May 2000

Algorithms for Generating Motion Trajectories Described by Prepositions

Yilun Dianna Xu
University of Pennsylvania

Norman I. Badler
University of Pennsylvania, badler@seas.upenn.edu

Recommended Citation

Xu, Y., & Badler, N. I. (2000). Algorithms for Generating Motion Trajectories Described by Prepositions. Retrieved from http://repository.upenn.edu/hms/46

Copyright 2000 IEEE. Reprinted from *Proceedings of the Computer Animation 2000 Conference*, May 2000, pages 30-35.
Publisher URL: http://dx.doi.org/10.1109/CA.2000.889029

Algorithms for Generating Motion Trajectories Described by Prepositions

Abstract
A list of representative directional prepositions of the English language is investigated to develop computation models that output some general motion trajectory or goal direction, given instructions involving prepositional phrases. Computation models are implemented through geometric definitions and procedures such as: centroid, quasi-centroid, convex-hull, closest, nearest-neighbor, and next-to. All algorithms are defined by or derived from standard computational geometry concepts.

Comments
Copyright 2000 IEEE. Reprinted from *Proceedings of the Computer Animation 2000 Conference,* May 2000, pages 30-35.
Publisher URL: http://dx.doi.org/10.1109/CA.2000.889029

Algorithms for Generating Motion Trajectories Described by Prepositions

Yilun Dianna Xu Norman I. Badler [*]

University of Pennsylvania

Abstract

A list of representative directional prepositions of the English language is investigated to develop computation models that output some general motion trajectory or goal direction, given instructions involving prepositional phrases. Computation models are implemented through geometric definitions and procedures such as: centroid, quasi-centroid, convex-hull, closest, nearest-neighbour, and next-to. All algorithms are defined by or derived from standard computational geometry concepts.

1 Introduction

Understanding spatial and directional prepositions is a complex cognitive task. It is a long standing problem and extensive research has been conducted in linguistics. We approach the interpretation of prepositions constructively rather than analytically.

In any graphics environment where human agents are present, the need of some motion guidance system arises naturally. We investigate a set of representative prepositions in English in an attempt to develop computational models that interpret them and output some general motion trajectory or goal direction. At this level, the interpretation does not involve collision avoidance. The trajectories computed are global directives: collision avoidance will take place locally, e.g., if the agent detects obstacles, it will take a detour to avoid collision but will go back on track once the obstacle is cleared. The situation studied typically involves two objects or agents, with few exceptions, and the space surrounding and between the two is assumed empty. Furthermore, objects are all assumed to have rigid bodies. The algorithms take a purely geometric approach, making use of many well-known computational geometry algorithms. It is our belief that despite the complexity of prepositions in natural language, in this particular application, relying entirely on geometric details is sufficient.

2 Related Work

Extensive work has been done on prepositions in the fields of linguistics, natural language processing, and geometric spatial modeling. Most of these studies focus on semantic and pragmatic analysis of locative prepositions, with close attention to the roles of prepositions in language and cognition. Bennett [4] and Herskovits [9] produced early linguistic space studies. The domain that spatial prepositional phrases define in natural language turns out to be extremely complex, as these authors and others clearly demonstrated. Our study is only concerned with what Herskovits called "ideal meanings" [9] of the prepositions: they have "normal"/"typical" meanings and we do not concern ourselves with usages that deviate from the norm either due to convention or pragmatic allowance. Our goal is to provide some generic guidance system for an agent or avatar living in a 3D graphics environment. In such environments, natural language tends to be used in *instructions* (rather than as *desciptions*) in which prepositional phrases are typically short and precise.

2.1 Description of a Scene

How to describe a scene to a graphics application is probably the most researched topic in language-based graphics applications. Understanding spatial prepositions is of great importance in this process. The Put placement system [6] proposed "progressively refining" a placement region based on well-established image schemas in cognitive linguistics. In particular, the spatial relations defined by prepositions "in", "on", "at", and others (alone or in a series) were studied and resolved to some final-location mappings. Fraczak [8] proposed a model to generate graphics models called "Mental Maps" from route descriptions, where typical prepositions in directions like "left of", "right of", "out of", "past", etc were considered. In this model, the "objects" in the environment are considered "landmarks" and individual geometric shapes are ignored.

---

[*] contact: badler@central.cis.upenn.edu

2.2 Scene Description

The other side of the description problem starts with a given scene and tries to generate appropriate and reasonable textual descriptions. Prepositions are again an important issue. The German dialogue system CITYTOUR from Andre et al. [2, 1] investigated the problem of describing spatial relations between physical objects (city buildings), when seen by a dynamic observer (the tour bus). Motion trajectories were analyzed in an effort to decide whether 2 objects satisfy relations defined by prepositions such as "past" and "along".

2.3 Motion Prepositions

Most of the prepositional relations studied are locative and stationary. Herskovits [10] discussed motion prepositions briefly in her study of schematization, where she pointed out that primarily only linear paths are involved, although in a few cases special verbs complicate matters. Edwards and Moulin [7] also mentioned motion prepositions in their study of modeling spatial relations with Voronoï diagrams. In particular, topological configurations of "around", "beyond" "through" and "toward" were given.

3 Schematization

Our study is not interested in the meaning nor the spatial/geometric description the prepositions imply, but the motions or motion trajectories they elicit. Therefore, even though most of the prepositions are polysemous, for us only the dynamic sense of such propositions is important. For example, "on" necessarily translates to "onto". Most of the prepositional phrases we study are not complicated and their meanings or uses are unambiguously defined. They typically take the form of "A verb preposition B", where A is the primary moving object (the agent), some literature calls A the *query object*, or the *Figure*. The preposition depicts the motion trajectories and B is often called the *reference object* or *Ground*, mostly stationary, with the exception of phrases like "A runs after B".

In this study, we have made a number of idealizations as deemed appropriate by our particular application. Talmy [12] presented a detailed study of linguistic *schematization*: the process of abstracting a reality into some manipulatable graphics objects and representations. It is clear that not all of the original geometry can be or needs to be preserved.

Rigid body motions consist of rotation and translation. Since our primary moving object is an agent and has humanoid motor skills, the local coordinate frame remains fixed in the vertical y axis in all normal situations when balance is not lost or deliberately sacrificed. Thus no true rotation occurs, and rotation around any center other than the rotating object's own centroid is essentially a series of translations. Rotation about the object's own centroid is never implied by prepositions, but by verbs such as "dance, spin, swirl, roll, revolve" etc. These verbs, of course can also combine with motion prepositions and define motions that consist of both rotation and translation, e.g. "The girl danced and swirled across the street". We observe that the rotation does not effect the translation trajectories, that is, the motion trajectories implied by preposition "across" remains the same whether the associated verb implies self-rotation or not. Therefore we conclude that the rotation does not play a role in motion trajectories implied by prepositional phrases, and when we say motion from now on, we mean translations only.

We also recognize that although we conduct our studies in the 3D world, with most of the prepositions that imply motion in natural language the third dimension is irrelevant. The motion trajectories occur in the 2D plane, with the third dimension staying constant. This is of course due to two facts: one, that gravity typically constrains us to periodic or constant contact with the ground plane, and second, that we cannot maintain balance if our vertical axis ever becomes more than a little tilted. There is a short list of exceptions with verbs like "climb, jump, lift, land" and prepositions like "up, upwards, down, downwards" which intrinsically imply changes in altitude. But also in these cases, due to the same constraint on the angle of body vertical, the motion primarily consists of changes in the altitude, as in "climb up"or "jump down". The motion trajectory is still approximated by a linear path. The only preposition that implies a real space curve is the preposition "over", especially when contact is required. We will look at "over" later.

Talmy [12] claimed that prepositions frequently treat the query object as a point or some other "related simple form". Herskovits [10] pointed out that this is an intuition that is not always true. However, all the exceptions seem to occur only when the prepositions involved are stationary. In the case of motion prepositions, since we established that only rigid body translations are involved, abstracting the moving object to a point will not cause too much loss of information. In a few cases one does need an extra parameter. For example, when dealing with "through", one needs to know the width of the moving object in order to determine whether "through" is at all feasible. Herskovits [10] states that "a rigid body undergoing translation has the shape of a generalized cylinder". Thus we need the diameter of this generalized cylinder.

4 Prepositions in the 2D plane

We then start by looking at the prepositions with our 3D world projected onto the 2D plane. First, some useful definitions:

Definitions:

- centroid(O) is the center of gravity of the object O, for all O, such that O is a simple polyhedron with uniformly distributed weights. It is computed as the $Barycenter(V(O))$, where $V(O)$ is the set of vertices of O, and $Barycenter$ is defined as:
$$\sum_{i=1}^{n} \lambda_i v_i$$
where $n = |V(O)|$, $v_i \in V(O)$ and $\lambda_1 = \lambda_2 = ... = \lambda_n = 1/n$.
- quasi-centroid(X,O) is the centroid of object O with respect to the minimum extent of O from point X.

  So if one is standing at the side of an infinitely long road and wants to go across, the quasi-centroid relative to where one is standing will be the midpoint of the road right in front. The other infinite extent is excluded.

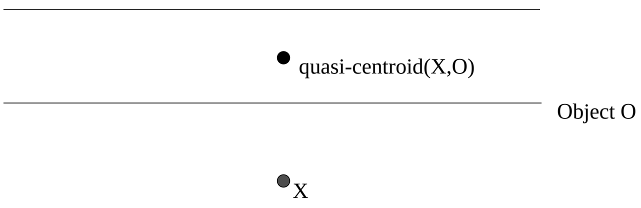


Figure 1: quasi-centroid(X,O).

- ref(O) is a reference point on object O. In all the following cases, unless otherwise specified, we pick the centroid(O) or centroid((convexhull(O)), if needed, to be the reference point.

The query object is approximated to a point, and we can assume it is the centroid of the object, or some other more convenient reference point, when the object is unusually shaped. We will call this point the query point from now on. The geometric details of the reference object is of crucial importance, as many prepositional motion trajectories largely depend on the shape of the reference object. Therefore its entire 2D outline is needed as input. As it is a closed polygonal chain, a list of all vertices in order (we choose counterclockwise) is sufficient to represent it. We make use of a database PAR (Parameterized Action Representation) [3] which stores all relative information on all objects and agents in a scene. The 2D outline of an object is stored as a parameter of that particular object and can be quickly retrieved by a query. Similarly, we also preprocess all orientation parameters such as if the object has a front, or a top, etc, and store the front/top face. Notice that this decision problem is not quite as complex as the one encountered in a placement problem such as discussed in [6], as our query object is a moving human agent. That means the existence of a top face is simply a large enough and mostly horizontal surface that a human being can maintain balance on, and the front face is a matter of whether the object has a predominate front face (such as a building with a front door). We also store the height of the reference object.

We make use of many of the standard computational geometry definitions, algorithms and implementations dealing with 2D points, lines and polygons, such as convex hull, centroid, quasi-centroid, nearest neighbor and point in polygon [11].

Surprisingly, except for some more complex ones, motion trajectories defined by most prepositions are fairly simple and straight forward. We will start by looking at some of the more interesting ones first. We assume that we always have the relations as “A verb preposition B”, with A as the query object, and B the reference object.

4.1 CLOSE TO/NEAR/NEXT TO

We discuss this relation first, because almost before any motion preposition can be executed, one has to be “near” or “close to” the reference object. In fact, if the agent is far away from the reference object, any instruction involving motion prepositions practically translates to an extra “go close to reference oject”. For example, if the instruction is “Go around the table”, and agent is far way from the “table”, the instructions splits to “Go close to the table” then “Go around the table”.

The motion trajectory implied by this relation is straight forward. Given the assumption that there are no obstacles, it is obviously just a linear path from the query point to the centroid of the reference object. The question is then, when does one stop? Or rather, how close is “close”?

It’s clear that going all the way until the centroid of the reference object is too far. In fact, “close to” does not imply contact, and one would have to stop before reaching the outline of the reference object. A simple intersection of the line from the query point to the centroid of the reference object and the outline of the reference object gives us the contact point.

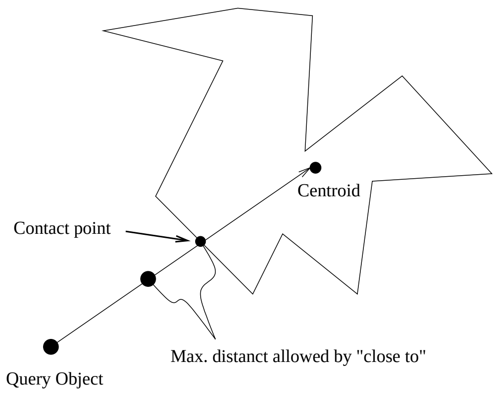


Figure 2: Close to

The interesting question is, how close to the contact point should one make the stop? It seems that "closeness" is not a strict function of distance, and one can not say that 10 inches apart is close and 20 is not. It seems that "closeness" depends at least to a certain extent on the relative size difference between the query object and the reference object. For example, we can say that we are standing "close to the World Trade Center" even when we are 10 feet away from it. But in order to make "close to a table" valid, we would need to be no more than a couple inches away from it. From the above observation, there is a proportional relationship: the larger the reference object is in regard to the query object, the more the maximum distance is allowed by "close to". The increase has to be significant, at least in an order of a magnitude. It seems that height is the slightly more dominant factor, perhaps due to human psyche, but not without support from some increases both in length and width. Any difference, even if significant, in a single dimension doesn't seem to suffice, e.g. "close to a pole", "close to a cable".

Since our query object is an agent of average adult size, and the increase has to be significant to make a difference, an estimated average will work for all agents. A function is established to calculated the maximum distance allowed by "close to", based on the dimensions of the reference object.

Another interesting point occurs when the reference object is another human agent, "closeness" in this case can very well be cultural and goes into the domain of "personal space" and social sience, which we will not elaborate here.

4.2 AROUND

The optimal trajectory implied by around is the convex hull of our reference object B, although it may not be the most human-like trajectory, as in most cases, we tend to round corners and would never move quite so geometrically. Therefore we attempt to fit first a circle, anchored at B's centroid, its radius the longest distance from the centroid to any points on B's outline. When B has a mostly convex shape, and its extremes are well balanced, a circle approximates quite well.

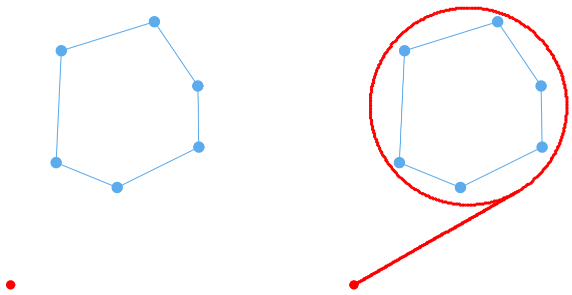


Figure 3: Around with circle

It is clear that when B is much longer in some dimension, the circle approach would be inaccurate. We determine whether or not we have a good fit by calculating the area of the circle against the area of the outline of B. If the circle turns out to be a bad fit, we next try an axes-aligned bounding box. When that also fails, we then employ the 2D convex hull.

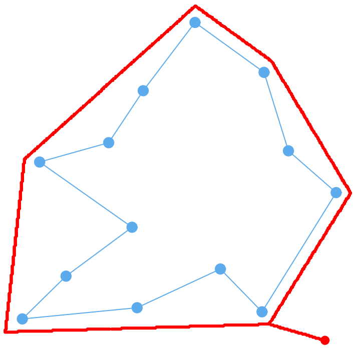


Figure 4: Around with convex hull

Geometry versus Natural Human Behavior

One might have noticed in the above example, that although the convex hull is the optimal trajectory implied by preposition "around", nevertheless the path appears too geometric and consequently unnatural. It's a natural tendency for any human to "cut the corners" while walking, thus any unforced sharp corner-turns appear robot-like. Such corners, can be easily eliminated by the use of a B-spline curve, using the corner-points as the control points of the spline, calculated with some subdivision algorithm, such as de Boor. How well this method works of course depends on the complexity of the original trajectory, i.e. the number of corners, and the desired smoothness, i.e. level of subdivision. So far tests have been conducted on "along", with satisfactory results.

The other behavior we notice, is that human beings tend to not decide their entire trajectories before they start walking, even if the goal object is in plain view and such a motion-planning is possible. We tend to make up our minds on the fly. This is certainly not optimal, but it is natural. Thus, we are also investigating methods to emulate such behaviors and decide our course one step at a time.

4.3 ALONG

Again, we need walk close to the reference object if we are far away from it, on which we will no longer elaborate.

Along requires that the reference object has a distinct "long" side. For example, "walk along the tree" is not acceptable. This "long" side can either be a straight long edge, as in "walk along the wall", or a series of edges that form some general "long" side with some gentle turns, as in "walk along the river". It is not difficult to identify this "long" side if it exits and store it. In many cases an object that has one "long" side also has another, e.g. "road" "river", etc. Therefore we also point out that further more, this "long" side must also be clearly visible when the instruction involving "along" is given.

Now once we query the database and obtain our "long" side, "along" should be easily computed. Ideally, "along" means to exactly trace the outline of this "long" side. But again in real life, no one walks so geometrically. Thus, we use the vertices that define the "long" side as the control polygon and compute a B-spline using the de Boor subdivison algorithm to obtain a B-spline. This curve still follows the general shape of the "long side", but is much smoother. We also check the angle at every vertex of the control polygon, if the angle is too sharp, then we drop it from the control polygon, so as to simulate the "rounding the corner" effect.

4.4 ACROSS

Across is one of the vague prepositions in the natural language. Ask 10 people to walk "across" a large enough region, 10 different trajectories will be presented. A phrase like "the tree across the river" is almost never enough to pin point any one tree, unless there is only a single tree in that environment. Therefore, there is a fair amount of freedom in picking the paths when it comes to computing trajectories for "across". One thing is certain is that "across" implies some passage through the "center" of the reference object, and to the "other side" of the reference object. We use the centroid or quasi-centroid of the reference object, whichever is more appropriate, as the center, and we define the "other side" as the area pointed to by the directional vector that goes from the query object's current position to the centroid of the reference object.

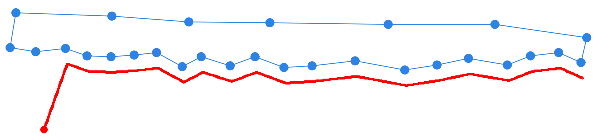


Figure 5: Along the river

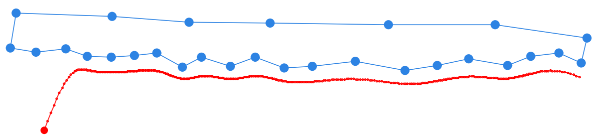
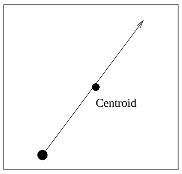


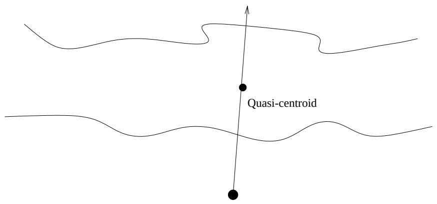
Figure 6: Smoother trajectories with splines

Trajectories implied by "across" divide into two types. It depends on whether the query object is inside the reference object or not. When the query object is physically inside the reference object, for example in "walk across the room" or "run across the baseball field", the query object is expected to move beyond to the opposite side, but remain inside the reference object. When we walk across a room, we don't go out of that particular room. When the query object is not physically inside the reference object, for example in "walk across the street" or "wade across the river", the query object is expected to enter the reference object, move beyond to the opposite side, and again leave the reference object.

Whether the query object is inside the reference object or not is easily determined by applying a simple "in polygon" algorithm from standard computational geometry libraries. Thus we can determine whether the trajectories end inside or outside of the reference object. The goal position is again checked with repeated application of the "in polygon" routine to make sure that the final destination is indeed on the



Across the room



Across a river

correct side.

5 Preposition in 3D

5.1 OVER

Over is probably the most complex motion preposition. It divides into two categories, contact and non-contact. Whether the usage is contact or non-contact in a specific instruction depends completely on the verb. The list of verbs that implies non-contact "over" is also the one that implies altitude change, for example "jump over the stone". This list is very short for human agents, since we are physically unable to maintain afloat in the air for more than a fraction of a second. The motion trajectory in such case is a fairly simple parabola, whose apex is slightly higher than the highest point of the reference object.

The contact case is much more complex, since it implies a true 3D space curve, as in "walk over the mountain". In this case, the motion trajectory is a geodesic on the curved surface of the reference object. There are also many possible paths to choose from, and the optimal is obviously the shortest path. Calculating shortest paths from a single source on the surface of a polyhedron is non-trivial. The best algorithm known today due to Chen & Han [5] has the complexity of $O(n^2)$ time and $\Theta(n)$ space, n being the number of vertices.

Most shortest path algorithms require that the polyhedron be triangulated beforehand. The complexity of triangulation is $n \log_n$, and it will have to be applied to each face of the polyhedron. After the triangulation, the main complexity of programming lies in the fact that each time one crosses an edge, one has to switch coordinate system.

6 Conclusions and Future Work

Currently we have implemented algorithms for "across", "along", "around", "awayfrom", "close to/next to" and "towards", which are also the more interesting and challenging ones in English. We will continue further implementations of the algorithms for the other, often simpler prepositions, and for "over", which is a bit more involved. These definitions will provide a library of robust routines that can be easily incorporated into a 3D agent animation system.

References

[1] E. Andre, G. Bosch, G. Herzog, and T. Rist. Characterizing trajectories of moving objects using natural language path descriptions. Technical report, Universität des Saarlandes, D-6600 Saarbrücken 11, West Germany, 1986.

[2] E. Andre, G. Bosch, G. Herzog, and T. Rist. Coping with the intrinsic and deictic uses of spatial prepositions. Technical report, Universität des Saarlandes, D-6600 Saarbrücken 11, West Germany, 1986.

[3] N. Badler, M. Palmer, and R. Bindiganavale. Animation control for real-time virtual humans. *Communications of ACM*, pages 64–73, 1999.

[4] David C Bennett. *Spatial and Temporal Uses of English Prepositions: An Essay in Stratificational Semantics.* Longman Group Limited London, first edition, 1975.

[5] J. Chen and Y. Han. Shortest paths on a polyhedron. In *Proc. 6th Annu. ACM Sympos. Comput. Geom.*, pages 360–369, 1990.

[6] Wilhelms Clay. Put: Language-based interactive mapnipulation of objects. *IEEE Computer Graphics and Applications*, 16(2):31–38, 1996.

[7] Geoffrey Edwards and Bernard Moulin. Towards the simulation of spatial mental images using the voronoï model. In Gapp Olivier, editor, *Presentation and Processing of Spatial Expressions*, pages 163–184. Laurence Erlbaum Associates, 1998.

[8] Lidia Fraczak. Generating "mental maps" from route descriptions. In Gapp Olivier, editor, *Presentation and Processing of Spatial Expressions*, pages 185–200. Laurence Erlbaum Associates, 1998.

[9] Annette Herskovits. *Language and Spatial Cognition: An Interdisciplinary Study of the Prepositions in English.* Cambridge University Press, first edition, 1986.

[10] Annette Herskovits. Schematization. In Gapp Olivier, editor, *Presentation and Processing of Spatial Expressions*, pages 149–162. Laurence Erlbaum Associates, 1998.

[11] Joseph O'Rourke. *Computational Geometry in C*. Cambridge University Press, first edition, 1995.

[12] Leonard Talmy. How language structures space. Technical report, Institute of Cognitive Studies, University of California at Berkley, 1983. Also appears in Herbert Pick and Linda Acredolo (Eds.), Spacial Orientation: Theory, Research, and Application, Plenum Press, 1983.