Ray Tracing of Quadratic Parametric Surface

Shinji Ogaki
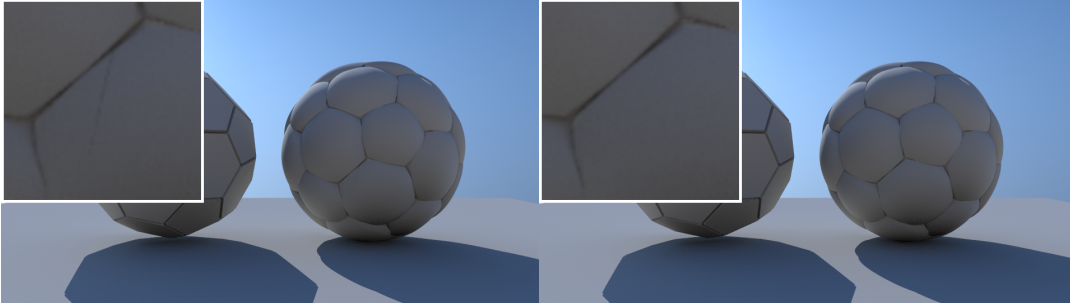
OLM Digital, Inc., Tokyo, Japan
and
JST CREST

FIGURE 1. Rendered images. The method of resultant (left) and method of pencil (right).

Over the past decades, vast research has been done on the ray-triangle intersect test but not much attention has been paid to the ray-quadratic parametric surface intersection test. In this paper we present two direct ray tracing methods for quadratic parametric surfaces and introduce a simple optimization technique for them.

1. INTRODUCTION

In the film industry there is an increasing demand for higher resolution images. The most common format has been Full HD(1920x1080pixels, 2K) but 4K monitors or even 8K monitors are now available on the market. Today's computer synthesized objects are mostly modeled with polygons and their silhouettes often appear non-smooth when rendered in very high resolution. To obtain high quality images, models must be highly tessellated.

Ray tracing is becoming increasingly popular for photo-realistic image creation as it is a natural way to simulate the behavior of photons. Although a significant amount of research has been done to accelerate ray tracing, handling a large number of polygons is still costly. Memory consumption is another serious problem because the number of polygons is normally quadrupled as resolution is doubled. Directly performing ray tracing for parametric surfaces helps to reduce a required memory amount.

However, we have to resort to numerical methods such as Bézier clipping [8] when the order of a parametric surface is high, we thus end up having heavy computational burden.

Our goal is to seek a sweet spot having a moderate amount of memory consumption and acceptable computation time. In what follows we focus on quadratic parametric surface since the intersection test is not so complicated as analytic solution exists.

2. Quadratic Parametric Surface

There are a variety of quadratic parametric surfaces including Steiner patch [3] and Phong tessellation [2]. In order to reduce memory consumption and achieve smooth rendering, we replace four triangle, one triangle and its adjacent three triangles, by a quadratic parametric surface.
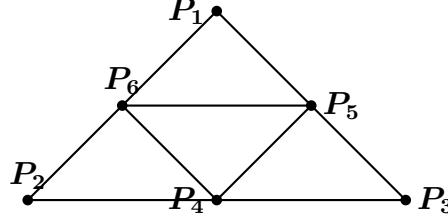

FIGURE 2. Four triangles defined by six points. Normally, these points are not on the same plane.

$$(1) \qquad Q(u, v) = \boldsymbol{A}u^2 + \boldsymbol{B}v^2 + \boldsymbol{C}w^2 + \boldsymbol{D}uv + \boldsymbol{E}uw + \boldsymbol{F}vw,$$

where $w = 1 - u - v$ and $0 \leq u, v, w \leq 1$. With the given six vertices $\boldsymbol{P_1}$, $\boldsymbol{P_2}$, $\boldsymbol{P_3}$, $\boldsymbol{P_4}$, $\boldsymbol{P_5}$, and $\boldsymbol{P_6}$ as in Figure 2, the coefficients are determined as $\boldsymbol{A} = \boldsymbol{P_1}$, $\boldsymbol{B} = \boldsymbol{P_2}$, $\boldsymbol{C} = \boldsymbol{P_3}$, $\boldsymbol{D} = 4\boldsymbol{P_6} - (\boldsymbol{P_1} + \boldsymbol{P_2})$, $\boldsymbol{E} = 4\boldsymbol{P_5} - (\boldsymbol{P_1} + \boldsymbol{P_3})$, and $\boldsymbol{F} = 4\boldsymbol{P_4} - (\boldsymbol{P_2} + \boldsymbol{P_3})$. This patch is C^0-continuous on edges.

3. Ray-Quadratic Parametric Surface Intersection Tests

There exist a number of methods for the ray-parametric surface intersection test including implicitization [4]. For higher order parametric surfaces we need numerical methods such as Newton's method and Beźier clipping [8].

Here we start with describing the algorithm developed by Kajiya [6] because this method gives elegant solutions especially for the ray-quadratic parametric surface intersection test (Sections 3.1 and 3.2).

In his method, each ray is represented as an intersection of two planes, say

$$(2) \qquad 0 = \boldsymbol{D_1} \cdot (x, y, z)^T + O_1$$
$$(3) \qquad 0 = \boldsymbol{D_2} \cdot (x, y, z)^T + O_2.$$

The intersection of a ray and quadratic parametric surface lies on both the planes. Substituting the right-hand side of Equation (1) for $(x, y, z)^T$ of Equations (2) and (3) gives two quadratic curves:

$$(4) \qquad 0 = F(u, v) = au^2 + bv^2 + c + duv + eu + fv$$
$$(5) \qquad 0 = G(u, v) = lu^2 + mv^2 + n + ouv + pu + qv,$$

where $a = (\boldsymbol{A} + \boldsymbol{C} - \boldsymbol{E}) \cdot \boldsymbol{D_1}$, $b = (\boldsymbol{B} + \boldsymbol{C} - \boldsymbol{F}) \cdot \boldsymbol{D_1}$, $c = \boldsymbol{C} \cdot \boldsymbol{D_1} + O_1$, $d = (\boldsymbol{D} - \boldsymbol{E} - \boldsymbol{F} + 2\boldsymbol{C}) \cdot \boldsymbol{D_1}$, $e = (\boldsymbol{E} - 2\boldsymbol{C}) \cdot \boldsymbol{D_1}$, $f = (\boldsymbol{F} - 2\boldsymbol{C}) \cdot \boldsymbol{D_1}$, $l = (\boldsymbol{A} + \boldsymbol{C} - \boldsymbol{E}) \cdot \boldsymbol{D_2}$, $m = (\boldsymbol{B} + \boldsymbol{C} - \boldsymbol{F}) \cdot \boldsymbol{D_2}$, $n = \boldsymbol{C} \cdot \boldsymbol{D_2} + O_2$, $o = (\boldsymbol{D} - \boldsymbol{E} - \boldsymbol{F} + 2\boldsymbol{C}) \cdot \boldsymbol{D_2}$, $p = (\boldsymbol{E} - 2\boldsymbol{C}) \cdot \boldsymbol{D_2}$, and $q = (\boldsymbol{F} - 2\boldsymbol{C}) \cdot \boldsymbol{D_2}$. Thus the intersection test for a bivariate quadratic parametric surface is rearranged into the intersection problem of two quadratic curves.

In the following subsections, we describe two interesting curve intersection finding algorithms.

3.1. **The Method of Resultant.** This method was introduced in [6]. Two quadratic curves have four intersections at a maximum and they are obtained analytically. We can find the intersections of the two planar curves $F(u, v)$ and $G(u, v)$ with the resultant by regarding them as quadratic curves of either u or v. If F and G have one or more common roots, the determinant of the Sylvester matrix is zero. If we treat the two curves as polynomials of u, we have

(6)
$$0 = a_4 u^4 + a_3 u^3 + a_2 u^2 + a_1 u + a_0,$$

where

$$
\begin{aligned}
a_4 \;=\; & abo^2 + a^2 m^2 + d^2 lm + b^2 l^2 - admo - bdlo - 2ablm \\
a_3 \;=\; & beo^2 + d^2 mp \\
& - admq - bdlq - bdop - afmo - demo - bflo \\
& + 2(aem^2 + b^2 lp + aboq + dflm - abmp - belm) \\
a_2 \;=\; & abq^2 + f^2 lm + bco^2 + d^2 mn + b^2 p^2 + e^2 m^2 \\
& - bfop - bdno - efmo - cdmo \\
& - bdpq - afmq - demq - bflq \\
& + 2(b^2 ln + acm^2 + beoq + dfmp - bemp - abmn - bclm) \\
a_1 \;=\; & beq^2 + f^2 mp \\
& - bfpq - bdnq - efmq - cdmq - bfno - cfmo \\
& + 2(cem^2 + b^2 np + bcoq + dfmn - bcmp - bemn) \\
a_0 \;=\; & bcq^2 + b^2 n^2 + f^2 mn + c^2 m^2 - bfnq - cfmq - 2bcmn.
\end{aligned}
$$

Solving this quartic equation, values for u are obtained. Substituting the values for u of Equation (4) or (5), we obtain values for v. Unfortunately, this method is not robust because of numerical error.

3.2. **The Method of Pencil.** A simpler and more robust method is to utilize a matrix pencil [7]. The linear matrix pencil M is a linear combination of two matrices defined as

(7)
$$
M = x \begin{pmatrix} a & d/2 & e/2 \\ d/2 & b & f/2 \\ e/2 & f/2 & c \end{pmatrix} + \begin{pmatrix} l & o/2 & p/2 \\ o/2 & m & q/2 \\ p/2 & q/2 & n \end{pmatrix}
$$

with $x \in \mathbb{R}$. The linear combination of F and G can then be written as

(8)
$$0 = P = xF(u, v) + G(u, v) = (u, v, 1)M(u, v, 1)^T.$$

Interestingly, P can be represented as a product of two lines if $0 = |M|$. For more details, see [5] for example. This is a special case of a hyperbola. By letting

(9)
$$L_1 \;=\; \alpha_1 u + \beta_1 v + \gamma_1$$

(10)
$$L_2 \;=\; \alpha_2 u + \beta_2 v + \gamma_2,$$

we have

(11)
$$(u, v, 1)M(u, v, 1)^T = L_1 L_2.$$

In this case the set of the intersections is decomposed as

$$
\begin{aligned}
\{(u, v)|0 = F\} \cap \{(u, v)|0 = G\} \;=\;\; & \{(u, v)|0 = F\} \cap \{(u, v)|0 = L_1\} \\
\cup\;\; & \{(u, v)|0 = F\} \cap \{(u, v)|0 = L_2\}
\end{aligned}
$$

since

$$\begin{aligned}
\{(u,v)|0=F\} \cap \{(u,v)|0=G\} &= \{(u,v)|0=F\} \cap \{(u,v)|0=F+G\} \\
&= \{(u,v)|0=F\} \cap \{(u,v)|0=xF+G\} \\
&= \{(u,v)|0=F\} \cap \{(u,v)|0=P\} \\
&= \{(u,v)|0=F\} \cap \{(u,v)|0=L_1L_2\}.
\end{aligned}$$

Thus, the problem can be simplified by finding the value of x such that $0=|M|$. The values of x are given by solving the following cubic equation

(12)
$$0 = a_3 x^3 + a_2 x^2 + a_1 x + a_0,$$

where

$$\begin{aligned}
a_3 &= abc + (def - af^2 - be^2 - cd^2)/4 \\
a_2 &= abn + amc + lbc - (afq + bep + cdo)/2 \\
&+ (oef + deq + dpf - lf^2 - me^2 - nd^2)/4 \\
a_1 &= amn + lbn + lmc - (lfq + mep + ndo)/2 \\
&+ (dpq + oeq + opf - aq^2 - bp^2 - co^2)/4 \\
\text{(13)} \quad a_0 &= lmn + (opq - lq^2 - mp^2 - no^2)/4.
\end{aligned}$$

We only need to obtain one value of x, which significantly reduces the cost of root finding. The factorization is done by comparing the coefficients of u^2, v^2, uv, u, v, and 1. Dividing the both sides of Equation (11) by M_{11} or M_{22}, whichever has a greater absolute value, makes the calculation easier and more robust. In the case of $M_{11} > M_{22}$, the two lines L_1 and L_2 are obtained as

$$L_1 = u + \left(M'_{12} + \sqrt{M'^2_{12} - M'_{22}}\right)v + \left(M'_{13} \pm \sqrt{M'^2_{13} - M'_{33}}\right)$$

$$L_2 = u + \left(M'_{12} - \sqrt{M'^2_{12} - M'_{22}}\right)v + \left(M'_{13} \mp \sqrt{M'^2_{13} - M'_{33}}\right),$$

where $M'_{ij} = M_{ij}/M_{11}$. The coefficients γ_1 and γ_2 are chosen so that $M'_{23} = \beta_1\gamma_2 + \beta_2\gamma_1$. Similary, in the case of $M_{22} > M_{11}$, L_1 and L_2 are obtained as

$$L_1 = \left(M'_{12} + \sqrt{M'^2_{12} - M'_{11}}\right)u + v + \left(M'_{23} \pm \sqrt{M'^2_{23} - M'_{33}}\right)$$

$$L_2 = \left(M'_{12} - \sqrt{M'^2_{12} - M'_{11}}\right)u + v + \left(M'_{23} \mp \sqrt{M'^2_{23} - M'_{33}}\right),$$

where $M'_{ij} = M'_{ij}/M_{22}$. The coefficients γ_1 and γ_2 are chosen so that $M'_{13} = \alpha_1\gamma_2 + \alpha_2\gamma_1$. Substituting Equations (9) and (10) into Equation (4) or (5) gives two quadratic equations. By solving them, we obtain values for u and v. These values give the actual intersection points. The parameters u, v, and $w = 1 - (u+v)$ must lie between 0 and 1, and intersection points must be in the viewing direction. Note also that the intersection test can be immediately terminated if $0 < M_{11}M_{22} - M_{12}M_{21}$ since $0 = (u,v,1)M(u,v,1)^T$ becomes an ellipsoid whose area is zero.

4. Optimization Technique

In the previous section two intersection tests were introduced. What we have over-looked is the way of choosing the two planes (2) and (3). Here we show that the computation can be simplified if we choose them cleverly. For example, if $\boldsymbol{D_1}$ and $\boldsymbol{D_2}$ satisfy $a = 0$ and $m = 0$, the coefficients (13) become

$$
\begin{aligned}
a_3 &= (def - be^2 - cd^2)/4 \\
a_2 &= lbc - (bep + cdo)/2 + (oef + deq + dpf - lf^2 - nd^2)/4 \\
a_1 &= lbn - (lfq + ndo)/2 + (dpq + oeq + opf - bp^2 - co^2)/4 \\
a_0 &= (opq - lq^2 - no^2)/4,
\end{aligned}
$$

which dramatically reduces the number of multiplications.

5. Result

We applied Phong tessellation to a soccer ball model and rendered it with the two different algorithms as in Figure 1. All calculations were done with single precision. A little artifact can be seen in the image rendered with the method of resultant. On the other hand, the method of pencil gives an ideal result.
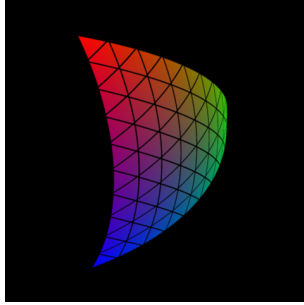

Figure 3. A quadratic parametric surface.

Resultant	Pencil	Pencil with Optimization Technique
1.00sec	0.84sec	0.80sec

Table 1. Comparison of three methods.

We also rendered a simple quadratic parametric surface (Figure 3) with three meth-ods for performance comparison. The method of pencil with the optimization technique is 20% faster than the method of resultant (Table 1).

6. Conclusion and Future Work

In this paper we described two ray-quadratic parametric surface intersection tests. The method of pencil has a couple of advantages: (1) we can avoid to solve a quartic equation hence more robust results are obtained and (2) early termination of com-putation is possible. We also showed that two well-chosen planes reduce the number of multiply operations. However, the intersection test still remains computationally expensive. We would like to explore a way to further improve the performance of the intersection test. Another interesting research avenue is to extend the pencil method for higher order parametric surfaces.

7. Acknowledgements

We would like to thank Ken Anjyo and Sampei Hirose for their valuable comments.

References

[1] Bajaj C. L., Chen J., Xu G.: Modeling with cubic a-patches. *ACM Trans. Graph. 14* (April 1995), pp. 103–133.

[2] Boubekeur T., Alexa M.: Phong tessellation. In *SIGGRAPH Asia '08: ACM SIGGRAPH Asia 2008 papers* (New York, NY, USA, 2008), ACM, pp. 1–5.

[3] Breen D. E.: Creation and smooth-shading of steiner patch tessellations. In *Proceedings of 1986 ACM Fall joint computer conference* (Los Alamitos, CA, USA, 1986), ACM '86, IEEE Computer Society Press, pp. 931–940.

[4] Hanrahan P.: *Ray tracing algebraic surfaces*. In *Proceedings of the 10th annual conference on Computer graphics and interactive techniques* (Detroit, Michigan, USA, 1983), SIGGRAPH '83, ACM, pp. 83–90.

[5] Hosaka M.: *Modeling of curves and surfaces in CAD/CAM*. Springer-Verlag New York, Inc., (New York, NY, USA, 1992).

[6] Kajiya J. T.: Ray tracing parametric patches. In *Proceedings of the 9th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1982), SIGGRAPH '82, ACM, pp. 245–254.

[7] Ogaki S. and Tokuyoshi Y.: Direct Ray Tracing of Phong Tessellation. In *EGSR'11 Proceedings of the Twenty-second Eurographics conference on Rendering*, pp. 1337-1344, 2011.

[8] Sederberg T. W., Nishita T.: Curve intersection using bézier clipping. *Computer-Aided Design 22*, 9 (1990), pp. 538–549.

[9] Vlachos A., Peters J., Boyd C., Mitchell J. L.: Curved pn triangles. In *SI3D '01: Proceedings of the 2001 symposium on Interactive 3D graphics* (New York, NY, USA, 2001), ACM Press, pp. 159–166.