Dynamic disk B-spline curves

Abstract

A disk B-spline curve (DBSC) is an extension of a B-spline curve and is used to represent a 2D region. DBSC is a useful 2D geometric representation and is widely applied in the 2D art design area, such as computer calligraphy, 2D computer animation, and nonphotorealistic rendering. To enhance the flexibility of DBSC, in this paper, we propose a physics-based generalization of DBSC — dynamic DBSC (D-DBSC), which extends the traditional DBSC in the time domain. We give the mathematical expression of D-DBSC and prove its several mathematical properties. We derive the motion equations of D-DBSC based on Lagrangian mechanics and investigate the motion equations when it is under linear geometric constraints. Last, a D-DBSC physical simulation system based on finite difference method is presented.

Keywords: Disk B-spline curves, Physics-based model, Dynamics, Finite difference method

1 Introduction

The disk B-spline curve (DBSC) was first proposed in 2005[1]. As a 2D geometric modeling method, DBSC has many attractive properties. Currently, DBSC is widely used in 2D computer animation, painting and other related art areas such as Chinese calligraphy. Current research approaches related to DBSC have all focused on the static form of DBSC. However, the static form of DBSC still encounters several drawbacks:

- In 2D geometric design area. Designers adjust the shape of a DBSC by manipulating its control disks. In general, this indirect method is a time-consuming task for obtaining the expected result.

- In 2D animation generation area. Animators need to create a DBSC for physics-based animation, such as a DBSC deforming under an external force. In this case, animators have to manually change the shape of DBSC frame by frame to generate a realistic simulation. Sometimes, the key frame interpolation technique can be helpful, but there is still an enormous amount of work involved.

To overcome these drawbacks, in this paper, we propose a physics-based generalization of DBSC – dynamic DBSC – which we name D-DBSC. A D-DBSC is an extension of a DBSC in the time domain, and several physical quantities are incorporated into D-DBSC, such as mass, tension, rigidity, and external force distribution. D-DBSC is based on Lagrangian mechanics; when there is an applied external force, D-DBSC will deform under physical laws.

The physics-based D-DBSC model has several important advantages:

- The physics-based model obeys physical laws; when an external force is applied, D-DBSC can deform under physical laws.

- By applying a proper external force on a DBSC, designers can manipulate the shape of DBSC directly. Because D-DBSC is physics-based, it will deform as expected.

- Using D-DBSC, we can generate a natural and realistic physics-based animation. As a result, animators can create physics-based animations automatically by choosing proper physical parameters and an external force.

In this paper, we propose the concept of D-DBSC and provide its mathematical expression. Based on Lagrangian mechanics, we derive motion equations of D-DBSC. Last, a D-DBSC physical simulation system using the finite difference method is implemented, and several results are demonstrated. We present several contributions in this paper:

- The concept of D-DBSC and the mathematical expression of D-DBSC are proposed. Several mathematical properties of D-DBSC are proved.

- We derive the motion equations of D-DBSC based on Lagrangian mechanics, and investigate the motion equations when it is under linear geometric constraints.

- A numerical implementation of D-DBSC based on the finite difference method is presented.

2 Related work

Seah et al. proposed the concept of DBSC in 2005 [1]. DBSC is a skeleton-based 2D geometric modeling method, which is used to represent a 2D region with varying radius. Besides DBSC, there are several other related geometric modeling methods. In 1997, Peternell and Pottmann proposed the canal surface which is a envelope of a interpolated skeleton curve with varying radius [2, 3], they further used a curve in $\mathbb{R}^4$ to represent the 3D rational canal surface and investigated its mathematical properties [4]. In 2010, Kunkli and Hoffmann investigated a similar problem, they proposed a G^1 continuous surface to represent the skin of an ordered set of discrete circles [5]. Lin and Rokne proposed disk Bézier curves in 1998, unlike DBSC, they used a Bézier curve to represent the skeleton [6]. Different from researches mentioned above, DBSC uses a B-spline curve to represent the skeleton curve. Thus, nice mathematical properties of B-spline curve can be adopted into DBSC.

Since the establishment of DBSC theory, several important works about DBSC have been accomplished. In 2015, Zhang et al. proposed an extension algorithm for DBSC with G^2 continuity [7]. In 2018, Ao et al. proposed a high-accuracy intersection algorithm for DBSC [8]. However, until now, works related to DBSC have all focused on the static form. As we mentioned above, due to the drawbacks of the static DBSC, the ability and flexibility of the DBSC geometric modeling method are limited.

A gap still remains regarding the physics-based model of DBSC. Since D-DBSC is a physics-based model and a DBSC can be regarded as an extension of B-spline curves, we will review previous work related to B-spline curve deformation modeling.

Terzopoulos et al. proposed elastically deformable models in 1987, wherein they employed the theory of elasticity and constructed differential equations to simulate the behavior of nonrigid curves, surfaces and solids under an external force [9]. Celniker and Gossard implemented a free-form design system based on the finite element method of energy optimization in 1991 [10]. Bloor and Wilson developed related models and used B-splines in their method in 1990 [11]. Celniker and Welch investigated deformable B-splines with linear constraints in 1992 [12]. Thingvold and Cohen proposed a deformable B-spline using a mass-spring model in 1990 [13]. In 1994, Terzopoulos and Qin proposed a physics-based generalization of NURBS curves and surfaces — dynamic NURBS (D-NURBS). The D-NURBS model was derived systematically through the application of Lagrangian mechanics and implemented using the finite element method [14, 15, 16].

In our paper, D-DBSC adopts the method in the D-NURBS model. We extend DBSC into a physics-based model using Lagrangian mechanics. We simplify the motion equations of D-DBSC using the properties of D-DBSC and solve our physics-based model by implementing the finite difference method (FDM).

Compared with B-spline and NURBS curves, developing the physics-based model of DBSC is more challenging. B-spline and NURBS curves possess a more compact geometric form; however, DBSC have a more complicated geometric form. The complicated geometric form makes it difficult to calculate some physical quantities (such as second-order partial derivatives). To solve this problem, we use a finite difference method to approximate a physical quantity when it is difficult to obtain an analytical solution. D-DBSC allows a continuous physical attribute distribution, such as the mass distribution and damping distribution; thus, we need to calculate the mass matrix, damping matrix, etc. Due to the existence of control radii (see Subsection 3.1), the dynamics of DBSC comprise a nonlinear system, and thus, physical quantities, such as the mass matrix and damping matrix, need to be recomputed at every time step during simulation.

3 The formulation of D-DBSC

In this section, we will review the general geometric representation of DBSC and give the formulation of D-DBSC.

3.1 Geometric representation of DBSC

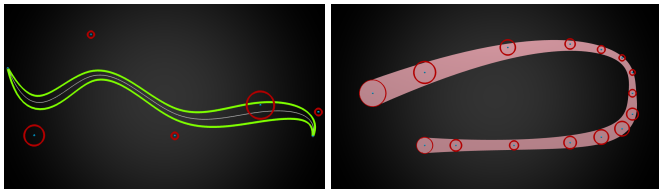
A disk is a circular area in 2D space, and we define a disk $\langle \mathbf{d}; r\rangle$ specifically as

$$\langle \mathbf{d}; r\rangle \equiv \left\{ \mathbf{x} \in R^2 \;\middle|\; \|\mathbf{x}-\mathbf{d}\| \le r, \mathbf{d} \in R^2, r \in R^+ \right\} \tag{1}$$

A DBSC is a 2D region that can be considered as the extension of a B-spline curve (see Figure 1). In a DBSC, every control point $\mathbf{d}_i = (x_i, y_i)$ has a control radius r_i, and we call $\langle \mathbf{d}_i; r_i\rangle$ a control disk. Denoting $N_{i,p}(u)$ as the i-th B-spline basis function of degree p, we can define a DBSC region as

$$\mathbf{s}(u) = \left\langle \sum_{i=0}^{n} N_{i,p}(u)\mathbf{d}_i; \sum_{i=0}^{n} N_{i,p}(u) r_i \right\rangle \tag{2}$$

Since Equation 2 is an implicit form, and is not suitable for analysis, we use an alternative geometric expression of DBSC in this paper. For convenience of writing, we use $\mathbf{p}_i = \begin{bmatrix} x_i & y_i & z_i & r_i \end{bmatrix}^\top$ to represent the i-th control disk and specifically use $\mathbf{p}_i^* = \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}^\top$ to represent the position of the i-th control disk. We use $\mathbf{c}(u)$ and $r(u)$ to represent the parametric expression of the skeleton curve and varying radii along that curve, respectively. Additionally, we define the tangent vector



(a) The boundary (green) and skeleton (white) of a DBSC

(b) The region (pink) of a DBSC

Figure 1: DBSC demonstrations

and normal vector along the skeleton curve of the DBSC as $\mathbf{T}(u)$, and $\mathbf{N}(u)$. As a result, we have the following symbols:

$$\begin{aligned} \mathbf{c}(u) &= \sum \mathbf{p}_i^* N_{i,p}(u) \\ r(u) &= \sum r_i N_{i,p}(u) \\ \mathbf{T}(u) &= \frac{\mathbf{c}'(u)}{\|\mathbf{c}'(u)\|} \\ \mathbf{N}(u) &= \begin{bmatrix} -\mathbf{T}_y & \mathbf{T}_x \end{bmatrix}^\top \end{aligned} \tag{3}$$

Then, the geometric expression of a DBSC can be defined in an explicit form:

$$\mathbf{s}(u,v) = \mathbf{c}(u) - v^2 \frac{r(u)r'(u)}{\|\mathbf{c}'(u)\|}\mathbf{T}(u) + v\, r(u)\sqrt{1-\left(\frac{vr'(u)}{\|\mathbf{c}'(u)\|}\right)^2}\,\mathbf{N}(u), \quad v \in [-1,1] \tag{4}$$

where u and v represent parameters along the skeleton curve and perpendicular to the skeleton curve, respectively.

Equation 4 describes the relationship between the parameter pair (u, v) and the position of an arbitrary point in the region of the DBSC. Using Equation 4, the geometric properties of any point in DBSC can be analyzed. In fact, Equation 2 and Equation 4 are not equivalent. Geometrically, Equation 2 includes one more sector-like area than Equation 4 at each end of DBSC. Since the deformation of two ends will follow the deformation of the main body, we do not take the two ends of DBSC into consideration and use Equation 4 as a general representation of DBSC.

3.2 The formulation of D-DBSC

The formulation of D-DBSC is based on several hypotheses:

- A deformable object can be represented with DBSC at every moment during deformation. We suppose that the shape of a deformable object can be represented with an element in DBSC space during deformation.
- The knot vector and degree of B-spline basis functions are fixed during deformation; these quantities will not change over time. In our physics-based model, we focus on the change of control disks during deformation, so the knot vector and degree of B-spline basis functions are not considered.

With these hypotheses, we can find that D-DBSC is determined by all control disks. If we denote t as the time parameter, we can use $\mathbf{p}_i(t)$ to represent the i-th control disk, and $\mathbf{p}_i(t)$ possesses three attributes: $x_i(t)$, $y_i(t)$ and $r_i(t)$. We can write all control disks in vector form, denoted $\mathbf{p}(t)$, and we have

$$\mathbf{p}(t) = \begin{bmatrix} \cdots & x_i(t) & y_i(t) & r_i(t) & \cdots \end{bmatrix}^\top \tag{5}$$

$\mathbf{p}(t)$ is a function of time, and a DBSC is determined with $\mathbf{p}(t)$. In Lagrangian mechanics, we call $\mathbf{p}(t)$ generalized coordinates. In Subsection 5.1, we will discuss generalized coordinates in detail.

D-DBSC is a dynamic generalization of DBSC in the time domain. Based on the geometric representation of DBSC and the above hypotheses, a D-DBSC is fully determined by generalized coordinates $\mathbf{p}(t)$; therefore, the expression of D-DBSC is given as

$$\begin{aligned} \mathbf{s}(u,v,t) = \mathbf{c}(u,t) - v^2 \frac{r(u,t)r'(u,t)}{\|\mathbf{c}'(u,t)\|}\mathbf{T}(u,t) + \\ v\, r(u,t)\sqrt{1-\left(\frac{vr'(u,t)}{\|\mathbf{c}'(u,t)\|}\right)^2}\,\mathbf{N}(u,t), \quad v \in [-1,1] \end{aligned} \tag{6}$$

4 Properties of D-DBSC

In this section, we will investigate the Jacobian matrix and properties of D-DBSC; these properties will used in subsequent sections for simplifying the motion equations of D-DBSC.

4.1 Jacobian matrix

The Jacobian matrix is a matrix of all first-order derivatives of a vector-value function. The Jacobian matrix between a D-DBSC $\mathbf{s}$ and its generalized coordinates $\mathbf{p}$ describes their mathematical relationship.

We use $\mathbf{p}$ to represent the generalized coordinates of a D-DBSC in vector form. If there are N control disks, $\mathbf{p}$ will be a $3N \times 1$ vector.

$$\mathbf{p}(t) = \begin{bmatrix} \cdots & x_i(t) & y_i(t) & r_i(t) & \cdots \end{bmatrix}^\top \tag{7}$$

Additionally, the Jacobian matrix $\mathbf{J}$ of D-DBSC is a $2 \times 3N$ matrix:

$$\mathbf{J} = \left(\frac{\partial \mathbf{s}}{\partial \mathbf{p}}\right)^\top = \begin{bmatrix} \cdots & \frac{\partial \mathbf{s}_x}{\partial x_i} & \frac{\partial \mathbf{s}_x}{\partial y_i} & \frac{\partial \mathbf{s}_x}{\partial r_i} & \cdots \\ \cdots & \frac{\partial \mathbf{s}_y}{\partial x_i} & \frac{\partial \mathbf{s}_y}{\partial y_i} & \frac{\partial \mathbf{s}_y}{\partial r_i} & \cdots \end{bmatrix} \tag{8}$$

where $\frac{\partial \mathbf{s}}{\partial \mathbf{p}}$ is an abbreviation of

$$\begin{bmatrix} \cdots & \frac{\partial \mathbf{s}}{\partial x_i} & \frac{\partial \mathbf{s}}{\partial y_i} & \frac{\partial \mathbf{s}}{\partial r_i} & \cdots \end{bmatrix}^\top \tag{9}$$

Note that we will use similar notations in subsequent sections.

4.2 Propositions of D-DBSC

Using the D-DBSC geometric representation in Equation 6, we can prove some important properties of D-DBSC. Now, we will prove the correctness of some identities, and these equations will be useful in subsequent sections.

Proposition 1.

$$\mathbf{s} = \mathbf{J}\mathbf{p} \tag{10}$$

where $\mathbf{s}$ *is the position of an arbitrary point in D-DBSC,* $\mathbf{J}$ *is a Jacobian matrix, and* $\mathbf{p}$ *are generalized coordinates.*

Proof. First we prove $\mathbf{s}(\mathbf{p})$ is a homogeneous function: A function $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ is said to be homogeneous of degree k if $\mathbf{f}(\lambda \mathbf{x}) = \lambda^k \mathbf{f}(\mathbf{x})$ for any scalar λ.

From Equation 3 and Equation 6, we have

$$\begin{aligned} \mathbf{s}(\lambda \mathbf{p}) &= \lambda \mathbf{c} - v^2 \frac{(\lambda r)(\lambda r')}{\|\lambda \mathbf{c}'\|}\mathbf{T} + v(\lambda r)\sqrt{1 - \left(\frac{v(\lambda r')}{\|\lambda \mathbf{c}'\|}\right)^2}\,\mathbf{N} \\ &= \lambda \left(\mathbf{c} - v^2 \frac{r r'}{\|\mathbf{c}'\|}\mathbf{T} + v r \sqrt{1 - \left(\frac{v r'}{\|\mathbf{c}'\|}\right)^2}\,\mathbf{N} \right) \\ &= \lambda\, \mathbf{s}(\mathbf{p}) \end{aligned} \tag{11}$$

Since $\mathbf{s}(\mathbf{p})$ is a homogeneous function of degree 1, then, according to Euler's homogeneous function theorem [17], Equation 10 is proved:

$$\mathbf{J}\mathbf{p} = \begin{bmatrix} \sum \frac{\partial \mathbf{s}_x}{\partial x_i} x_i + \sum \frac{\partial \mathbf{s}_x}{\partial y_i} y_i + \sum \frac{\partial \mathbf{s}_x}{\partial r_i} r_i \\ \sum \frac{\partial \mathbf{s}_y}{\partial x_i} x_i + \sum \frac{\partial \mathbf{s}_y}{\partial y_i} y_i + \sum \frac{\partial \mathbf{s}_y}{\partial r_i} r_i \end{bmatrix} = \mathbf{s} \tag{12}$$

□

Equation 10 shows that the position of an arbitrary point in D-DBSC can be represented by the Jacobian matrix $\mathbf{J}$ and generalized coordinates $\mathbf{p}$.

Proposition 2.

$$\dot{\mathbf{s}} = \mathbf{J}\dot{\mathbf{p}} \tag{13}$$

$\dot{\mathbf{s}}$ *and* $\dot{\mathbf{p}}$ *are derivatives of* $\mathbf{s}$ *and* $\mathbf{p}$*, respectively, with respect to time* t*.*

Proof. With the chain rule of derivatives,

$$\dot{\mathbf{s}} = \left(\frac{\partial \mathbf{s}}{\partial \mathbf{p}}\right)^\top \dot{\mathbf{p}} = \mathbf{J}\dot{\mathbf{p}}$$

□

Equation 13 shows that the velocity of an arbitrary point in D-DBSC can be represented by the Jacobian matrix $\mathbf{J}$ and the velocity of generalized coordinates $\mathbf{p}$.

Proposition 3.

$$\dot{\mathbf{J}}\mathbf{p} = 0 \tag{14}$$

$\dot{\mathbf{J}}$ *is the derivative of Jacobian matrix* $\mathbf{J}$ *with respect to time* t*.*

Proof. With Equation 10 and Equation 13, we have

$$\dot{\mathbf{s}} = \dot{\mathbf{J}}\mathbf{p} + \mathbf{J}\dot{\mathbf{p}} \qquad \dot{\mathbf{s}} = \mathbf{J}\dot{\mathbf{p}}$$

Therefore, Equation 14 is proved.

□

Equation 14 is an important equation of D-DBSC, which will be used to simplify the discretized motion equations of D-DBSC in Subsection 6.2.

5 Dynamics of D-DBSC

In this section, we will introduce Lagrangian equations to D-DBSC and derive motion equations of D-DBSC. Then, we will investigate D-DBSC under linear geometric constraints and provide the corresponding motion equations of D-DBSC.

5.1 Lagrangian mechanics

D-DBSC is based on the work-energy version of Lagrangian equations. In Lagrangian mechanics, a physical system is determined by a set of parameters called generalized coordinates. We denote generalized coordinates as $\mathbf{p}$ and their derivatives with time t as $\dot{\mathbf{p}}$; we have

$$\mathbf{p} = \begin{bmatrix} \cdots & p_i(t) & \cdots \end{bmatrix}^\top, \quad \dot{\mathbf{p}} = \begin{bmatrix} \cdots & \dot{p}_i(t) & \cdots \end{bmatrix}^\top$$

In a physical system, if we define the kinetic energy, potential energy and Raleigh dissipation energy as T, U, and D, respectively, and denote the generalized external force exerted on generalized coordinate p_i as $\mathbf{f}_i$, then we can express the work-energy version of Lagrangian equations [18, 19, Section 1,2] as

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{p}_i} - \frac{\partial T}{\partial p_i} + \frac{\partial D}{\partial \dot{p}_i} + \frac{\partial U}{\partial p_i} = \mathbf{f}_i \tag{15}$$

where T, U, and D are functions of generalized coordinates $\mathbf{p}$ or their derivatives $\dot{\mathbf{p}}$, and $\mathbf{f}_i$ are functions of time t. We can also express Equation 15 in matrix form as

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{\mathbf{p}}} - \frac{\partial T}{\partial \mathbf{p}} + \frac{\partial D}{\partial \dot{\mathbf{p}}} + \frac{\partial U}{\partial \mathbf{p}} = \mathbf{f}_p \tag{16}$$

where $\frac{\partial T}{\partial \dot{\mathbf{p}}} = \begin{bmatrix} \cdots & \frac{\partial T}{\partial \dot{p}_i} & \cdots \end{bmatrix}^\top$, $\mathbf{f}_p = \begin{bmatrix} \cdots & f_i & \cdots \end{bmatrix}^\top$, and other terms are similar.

5.2 Motion equations of D-DBSC

The motion equations of D-DBSC are based on Lagrangian equations Equation 16. From the geometric representation of D-DBSC, we can find that when the knot vector and the degree of the basis function are fixed, a D-DBSC system is determined by control disks. Therefore, a D-DBSC system with N control disks possesses $3N$ generalized coordinates, and we use $\mathbf{p}$ to represent the generalized coordinates of D-DBSC in vector form:

$$\mathbf{p}(t) = \begin{bmatrix} \cdots & x_i(t) & y_i(t) & r_i(t) & \cdots \end{bmatrix}^\top$$

In addition, with the geometric representation of D-DBSC (Equation 6), we can determine every term in Equation 16.

For a D-DBSC, we use $\mu(u, v)$ and $\mathbf{M}(\mathbf{p})$ to represent its mass density distribution and mass matrix, respectively, and the kinetic energy T can be expressed as

$$T = \frac{1}{2}\iint \mu \dot{\mathbf{s}}^\top \dot{\mathbf{s}} du dv = \frac{1}{2}\dot{\mathbf{p}}^\top \mathbf{M} \dot{\mathbf{p}}$$

With Equation 10 and Equation 14, we can obtain the expression of mass matrix $\mathbf{M}(\mathbf{p})$:

$$\mathbf{M} = \iint \mu \mathbf{J}^\top \mathbf{J} du dv$$

Similarly, we use $\gamma(u, v)$ and $\mathbf{D}(\mathbf{p})$ to represent the damping density distribution and damping matrix, respectively, and the dissipation energy D can be expressed as

$$D = \frac{1}{2}\iint \gamma \dot{\mathbf{s}}^\top \dot{\mathbf{s}} du dv = \frac{1}{2}\dot{\mathbf{p}}^\top \mathbf{D} \dot{\mathbf{p}}$$

The damping matrix $\mathbf{D}(\mathbf{p})$ is

$$\mathbf{D} = \iint \gamma \mathbf{J}^\top \mathbf{J} du dv$$

For calculating the potential energy of D-DBSC, we adopt the energy model proposed in [20]. We use $\alpha(u, u)$, $\beta(u, v)$ and $\mathbf{K}(\mathbf{p})$ to represent the local tension function, rigidity function and stiffness matrix, respectively, and using this energy model, the potential energy of D-DBSC can be expressed as

$$U = \frac{1}{2}\iint \left(\alpha_{1,1} \frac{\partial \mathbf{s}^\top}{\partial u}\frac{\partial \mathbf{s}}{\partial u} + \alpha_{2,2}\frac{\partial \mathbf{s}^\top}{\partial v}\frac{\partial \mathbf{s}}{\partial v} + \beta_{1,1}\frac{\partial^2 \mathbf{s}^\top}{\partial u^2}\frac{\partial^2 \mathbf{s}}{\partial u^2} + \beta_{1,2}\frac{\partial^2 \mathbf{s}^\top}{\partial u \partial v}\frac{\partial^2 \mathbf{s}}{\partial u \partial v} + \beta_{2,2}\frac{\partial^2 \mathbf{s}^\top}{\partial v^2}\frac{\partial^2 \mathbf{s}}{\partial v^2} \right) du dv = \frac{1}{2}\mathbf{p}^\top \mathbf{K} \mathbf{p}$$

With Equation 10 and Equation 14, the expression of stiffness matrix $\mathbf{K}(\mathbf{p})$ is

$$\mathbf{K}(\mathbf{p}) = \iint (\alpha_{1,1}\mathbf{J}_u^\top \mathbf{J}_u + \alpha_{2,2}\mathbf{J}_v^\top \mathbf{J}_v + \beta_{1,1}\mathbf{J}_{uu}^\top \mathbf{J}_{uu} + \beta_{1,2}\mathbf{J}_{uv}^\top \mathbf{J}_{uv} + \beta_{2,2}\mathbf{J}_{vv}^\top \mathbf{J}_{vv}) du dv \tag{17}$$

where the subscripts of $\mathbf{J}$ indicate the partial derivatives.

The generalized external force $\mathbf{f}_p$ can be obtained through the principle of virtual work [18]. We denote the external distribution force as $\mathbf{f}(u, v, t)$, and the generalized external force of D-DBSC can be expressed as

$$\mathbf{f}_p = \iint \mathbf{J}^\top \mathbf{f}(u, v, t) du dv$$

Therefore, in a D-DBSC system, Equation 16 can be expressed as

$$\mathbf{M}\ddot{\mathbf{p}} + \dot{\mathbf{M}}\dot{\mathbf{p}} - \frac{1}{2}\dot{\mathbf{p}}^\top \frac{\partial \mathbf{M}}{\partial \mathbf{p}}\dot{\mathbf{p}} + \mathbf{D}\dot{\mathbf{p}} + \mathbf{K}\mathbf{p} + \frac{1}{2}\mathbf{p}^\top \frac{\partial \mathbf{K}}{\partial \mathbf{p}}\mathbf{p} = \mathbf{f}_p \tag{18}$$

According to the properties of the Jacobian matrix, Equation 18 can be simplified (see [14]), and the simplified motion equations of D-DBSC are given by

$$\mathbf{M}\ddot{\mathbf{p}} + \mathbf{D}\dot{\mathbf{p}} + \mathbf{K}\mathbf{p} = \mathbf{f}_p - \mathbf{I}\dot{\mathbf{p}} \tag{19}$$

where

$$\mathbf{I} = \iint \mu \mathbf{J}^\top \dot{\mathbf{J}} du dv$$

5.3 Linear geometric constraints

Applying linear geometric constraints on a physical system is useful for geometric design and animation generation. Linear geometric constraints can be expressed generally as

$$\mathbf{C}(\mathbf{p}) = \mathbf{A}\mathbf{p} + \mathbf{b} = 0$$

If there are M independent linear constraints, $\mathbf{A}$ will be an $M \times N$ coefficient matrix. Moreover, we can express generalized coordinates $\mathbf{p}$ as

$$\mathbf{p} = \mathbf{G}\mathbf{q} + \mathbf{q}_0$$

where $\mathbf{G}$ is an $N \times (N - M)$ matrix, $\mathbf{q}_0$ is a constant vector, and $\mathbf{q}$ represents new lower-dimensional generalized coordinates, which can be computed by Gaussian elimination or other methods.

For new generalized coordinates $\mathbf{q}$, we can express the motion equations of D-DBSC with linear geometric constraints as

$$\mathbf{M}_q\ddot{\mathbf{q}} + \mathbf{D}_q\dot{\mathbf{q}} + \mathbf{K}_q\mathbf{q} = \mathbf{f}_q + \mathbf{g}_q - \mathbf{I}_q\dot{\mathbf{q}} \tag{20}$$

where

$$\mathbf{M}_q = \mathbf{G}^\top \mathbf{M}\mathbf{G}, \quad \mathbf{D}_q = \mathbf{G}^\top \mathbf{D}\mathbf{G}, \quad \mathbf{K}_q = \mathbf{G}^\top \mathbf{K}\mathbf{G}$$
$$\mathbf{f}_q = \iint \mathbf{L}^\top \mathbf{f}(u, v, t) du dv, \quad \mathbf{g}_q = -\mathbf{G}^\top \mathbf{K}\mathbf{q}_0, \quad \mathbf{I}_q = \mathbf{G}^\top \mathbf{I}\mathbf{G}$$

and $\mathbf{L} = \mathbf{J}\mathbf{G}$ is the new Jacobian matrix of $\mathbf{s}$ with respect to $\mathbf{q}$. The detailed proof of Equation 20 can be found in [14].

6 Numerical implementation

The complicated geometric form of D-DBSC makes it difficult to calculate some physical quantities in its implementation such as the stiffness matrix. To solve this problem, we use a finite difference method (FDM) to approximate a physical quantity when it is difficult to obtain an analytical solution. The FDM is a numerical method used to solve differential equations by the finite difference approximation. In the FDM, a continuum is approximated by a finite set of nodes ; given that a function's derivatives in the geometric domain can be approximated by finite differences of function values at nodes, differential equations can be converted into linear algebraic equations by using the FDM. Additionally, our research focuses on the deformation of the body of D-DBSC, and the two ends of the D-DBSC are not included when calculating the deformation . The two ends will be added naturally according to the shape of the D-DBSC after deformation.

6.1 Calculation of the stiffness matrix

Since the expression of stiffness matrix $\mathbf{K}$ (Equation 17) is complicated. To avoid the cost of deriving the analytical solutions of $\mathbf{J}_u, \mathbf{J}_v, \mathbf{J}_{uv}, \mathbf{J}_{uu}$, and $\mathbf{J}_{vv}$, we use a finite difference method to approximate these variables. For the internal nodes, the central difference approximation is used:

$$\begin{aligned}
\mathbf{J}_u(u,v) &= \frac{\mathbf{J}(u+\Delta u,v)-\mathbf{J}(u-\Delta u,v)}{2\Delta u}\\
\mathbf{J}_v(u,v) &= \frac{\mathbf{J}(u,v+\Delta v)-\mathbf{J}(u,v-\Delta v)}{2\Delta v}\\
\mathbf{J}_{uu}(u,v) &= \frac{\mathbf{J}(u+\Delta u,v)-2\mathbf{J}(u,v)+\mathbf{J}(u-\Delta u,v)}{\Delta u^2}\\
\mathbf{J}_{uv}(u,v) &= \frac{\mathbf{J}(u+\Delta u,v+\Delta v)-\mathbf{J}(u-\Delta u,v)-\mathbf{J}(u,v-\Delta v)+\mathbf{J}(u,v-\Delta v)}{4\Delta u\Delta v}\\
\mathbf{J}_{vv}(u,v) &= \frac{\mathbf{J}(u,v+\Delta v)-2\mathbf{J}(u,v)+\mathbf{J}(u,v-\Delta v)}{\Delta v^2}
\end{aligned} \tag{21}$$

Similarly, for nodes on the boundary, the forward/backward difference is used for the approximation.

6.2 Time integration

Considering that Equation 19 is a second-order partial difference equation that has no an analytical solution, in our implementation, we discretize time t and use $\mathbf{p}$ to approximate $\dot{\mathbf{p}}$ and $\ddot{\mathbf{p}}$ using the finite difference method in Equation 19 so that we can obtain a numerical solution. Given some specified physical parameters, Equation 19 can be a stiff system; in this case, the explicit time integration is not stable. Therefore, we use an implicit Euler method to ensure the stability:

$$\begin{aligned}
\ddot{\mathbf{p}} &= \frac{\mathbf{p}^{(t+\Delta t)}-2\mathbf{p}^{(t)}+\mathbf{p}^{(t-\Delta t)}}{\Delta t^2}\\
\dot{\mathbf{p}} &= \frac{\mathbf{p}^{(t+\Delta t)}-\mathbf{p}^{(t-\Delta t)}}{2\Delta t}
\end{aligned}$$

Using Equation 14, we can derive the discretized form of Equation 19:

$$\begin{aligned}
\left(4\mathbf{M}+2\Delta t\mathbf{D}+4\left(\Delta t\right)^2\mathbf{K}\right)\mathbf{p}^{(t+\Delta t)} =\\
4\left(\Delta t\right)^2\mathbf{f}_p+8\mathbf{M}\mathbf{p}^{(t)}-(3\mathbf{M}-2\Delta t\mathbf{D})\mathbf{p}^{(t-\Delta t)}-\iint\mu\mathbf{J}^\top\mathbf{s}^{(t-\Delta t)}dudv
\end{aligned} \tag{22}$$

The detailed derivation of Equation 22 can be found in [21]. In Equation 22, matrices without a superscript should be evaluated at time $t+\Delta t$, and for simplicity, the value of the matrix at time $t+\Delta t$ can be approximated using its value at time t, for example:

$$\mathbf{M}^{(t+\Delta t)}=\mathbf{M}^{(t)}+\Delta t\dot{\mathbf{M}}=2\mathbf{M}^{(t)}-\mathbf{M}^{(t-\Delta t)}$$

A simpler approximation $\mathbf{M}^{(t+\Delta t)}=\mathbf{M}^{(t)}$ is also acceptable according to [22, Section 8.6].

Similarly, we can derive discretized motion equations of D-DBSC with geometric constraints using Equation 20:

$$\begin{aligned}
\left(4\mathbf{M}_q+2\Delta t\mathbf{D}_q+4\left(\Delta t\right)^2\mathbf{K}_q\right)\mathbf{q}^{(t+\Delta t)} =\\
4(\Delta t)^2(\mathbf{f}_q+\mathbf{g}_q)+8\mathbf{M}_q\mathbf{q}^{(t)}-(3\mathbf{M}_q-2\Delta t\mathbf{D}_q)\mathbf{q}^{(t-\Delta t)}-\mathbf{G}^\top\mathbf{M}\mathbf{q}_0-\\
\iint\mu\mathbf{L}^\top\mathbf{s}^{(t-\Delta t)}dudv
\end{aligned} \tag{23}$$

Given some specified physical parameters, Equation 19 can be a nonstiff system, in which case an explicit time integration method can be used to simplify Equation 22 and Equation 23; see [14] for more details.

7 Experiments and analysis

In this section, we will demonstrate several experiments on D-DBSC, aiming to analyze the deformation of D-DBSC with different physical parameters and under different geometric constraints. Last, we will demonstrate several D-DBSC applications.

7.1 Geometric constraints on control radii

When the control radii of a D-DBSC are not restricted, the control radii change during deformation. Sometimes, varying radii can distort the shape of D-DBSC. To avoid this issue, geometric constraints can be applied on the control radii.

Here, two D-DBSC experiments with the same experimental configurations are performed (see Table 1) except for the radii constraint. In these experiments, D-DBSC is restricted with a two-end-fixed geometric constraint, and a uniform distributed force $\mathbf{f}$ is applied. We assign a uniform value to the other physical parameters for simplicity.

The result of Experiment 1 is shown in Figure 2(a). As time evolves, D-DBSC deforms under the influence of the external force, and the two ends remain fixed.

Table 1: Experimental configurations

	Experiment 1	**Experiment 2**
Geometric constraints	Two ends fixed	Two ends fixed
Constant control radii	No	Yes
μ	30	30
γ	10	10
α	40	40
β	10	10
Δt	0.008	0.008
$\mathbf{f}$	$[0\ -10^5\ 0]^\top$	$[0\ -10^5\ 0]^\top$

This experimental phenomenon is consistent with our expectation.

From the result of Experiment 1, we can observe that the control radii of D-DBSC change over time. Such changes can be problematic when D-DBSC is used for designing art because we expect D-DBSC to retain its original shape features in some art design applications. Therefore, to maintain the original shape features of D-DBSC, geometric constraints can be applied on the control radii.

In Experiment 2, we restrict the control radii to be constants, keeping them invariant over time, while the other experimental parameters are kept the same as those in Experiment 1. The result of Experiment 2 is shown in Figure 2(b). From the experimental result, we can observe that the original shape features are preserved.

(a) Varying radii

(b) Constant radii

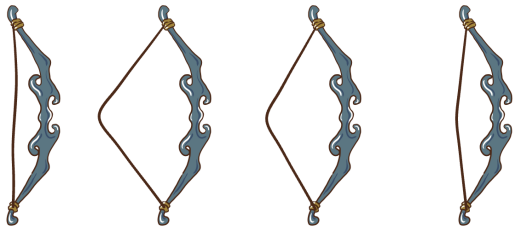
Figure 2: Deformation of a two-end-fixed D-DBSC with different radii constraints. A gravity-like force is applied to D-DBSC. The green curve is the boundary of D-DBSC, and red circles are control disks.

7.2 Tension and rigidity

The tension function α and rigidity function β describe the local physical properties of D-DBSC. By changing the values and distribution of α and β, we can simulate objects made of different materials. Now, we will analyze the effects of different tension and rigidity values.

Figure 3 shows the simulation of drawing a bow. A right-to-left force is applied at the center of the bowstring, while the ends of the bowstring are fixed. From the simulation result, we find that a larger tension makes the bowstring "tighter".

Figure 4 shows the deformation of a metal pole under an external force. The external force is top-to-bottom and is applied at the ends of the metal pole. The center of the pole is fixed. The simulation result shows that a higher rigidity makes the metal pole more difficult to deform.



(a) Initial state (b) $1 \times \alpha_0$ (c) $2 \times \alpha_0$ (d) $10 \times \alpha_0$

Figure 3: Drawing a bow under different tension conditions. α_0 is a base value.

(a) Initial state (b) $1 \times \beta_0$ (c) $10 \times \beta_0$ (d) $100 \times \beta_0$

Figure 4: Bending a metal pole under different rigidity conditions. β_0 is a base value.

7.3 D-DBSC application demonstration

Figure 5 shows a helical band that deforms under an external force. One end of this band is fixed, and a nonuniform force is applied on the band along the normal direction. The shape of the band gradually changes from a helix to a straight line. Due to the tension of the band, we can observe that the length of the band shortens over time.

Figure 5: A helical band with an applied normal direction external force

Figure 6 shows an application of D-DBSC in computer animation. In Figure 6, a carrying pole is bent under a burden. The center of the pole is fixed, and a downward external force is applied at the ends of the pole. The deformation of the carrying pole is simulated using a D-DBSC, and the position of the person and the burden are adjusted by the animators.

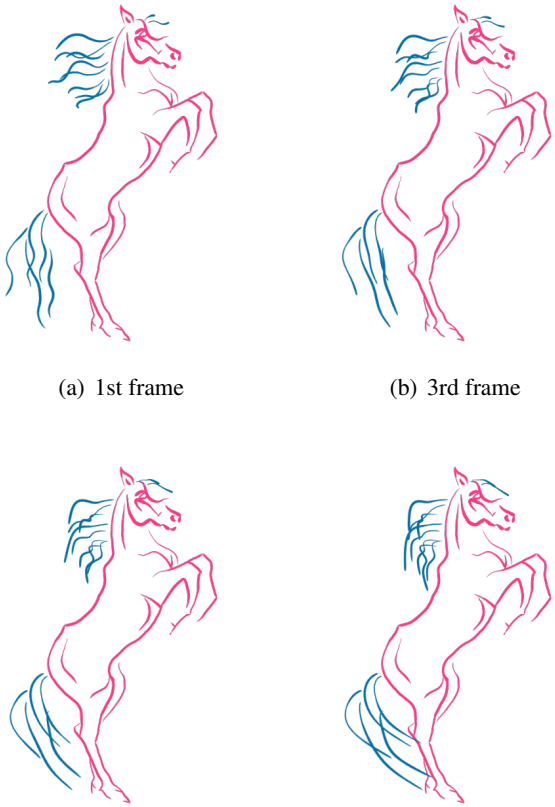
(a) 1st frame (b) 2nd frame (c) 3rd frame (d) 4th frame

Figure 6: A carrying pole is bent under a burden

Figure 7, Figure 8 and Figure 9 simulate a wind scene, in which objects are waving in the wind. The leaves of the tree, human hair and horse hair are fixed on one end, and a rightward external force is applied on them. All objects in Figure 7, Figure 8 and Figure 9 are represented using D-DBSC. In Figure 7 and Figure 8, the geometric models of the tree and human are designed using a computer, and in Figure 9, the geometric model of the horse is hand-painted by artists.

Figure 7: Wind blowing a palm tree's leaves

Figure 8: Human hair waving in the wind



(a) 1st frame (b) 3rd frame

(c) 5th frame (d) 7th frame

Figure 9: Wind blowing a horse's hair

8 Conclusion

We propose a physics-based generalization of DBSC — D-DBSC, which extends the traditional DBSC representation to time domain and can describe DBSC deformation under physical laws over time. By regarding every physics quantity as a function of time, we give the formulation of D-DBSC, which provides a theory foundation for the analysis of D-DBSC. Based on Lagrangian mechanics and the properties of D-DBSC, we derive the motion equations of D-DBSC, and investigate D-DBSC with linear geometric constraints. At last, we implement our D-DBSC theory using the finite difference method and analyze several D-DBSC experiments.

Experimental results show that D-DBSC can realistically describe the deformation of DBSC under physical laws, while our method can be easily implemented and can obtain sufficient accuracy for entertainment applications. Experiments show that D-DBSC is useful in many areas, for example, 2D dynamic geometric modeling, art design, computer calligraphy, and physics-based animations.

Moreover, D-DBSC possesses a solid theoretical foundation and provides a foundation for future related research. Since D-DBSC significantly enhances the DBSC geometric modeling method, in the foreseeable future, D-DBSC will be a useful 2D dynamic geometric modeling method.

References

[1] Hock Soon Seah, Zhongke Wu, Feng Tian, Xian Xiao, and Boya Xie. Artistic brushstroke representation and animation with disk B-spline curve. In *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, pages 88–93. ACM, 2005.

[2] Martin Peternell and Helmut Pottmann. Computing rational parametrizations of canal surfaces. *Journal of Symbolic Computation*, 23(2-3):255–266, 1997.

[3] Helmut Pottmann and Martin Peternell. Applications of laguerre geometry in cagd. *Computer Aided Geometric Design*, 15(2):165–186, 1998.

[4] Martin Peternell. Rational two-parameter families of spheres and rational offset surfaces. *Journal of Symbolic Computation*, 45(1):1–18, 2010.

[5] Roland Kunkli and Miklós Hoffmann. Skinning of circles and spheres. *Computer Aided Geometric Design*, 27(8):611–621, 2010.

[6] Qun Lin and Jon G Rokne. Disk bézier curves. *Computer Aided Geometric Design*, 15(7):721–737, 1998.

[7] Ting Zhang, Xingce Wang, Qianqian Jiang, Zhongke Wu, Mingquan Zhou, and Hock Soon Seah. An extension algorithm for disk B-spline curve with g2 continuity. *Computer-Aided Design and Applications*, 12(5):519–525, 2015.

[8] Xuefeng Ao, Qian Fu, Zhongke Wu, Xingce Wang, Mingquan Zhou, Quan Chen, and Hock Soon Seah. An intersection algorithm for disk B-spline curves. *Computers & Graphics*, 70:99–107, 2018.

[9] Demetri Terzopoulos, John Platt, Alan Barr, and Kurt Fleischer. Elastically deformable models. *ACM Siggraph Computer Graphics*, 21(4):205–214, 1987.

[10] George Celniker and Dave Gossard. Deformable curve and surface finite-elements for free-form shape design. *ACM SIGGRAPH computer graphics*, 25(4):257–266, 1991.

[11] Malcolm IG Bloor and Michael J Wilson. Representing pde surfaces in terms of B-splines. *Computer-Aided Design*, 22(6):324–331, 1990.

[12] George Celniker and Will Welch. Linear constraints for deformable non-uniform B-spline surfaces. In *Proceedings of the 1992 symposium on Interactive 3D graphics*, pages 165–170. ACM, 1992.

[13] Jeffrey A Thingvold and Elaine Cohen. Physical modeling with B-spline surfaces for interactive design and animation. In *ACM Siggraph Computer Graphics*, volume 24, pages 129–137. ACM, 1990.

[14] Demetri Terzopoulos and Hong Qin. Dynamic NURBS with geometric constraints for interactive sculpting. *ACM Transactions on Graphics (TOG)*, 13(2):103–136, 1994.

[15] Hong Qin. *Dynamic Non-Uniform Rational B-Splines*. PhD thesis, University of Toronto, 1995.

[16] Hong Qin and Demetri Terzopoulos. D-NURBS: a physics-based framework for geometric design. *IEEE Transactions on Visualization and Computer Graphics*, 2(1):85–96, 1996.

[17] Eric W. Weisstein. Euler's homogeneous function theorem.

[18] Ben Roger Gossick. Hamilton's principle and physical systems. 1967.

[19] Herbert Goldstein, Charles Poole, and John Safko. Classical mechanics, 2002.

[20] Demetri Terzopoulos. Regularization of inverse visual problems involving discontinuities. *IEEE Transactions on pattern analysis and Machine Intelligence*, (4):413–424, 1986.

[21] Josildo Pereira da Silva, Antônio Lopes Apolinário Júnior, and Gilson A. Giraldi. A Review of Dynamic NURBS Approach. *arXiv e-prints*, page arXiv:1303.6683, Mar 2013.

[22] Hayrettin Kardestuncer and Douglas H Norrie. *Finite element handbook*. McGraw-Hill, Inc., 1987.