDGP with $T = \{1\}$ is a "standard" distance geometry problem [12].

This article focuses on a particular application of the dynDGP, where the main interest is in analyzing human motions through the study of animated skeletal structures which resemble the human body [20]. Some initial works have shown that tools for distance geometry can be employed in this context [2, 17], but they consider, as in the majority of the applications in distance geometry, the Euclidean distance as a metric.

In fact, only a few examples of use of non-Euclidean distances in the context of distance geometry can currently be found in the scientific literature: in [4], for example, the L_∞ distance is used to reformulate the distance geometry problem as an optimization problem whose constraints only include linear terms; in [13], distance geometry on the sphere is studied, where the distances can be either Euclidean, or geodesic distances. In the present work, the possibility to replace the Euclidean distance with another distance function, which is able to "better" capture the features of the motion adaptation problem, is discussed. To this aim, the *interaction distance* will be introduced, and some of its basic properties will be studied.

The rest of the article is organized as follows. Section 2 will introduce more in details the mathematical objects that are involved in motion adaptation. Section 3 will propose a definition of the human motion adaptation problem which is based on the formulation of a dynDGP. Then, the interaction distance will be introduced in Section 4, and some initial properties will be presented. Finally, Section 5 will conclude the paper.

2. The objects that come to play

Human-like characters are sets of bars that represent "bones" (i.e. the rigid components) of a human body, while the intersection points between two bars are named "joints". This kind of structure is widely used in applied mathematics and can be represented by a simple graph $S = (V, E_S)$, where vertices in V are the joints and the existence of an edge $\{u, v\} \in E_S$ between two joints u and v indicates that they are connected by a bar [11].

The graph S provides information about the structure (a hand is connected through a bar to an elbow, which is then connected through another bar to the shoulder, and so on), but it gives no information about the morphology of the character. For a predefined accuracy level of the skeletal representation of the character (not all human bones are generally represented, but only a few of them, often a little more than 20 bones), in fact, it is actually possible to consider the graph S as a constant for an entire group of characters. Then, the difference between pairs of characters can be evaluated by simply comparing their morphology. Notice that, for the majority of the skeletal representations, the graph S is a tree.

The simplest way to add morphology information to a character is to include in the graph S a weight function that associates a real positive value (the bone length) to all its edges in E_S. This way, a "tall guy" has larger values associated to the weights of its "leg" edges, in comparison with a normal-sized guy. Naturally, lengths can be provided in absolute values for all edges:

$$w : \{u, v\} \in E \longrightarrow w(u, v) \in \mathbb{R}_+,$$

or rather the propositions among them can be provided in order to obtain a more efficient character description which is independent from the metric scale.

Even though the simple, and weighted, graph $S = (V, E, w)$ provides enough information to represent a character bone structure and morphology, it is typical (in the works signed by the computer graphics community) to represent characters by *skeletal structures* [1, 11]. A skeletal structure is a pair consisting of a graph (S in our case, without weights), and a function χ that is able to assign a position in $\mathbb{R}^3$ to every joint $v \in V$. This initial set of positions corresponds to the
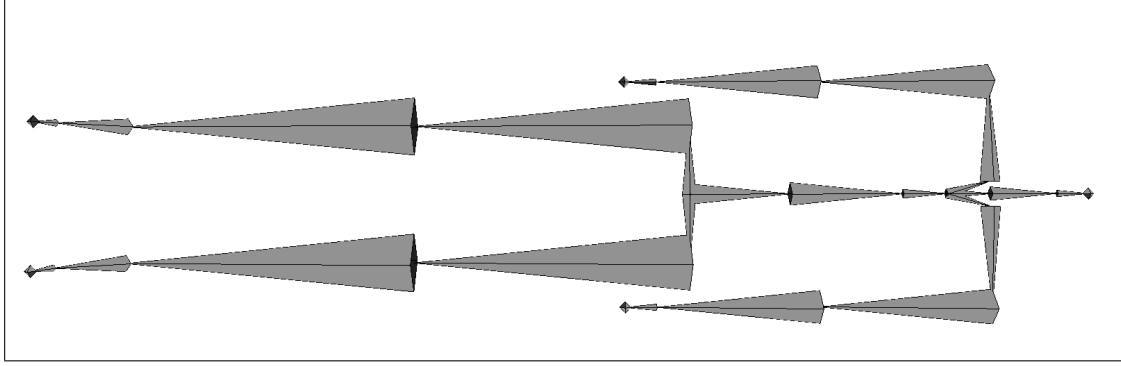
Рис. 1: *The initial pose for the skeletal structure* (S, χ). *S is a tree and, in its realization shown in this figure, the joint positions are computed by setting the position of its root to* $(0,0,0)$ *and by using the displacement function* χ *to compute the coordinates of all other joints.*

"initial pose" of the character (Fig. 1 shows an example of initial pose for a human-like character). The function χ takes into consideration the hierarchical information in the graph S, and actually provides the *displacement* $(\xi_v^x, \xi_v^y, \xi_v^z)$ of every vertex $v \in V$ w.r.t. its parent (when S is a tree, every vertex has only one parent):

$$\chi : v \in V \longrightarrow (\xi_v^x, \xi_v^y, \xi_v^z) \in \mathbb{R}^3,$$

from which all vertex positions in the initial pose can be obtained.

The most common way to represent the animation of the character (S, χ) is through the function:

$$\rho : (v, t) \in V \times T \longrightarrow (\theta_v^t, \phi_v^t, \eta_v^t) \in \mathbb{R}^3, \tag{3}$$

where θ_v^t, ϕ_v^t and η_v^t are three Euler angles (pitch: θ, roll: ϕ and yaw: η) used to control the relative rotations of every character edge in the local coordinate system of the hierarchically lowest (closest to the root) joint of the edge [14]. This representation has the great advantage of keeping the skeletal structure, as well as its morphology, constant during the animation.

From the initial pose for the character, and by using the Euler angles $(\theta_v^t, \phi_v^t, \eta_v^t)$ for all following time steps t, it is possible to calculate the vertex coordinates of all subsequent character poses, so that the entire animation can be represented through a function x (see equ. (1)). This procedure is detailed, for example, in [14]. A solutions (realization) x to a dynDGP has exactly the same form.

3. Human motion adaptation

Character animation is nowadays largely used in movie and video game industries. Typically, a 3D model and its associated skeletal structure (S, χ) are designed by an artist, then animated either manually or using recorded motion capture data. However, it is often the case that animations created for a specific character, or captured from a given actor, need to be reused on characters with a different morphology: this problem is known in the specialized literature as the *motion adaptation problem*.

Since the early works of Gleicher [8], and Choi and Ko [3], the problem of adapting a given animation to the special features of a given character is taking particular attention from the research community. The main problem consists *extracting* the information about the animation played by character A (the actor), and to *transfer* this piece of information to another character B (the character to be animated), which is generally morphologically different from character A (see Section 2 for the representation of the characters and the animations).

Let (S, χ_A) be the skeletal structure that describes the morphology of character A, and suppose that the initial animation is captured from character A; let (S, χ_B) be the skeletal structure of the target character B. The two characters have a common graph S, but the two functions χ_A and χ_B are different. When the two initial poses described by the two displacement functions χ_A and χ_B are the same (different numerical values for the displacements over the three components may represent the same pose, such as the typical **T** pose), one possible (and easy) approach to motion adaptation is to simply *transfer* the function ρ_A (see equ. (3), representing the animation of character A, to character B (in other words, it is imposed that $\rho_B \equiv \rho_A$). However, the Euler angles in ρ_A are not able to take into consideration the spatial relationship between pairs of character joints [2, 10], and as a consequence undesired effects may be introduced in the animation. Typical undesired effects include creating body contacts that were not present in the original animation, or to miss body contacts that actually occur in the original animation.

As already remarked in Section 2, the coordinates (x_v^t, y_v^t, z_v^t) for every joint and for every time t can be computed by combining the displacement function of the skeletal structure (S, χ_A) with the function ρ_A. With the obtained representation x_A of the animation, Euclidean distances between pairs of joints, at every time frame t, can be computed. As it is well-known, the Euclidean distance is widely used in several scientific disciplines, and provides information that is easily understandable by humans, which is, the simple measure of proximity between object pairs. A character that flaps its hands, for example, needs to bring, at particular time frames, the Euclidean distance between the two hands to zero in order to perform this movement; the distance provides this simple and fundamental feature of the movement.

When the animations are represented by functions having the form of equ. (1), the motion adaptation problem can be formulated as follows [2, 17]. Let x_A be the initial animation of (S, χ_A), given in terms of absolute vertex positions. In order to capture the spatial relationships between joint pairs in the animation, *all* relative distances δ, for every time frame, can be computed. If some distances are considered to be more important than others, then a higher priority value π can be associated to them. This procedure allows to define the simple undirected graph $G = (V \times T, E, \{\delta, \pi\})$, which represents the animation in terms of distances.

DEFINITION 11. (dynDGP-based human motion adaptation).
Given a skeletal structure (S, χ_B) and a simple weighted undirected graph $G = (V \times T, E, \{\delta, \pi\})$ obtained from an animation of (S, χ_A), find an animation x_B for the skeletal structure (S, χ_B) that is solution to dynDGP(G).

It is important to remark that "an animation for (S, χ_B)" is meant to be an animation where the morphology described by the skeletal structure is preserved. Naturally, finding an animation for (S, χ_A), when G was obtained from an animation of the same skeletal structure, is an easy problem, because all distance constraints are precise and compatible to each other [15]. When the morphology changes from the original character to the target character, instead, it is unlikely that the all distance constraints can be satisfied, so that the best *trade-off* needs to be identified by minimizing the function $\sigma(x)$ (see equ. (2)).

Some initial studies where the motion adaptation problem is formulated as a dynDGP can be found in [2, 15, 17]. However, these studies pointed out some limitations in the use of the Euclidean distance for this particular application. The Euclidean distance, in fact, can fail in providing information about the dynamics of the considered animation, because it focuses only on one snapshot of the animation per time. This observation motivated the main aim of this paper: the next section will introduce the concept of interaction distance, and will study some basic properties.

4. The interaction distance

Immanuel Kant, whose works preceding the *criticism* were influenced by his readings of Newton and Leibniz, compared *time* (in his early writings on the topic) to a point moving on an infinite line [22]. Moreover, such a line was oriented, because "time instants do not depend reciprocally on each other, it is rather the instant that precedes which determines the one that follows; this is the reason why time admits only one dimension" [7].

In the same spirit, the main idea behind the interaction distance is to consider the dynamics of the problem through the semi-straight lines (only the future information is taken into consideration) defined by given joint positions at two consecutive time frames. Let u and $v \in V$ be two distinct joints. The orientation of the joint u at the current frame $t > 1$ can be obtained by using two positions: the position of u at the previous frame $p_u^{t-1} = (x_u^{t-1}, y_u^{t-1}, z_u^{t-1})$, and its current position p_u^t. The semi-straight line ℓ_u defined by p_u^{t-1} and p_u^t, having as a boundary p_v^t and *not* including p_u^{t-1}, provides an approximation of the trajectory of u at further frames in the hypothesis there will be no changes in the joint orientation. Moreover, in the hypothesis the joint speed is also constant, it is possible to select a discrete subset of equidistant points on ℓ_u representing potential positions for u at future time steps. The discrete set of potential future positions for u is therefore given by the equation:

$$p(u, \alpha_u) = p_u^{t-1} + \alpha_u \cdot (p_u^t - p_u^{t-1}), \quad \alpha_u \in \{1, 2, \ldots, +\infty\}.$$

Once a similar discrete set of potential positions for the joint v is computed, the interaction distance between u and v can be measured by studying how these potential position points on the two semi-straight lines ℓ_u and ℓ_v approach one another.

DEFINITION 12. (The interaction distance).
Given a skeletal structure (S, χ) and an animation x, the "interaction distance" $D(p_u^t, p_v^t)$ at frame $t > 1$ between two joints $u, v \in V$ is:

$$\min_{\alpha \in \mathbb{N} \setminus \{0\}} d(p(v, \alpha), p(u, \alpha)), \quad \text{if } p_u^{t-1} \neq p_u^t \text{ and } p_v^{t-1} \neq p_v^t,$$

$$\min_{\alpha \in \mathbb{N} \setminus \{0\}} d(p(v, \alpha), p_u^t), \quad \text{if } p_u^{t-1} = p_v^t \text{ and } p_v^{t-1} \neq p_v^t,$$

$$d(p_v^t, p_u^t), \quad \text{if } p_u^{t-1} = p_v^t \text{ and } p_v^{t-1} = p_v^t,$$

where d is the Euclidean distance.

From a geometric point of view, the interaction distance gives a measure of the relative approach between two joints by using only the information about their current position and orientation. It follows immediately from the definition that interaction distances between joints can be computed at every frame of the animation, except the frame 1, and that two consecutive frames are involved in the computation of this distance. Fig. 2 shows a graphical representation of the interaction distance for a simple animation.

It is important to remark that the idea to use semi-straight lines to define a distance function is not completely new. A distance with exactly the same name ("interaction distance") was already introduced in the context of ballistics [6] (see Ch.24, Distances in Physics), but it differs from the definition given above for the domain of α (which is discrete in the definition above). Moreover, in the context of robotics, semi-straight lines were also used in [21] with the aim of verifying an entire set of intermediary configurations for robot arms, instead of only a few positions. This use of the semi-straight lines is evidently different from the one in the definition of interaction distance.

Since the interaction distance is, by definition, the smallest Euclidean distance between selected points on two straight lines, which also include the two points p_u^t and p_v^t, it follows that the interaction distance between two joints u and v at frame $t > 1$ is always smaller than, or at most equal to, the Euclidean distance between the current positions of the two joints at frame t.
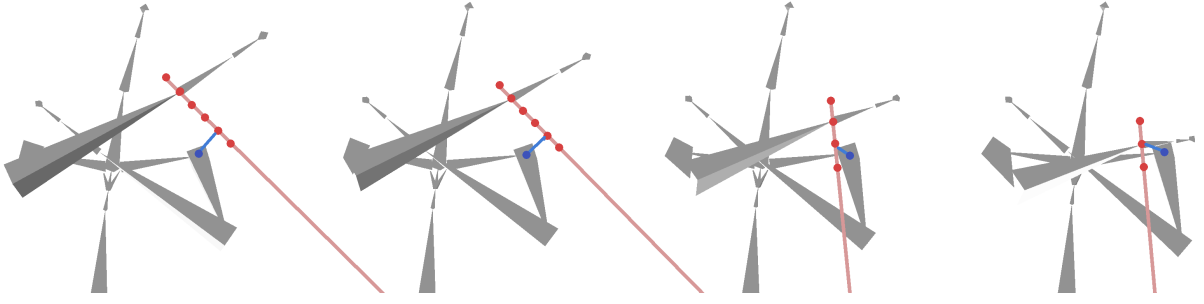
Рис. 2: *A graphical representation of the interaction distance between two joints (wrist of approaching arm in red, and steady opposite shoulder in blue) in a simple animation. Since one joint is steady, it is necessary to compute the semi-straight line (light red) only for one of the two joints. It is possible to notice that the interaction distance (blue segment) is able to "anticipate" the detection of potential joint contacts. The potential interaction is finally not completely attained, because the value of the interaction distance at the third shown frame is smaller than the Euclidean distance in the last frame, where the Euclidean and interaction distances coincide.*

PROPOSITION 2. *The interaction distance is not a metric, but rather a symmetric quasi-distance.*

Proof The three properties that a symmetric quasi-distance and a metric have in common can be easily verified:

- nonnegativity: $\forall t \in T, \forall u, v \in V, \quad D(p_u^t, p_v^t) \geqslant 0$;

- reflexivity: $\forall t \in T, \forall v \in V, \quad D(p_v^t, p_v^t) = 0$;

- symmetry: $\forall t \in T, \forall u, v \in V : u \neq v, \quad D(p_u^t, p_v^t) = D(p_v^t, p_u^t)$.

The other two additional properties of a metric are instead not satisfied. The interaction distance does not satisfy the "identity of indiscernibles" property, because a distance equal to 0 does not necessarily imply that p_u^t and p_v^t are actually the same. As the reflexivity property shows, in fact, the distance between one joint and itself is 0 (because the two straight lines ℓ_u and ℓ_v coincide), but the converse implication is not true in general: $D(p_u^t, p_v^t)$ can be equal to 0 when two distinct joints will collide when continuing to travel along their current trajectories in future frames (in other words, when the two straight lines ℓ_u and ℓ_v intersect in the common point where the collision occurs).

The second property of a metric that is not satisfied is the triangular inequality. Suppose that the joints u and v are fixed in two distinct positions during the animation, while the joint w "travels" on the straight line defined by p_u^t and p_v^t in their direction, so that $D(p_u^t, p_w^t)$ and $D(p_v^t, p_w^t)$ are both equal to 0. Let c be the Euclidean distance between p_u^t and p_v^t, which is strictly larger than 0 by hypothesis. The triangular inequality would imply:

$$D(p_u^t, p_v^t) \leqslant D(p_u^t, p_w^t) + D(p_w^t, p_v^t) \Rightarrow c \leqslant 0,$$

which brings to a contradiction. □

Notice that, for any given distance function $d(p_u^t, p_v^t)$, if the distance value $\hat{d}$ is known and one of the two points is fixed, say p_u^t, then the set

$$\Pi = \left\{ p \in \mathbb{R}^3 : d(p_u^t, p) = \hat{d} \right\}$$

contains all points p such that the distance between p_u^t and p is equal to $\hat{d}$. When d is the Euclidean distance, it is well-known that Π is a sphere centered in p_u^t and having radius $\hat{d}$. When the interaction distance is concerned, fixing p_u^t implies the definition of the semi-straight line ℓ_u, from which all potential *interaction points* $p(u, \alpha)$, for every α, can be extracted (because the step $p_u^t - p_u^{t-1}$ is known).

Let $\hat{D}$ be the known value for the interaction distance between the two joints u and v at the time step t. Every interaction point $p(u, \alpha)$ on ℓ_u corresponds to a sphere $\mathcal{S}_\alpha(p_u^{t-1})$ centered in $p(u, \alpha)$ and having radius $\hat{D}$. The points in Π are therefore those lying on the semi-straight lines passing through p_v^{t-1} and any other point of $\mathcal{S}_\alpha(p_u^{t-1})$, for any α. Once one semi-straight line ℓ_v is selected, the corresponding value of α can be deduced from the sphere $\mathcal{S}_\alpha(p_u^{t-1})$ that ℓ_v intersects. This value is necessary to set the intersection point α time steps after p_v^{t-1}: the position p_v^t corresponds to the point of ℓ_v placed only one time step further ($\alpha = 1$).

In the hypothesis that more than one distance constraint concerns the same point p_v^t (the unknown), so that the system of equations

$$\begin{cases} D(p_u^t, p_v^t) = \hat{D}_{uv}, \\ \dots \\ D(p_w^t, p_v^t) = \hat{D}_{wv}, \end{cases}$$

can be defined, the feasible points for p_v^t lie on the intersection of the several sets Π. Intuitively, this intersection is related to a set of semi-straight lines which do not only pass through the point p_v^t, but also through the points of $\mathcal{S}_\alpha(p_u^{t-1}) \cap \dots \cap \mathcal{S}_\alpha(p_w^{t-1})$, for every α. Since the value of α needs to be the same for every semi-straight line, each intersection concerns only the spheres related to the same α: the number of spheres to intersect, for every α, depends on the number of distance constraints (see Fig. 3 for a graphical representation in 2D). These sphere intersections may be for example empty (indicating that the target distance cannot be attained for the fixed value of α), or they could enjoy special properties, such as the ones studied in [18] in the context of distance geometry with the Euclidean distance. This is the current research direction.

5. Conclusions

Some initial empirical studies on the interaction distance computed for the joints of human-like character animations have shown that this new distance can, differently from the Euclidean distance, anticipate joint contacts and collisions of several frames before they actually occur (see Fig. 2). This is of fundamental importance in real-time applications, where the *future* is actually not known yet (differently from other applications where an animation may be preprocessed in order to find out its critical parts).

Even if the interaction distance is introduced in this article in the context of human motion adaptation, it is important to point out that the problem of "collision detection" arises in many other contexts, such as aircraft and robot trajectory planning, and crowd simulations, to name a few important examples. In some early studies of pedestrians collision avoidance [5], a short survey on previous works proposed a classification on the two main approaches to the problem: those which focus on the idea to predict *when* a collision is going to occur; the others where the main question is *whether* the collision is going to take place or not. The interaction distance introduced in this work makes a bridge between these two main approaches: small values of the distance indicates in fact that joints are going to approach each other; moreover, the value of α (see Def. 12) corresponding to the distance value indicates how many time steps are necessary for the contact to actually take place.

However, while the interaction distance already finds its interest from a mathematical point of view, the real practical benefits in using it in real-life applications still need to be verified.
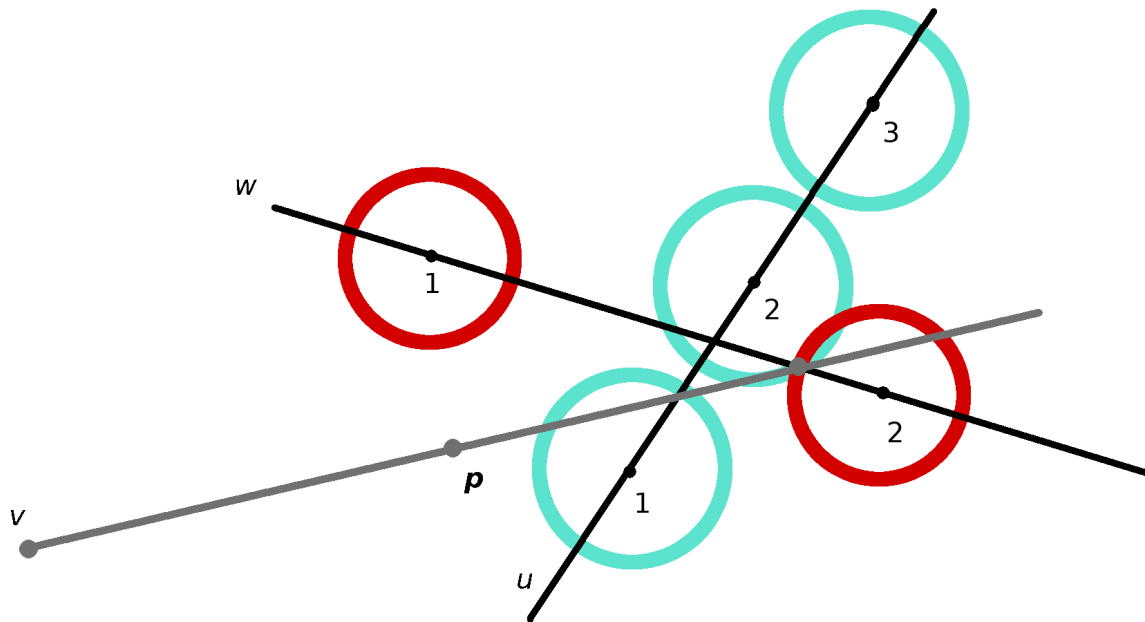
Рис. 3: *An example in 2D where the set of feasible semi-straight lines for v reduces to two lines when $\alpha = 2$ (only one of the two is reported in gray in the figure). The spheres $\mathcal{S}_\alpha(p_u^t)$ on ℓ_u, as well as the spheres $\mathcal{S}_\alpha(p_w^t)$ on ℓ_w, intersect in fact only when α is set to 2 (the values of α are reported inside the spheres). Once one semi-straight line ℓ_v is selected, the point p (the unknown) is the point on the line corresponding to $\alpha = 1$.*

In particular, the definition and solution of dynDGP instances related to human-like character animations, where the interaction distance replaces the classical Euclidean distance, will be the subject of future publications.

Dedication

This article is dedicated to the memory of Michel Deza (1939–2016). The reason to dedicating this work to him is not only because it is going to appear in a journal special issue where contributions from a recent conference, dedicated to his 80th birthday, are collected. The main reason is actually that most of this work was inspired by the reading of the wonderful Encyclopedia of Distances, that Michel authored with his wife.

REFERENCES

1. A.Y. Alfakih, *On Dimensional Rigidity of Bar-and-Joint Frameworks*, Discrete Applied Mathematics **155**(10), 1244–1253, 2007.

2. A. Bernardin, L. Hoyet, A. Mucherino, D.S. Gonçalves, F. Multon, *Normalized Euclidean Distance Matrices for Human Motion Retargeting*, ACM Conference Proceedings, Motion in Games 2017 (MIG17), Barcelona, Spain, November 2017.

3. K.J. Choi, H.S. Ko, *Online Motion Retargetting*, The Journal of Visualization and Computer Animation **11**(5), 223–235, 2000.

4. G.M. Crippen, *An Alternative Approach to Distance Geometry using L^∞ Distances*, Discrete Applied Mathematics **197**, 20–26, 2015.

5. J.E. Cutting, P.M. Vishton, P.A. Braren, *How We Avoid Collisions With Stationary and Moving Objects*, Psychological Review **102**(4), 627–651, 1995.

6. M.M. Deza, E. Deza, *Encyclopedia of Distances*, 756 pages, Springer, 2016.

7. B. Erdmann, *Reflexionen Kants zur Kritischen Philosophie*, Part II "Reflexionen Kants zur Kritik der Reinen Vernunft, aus Kants Handschriftlichen Aufzeichnungen", Leipzig, Fues's Verlag, 1884.

8. M. Gleicher, *Retargetting Motion to New Characters*. ACM Proceedings of the 25th annual conference on Computer Graphics and Interactive Techniques, 33–42, 1998.

9. W. Glunt, T.L. Hayden, M. Raydan, *Molecular Conformations from Distance Matrices*, Journal of Computational Chemistry **14**(1), 114–120, 1993.

10. E.S.L Ho, T. Komura, C-L. Tai, *Spatial Relationship Preserving Character Motion Adaptation*, Proceedings of the 37th International Conference and Exhibition on Computer Graphics and Interactive Techniques, ACM Transactions on Graphics **29**(4), 8 pages, 2010.

11. G. Laman, *On Graphs and Rigidity of Plane Skeletal Structures*, Journal of Engineering Mathematics **4**(4), 331–340, 1970.

12. L. Liberti, C. Lavor, N. Maculan, A. Mucherino, *Euclidean Distance Geometry and Applications*, SIAM Review **56**(1), 3–69, 2014.

13. L. Liberti, G. Swirszcz, C. Lavor, *Distance Geometry on the Sphere*, Proceedings of the Japanese Conference on Discrete and Computational Geometry and Graphs (JCDCGG15), Lecture Notes in Computer Science **9943**, J. Akiyama, H. Ito, T. Sakai Y. Uno (Eds.), Kyoto, Japan, 204–215, 2015.

14. M. Meredith, S. Maddock, *Motion Capture File Formats Explained*, Technical Report 211, Department of Computer Science, University of Sheffield, 36 pages, 2001.

15. A. Mucherino, *On the Discretization of Distance Geometry: Theory, Algorithms and Applications*, HDR Monograph, University of Rennes 1. INRIA Hal archive id: `tel-01846262`. July 17, 2018.

16. A. Mucherino, D.S. Gonçalves, *An Approach to Dynamical Distance Geometry*, Lecture Notes in Computer Science **10589**, F. Nielsen, F. Barbaresco (Eds.), Proceedings of Geometric Science of Information (GSI17), Paris, France, 821–829, 2017.

17. A. Mucherino, D.S. Gonçalves, A. Bernardin, L. Hoyet, F. Multon, *A Distance-Based Approach for Human Posture Simulations*, IEEE Conference Proceedings, Federated Conference on Computer Science and Information Systems (FedCSIS17), Workshop on Computational Optimization (WCO17), Prague, Czech Republic, 441–444, 2017.

18. A. Mucherino, C. Lavor, L. Liberti, *The Discretizable Distance Geometry Problem*, Optimization Letters **6**(8), 1671–1686, 2012.

19. A. Mucherino, C. Lavor, L. Liberti, N. Maculan (Eds.), *Distance Geometry: Theory, Methods and Applications*, 410 pages, Springer, 2013.

20. A. Mucherino, J. Omer, L. Hoyet, P. Robuffo Giordano, F. Multon, *An Application-based Characterization of Dynamical Distance Geometry Problems*, to appear in Optimization Letters, Springer, 2019.

21. F. Schwarzer, M. Saha, J-C. Latombe, *Exact Collision Checking of Robot Paths*. In: Algorithmic Foundations of Robotics V, Springer, 25–41, 2004.

22. J. Sivadjian, *Le Temps. Étude Philosophique Physiologique et Psychologique*, Part III "Le Problème Psychologique", Hermann & C^{le} (Eds.), 1938.