Acta Univ. Sapientiae Informatica **13,** 2 (2021) 303–323

DOI: 10.2478/ausi-2021-0014

Dual quaternion-based osculating circle algorithm for finding intersection curves

Vahide BULUT
Izmir Katip Celebi University,
Department of Engineering Sciences,
Cigli, Izmir, 35620, Turkey
email: vahide.bulut@ikcu.edu.tr

Abstract. The intersection of surfaces is a fundamental process in computational geometry and computer-aided design applications to build and interrogate complex shapes in the computer. This paper presents a novel and simple dual quaternion-based osculating circle DQOC algorithm to find the intersection curve of two regular surfaces based on the osculating circle concept and dual quaternions. Additionally, we expressed the natural equations of the intersection curve. We have also demonstrated the superiority of our method through numerical examples.

1 Introduction

Surface/surface intersections are widely used in computer-aided manufacturing (CAM) / computer-aided design (CAD), such as path planning, animations, and modeling some shapes. Numerical methods are generally preferred for the intersection of two surface cases.

The marching method provides an intersection curve's points sequences by utilizing the local differential geometry [8],[10]. First, the initial point must be determined to proceed through this method, and next, the point continues along the intersection curve by marching. The intersection curve's local

Computing Classification System 1998: I.3.3, I.3.5, J.2
Mathematics Subject Classification 2010: 65D17, 65D05, 53A04, 53A05
Key words and phrases: Surface intersection, Marching method,osculating circle, dual quaternion, intersection curve

geometric properties are used to compute the marching direction plus steps. The marching step is computed via the sequence of points that originated from the first point. Several possible solutions exist for marching directions, including moving in a tangential direction [2], [9] or traveling along a circle [1] or a parabola [12]. However, the most commonly used solution is step length, based on the curve's curvature.

Not only are dual quaternions vital because they perform the solution more robust and straightforward, but they also provide a compact, unambiguous, singularity - free rigid transform using minimal computations. Another positive feature of dual quaternions is that they are the most efficient and most compact form that can be utilized to represent rotation and translation. Additionally, they can solve a problem more rapidly while doing it in fewer steps, and they show the result more clearly. Also, fewer code lines are used for dual quaternions to practice [7].

In literature, some authors have used lengthy calculations to determine the step length and next intersection point. This study presents a new and applicable DQOC algorithm that employs fewer computational calculations to obtain the solution less complicated. Additionally, we obtain closer points to the intersection curve. On the other hand, we present the natural equations of the intersection curve. We also compare our algorithm with the method that Wu and Andrade used [15].

2 Preliminaries

2.1 Dual quaternions

Definition 1 *A dual number is written as*

$$A = a + \varepsilon \bar{a}$$

in which, a *and* $\bar{a}$ *are real numbers and* $\varepsilon^2 = 0$*,* $\varepsilon \neq 0$ *[13], [16].*

Definition 2 *An ordinary quaternion is defined as*

$$q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$$

where $\mathbf{i}, \mathbf{j}, \mathbf{k}$ *are the standard orthonormal basis in* R^3*, providing* $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$*, and* a, b, c, d *are real numbers.*

Quaternions can present a rigid body's rotation according to an axis. Quaternions do not cause any singularity problem and provide the keyframe interpolation better [5], [11], [6].

Definition 3 *A dual quaternion can be expressed by*

$$\hat{Q} = Q + \varepsilon Q^* \tag{1}$$

in which

$$Q = q_r + \vec{q} \qquad Q^* = q_r^* + \vec{q}^*$$

and $\varepsilon^2 = 0$.

Dual quaternions were first described by W. Kingdon Clifford in 1873 [3]. They are different from the real quaternions because they are utilized for both translation and rotation. A dual quaternion's four dual number terms can be interpreted as

$$\hat{Q} = \hat{q}_r + \hat{q}_x\mathbf{i} + \hat{q}_y\mathbf{j} + \hat{q}_z\mathbf{k} \tag{2}$$

or

$$\hat{Q} = s + x\mathbf{i} + y\mathbf{j} + z\mathbf{k} + \varepsilon\left(s_\varepsilon + x_\varepsilon\mathbf{i} + y_\varepsilon\mathbf{j} + z_\varepsilon\mathbf{k}\right).$$

Lemma 4 *If we take the two dual quaternions as*

$$\hat{Q}_1 = Q_1 + \varepsilon Q_1^*, \qquad \hat{Q}_2 = Q_2 + \varepsilon Q_2^*,$$

then, the dual quaternion multiplication can be given with

$$\hat{Q}_1\hat{Q}_2 = Q_1Q_2 + \varepsilon\left(Q_1Q_2^* + Q_2Q_1^*\right).$$

Lemma 5 *Dual conjugate* $\overline{\hat{Q}}$ *and dual quaternion norm can be defined as*

$$\overline{\hat{Q}} = Q - \varepsilon Q^*$$

and

$$\|\hat{Q}\| = \|Q\| + \varepsilon\frac{< Q, Q^* >}{\|Q\|}.$$

Definition 6 *A dual quaternion that satisfies the conditions* $\|\hat{Q}\| = 1$ *and* $< Q, Q^* >= 0$ *is called a unit dual quaternion.*

Lemma 7 *The inverse of a dual quaternion is*

$$\hat{Q}^{-1} = \frac{1}{Q} - \varepsilon \frac{Q^*}{Q^2}$$

in which, $Q \neq 0$.

Lemma 8 *A second conjugation operator for a dual quaternion is*

$$\overline{\hat{Q}}^* = (s, -x, -y, -z, -s_\varepsilon, x_\varepsilon, y_\varepsilon, z_\varepsilon)\,.$$

Lemma 9 *Transformations can be represented by only one dual quaternion. Let* $\hat{Q}$ *and* $\hat{P}$ *be two transformation dual quaternions and* Q_v *be a position vector dual quaternion. Then, the combined transformation* C *can be applied to* Q_v *as*

$$\hat{Q}_v' = \hat{P}\left(\hat{Q}Q_v\overline{\hat{Q}}^*\right)\overline{\hat{P}}^* = \left(\hat{P}\hat{Q}\right)(Q_v)\left(\overline{\hat{Q}}^*\overline{\hat{P}}^*\right) \tag{3}$$

or

$$\hat{C} = \hat{P}\hat{Q} \Rightarrow \hat{Q}_v' = \hat{C}Q_v\overline{\hat{C}}^*.$$

Lemma 10 *Unit dual quaternions represent the three-dimensional 3D rotation with an angle* θ *and a unit axis* $\mathbf{n}$ *when the dual part* $Q^* = 0$.

$$\begin{aligned}\hat{Q}_r = \quad & \left[\cos\left(\tfrac{\theta}{2}\right), \mathbf{n}_x \sin\left(\tfrac{\theta}{2}\right), \mathbf{n}_y \sin\left(\tfrac{\theta}{2}\right),\right. \\ & \left.\mathbf{n}_z \sin\left(\tfrac{\theta}{2}\right)\right][0, 0, 0, 0]\end{aligned}$$

Lemma 11 *A pure translation can be expressed in terms of a dual quaternion by*

$$\hat{Q}_t = [1, 0, 0, 0]\left[0, \frac{t_x}{2}, \frac{t_y}{2}, \frac{t_z}{2}\right]$$

Lemma 12 *On the following, a single unit quaternion can be used to denote a rotation followed by a translation as* [7], [18]

$$\hat{Q} = \hat{Q}_t \times \hat{Q}_r. \tag{4}$$

3 Intersection curve

3.1 Geometric concepts of the intersection curve

First and second derivatives of the parametric curve $\boldsymbol{\alpha} = \boldsymbol{\alpha}(s)$ in R^3 in terms of arc-length parameter are given by

$$\boldsymbol{\alpha}'(s) = \mathbf{T}, \qquad \boldsymbol{\alpha}''(s) = \kappa\mathbf{N} \tag{5}$$

in which the vectors **T**, **N** correspond to the components of the Frenet-Serret frame, and κ represent the curvature of this curve.

Definition 13 *Assume that the points* P*,* Q*, and* M *be on the same curve. If the points* P *and* M *tend to* Q*, the circle's limit that passes through all these points is called the osculating circle at* Q*.*

The osculating circle can then be used as an approximation of the intersection curve because the curve's osculating circle at a point has the same tangent and curvature as the curve at that point [4].

Suppose that the intersection curve $\boldsymbol{\alpha}(s)$ is given as of regular surfaces $\mathbf{S}_1(u,v)$ and $\mathbf{S}_2(r,s)$. The tangent vector of this curve can be written using the surfaces' unit normal vectors $\mathbf{n}_1$ and $\mathbf{n}_2$ at the point P with following equation [17].

$$\mathbf{T} = \frac{\mathbf{n}_1 \times \mathbf{n}_2}{\|\mathbf{n}_1 \times \mathbf{n}_2\|}. \tag{6}$$

In this paper, the intersection curve is obtained by the transversal intersection of two regular surfaces.

3.2 Marching algorithm

Finding the starting points of the intersection curves plays a vital role in tracing methods. Analytically, we need to compute the below equation to determine these two parametric surfaces' intersection.

$$\mathbf{S}_1(u,v) = \mathbf{S}_2(r,s). \tag{7}$$

In the case of not intersecting these surfaces, the minimum distance is computed between two surfaces to determine the starting points. For instance, if the two surfaces exist in the same bounding box, we can compute a starting point using the bounding box method.

For every next point, we can find the step vector at each point by using the marching algorithm and the marching process can be performed in parametric or Cartesian space. In Cartesian spaces, the marching method uses a fixed-step. However, with our algorithm, we consider the curve's local geometry instead of a fixed step [14].

This paper utilizes the osculating circle to compute the length of each step. The reason is that the intersection curve's osculating circle $\boldsymbol{\alpha}(s)$ can best approximate the curve at the same point [15].

4 Proposed method

Our proposed method focused on the osculating circle of the intersection curve via using dual quaternions:

- We need to find initial points.
- We can compute the center and the osculating circle's radius by using these initial points.
- We obtain the next intersection point using dual quaternions.

In our method, we begin with determining the starting points by applying the steps as follows:

Step 1: Determine the point P to be a starting point using the equation (7).

Step 2: Obtain the point Q. It is accepted that the distance between the points P and Q is $\frac{\mathsf{L}}{(\mathsf{k}.10)}$. The initial value of k is taken as 2. Here, L indicates the step length.

Step 3: Increase the value of k by one, and continue by Step 2 if the point Q becomes a singular point. On the other hand, if the singular case remains the same, the value of k can be increased sequentially till 10.

Step 4: Continue with step 1 if the point Q can not be found because of the singularity.

Next, to find the osculating circle's center and radius, we utilized the method in [15]. We can compute the osculating circle's center with the following linear system of equations.

$$\begin{cases} \mathsf{C}\mathbf{u} = \mathsf{P}\mathbf{u} \\ \mathsf{C}\mathbf{v} = \mathsf{Q}\mathbf{v} \\ \mathsf{C}\mathbf{w} = \mathsf{Q}\mathbf{w} \end{cases} \tag{8}$$

in which C is the osculating circle's center point at Q. This center point C is obtained via the intersection of three planes that have the normal vectors $\mathbf{u}$, $\mathbf{v}$ and $\mathbf{w}$. Also, the distance between C and Q indicates the osculating circle's radius.

The step length can be obtained right after the osculating circle approximation. Wu and Andrade first computed the normal vector to the circumference plane. After the transformations of translation and rotation, they moved the osculating circle to the XOY plane by placing the center to $\mathsf{O} = (0, 0, 0)$. Consequently, they obtained the transformed points and then determined whether

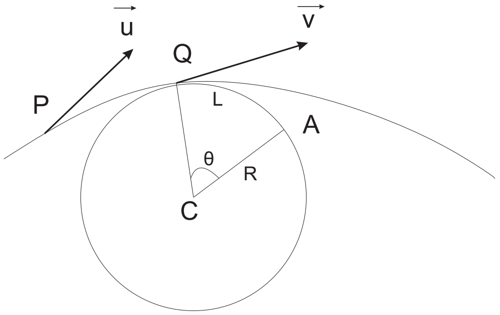


Figure 1: The circular step is obtained by the proposed method.

the orientation of the arc $\left(\widehat{PQ}\right)$ was clockwise or counterclockwise. Next, they found point $A^{'}$ after some computations and applied the inverse transformations to point $A^{'}$ to ascertain the next intersection point A [15]. For this process, they used very long calculations to find the next intersection point A.

In our new method, we determined θ as an increment L as shown in Fig. 1. We can compute the next intersection point A using only the equation (3) below as

$$\begin{aligned} A = \hat{Q}Q\overline{\hat{Q}}^{*} &= [1,\ 0,\ 0,\ 0] + \varepsilon[0,\ c_1 + (q_1 - c_1)\cos\theta + (c_2 - q_2)\sin\theta, \\ &\quad c_2 + (q_2 - c_2)\cos\theta + (q_1 - c_1)\sin\theta,\ q_3]. \end{aligned} \tag{9}$$

Using the equation (9), we translate the osculating circle's center C to the origin O, implement the rotation, and then re-translate C to its initial position and find the next intersection point applying only unit dual quaternions. The proposed dual quaternion based osculating circle DQOC algorithm is given in Algorithm 1.

5 Natural equation of the intersection curve

We can obtain the natural equation of the intersection curve obtained by the transversal intersection of two regular surfaces, using the Darboux frames of these surfaces.

Theorem 14 *Let two surfaces be given as* $\mathbf{S}_1(u,v)$ *and* $\mathbf{S}_2(r,s)$*, and the intersection curve of these surfaces be* $\boldsymbol{\alpha}(s)$ *parametrized with arc length parameter.*

We can expand series of the intersection curve $\boldsymbol{\alpha}(s)$ *at* $s = 0$ *as*

$$\boldsymbol{\alpha}(s) = \frac{s}{1!}\frac{d\boldsymbol{\alpha}}{ds} + \frac{s^2}{2!}\frac{d^2\boldsymbol{\alpha}}{ds^2} + \frac{s^3}{3!}\frac{d^3\boldsymbol{\alpha}}{ds^3} + \ldots \tag{10}$$

We can write the natural equation of the intersection curve $\boldsymbol{\alpha}(s)$ *at* $s = 0$ *in terms of the Darboux frame coefficients of the surfaces* $\mathbf{S}_1$ *and* $\mathbf{S}_2$ *with the following equation.*

$$\begin{aligned}
\boldsymbol{\alpha}(s) \quad &= \frac{s}{1!}t + \frac{s^2}{2!}\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[-a\left(\tau_{g_1} - \tau_{g_2}\right)\mathbf{n}\right. \\
&\left. -k_{n_2}\mathbf{g}_1 + \tau_{g_1} b\mathbf{n}_1 + k_{n_1}\mathbf{g}_2 - \tau_{g_2} c\mathbf{n}_2\right] \\
&= +\frac{s^3}{3!}\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[-a'\left(\tau_{g_1} - \tau_{g_2}\right) - a\left(\tau'_{g_1} - \tau'_{g_2}\right)\right. \\
&+\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}a^2\left(\tau_{g_1} - \tau_{g_2}\right)^2 + k_{n_2}\mathbf{g}_1 \\
&\left. -\tau_{g_1} b k_{n_1} - k_{n_1}k_{g_2} - \tau_{g_2}k_{n_2}\right]\mathbf{t} \\
&+\frac{s^3}{3!}\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}a\left(\tau_{g_1} - \tau_{g_2}\right)k_{n_2}\right. \\
&\left. -k'_{n_2} - \tau^2_{g_1} b\right]\mathbf{g}_1 \\
&+\frac{s^3}{3!}\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[-\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}a\left(\tau_{g_1} - \tau_{g_2}\right)\tau_{g_1} b\right. \\
&\left. -k_{n_2}\tau_{g_1} + \tau'_{g_1} b + \tau_{g_1} b'\right]\mathbf{n}_1 \\
&+\frac{s^3}{3!}\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[-\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}a\left(\tau_{g_1} - \tau_{g_2}\right)k_{n_1} + k'_{n_1} + \tau^2_{g_2}\right]\mathbf{n}_2 \\
&+\frac{s^3}{3!}\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[+\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}a\left(\tau_{g_1} - \tau_{g_2}\right)\tau_{g_2} c\right. \\
&\left. +k_{n_1}\tau_{g_2} - \tau'_{g_2} c - \tau_{g_2} c'\right]\mathbf{n}_2 \\
&+\ldots
\end{aligned} \tag{11}$$

Proof. Since the intersection curve will take place on both surfaces, Darboux frames of this curve on the surfaces $\mathbf{S}_1$ and $\mathbf{S}_2$ can be taken as $\{\mathbf{t}_1, \mathbf{g}_1, \mathbf{n}_1\}$ and $\{\mathbf{t}_2, \mathbf{g}_2, \mathbf{n}_2\}$. These frames are right-handed orthogonal frames associated with each point of the intersection curve $\boldsymbol{\alpha}(s)$, where $\mathbf{t}_1$ and $\mathbf{t}_2$ are the unit tangent vectors of $\boldsymbol{\alpha}(s)$ at the surfaces $\mathbf{S}_1$ and $\mathbf{S}_2$; $\mathbf{n}_1$ and $\mathbf{n}_2$ are the unit normal vectors of the surfaces $\mathbf{S}_1$ and $\mathbf{S}_2$; $\mathbf{g}_1$ and $\mathbf{g}_2$ are the geodesic normals of the surfaces $\mathbf{S}_1$ and $\mathbf{S}_2$, respectively. On the other hand, we can write the unit tangent vector of the intersection curve using the equation (6) as

$$\frac{d\boldsymbol{\alpha}}{ds} = \frac{\mathbf{n}_1 \wedge \mathbf{n}_2}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|} = \mathbf{t} = \mathbf{t}_1 = \mathbf{t}_2. \tag{12}$$

Next, we can get the second derivative of the above equation as follows:

$$\begin{aligned}
\frac{d^2\boldsymbol{\alpha}}{ds^2} &= \frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[\mathbf{n}_1^{'} \wedge \mathbf{n}_2 + \mathbf{n}_1 \wedge \mathbf{n}_2^{'}\right] \\
&= \frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[\left(-k_{n_1}\mathbf{t} - \tau_{g_1}\mathbf{g}_1\right) \wedge \mathbf{n}_2 + \mathbf{n}_1 \wedge \left(-k_{n_2}\mathbf{t} - \tau_{g_2}\mathbf{g}_2\right)\right] \\
&= \frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[k_{n_1}\mathbf{n}_2 - \tau_{g_1}\left(\left(\mathbf{n}_1\mathbf{n}_2\right)\mathbf{t} - \left(\mathbf{t}\mathbf{n}_2\right)\mathbf{n}_1\right)\right. \\
&\left. -k_{n_2}\mathbf{g}_1 - \tau_{g_2}\left(\left(\mathbf{n}_1\mathbf{t}\right)\mathbf{n}_2 - \left(\mathbf{n}_1\mathbf{n}_2\right)\mathbf{t}\right)\right] \\
&= \frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[-a\left(\tau_{g_1} - \tau_{g_2}\right)\mathbf{t}\right. \\
&\left. -k_{n_2}\mathbf{g}_1 + \tau_{g_1}b\mathbf{n}_1 + k_{n_1}\mathbf{g}_2 - \tau_{g_2}c\mathbf{n}_2\right]
\end{aligned} \tag{13}$$

where, $\mathbf{n}_1\mathbf{n}_2 = a$, $\mathbf{t}\mathbf{n}_2 = b$ and $\mathbf{n}_1\mathbf{t} = c$. Also, we can write the third derivative of the equation (12) with below equation as

$$
\begin{aligned}
\frac{d^3\boldsymbol{\alpha}}{ds^3} &= \frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}[-a'\left(\tau_{g_1} - \tau_{g_2}\right)\mathbf{t} \\
&-a\left(\tau'_{g_1} - \tau'_{g_2}\right)\mathbf{t} - a\left(\tau_{g_1} - \tau_{g_2}\right)\mathbf{t}' - k'_{n_2}\mathbf{g}_1 - k_{n_2}\mathbf{g}'_1 \\
&+\tau'_{g_1} b\mathbf{n}_1 + \tau_{g_1} b'\mathbf{n}_1 + \tau_{g_1} b\mathbf{n}'_1 + k'_{n_1}\mathbf{g}_2 \\
&+k_{n_1}\mathbf{g}'_2 - \tau'_{g_2} c\mathbf{n}_2 - \tau_{g_2} c'\mathbf{n}_2 - \tau_{g_2} c\mathbf{n}'_2] \\
&= \frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\Big[-a'\left(\tau_{g_1} - \tau_{g_2}\right) - a\left(\tau'_{g_1} - \tau'_{g_2}\right) \\
&+\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|} a^2\left(\tau_{g_1} - \tau_{g_2}\right)^2 + k_{n_2}\mathbf{g}_1 - \tau_{g_1} b k_{n_1} \\
&-k_{n_1} k_{g_2} - \tau_{g_2} k_{n_2}]\,\mathbf{t} \\
&+\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|} a\left(\tau_{g_1} - \tau_{g_2}\right) k_{n_2} - k'_{n_2} - \tau^2_{g_1} b\right]\mathbf{g}_1 \\
&+\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\Big[-\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|} a\left(\tau_{g_1} - \tau_{g_2}\right)\tau_{g_1} b \\
&-k_{n_2}\tau_{g_1} + \tau'_{g_1} b + \tau_{g_1} b'\Big]\,\mathbf{n}_1 \\
&+\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\left[-\frac{1}{\|n_1 \wedge n_2\|} a\left(\tau_{g_1} - \tau_{g_2}\right) k_{n_1} + k'_{n_1} + \tau^2_{g_2}\right]\mathbf{g}_2 \\
&+\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\Big[+\frac{1}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|} a\left(\tau_{g_1} - \tau_{g_2}\right)\tau_{g_2} c \\
&+k_{n_1}\tau_{g_2} - \tau'_{g_2} c - \tau_{g_2} c'\Big]\,\mathbf{n}_2.
\end{aligned} \tag{14}
$$

If we continue taking this derivative process as above, we can find the n^{th} derivative with following equation.

$$
\frac{d^n\boldsymbol{\alpha}}{ds^n} = u_n\mathbf{t} + v_n\mathbf{g}_1 + w_n\mathbf{n}_1 + v^*_n\mathbf{g}_2 + w^*_n\mathbf{n}_2 \tag{15}
$$

where, u_n, v_n, w_n, v_n^* and w_n^* are the known functions of sequential derivatives of k_{n_i}, k_{g_i} and τ_{g_i}, $i = 1, 2$. Taking derivative of (15), we can get the equation below.

$$
\begin{aligned}
\frac{d^{n+1}\boldsymbol{\alpha}}{ds^{n+1}} &= \frac{du_n}{ds}\mathbf{t} + u_n\mathbf{t}' + \frac{dv_n}{ds}\mathbf{g}_1 + v_n\mathbf{g}_1^{'} + \frac{dw_n}{ds}\mathbf{n}_1 \\
&+ w_n\mathbf{n}_1^{'} + \frac{dv_n^*}{ds}\mathbf{g}_2 + v_n^*\mathbf{g}_2^{'} + \frac{dw_n^*}{ds}\mathbf{n}_2 + w_n^*\mathbf{n}_2^{'} \\
&= \Bigg[\frac{du_n}{ds} - k_{g_1}v_n - k_{n_1}w_n - k_{g_2}v_n^* - k_{n_2}w_n^* \\
&- \frac{u_n}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}a\left(\tau_{g_1} - \tau_{g_2}\right)\Bigg]\mathbf{t} \\
&+ \left(\frac{dv_n}{ds} - \tau_{g_1}w_n - \frac{u_n}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}ak_{n_2}\right)\mathbf{g}_1 \\
&+ \left(\frac{dw_n}{ds} + \tau_{g_1}v_n + \frac{u_n}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\tau_{g_1}b\right)\mathbf{n}_1 \\
&+ \left(\frac{dv_n^*}{ds} - \tau_{g_2}w_n^* + \frac{u_n}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}k_{n_1}\right)\mathbf{g}_2 \\
&+ \left(\frac{dw_n^*}{ds} + \tau_{g_2}v_n^* - \frac{u_n}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|}\tau_{g_2}c\right)\mathbf{n}_2.
\end{aligned}
\tag{16}
$$

Thus, we can obtain the recurrence formulas as

$$\begin{aligned}
u_{n+1} &= \frac{du_n}{ds} - k_{g_1} v_n - k_{n_1} w_n - k_{g_2} v_n^* - k_{n_2} w_n^* \\
&\quad - \frac{u_n}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|} a\left(\tau_{g_1} - \tau_{g_2}\right), \\
v_{n+1} &= \frac{dv_n}{ds} - \tau_{g_1} w_n - \frac{u_n}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|} a k_{n_2}, \\
w_{n+1} &= \frac{dw_n}{ds} + \tau_{g_1} v_n + \frac{u_n}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|} \tau_{g_1} b, \\
v_{n+1}^* &= \frac{dv_n^*}{ds} - \tau_{g_2} w_n^* + \frac{u_n}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|} k_{n_1}, \\
w_{n+1}^* &= \frac{dw_n^*}{ds} + \tau_{g_2} v_n^* - \frac{u_n}{\|\mathbf{n}_1 \wedge \mathbf{n}_2\|} \tau_{g_2} c.
\end{aligned} \tag{17}$$

Finally, using the equations (12), (13), (14), (15), (16) and (17) the natural equation of the intersection curve given in (11) can be written. □

6 Experimental results

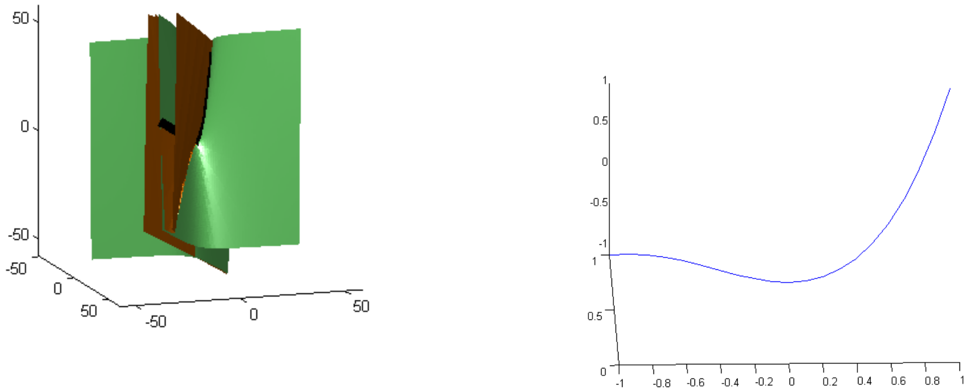
This study compares our proposed dual quaternion-based osculating circle DQOC algorithm with Wu and Andrade's [15] method. To determine the intersection curve points, we need to check the parallelism between the tangent vectors **u** and **v** at the points P and Q, and so on. If **u** and **v** are parallel, the curvature radius goes to infinity. Therefore, the best choice for step vector is the tangent vector at Q. We used the value $\cos\theta$, which is the angle between the tangent vectors of consecutive intersection points. If $\cos\theta \geq 1-\varepsilon$ or $\cos\theta \leq -1+\varepsilon$, we can assume that the vectors **u** and **v** are parallel. If, $\cos\theta \longrightarrow 1$ or $\cos\theta \longrightarrow -1$, then radius of the osculating circle $\longrightarrow \infty$. Because of this situation, we defined a parallel threshold value called ε above. In the following applications, we obtained closer points to the intersection curve than their method did. On the other hand, the number of next intersection points on the tangential direction of the point Q can change according to the threshold value. If the threshold value grows steadily, then, the number of next intersection points may increase on the tangential direction of the point Q. If the threshold value is taken as very small, next intersection points are

Algorithm 1 Dual Quaternion based Osculating Circle DQOC Algorithm

```
Input: Two surfaces, $S_i$ $(i = 1, 2)$
Output: Points on the intersection curve
global Points, pointA, pointP, pointQ, pointM, center
global Tangent vectors, pointPtan, pointQtan
global Normal vector of plane, normal; radius of osculating circle, R
global Transformation matrices and their elements, lambda, V, TMatrix, RMatrix, pointPTrans, pointQTrans, directionValue, beta, dist, L;
 ▷ Find the initial points of the intersection of two surfaces $S_i$ $(i = 1, 2)$
$count \leftarrow 1$;
$tValue1 \leftarrow 0.0$;
$tStartValue \leftarrow 0.0$;
$tValue1 \leftarrow tStartValue + count * 0.005$;
findInitialPointsPQM(tValue1);
while $count < 2$ do
    $count \leftarrow count + 1$;
    $parallel \leftarrow isParallel(pointPtan, pointQtan)$;
    if parallel then
        $pointA \leftarrow getNewPointAlongTangent(pointQ, L)$;
    else
        findCenterOfOsculatingCircle();
        $R \leftarrow getRadiusOfOsculator(pointQ, center)$;
        $normal \leftarrow getNormalVector(center, pointQ, pointP)$;
        $lambda \leftarrow getLambda(normal)$;
        $V \leftarrow getV(normal)$;
        $TMatrix \leftarrow getTMatrix(center)$;  ▷ Obtain translation matrix
        $RMatrix \leftarrow getRMatrix(normal)$;  ▷ Obtain rotation matrix
        findTransformedPoints();
        $directionValue \leftarrow getDirectionValue(pointPTrans, pointQTrans)$;
        $beta \leftarrow getBeta(directionValue)$;
        $pointA \leftarrow getPointA(R, beta, TMatrix, RMatrix)$;
        $dist \leftarrow getDistanceToCurve(pointA)$;
    end if
    $pointAtan \leftarrow getTangentOfPoint(pointA)$;
    updatePoints();
end while
```

going to be far from the intersection curve. First, we present intersection of two surfaces and intersection curves of these surfaces as seen from the figures (6) and (6) for example 1, figures (6(a)) and (6) for example 2, and figures (10(a)) and (10) for example 3. Next, we compared our method with Wu and Andrade's [15] method for different parallel threshold values as seen from the figures (3), (4) and (5) for example 1, figures (7), (8) and (9) for example 2, and figures (11), (12) and (13) for example 3.

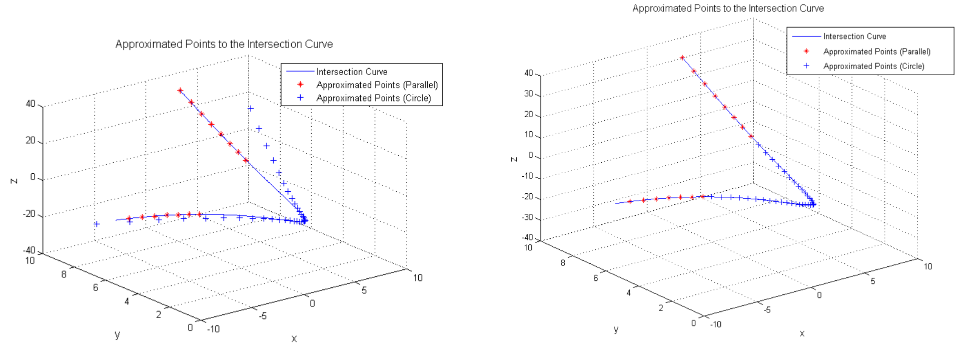
Example 1: Let two surfaces be given as $\mathbf{S}_1 : xy - z = 0$ and $\mathbf{S}_2 : z = x^3 + xy - 2z = 0$. Then, the intersection curve of these two surfaces can be expressed as $\boldsymbol{\alpha}(t) = \left(t, t^2, t^3\right)$.



(a) The intersection of surfaces $\mathbf{S}_1 : xy - z = 0$ $\mathbf{S}_2 : z = x^3 + xy - 2z = 0$

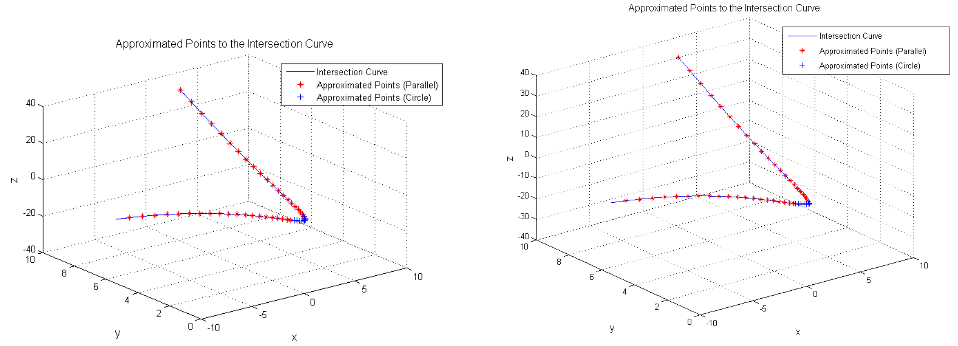
(b) The intersection curve $\boldsymbol{\alpha}(t) = \left(t, t^2, t^3\right)$

Figure 2: Intersection curve of two given surfaces.



(a) (b)

Figure 4: The figure in (a) shows the approximated points found by the method in ([15]), and the figure (b) shows the approximated points found by the proposed method based on the threshold value 10^{-9}.



(a) (b)

Figure 3: The figure in (a) shows the approximated points found by the method in ([15]), and the figure (b) shows the approximated points found by our method based on the threshold value 10^{-7}.

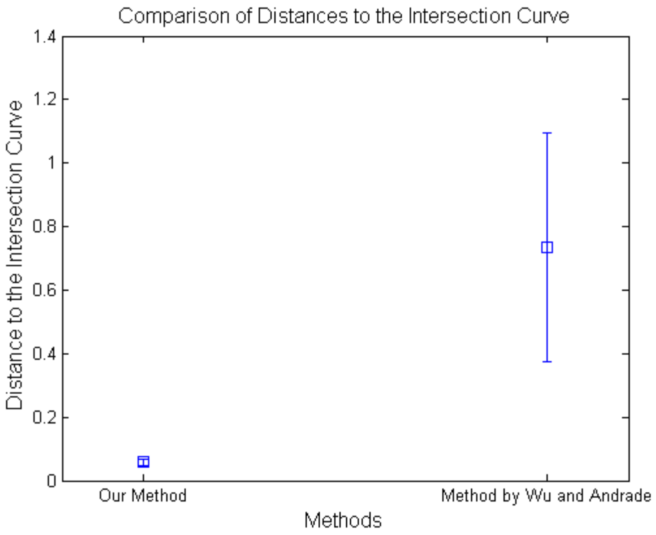
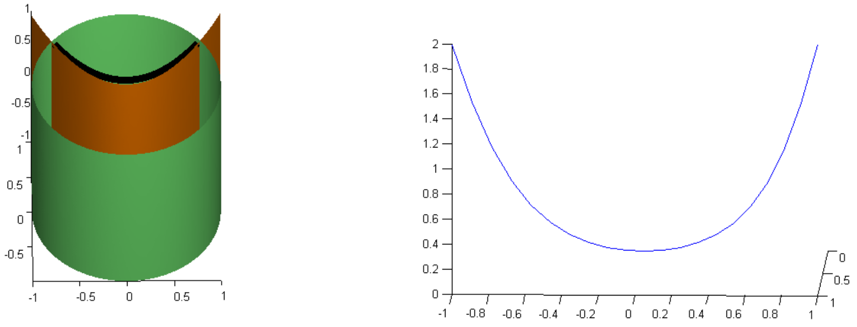


Figure 5: Comparison of distances to the intersection curve.

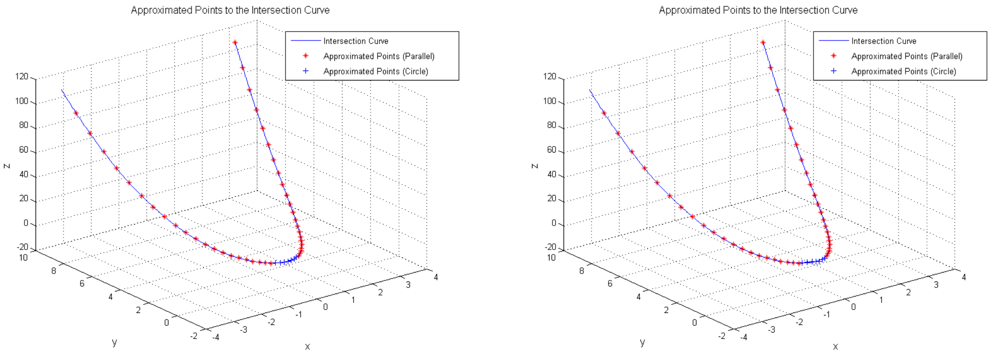
Example 2: The surfaces are given as $\mathbf{S}_1 : x^2 + y^2 = z$ and $\mathbf{S}_2 : y = x^2$. We can define the intersection curve with $\boldsymbol{\alpha}(t) = \left(t,\, t^2,\, t^2 + t^4\right)$.



(a) The intersection of surfaces $\mathbf{S}_1 : x^2 + y^2 = z$ and $\mathbf{S}_2 : y = x^2$.

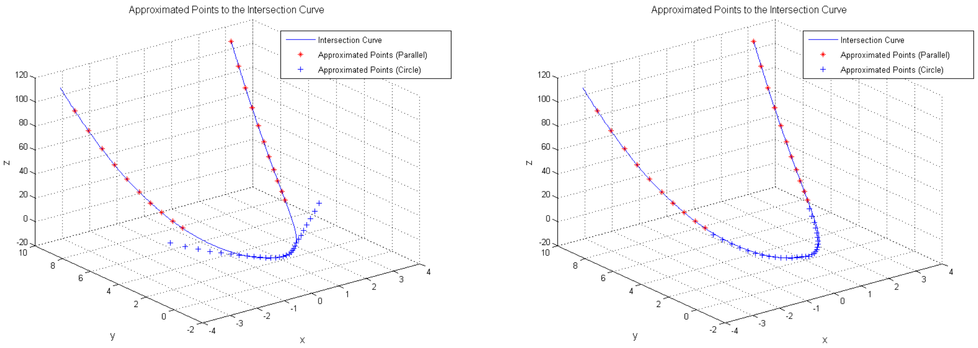
(b) The intersection curve $\boldsymbol{\alpha}(t) = \left(t,\, t^2,\, t^2 + t^4\right)$.

Figure 6: Intersection curve of two given surfaces.



(a) (b)

Figure 7: The figure in (a) shows the approximated points found by the method in ([15]), and the figure (b) shows the approximated points found by our method based on the threshold value 10^{-7}.



(a) (b)

Figure 8: The figure in (a) shows the approximated points found by the method in ([15]), and the figure (b) shows the approximated points found by the proposed method based on the threshold value 10^{-9}.

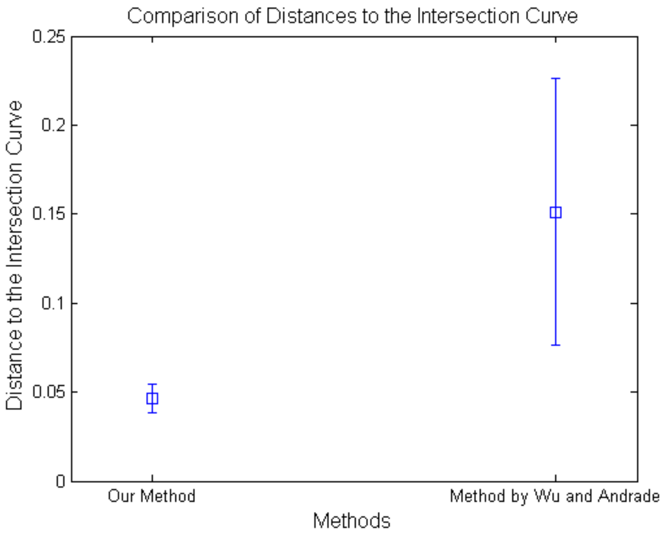
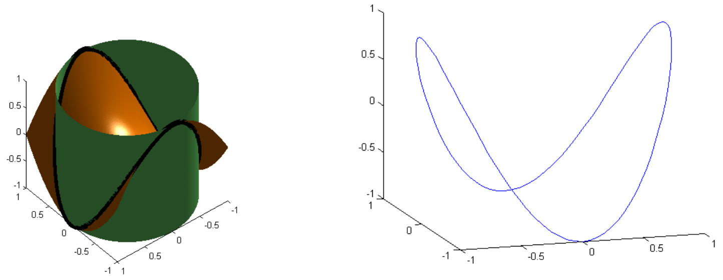


Figure 9: Comparision of distances to the intesection curve.

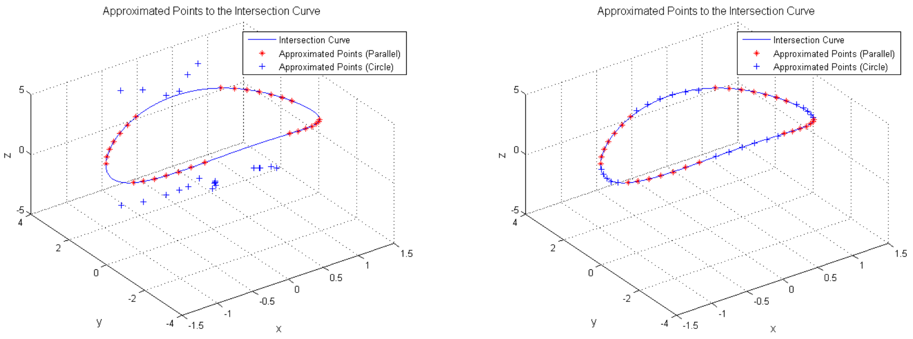
Example 3: Let's take two surfaces as $\mathbf{S}_1 : x^2 + y^2 = 1$ and $\mathbf{S}_2 : z = x^2 - y^2$. We can find the intersection curve as $\boldsymbol{\alpha}(t) = (\cos t,\ \sin t,\ \cos 2t,)$.



(a) The intersection of surfaces $\mathbf{S}_1 : x^2 + y^2 = 1$ and $\mathbf{S}_2 : z = x^2 - y^2$.

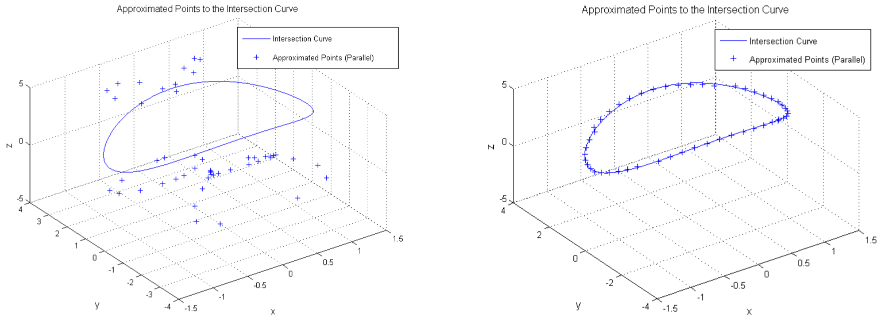
(b) The intersection curve $\boldsymbol{\alpha}(t) = (\cos t,\ \sin t,\ \cos 2t,)$.

Figure 10: Intersection curve of two given surfaces.



(a) (b)

Figure 11: The figure on the left shows the approximated points found by the method in ([15]), and the one on the right shows the approximated points found by our method based on the threshold value 10^{-7}.



(a) (b)

Figure 12: The figure in (a) shows the approximated points found by the method in ([15]), and the figure (b) shows the approximated points found by the proposed method based on the threshold value 10^{-9}.

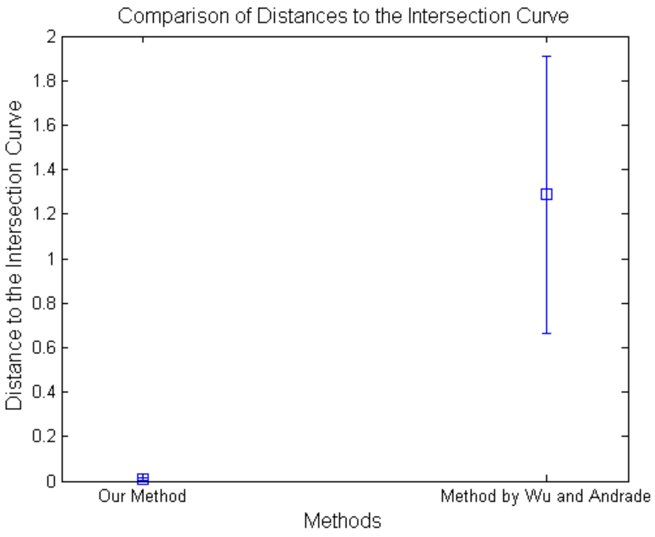


Figure 13: Comparision of distances to the intesection curve.

7 Conclusion

In this study, we presented a novel dual quaternion-based osculating circle DQOC algorithm for finding the intersection curve points of regular two surfaces based on the differential geometric properties of the curve and the use of dual quaternions. We obtained closer points to the intersection curve when we compare our method with Wu and Andrade's [15] method. Our proposed method is more efficient and accurate when we compare our method with Wu and Andrade's method for different parallel threshold values. Also, we used fewer calculations for this process. Moreover, we gave the natural equations of the intersection curve.

References

[1] C. Asteasu, Intersection of arbitrary surfaces. *Computer Aided Design*, **20** (1988) 533–538. ⇒304

[2] R. E. Barnhill, S. N. Kersey, A marching method for parametric surface/surface intersection. *Computer Aided Geometric Design*, **7** (1990) 257–280. ⇒304

[3] W. Clifford, Preliminary sketch of bi-quaternions. *Proc. London Math. Soc.*, **4,** 1 (1873) 381–395. ⇒305

[4] M. Do Carmo, *Differential geometry of curves and surfaces.* Prentice-Hall, New Jersey, 1976. ⇒307

[5] W. R. Hamilton, On quaternions; or a new system of imaginaries in algebra.v*Phil. Mag. J.* **25** (1844) 489–495. ⇒305

[6] W. R. Hamilton, *Elements of quaternions, Vol. I and II.* Chelsea Publishing Company, New York, 1869. ⇒305

[7] B. Kenwright A beginners guide to dual-quaternions: What they are, how they work and how to use them for 3d character hierarchies. *In The 20th International Conference on Computer Graphics, Visualization and Computer Vision,* 2012. ⇒304, 306

[8] R. E. Lynch, C. L. Bajaj, C. M. Hoffman, J. E. H. Hopcroft, Tracing surface intersections. *Computer Aided Geometric Design*, **5** (1988) 285–307. ⇒303

[9] M. E. Mortenson, *Geometric Modeling.* Wiley, USA, 1 st ed. 1985. ⇒304

[10] N. M. Patrikalakis, Surface-to-surface intersections. *Computer Graphics and Applications, IEEE*, **13,** 1 (1993) 89–95. ⇒303

[11] E. Salamin, Application of quaternions to computation with rotations. *Technical Report, Stanford Univ*, 1979. ⇒305

[12] Tz. E. Stoyanov, Marching along surface/surface intersection curves with an adaptive step length. *Computer Aided Geometric Design*, **9** (1992) 485–489. ⇒ 304

[13] G. R. Veldkamp On the use of dual numbers, vectors, and matrices in instantaneous spatial kinematics. *Mech. Mach. Theory*, **11** (1976) 141–156. ⇒304

[14] M. Ventura, C. G. Soares, Surface intersection in geometric modeling of ship hulls. *Journal of Marine Science and Technology*, **17,** 1 (2012) 114–124. ⇒307

[15] S. T. Wu, L. Andrade, Marching along a regular surface/surface intersection with circular steps. *Computer Aided Geometric Design*, **16,** 4 (1999) 249–268. ⇒304, 307, 308, 309, 314, 316, 317, 319, 321, 322

[16] A. T. Yang, *Application of Quaternion Algebra and Dual numbers to the Analysis of Spatial Mechanisms.* PhD thesis, Columbia University, 1963. ⇒304

[17] X. Ye, T. Maekawa Differential geometry of intersection curves of two surfaces. *Computer Aided Geometric Design*, **16** (1999) 767–788. ⇒307

[18] J. Zara, L. Kavan, S. Collins, C. O'Sullivan, Skinning with dual quaternions. *In Proceedgings of ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 2007. ⇒306

Received: November 3, 2021 • Revised: November 13, 2021