A DE CASTELJAU ALGORITHM FOR GENERATING OBJECT SHAPE-DEPENDENT FREEFORM PLANAR MOTIONS

Huan Liu, Qiaode Jeffrey Ge
Department of Mechanical Engineering, Stony Brook University, Stony Brook, NY, U.S.A.
Email: huan.liu.1@stonybrook.edu; Qiaode.Ge@stonybrook.edu

ABSTRACT

This paper follows our recent work that incorporates the shape of a bounded object in formulating the distance between two positions of the object as well as interpolating motions between them. In this paper, we show that such a shape-dependent interpolating motion can be applied recursively in the framework of de Casteljau algorithm for the generation of Bézier-like freeform motions. Examples are presented to illustrate the effect of the object and the weight factors on the path of the resulting Bézier-like motions.

Keywords: Shape-dependent distance measure; de Casteljau algorithm; motion interpolation, Bézier motion.

UN ALGORITHME DE DE CASTELJAU POUR GÉNÉRER DES MOUVEMENTS PLANAIRES DE FORME LIBRE DÉPENDANT DE LA FORME D'UN OBJET

RÉSUMÉ

Cet article fait suite à nos travaux récents qui intègrent la forme d'un objet délimité dans la formulation de la distance entre deux positions de l'objet ainsi que les mouvements d'interpolation entre eux. Dans cet article, nous montrons qu'un tel mouvement d'interpolation dépendant de la forme peut être appliqué récursivement dans le cadre de l'algorithme de de Casteljau pour la génération de mouvements de forme libre de type Bézier. Des exemples sont présentés pour illustrer l'effet de l'objet et des facteurs de poids sur la trajectoire des mouvements de type Bézier résultants.

Mots-clés : Mesure de distance dépendante de la forme ; algorithme de Casteljau ; interpolation de mouvement, mouvement de Bézier.

1. INTRODUCTION

Kinematic theories such as rigid transformations, Lie Groups, and quaternions are mostly concerned with motions of unbounded infinite spaces and their trajectories traced out by idealized geometric elements such as points, lines and planes. When dealing with practical kinematic problems, however, there is often a strong interplay between object shape and motion and thus the need for taking into account some aspects of the object shape and size, not just the position and orientation of the moving space that contains the object. For example, motion animation in computer graphics is concerned with the visual representation of the deformation and movement of objects with bounded shapes and sizes. Swept volume analysis, which plays a key role in NC tool motion simulation and collision avoidance in robot motion planning, is another example where the outcome is determined by a combination of a rigid-body motion and the shape and size of an object.

Kazerounian and Rastegar [1–3] were the first to consider the notion of "object norm" as the average distance of all points of a moving object between two positions. In this way, they obtain a distance function that includes not only the distance of translation, the angle of rotation but also the area moments of inertia. This concept of object shape-dependent distance measure was recently extended and adapted to the development of planar motion interpolants between two object positions where the area moments of inertia of the object contribute to the path of the resulting motion [4].

The purpose of this paper is to show that such a shape-dependent interpolating motion can be applied recursively in the framework of the de Casteljau algorithm for the generation of Bézier-like freeform motions. Examples are presented to illustrate the effect of the object and the weight factors on the path of the resulting Bézier-like motions. The research described in this paper can therefore be considered as an extension of methods in the field of Computer Aided Geometric Design (or CAGD) to freeform motion generation in kinematics. To this end, Shoemake [5] was the first to develop the so-called "spherical-linear interpolation" or *Slerp* by extending the notion of linear interpolation to a unit hypersphere defined by unit quaternions of $SO(3)$. He then combined *Slerp* with the de Casteljau algorithm to generate a Bézier-like curve on the hypersphere that corresponds to a spherical Bézier motion. Kim et al. [6] extended Shoemake's work to include higher order derivatives. This scheme was later extended to dual quaternions for generating dual hyperspherical curves for the generation of spatial Bézier motions [7]. Ge and Ravani [8] provided a formulation for designing rational Bézier motions using dual quaternions. This was extended to rational B-spline motions in [9–11]. A comprehensive review for rational motion design can be found in [12]. Another school of thought for combining CAGD with motion synthesis is to use exponential coordinates of Lie groups instead of quaternions and dual quaternions [13–16]. None of these works, however, took into account the object shape in motion generation.

The organization of the paper is as follows. Section 2 reviews the concept of shape-dependent distance measure. Section 3 presents shape-dependent motion interpolants for two positions of an object with a shape parameter. Section 4 shows how de Casteljau algorithm can be combined with shape-dependent motion interpolants to generate Bézier-like freeform motions. Section 5 presents two examples to illustrate the properties of the resulting motions.

2. SHAPE-DEPENDENT DISTANCE MEASURE

Consider a simply connected planar object with an arbitrary shape (Figure 1). Let I_u, I_v denote the area moments of inertia along the principal directions. It has been shown in [4] that for planar displacements, the shape-dependent distance measure introduced in [1] depends only on I_u, I_v and not on the detailed shape of the object. In this way, when studying shape-dependent distance, we can consider any simply connected planar object to be equivalent to the inertia ellipse defined by I_u and I_v.

Now consider two positions of an object shown in Figure 2. Let $O_1U_1V_1$ and $O_2U_2V_2$ be the principal

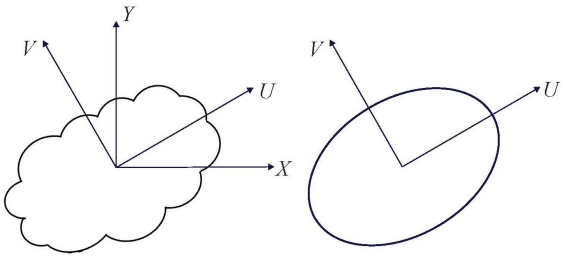

Fig. 1. A planar object and its inertia ellipse.

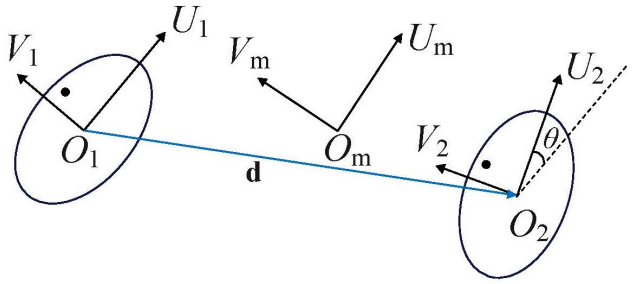

Fig. 2. Two positions of a planar object along with an interpolated position with its PCF. Note that the object shape is not limited to ellipse and can be of any simply connected shape.

coordinate frames of the object at its initial and final positions. The relative displacement of $O_2U_2V_2$ with respect to $O_1U_1V_1$is defined by the vector $\mathbf{d} = (d_u, d_v)$ connecting the two positions of the centroid of the ellipse and the angle between the axes U_1 and U_2 is denoted as θ. The average distance between all points of the object at the two given positions is defined as [1, 2]:

$$D^2 = \frac{1}{S}\iint_S (\Delta u^2 + \Delta v^2)\, dS \tag{1}$$

where S is the total area of the object, dS is an infinitesimal element of area and $\Delta u, \Delta v$ are the displacements of dS in U_1 and V_1 directions, respectively.

Let k_u and k_v ($k_u \leq k_v$) be the radii of gyration, i.e.,

$$I_u = k_u^2 S, \quad I_v = k_v^2 S. \tag{2}$$

It has been shown in [4] that the distance measure (1) reduces to:

$$D^2 = d^2 + 4k^2 \sin^2\frac{\theta}{2}, \quad \text{where } d^2 = d_u^2 + d_v^2,\ \ k^2 = k_u^2 + k_v^2. \tag{3}$$

Let a and b ($a \geq b$) be the length of semi-major and semi-minor axis of the inertia ellipse. The radii of gyration for the ellipse can be expressed as:

$$k_u = \frac{b}{2}, \quad k_v = \frac{a}{2}. \tag{4}$$

Substituting Eqs.(4) into Eq. (3), the distance measure is rewritten as:

$$D^2 = d^2 + (a^2 + b^2)\sin^2\frac{\theta}{2}. \tag{5}$$

Eq.(5) indicates that shape-dependent distance D^2 is a combination of the translational distance d and the rotation angle θ but weighted by the shape parameter a^2+b^2.

The shape-dependent distance D^2 can be further decomposed into two components in the U_1 and V_1 directions by separating D^2 into D_u^2 and D_v^2 such that

$$D^2 = D_u^2 + D_v^2, \tag{6}$$

where

$$D_u^2 = \frac{1}{S}\iint_S \Delta u^2\, dS, \quad D_v^2 = \frac{1}{S}\iint_S \Delta v^2\, dS. \tag{7}$$

Following a procedure similar to that from (1) to (5), we obtain ([4]):

$$\begin{aligned} D_u^2 &= d_u^2 + (b^2\cos^2\frac{\theta}{2} + a^2\sin^2\frac{\theta}{2})\sin^2\frac{\theta}{2}, \\ D_v^2 &= d_v^2 + (b^2\sin^2\frac{\theta}{2} + a^2\cos^2\frac{\theta}{2})\sin^2\frac{\theta}{2}. \end{aligned} \tag{8}$$

Introducing the angle ϕ such that

$$\cos^2\phi = \frac{1}{a^2+b^2}(b^2\cos^2\frac{\theta}{2} + a^2\sin^2\frac{\theta}{2}), \;\; \sin^2\phi = \frac{1}{a^2+b^2}(b^2\sin^2\frac{\theta}{2} + a^2\cos^2\frac{\theta}{2}), \tag{9}$$

Eq. (8) can be alternatively expressed as

$$\begin{aligned} D_u^2 &= d_u^2 + (a^2+b^2)\cos^2\phi\sin^2\frac{\theta}{2}, \\ D_v^2 &= d_v^2 + (a^2+b^2)\sin^2\phi\sin^2\frac{\theta}{2}. \end{aligned} \tag{10}$$

3. INTERPOLATING TWO POSITIONS

It is assumed that the orientation of an object is independent of its shape so the following linear interpolation without shape parameters is used:

$$\theta_m(t) = t\theta. \tag{11}$$

First, let us consider an interpolating function $f(t)$ for the shape-dependent distance D, i.e., we let $D_m(t) = f(t)D$ where $f(0)=0$ and $f(1)=1$. In view of (5), we have

$$D_m^2(t) = f^2(t)(d^2 + (a^2+b^2)\sin^2\frac{\theta}{2}) = d_m^2(t) + (a^2+b^2)\sin^2\frac{t\theta}{2}. \tag{12}$$

By equating each corresponding term, Eq.(12) leads to

$$d_m(t) = f(t)d, \quad \sin\frac{t\theta}{2} = f(t)\sin\frac{\theta}{2}. \tag{13}$$

Thus, the interpolant and distance functions are determined as follows:

$$f(t) = \frac{\sin\frac{t\theta}{2}}{\sin\frac{\theta}{2}}, \tag{14}$$

and

$$D_m(t) = \frac{\sin\frac{t\theta}{2}}{\sin\frac{\theta}{2}}D, \quad d_m(t) = \frac{\sin\frac{t\theta}{2}}{\sin\frac{\theta}{2}}d. \tag{15}$$

This means that the same interpolant (14) works for both the shape-dependent distance function $D_m(t)$ as well as the translational distance function $d_m(t)$.

Now turn our attention to the problem of finding the interpolating functions $f_u(t)$ and $f_v(t)$ such that

$$D_{mu}(t) = f_u(t)D_u, \quad D_{mv}(t) = f_v(t)D_v, \tag{16}$$

where D_u and D_v are two shape-dependent distance components in U and V directions, respectively, as given by (8). The end conditions for the interpolation (16) are

$$D_{mu}(0) = 0, \;\; D_{mv} = 0; \;\; D_{mu}(1) = D_u, \;\; D_{mv}(1) = D_v. \tag{17}$$

A procedure, which is similar to that from (12) to (15), can be applied to interpolate two shape-dependent distance components. And it has been shown in [4] that the following interpolating functions

$$\begin{aligned} f_u(t) &= \frac{\sin\frac{t\theta}{2}\sqrt{b^2\cos^2\frac{t\theta}{2} + a^2\sin^2\frac{t\theta}{2}}}{\sin\frac{\theta}{2}\sqrt{b^2\cos^2\frac{\theta}{2} + a^2\sin^2\frac{\theta}{2}}}, \\ f_v(t) &= \frac{\sin\frac{t\theta}{2}\sqrt{b^2\sin^2\frac{t\theta}{2} + a^2\cos^2\frac{t\theta}{2}}}{\sin\frac{\theta}{2}\sqrt{b^2\sin^2\frac{\theta}{2} + a^2\cos^2\frac{\theta}{2}}}, \end{aligned} \tag{18}$$

work for both (16) and $\mathbf{d}_m = (d_{mu}(t), d_{mv}(t))$ where

$$d_{mu}(t) = f_u(t)d_u, \quad d_{mv}(t) = f_v(t)d_v. \tag{19}$$

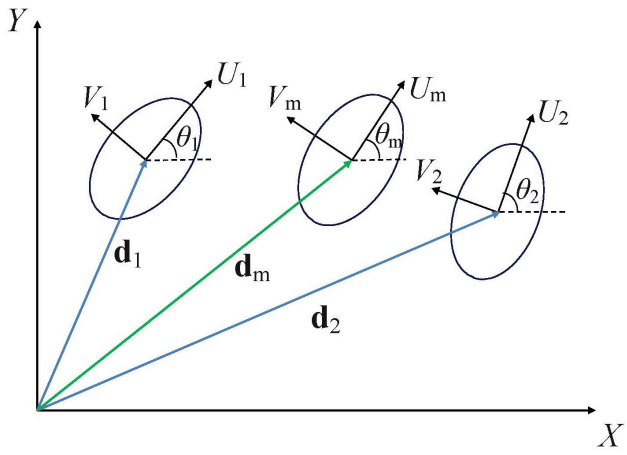

Fig. 3. Two positions of a planar object along with an interpolated position with respect to a fixed coordinate frame.

When a general fixed frame is used, such that U_1-axis makes an angle θ_1 and U_2-axis makes an angle θ_2 relative to its X-axis (Figure 3), then we have, for the angular interpolation:

$$\theta_m(t) = (1-t)\theta_1 + t\theta_2 = \theta_1 + t\theta_{12} \tag{20}$$

where $\theta_{12} = \theta_2 - \theta_1$. The interpolations of the translational components, $d_{mx}(t)$ and $d_{my}(t)$, become

$$\begin{aligned} d_{mx}(t) &= d_{1x} + f_x(t)(d_{2x} - d_{1x}), \\ d_{my}(t) &= d_{1y} + f_y(t)(d_{2y} - d_{1y}), \end{aligned} \tag{21}$$

where

$$f_x(t) = \frac{\sin\frac{t\theta_{12}}{2}\sqrt{e^2\cos^2(\theta_1+\frac{t\theta_{12}}{2})+\sin^2(\theta_1+\frac{t\theta_{12}}{2})}}{\sin\frac{\theta_{12}}{2}\sqrt{e^2\cos^2(\theta_1+\frac{\theta_{12}}{2})+\sin^2(\theta_1+\frac{\theta_{12}}{2})}},$$
$$f_y(t) = \frac{\sin\frac{t\theta_{12}}{2}\sqrt{e^2\sin^2(\theta_1+\frac{t\theta_{12}}{2})+\cos^2(\theta_1+\frac{t\theta_{12}}{2})}}{\sin\frac{\theta_{12}}{2}\sqrt{e^2\sin^2(\theta_1+\frac{\theta_{12}}{2})+\cos^2(\theta_1+\frac{\theta_{12}}{2})}}. \tag{22}$$

where $e = b/a \leq 1$. When $e = 1$, Eq.(22) reduces to

$$f_x(t) = f_y(t) = \frac{\sin\frac{t\theta}{2}}{\sin\frac{\theta}{2}}, \tag{23}$$

and the path (21) becomes a straight line.

It is noted that $f_x(t), f_y(t)$ in Eq.(22) satisfy the end conditions, $f_x(0) = f_y(0) = 0$ and $f_x(1) = f_y(1) = 1$, regardless the value of θ_1. For the remainder of the paper, we set $\theta_1 = 0$ so that the resulting interpolants are independent of the choice of the fixed reference frame. Furthermore, we let

$$s = \frac{\sin\frac{t\theta_{12}}{2}}{\sin\frac{\theta_{12}}{2}}, \quad s \in [0,1], \tag{24}$$

to obtain the final form of the interpolants:

$$f_x(s) = s\sqrt{\frac{e^2+s^2(1-e^2)\sin^2\frac{\theta_{12}}{2}}{e^2+(1-e^2)\sin^2\frac{\theta_{12}}{2}}},$$
$$f_y(s) = s\sqrt{\frac{1-s^2(1-e^2)\sin^2\frac{\theta_{12}}{2}}{1-(1-e^2)\sin^2\frac{\theta_{12}}{2}}}. \tag{25}$$

The above interpolants are independent of choice of the fixed coordinate frame as well as more compact and computationally more efficient.

For the rest of the paper, Eq. (25) will be used as the standard form for two-position interpolation as well as in recursive form when more than two positions are to be interpolated (in Bézier sense).

4. A DE CASTELJAU ALGORITHM FOR SHAPE-DEPENDENT MOTION GENERATION

In the field of Computer Aided Geometric Design (or CAGD), the de Casteljau algorithm is a classical tool for generating points on a Bézier curve via repeated linear interpolation [17]. Given a set of Bézier control points $\mathbf{b}_0, \mathbf{b}_1, \ldots, \mathbf{b}_n$, a Bézier curve can be generated recursively using the following repeated linear interpolation:

$$\mathbf{b}_i^r(t) = (1-t)\mathbf{b}_i^{r-1}(t) + t\mathbf{b}_{i+1}^{r-1}(t), \tag{26}$$

for $r = 1, \ldots, n$, $i = 0, 1, \ldots, n-r$, and $\mathbf{b}_i^0(t) = \mathbf{b}_i$. The end of this recursion produces a point $\mathbf{b}_0^n(t)$ on a Bézier curve of degree n with parameter t. A Bézier curve interpolates only the two end points but not the rest of the control points. If rational linear interpolations are used instead of linear interpolations, then the above algorithm generates a rational Bézier curve whose shape is determined not only by the Bézier control points but also the weight factors.

In this paper, we seek to apply the standard form (25) for two position interpolation recursively in the same manner as the de Casteljau algorithm to generate Bézier-like freeform motions of an object that are shape-dependent. We incorporate the weight factors in our formulation in order to explore the effect of the weights on the resulting motion.

Instead of presenting the de Casteljau algorithm in its most general form similar to (26), here we present only the case when there are three control positions $\mathbf{P}_i = (d_{ix}, d_{iy}, \theta_i)$ $(i = 0, 1, 2)$, where $\mathbf{d} = (d_{ix}, d_{iy})$ is the position vector of the centroid of an object in its i^{th} position and θ_i is the corresponding orientation of the object. The weight factors are denoted by w_i $(i = 0, 1, 2)$. The shape of the object is defined by the ratio $e = b/a$ as stated earlier.

Let $\mathbf{P}_0^1 = (d_{0x}^1, d_{0y}^1, \theta_0^1)$ and $\mathbf{P}_1^1 = (d_{1x}^1, d_{1y}^1, \theta_1^1)$be the two intermediate positions after the first level of de Casteljau algorithm. They are given by

$$\begin{aligned} d_{0x}^1(s) &= \frac{w_0 d_{0x}(1-f_{01x}) + w_1 d_{1x} f_{01x}}{w_0(1-f_{01x}) + w_1 f_{01x}}, \\ d_{0y}^1(s) &= \frac{w_0 d_{0y}(1-f_{01y}) + w_1 d_{1y} f_{01y}}{w_0(1-f_{01y}) + w_1 f_{01y}}, \\ \theta_0^1(s) &= \frac{w_0 \theta_0(1-s) + w_1 \theta_1 s}{w_0(1-s) + w_1 s}, \end{aligned} \tag{27}$$

and

$$\begin{aligned} d_{1x}^1(s) &= \frac{w_1 d_{1x}(1-f_{12x}) + w_2 d_{2x} f_{12x}}{w_1(1-f_{12x}) + w_2 f_{12x}}, \\ d_{1y}^1(s) &= \frac{w_1 d_{1y}(1-f_{12y}) + w_2 d_{2y} f_{12y}}{w_1(1-f_{12y}) + w_2 f_{12y}}, \\ \theta_1^1(s) &= \frac{w_1 \theta_1(1-s) + w_2 \theta_2 s}{w_1(1-s) + w_2 s}. \end{aligned} \tag{28}$$

At the second level, the position $\mathbf{P}_0^2(s) = (d_{0x}^2, d_{0y}^2, \theta_0^2)$ is interpolated between $\mathbf{P}_0^1(s)$ and $\mathbf{P}_1^1(s)$ based on shape-dependent interpolants f_{01x}^1, f_{01y}^1 associated with the two positions:

$$\begin{aligned} d_{0x}^2(s) &= \frac{w_{0x}^1 d_{0x}^1(1-f_{01x}^1) + w_{1x}^1 d_{1x}^1 f_{01x}^1}{w_{0x}^1(1-f_{01x}^1) + w_{0x}^1 f_{01x}^1}, \\ d_{0y}^2(s) &= \frac{w_{0y}^1 d_{0y}^1(1-f_{01y}^1) + w_{1y}^1 d_{1y}^1 f_{01y}^1}{w_{0y}^1(1-f_{01y}^1) + w_{0y}^1 f_{01y}^1}, \\ \theta_0^2(s) &= \frac{w_0^1 \theta_0^1(1-s) + w_1^1 \theta_1^1 s}{w_0^1(1-s) + w_1^1 s}, \end{aligned} \tag{29}$$

where

$$\begin{aligned} w_{0x}^1 &= w_0(1-f_{01x}) + w_1 f_{01x}, \quad w_{1x}^1 = w_1(1-f_{12x}) + w_2 f_{12x}, \\ w_{0y}^1 &= w_0(1-f_{01y}) + w_1 f_{01y}, \quad w_{1y}^1 = w_1(1-f_{12y}) + w_2 f_{12y}, \\ w_0^1 &= w_0(1-s) + w_1 s, \quad w_1^1 = w_1(1-s) + w_2 s. \end{aligned} \tag{30}$$

After the substitution of (27) and (28) into (29), and after some algebra, the object position $\mathbf{P}_0^2(s) =$

$(d^2_{0x}, d^2_{0y}, \theta^2_0)$ as the result of repeated shape-dependent interpolations is given by

$$
\begin{aligned}
d^2_{0x}(s) &= \frac{w_0 d_{0x}(1-f^1_{01x})(1-f_{01x}) + w_1 d_{1x}[(1-f^1_{01x})f_{01x} + f^1_{01x}(1-f_{12x})] + w_2 d_{2x} f^1_{01x} f_{12x}}{w_0(1-f^1_{01x})(1-f_{01x}) + w_1[(1-f^1_{01x})f_{01x} + f^1_{01x}(1-f_{12x})] + w_2 f^1_{01x} f_{12x}}, \\
d^2_{0y}(s) &= \frac{w_0 d_{0y}(1-f^1_{01y})(1-f_{01y}) + w_1 d_{1y}[(1-f^1_{01y})f_{01y} + f^1_{01y}(1-f_{12y})] + w_2 d_{2y} f^1_{01y} f_{12y}}{w_0(1-f^1_{01y})(1-f_{01y}) + w_1[(1-f^1_{01y})f_{01y} + f^1_{01y}(1-f_{12y})] + w_2 f^1_{01y} f_{12y}}, \\
\theta^2_0(s) &= \frac{w_0\theta_0(1-s)^2 + 2w_1\theta_1(1-s)s + w_2\theta_2 s^2}{w_0(1-s)^2 + 2w_1(1-s)s + w_2 s^2}.
\end{aligned}
\tag{31}
$$

When the shape parameter $e = 1$, we have $f_x(s) = f_y(s) = s$ from the standard form (25). Therefore, the above repeated shape-dependent interpolations will become the linear interpolation as shown in (26).

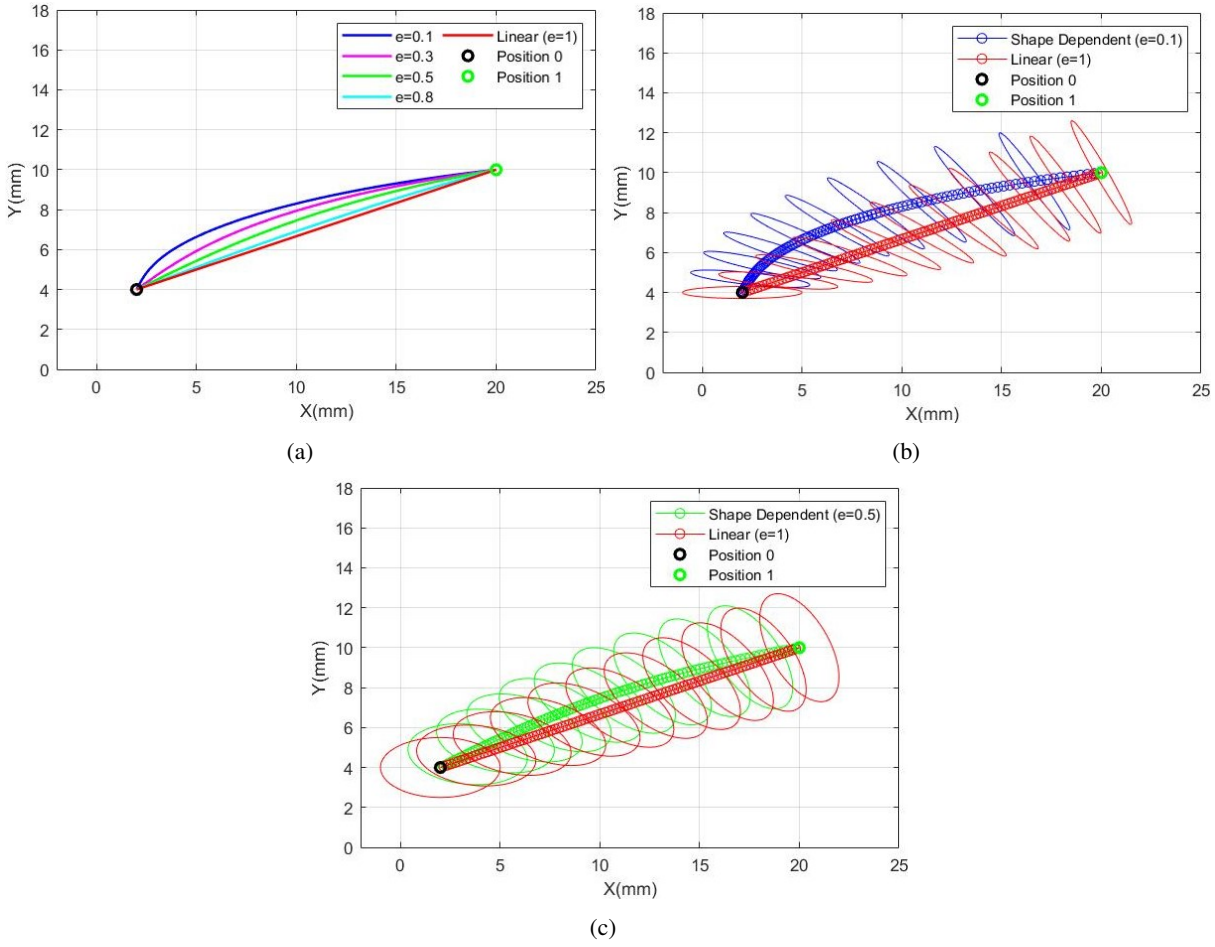

Fig. 4. Two-position interpolations with $w_0 = w_1 = 1$: (a) various values of e; (b) $e = 0.1$ with the shape of the object at eleven positions; (c) $e = 0.5$ with the shape of the object at eleven positions.

5. EXAMPLES

Consider the first example of interpolating two positions given by:

$$
\mathbf{P}_0 = (2, 4, 0^\circ), \quad \mathbf{P}_1 = (20, 10, -60^\circ). \tag{32}
$$

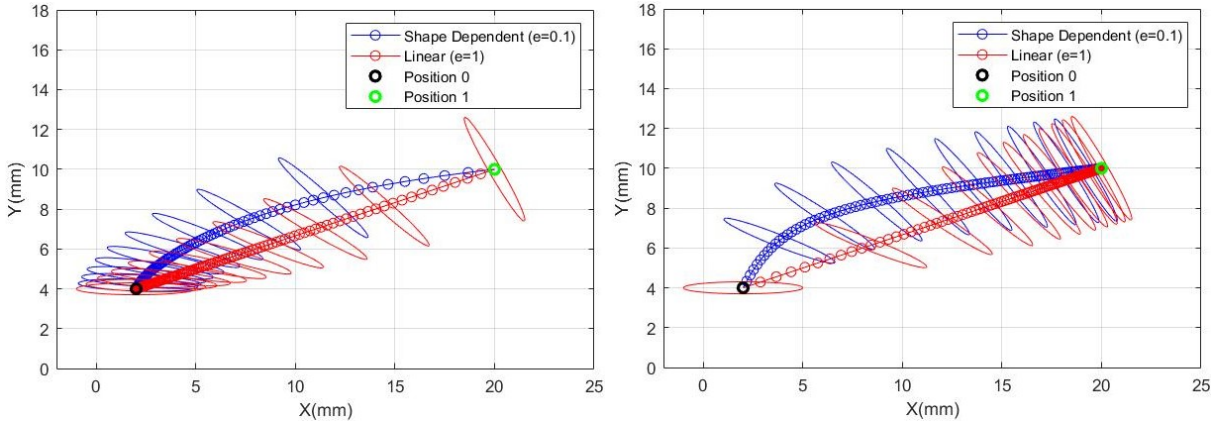

Fig. 5. Two-position interpolations with $e = 0.1$ but unequal weights: (a) $w_0 = 1, w_1 = 0.25$, (b) $w_0 = 1, w_1 = 5$.

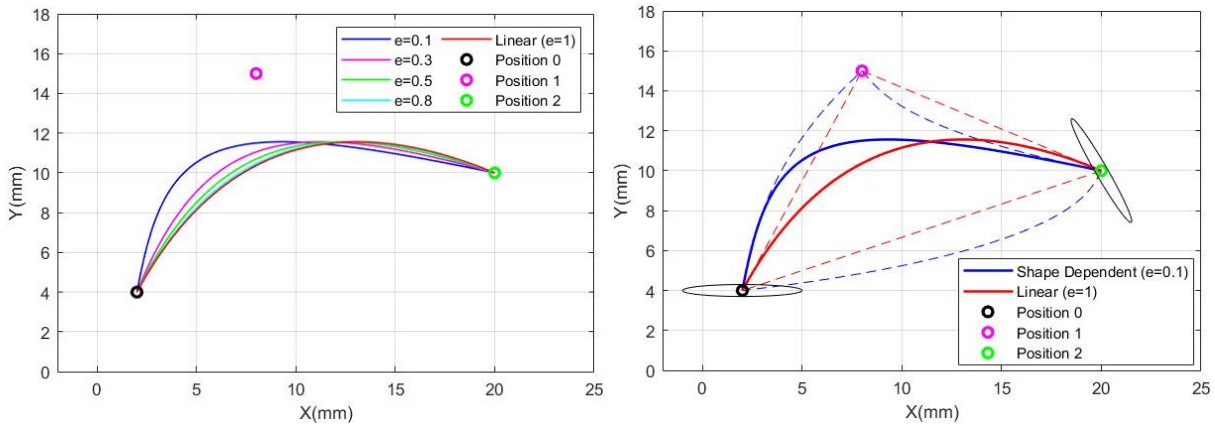

Fig. 6. (a) Three-position Bézier-like motions with various values of e (in blue) and quadratic Bézier curve (in red); (b) For $e = 0.1$, Bézier control triangle (in red) and Bézier-like control "curved triangle" (in blue).

Four shape parameter e of the objects in the range of $[0,1]$ are selected for comparison:

$$e = [0.1, 0.3, 0.5, 0.8]. \tag{33}$$

Eqs. (27) and the standard form (25) are used for interpolating the two positions.

In Figure 4(a), both weights are equal, i.e., $w_0 = w_1 = 1$, and four curves are the paths of shape-dependent interpolating motions with various shape parameters as given by (33). Included for comparison is the straight line (in red), which is the path of a linear interpolation $\mathbf{d} = (1-s)\mathbf{d}_1 + s\mathbf{d}_2$. As the values of e decrease, the paths of motion increasingly move away from that of the straight-line motion. Figure 4(b) and Figure 4(c) show the object motion by displaying eleven positions for the case of $e = 0.1$ (in blue) and $e = 0.5$ (in green), respectively. The effect of weights on the interpolating motions between two positions is shown in Figure 5. For both shape-dependent interpolation and linear interpolation, the paths are pulled to the position with larger weight factors.

For the second example, the starting and the final positions remain the same, so are the shape parameters

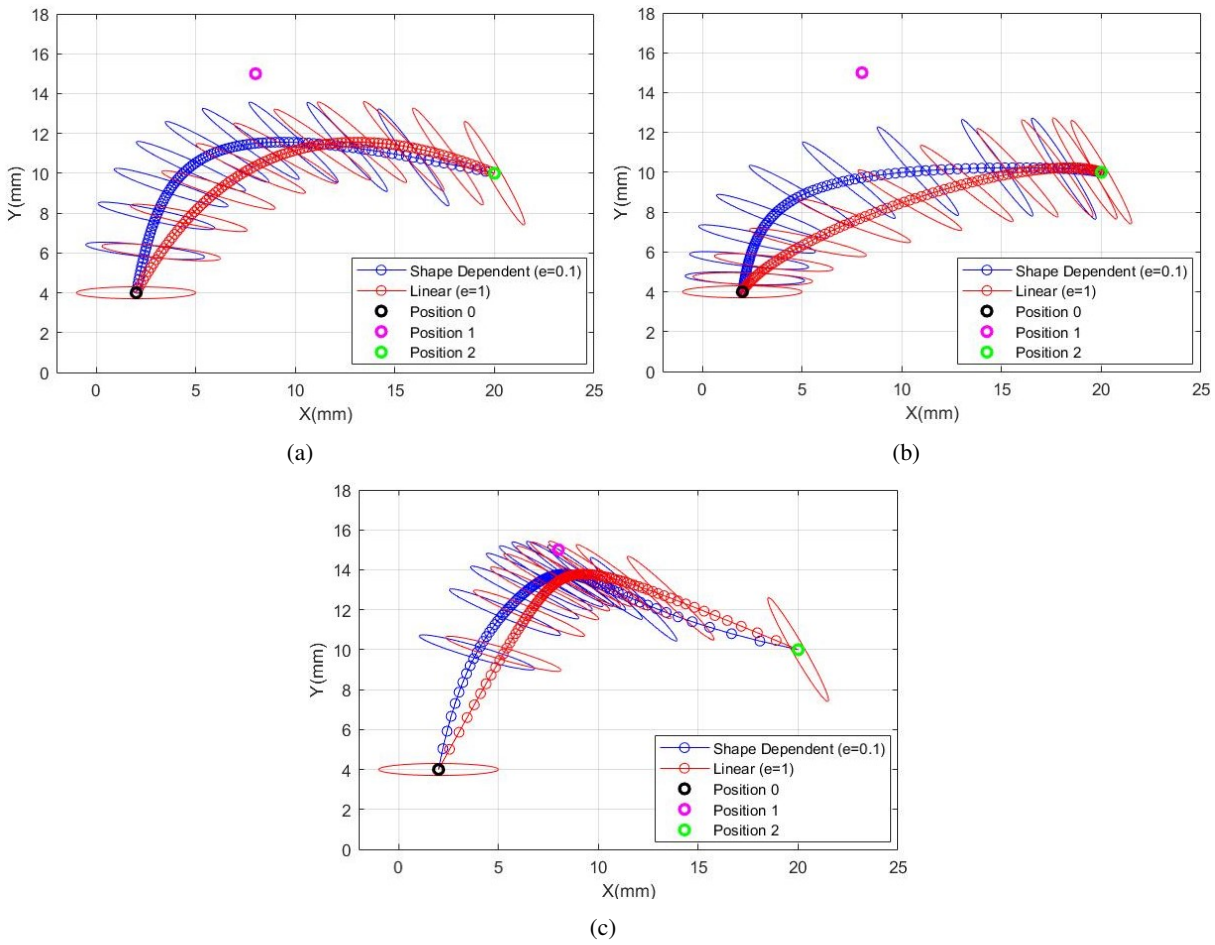

Fig. 7. Compare the effect of weights on the Bézier-like motions for $e = 0.1$ (in blue) and quadratic Bézier motion (in red): (a) $w_0 = 1, w_1 = 1, w_2 = 1$, (b) $w_0 = 1, w_1 = 0.25, w_2 = 1$, (c) $w_0 = 1, w_1 = 5, w_2 = 1$.

(33) but a middle position is added:

$$\mathbf{P}_0 = (2, 4, 0^\circ), \quad \mathbf{P}_1 = (8, 15, -30^\circ) \quad \mathbf{P}_2 = (20, 10, -60^\circ). \tag{34}$$

In Figure 6(a), the weights are all the same, $w_0 = w_1 = w_2 = 1$, and four curves are the paths of Bézier-like motions with various shape parameters as given by (33). Included for comparison is the quadratic Bézier curve (in red), which is the path of a quadratic Bèzier curve $\mathbf{d}(t) = (1-t)^2\mathbf{d}_1 + 2t(1-t)\mathbf{d}_2 + t^2\mathbf{d}_3$. Figure 6(b) compares Bézier control triangle (in red) and Bézier-like control "curved triangle" (in blue) for the two resulting motions. As expected, both paths are inside their respective control "triangles". Figure 7 compares the effect of weights on the resulting motions. Bézier-like shape-dependent motions are shown in blue while rational quadratic Bézier motions are shown in red. In both cases, the paths of the motion are pulled towards the mid control position $\mathbf{P}_2$ when the corresponding weight w_2 increases.

6. CONCLUSIONS

In this paper, we presented for the first-time the application of de Casteljau algorithm with shape-dependent interpolants for generating shape-dependent Bézier-like motions. It has been demonstrated that the paths of the resulting motions are made dependent on the shape parameter, they in general exhibit similar behavior as

Bézier curves. These shape-dependent motions could have applications in computer graphics, swept volume analysis and optimization, as well as new formulations for motion approximation problems that take into account object shapes.

ACKNOWLEDGEMENTS

Research reported in this publication was supported by the National Cancer Institute of the National Institutes of Health under Award Number R03CA249545. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

REFERENCES

1. Kazerounian, K. and Rastegar, J. "Object norms: A class of coordinate and metric independent norms for displacements." In "International Design Engineering Technical Conferences and Computers and Information in Engineering Conference," Vol. 9419, pp. 271–275. American Society of Mechanical Engineers, 1992.
2. Kazerounian, K. and Rastegar, J. "On the differential forms of coordinate and metric independent object norms." In "International Design Engineering Technical Conferences and Computers and Information in Engineering Conference," Vol. 80302, p. V01AT01A044. American Society of Mechanical Engineers, 1998.
3. Rastegar, J., Zhang, S. and Kazerounian, K. "An object shape dependent kinematic manipulability measure for path and trajectory synthesis and shape optimization." *Journal of Mechanical Design*, Vol. 120, No. 2, 1998.
4. Liu, H., Ge, Q.J. and Langer, M. "Shape dependent motion interpolants for planar objects." In "Proceedings of the IDETC/CIE, Paper No. IDETC2023/116793," ASME, 2023.
5. Shoemake, K. "Animating rotation with quaternion curves." In "Proceedings of the 12th annual conference on Computer graphics and interactive techniques," pp. 245–254, 1985.
6. Kim, M.J., Kim, M.S. and Shin, S.Y. "A general construction scheme for unit quaternion curves with simple high order derivatives." In "Proceedings of the 22nd annual conference on Computer graphics and interactive techniques," pp. 369–376, 1995.
7. Ge, Q.J. and Ravani, B. "Geometric construction of bézier motions." *Journal of mechanical design (1990)*, Vol. 116, No. 3, pp. 749–755, 1994.
8. Ge, Q.J. and Ravani, B. "Computer aided geometric design of motion interpolants." *Journal of mechanical design (1990)*, Vol. 116, No. 3, pp. 756–762, 1994.
9. Jüttler, B. and Wagner, M. "Computer-aided design with spatial rational b-spline motions." *Journal of mechanical design (1990)*, Vol. 118, No. 2, pp. 193–201, 1996.
10. Purwar, A. and Ge, Q.J. "On the effect of dual weights in computer aided design of rational motions." *Journal of Mechanical Design*, Vol. 127, p. 967, 2005.
11. Purwar, A., Jin, Z. and Ge, Q. "Rational motion interpolation under kinematic constraints of spherical 6r closed chains." *Journal of Mechanical Design*, Vol. 130, No. 6, 2008.
12. Röschel, O. "Rational motion design—a survey." *Computer-Aided Design*, Vol. 30, No. 3, pp. 169–178, 1998.
13. Park, F. and Ravani, B. "Bézier curves in riemannian manifolds and lie groups with kinematics applications." *Journal of mechanical design (1990)*, Vol. 117, No. 1, pp. 36–40, 1995.
14. Park, F. and Ravani, B. "Smooth invariant interpolation of rotations." *ACM Transactions on Graphics (TOG)*, Vol. 16, No. 3, pp. 277–295, 1997.
15. Popiel, T. and Noakes, L. "Bézier curves and c2 interpolation in riemannian manifolds." *Journal of Approximation Theory*, Vol. 148, No. 2, pp. 111–127, 2007.
16. Altafini, C. "The de casteljau algorithm on SE(3)." In "Nonlinear control in the year 2000," pp. 23–34. Springer, 2007.
17. Farin, G. *Curves and surfaces for CAGD: a practical guide*. Morgan Kaufmann, 2002.