Parametric Circles and Spheres

A. Ardeshir Goshtasby

Computer Science and Engineering Department
Wright State University

Formulations for parametric circles and spheres in terms of rational Gaussian (RaG) curves and surfaces are introduced. With the proposed formulations, a full circle is generated by interpolating a closed RaG curve to the vertices of an equilateral triangle, and a full sphere is generated by interpolating a closed RaG surface to the vertices of an octahedron with equilateral triangular faces. Generation of circles and spheres in this manner is very intuitive and easy to remember as the weights are all 1 and the nodes are all unique and uniformly spaced.

1 Introduction

Circles and spheres are very important primitives in geometric modeling. Although easy and intuitive methods for creating circles and spheres in implicit form [16] and by subdivision methods [11] exist, determination of efficient and intuitive methods that can generate circles and spheres in parametric form is still an area of research. Both approximation and exact methods have been proposed. Approximation methods use B-spline or Bézier formulations, while exact methods use non-uniform rational B-spline (NURBS) or rational Bézier formulations.

Current parametric representations for circles and spheres are not very intuitive. Although a full circle can be generated by a single NURBS curve [22], the knot vectors and the weights to be remembered are quite extensive. A parametric representation that can generate a full sphere from a single NURBS or rational Bézier surface is yet to be determined. Presently, a NURBS or a rational Bézier surface can create only a segment of a sphere.

In this paper, first existing methods for generating circles and spheres are reviewed and then new formulations for creating circles and spheres with rational Gaussian (RaG) curves and surfaces are presented. With the proposed formulations, a full circle or sphere is generated with a single RaG curve or surface. Moreover, the proposed methods are intuitive and are very easy to remember and use. A circle is obtained when a closed RaG curve is interpolated to the vertices of an equilateral triangle with uniformly spaced nodes and equal

weights, and a sphere is obtained when a closed RaG surface is interpolated to the vertices of an octahedron with equilateral triangular faces, again with uniformly spaced nodes and equal weights.

In this paper, the discussions focus on a circle of radius 1 centered at the origin and a sphere of radius 1 centered at the origin. An arbitrary circle or sphere can be obtained by appropriately scaling and translating the unit circle or sphere.

2 Past Work

2.1 Circles

Formulas for approximately and exactly generating circles have been proposed. Faux and Pratt [5] approximated a circular segment of exactly 90° by a cubic Bézier curve achieving an accuracy of 2.7E-4. Dokken *et al.* [3] approximated 90° and smaller circular segments by cubic Bézier curves, achieving an accuracy of 1.4E-4 and better, with smaller segments producing better accuracies. For example, an accuracy of 5.2E-10 was achieved when 32 Bézier curves were used to create a circle. Goldapp [7] approximated a quarter of a circle by a cubic Bézier curve with various boundary conditions, achieving an accuracy of 5.5E-5. Mørken [15] used quadratic Bézier curves to generate circular segments, creating a whole circle out of 4 quadratic Bézier segments and achieving an accuracy of 2.9E-2. Using smaller circular segments, he was able to approximate a circle with 64 quadratic Bézier curves and achieve an accuracy of 3.6E-7. Blinn [1] approximated a 90° circular segment by a polynomial of degree 3 achieving an accuracy of 4.2E-4.

To generate exact circles, rational Bézier and NURBS curves are used. In Geomed [13], a 90° circular arc was defined by rational polynomials $x(t) = (1-t^2)/(1+t^2)$ and $y(t) = 2t/(1+t^2)$ with $0 \leq t \leq 1$. This is equivalent to a quadratic NURBS curve with homogeneous coordinates $(1,0,1)$, $(1,1,1)$, $(0,2,2)$ and knot vector $\{0,0,0,1,1,1\}$ [23]. A full circle was defined using the vertices as well as the edge midpoints of the square circumscribing the circle as the control points of a quadratic NURBS curve, starting with an edge midpoint and arriving at the same edge midpoint. The knot vector was $\{0,0,0,1,1,2,2,3,3,4,4,4\}$ and the weights were $\{1,1,2,1,1,1,2,1,1\}$ [23].

Böhm *et al.* [2] created a circle from 3 circular arcs, each defined by a quadratic NURBS curve. The control points of the NURBS curve were the vertices and the midpoints of the equilateral triangle circumscribing the circle, starting from an edge midpoint and arriving at the same edge midpoint. The knot vector for the curve was $\{0,0,0,\frac{1}{3},\frac{1}{3},\frac{2}{3},\frac{2}{3},1,1,1\}$ and the weights were $\{1,\frac{1}{2},1,\frac{1}{2},1,\frac{1}{2},1\}$.

Given the vertices of an equilateral triangle, the circular segment passing through two of

its vertices and tangent to the edges that meet at the third vertex was obtained by a rational quadratic Bézier curve when letting the vertices of the triangle represent the control points and the weights be $\{1, \frac{1}{2}, 1\}$ [21]. A complementary circular arc was obtained from the same control points but with weights $\{1, -\frac{1}{2}, 1\}$ and knot vector $\{0, 0, 0, 1, 1, 1\}$. In this manner, a full circle was defined from two complementary circular segments using the vertices of an equilateral triangle. Lee [14] described circular arcs of any subtending angle between 0 and π by a rational quadratic Bézier curve, enabling design of a full circle from two or more rational Bézier curves.

Piegl and Tiller [20] described a full circle with a 7-control-point quadratic NURBS curve. The control points were obtained from the vertices of the circumscribing square and midpoints of two of its opposing edges, starting from an edge midpoint and returning to the same edge midpoint. The knot vector was $\{0, 0, 0, \frac{1}{4}, \frac{1}{2}, \frac{1}{2}, \frac{3}{4}, 1, 1, 1\}$ and the weights were $\{1, \frac{1}{2}, \frac{1}{2}, 1, \frac{1}{2}, \frac{1}{2}, 1\}$. They obtained a circle from 6 control points also through degree elevation of a Bézier curve with control points $(1, 0), (1, 2), (-1, 2), (-1, -2), (1, -2), (1, 0)$, weights $\{9, 1, \frac{1}{3}, \frac{1}{3}, 1, 9\}$ and knots $\{0, 0, 0, 0, \frac{1}{2}, \frac{1}{2}, 1, 1, 1, 1\}$.

2.2 Spheres

Formulas to approximate spheres as well as to create them exactly have been proposed. Herron [10] approximated a sphere with 20 triangular patches using the vertices of a 20-sided polyhedron, providing C^1 continuity. Eisele [4] used a quadratic Bézier patch to approximate a rectangular segment of a sphere that was symmetric with respect to xz and yz planes, producing an accuracy of 2.0E-4.

Exact spheres have been created using quadratic NURBS and quadratic rational Bézier surfaces. Piegl [17] described a spherical cap using a quadratic rational Bézier patch from 9 control points that formed the vertices of a pyramid with a square base. One of the control points represented the tip of the pyramid and the remaining 8 control points represented the vertices and the edge midpoints of the square base. The base of the spherical cap, which was a circle, was tangent to the edges of the square base. By negating the weight of the control point at the tip of the pyramid, a hemisphere was obtained whose circular base was the same as the circular base of the spherical cap obtained from the pyramid. The ability to create a hemisphere from such a pyramid enabled the creation of a full sphere from two opposing pyramids with a common base [19].

Farin *et al.* [6] described an octant of a sphere using a network of quadratic triangular Bézier patches from 16 control points. Piegl [18] constructed an octant of a sphere as a special case of an octant of a torus by a rational quadratic Bézier surface, using the vertices of the cube that contained the octant of the sphere as the control points of the surface.

Although full circles have been represented by quadratic NURBS and rational Bézier curves [20], representation of a full sphere by a single parametric surface is yet to be determined. In this paper, we will show how to create a full circle from a single RaG curve and how to create a full sphere from a single RaG surface.

3 RaG Circles

Given a set of ordered points $\{\mathbf{V}_i : i = 1, \ldots, N\}$, the RaG curve that approximates the points is given by [8, 9]

$$\mathbf{P}(u) = \sum_{i=1}^{N} W_i \mathbf{V}_i g_i(u) \tag{1}$$

where W_i is the weight of point $\mathbf{V}_i$, and $g_i(u)$ is the ith basis function of the curve defined by

$$g_i(u) = \frac{G_i(u)}{\sum_{j=1}^{N} G_j(u)} \tag{2}$$

and $G_i(u)$ is a Gaussian of height 1 centered at u_i (the node associated with $\mathbf{V}_i$):

$$G_i(u) = \exp\{-\frac{(u-u_i)^2}{2\sigma^2}\}. \tag{3}$$

σ is the standard deviation of the Gaussians. By increasing it, a smoother curve will be obtained, and by decreasing it, a curve that more closely follows the points will be obtained. To achieve the same effect when using NURBS or rational Bézier curves, the degree of the B-spline or Bézier basis functions should be varied.

Formulas (1)–(3) show an open curve. To obtain a closed curve, formula (3) should be replaced with

$$G_i(u) = \sum_{k=-\infty}^{\infty} \exp\{-\frac{(u-u_i+k)^2}{2\sigma^2}\}. \tag{4}$$

This is because a Gaussian extends to infinity on both sides, and when a curve is closed, it makes infinite cycles around the curve. Because a Gaussian approaches 0 exponentially, in practice k is a small number. For instance, when $\sigma = 1.0$, $(u - u_i) = 1.0$, and our computational accuracy is limited to 15 decimal places, we obtain $\exp\{-\frac{(u-u_i+k)^2}{2\sigma^2}\} = 10^{-15}$, from which we find $k = 9$.

A RaG curve approximates the given set of points. If it is desired to interpolate the points, we have to let $\mathbf{P}(u_i) = \mathbf{V}_i$ for $i = 1, \ldots, N$ and determine the new control points of the curve, $\{\mathbf{v}_i : i = 1, \ldots, N\}$, by solving the following system of linear equations:

$$\mathbf{V}_i = \sum_{j=1}^{N} W_j \mathbf{v}_j g_j(u_i), \qquad i = 1, \ldots, N. \tag{5}$$

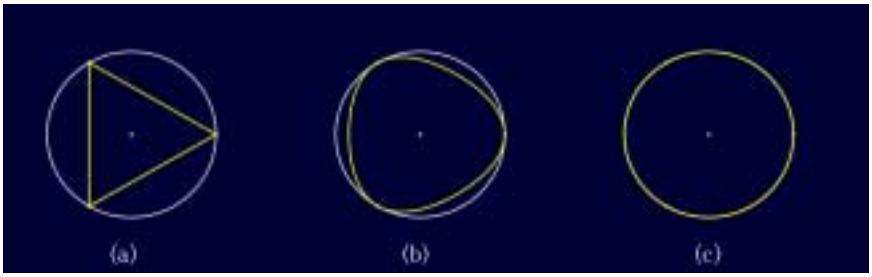

Figure 1: A closed RaG curve interpolating the vertices of an equilateral triangle. (a) $\sigma = 0.01$. (b) $\sigma = 0.2$. (c) $\sigma = 0.7$.

Table 1: Maximum radial error of 100 uniformly spaced samples from a closed RaG curve interpolating the vertices of an equilateral triangle as a function of parameter σ.

σ	0.1	0.2	0.3	0.4	0.5	0.6
E_1	3.0E-1	1.3E-1	7.3E-3	9.1E-5	2.6E-7	1.3E-9
σ	0.7	0.8	0.9	1.0	1.1	1.2
E_1	6.3E-13	1.3E-10	4.3E-9	3.6E-7	3.2E-6	1.6E-4

When the control points represent the vertices of an equilateral triangle, the closed RaG curve interpolating the vertices will fall between the triangle and the circle that interpolates the triangle. As the standard deviation of Gaussians is increased, the curve becomes smoother and approaches the circle. As the standard deviation of Gaussians is decreased, the curve gets closer to the triangle (see Fig. 1). The error between the RaG curve and the circle as a function of the standard deviation of Gaussians is shown in Table 1. These errors represent the maximum of $E_1(u) = |\sqrt{X(u)^2 + Y(u)^2} - 1|$ when varying u from 0 to 1 with increments of 0.01. $X(u)$ and $Y(u)$ are the components of $\mathbf{P}(u)$.

It is interesting to note that this error initially decreases as the standard deviation of Gaussians is increased. Then it increases as the standard deviation of Gaussians is increased further. This can be explained from the signal processing point of view by treating a curve like a signal. If we draw $X(u)$ or $Y(u)$, we will see a sinusoidal curve with period 1. When σ is close to 0, the curve approaches a polyline where the slopes of its segments are either 1 or -1. As σ is increased, the curve gets smoother. At a particular value of σ the curve becomes most similar to a sinusoid. By increasing σ further, the curve becomes smoother than the sinusoid, and that increases the error between the curve and the sinusoid again. As σ is further increased, we will reach a point where it is no longer possible to fit a curve with the required smoothness to the points, and the system of equations to be solved becomes singular.

Table 2: Maximum tangential accuracy of 100 points at uniformly spaced parameters on a unit circle as a function of parameter σ.

σ	0.1	0.2	0.3	0.4	0.5	0.6
E_2	4.9E-1	1.2E-1	2.8E-3	1.8E-4	8.6E-7	7.5E-10
σ	0.7	0.8	0.9	1.0	1.1	1.2
E_2	2.4E-12	6.0E-11	8.9E-10	3.7E-7	3.0E-5	1.5E-4

It is not currently known whether there is a smoothness parameter where $X(u)$ and $Y(u)$ exactly reproduce sinusoids. It was not possible to experimentally verify this because of inherent inaccuracies in exp(), sin(), cos(), and sqrt() used in the computations.

It is sometimes required to generate a circular arc rather than a full circle. To determine whether a RaG curve can generate a desired circular arc with a reasonably good accuracy, the following error measure was computed:

$$E_2(u) = \sqrt{E_x(u)^2 + E_y(u)^2}, \tag{6}$$

where

$$E_x(u) = X(u) - \cos(2\pi u), \tag{7}$$

$$E_y(u) = Y(u) - \sin(2\pi u). \tag{8}$$

$E_2(u)$ was computed for $0 \leq u \leq 1$ with increments of 0.01 and its maximum was determined. This was repeated for different values of σ and the obtained results were entered into Table 2. This tangential accuracy is at the same level as the radial accuracy, shown in Table 1. Therefore, circular arcs of arbitrary subtending angles can be reproduced with an accuracy of 2.4E-12 using regular computational tools available on PCs. It should be mentioned that the errors reported in Tables 1 and 2 contain the errors inherent in functions exp(), sqrt(), sin(), and cos() in the Microsoft Visual C programming library that we used. Actual errors may be considerably smaller.

Since the quality of an approximated circle is often judged by its curvature, the curvature accuracy of RaG curves creating unit circles is investigated. The curvature of a parametric curve in the plane is given by [16],

$$K = \frac{x^u y^{uu} - x^{uu} y^u}{[(x^u)^2 + (y^u)^2]^{3/2}}, \tag{9}$$

where

$$x^u = \sum_{i=1}^{N} W_i V_{xi} g_i^u(u), \tag{10}$$

$$y^u = \sum_{i=1}^{N} W_i V_{y_i} g_i^u(u), \tag{11}$$

$$x^{uu} = \sum_{i=1}^{N} W_i V_{xi} g_i^{uu}(u), \tag{12}$$

$$y^{uu} = \sum_{i=1}^{N} W_i V_{y_i} g_i^{uu}(u). \tag{13}$$

V_{xi} and V_{y_i} are the x and y components of $\mathbf{V}_i$, and

$$g_i^u(u) = \frac{G_i^u(u)H(u) - G_i(u)H^u(u)}{[H(u)]^2}, \tag{14}$$

$$g_i^{uu}(u) = \frac{I^u(u)J(u) - I(u)J^u(u)}{[J(u)]^2}, \tag{15}$$

where $I(u) = G_i^u(u)H(u) - G_i(u)H^u(u)$; $J(u) = [H(u)]^2$; $H(u) = \sum_{i=1}^{N} G_i(u)$; $H^u(u) = \sum_{i=1}^{N} G_i^u(u)$; $H^{uu}(u) = \sum_{i=1}^{N} G_i^{uu}(u)$; $I^u(u) = G_i^{uu}(u)H(u) - G_i(u)H^{uu}(u)$; $J^u(u) = 2H^u(u)H(u)$; and finally

$$G_i(u) = \sum_{k=-l}^{l} \exp\{(u - u_i + k)^2/2\sigma^2\}, \tag{16}$$

$$G_i^u(u) = \sum_{k=-l}^{l} \frac{1}{\sigma^2}(u - u_i + k)\exp\{(u - u_i + k)^2/2\sigma^2\}, \tag{17}$$

$$G_i^{uu}(u) = \frac{1}{\sigma^2}G_i(u) + \sum_{k=-l}^{l} \frac{1}{\sigma^4}(u - u_i + k)^2\exp\{(u - u_i + k)^2/2\sigma^2\}. \tag{18}$$

l is a small number between 1 and 10, representing the required accuracy in computation as discussed above.

Since the curvature along a perfect circle of radius 1 is 1, to determine the error in curvature K of a RaG curve approximating a unit circle, we compute

$$E_3 = |1 - K|. \tag{19}$$

Table 3 shows the maximum curvature error obtained according to equation (19) by varying parameter u from 0 to 1 with increments of 0.01 and for different values of σ. Comparing Tables 1, 2, and 3, we see that curvature error E_3 exhibits properties similar to those of radial and tangential errors computed by E_1 and E_2. More detailed information about change in curvature error as a function of change in u is shown in Fig. 2. Curvature plots for $\sigma = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6$, and 0.7 are shown. The thickness of the plotted curves is inversely proportional to the value of σ. For $\sigma > 0.4$ the curvature plots are very close to 1.0 and cannot be distinguished from each other. Curvature magnitudes larger than 2

Table 3: Maximum curvature error of 100 points at uniformly spaced parameters on a unit circle as a function of parameter σ.

σ	0.1	0.2	0.3	0.4	0.5	0.6
E_3	1.1E+2	1.1E+0	4.4E-2	6.7E-6	2.2E-6	1.8E-11
σ	0.7	0.8	0.9	1.0	1.1	1.2
E_3	3.8E-12	5.9E-11	1.1E-9	2.6E-8	5.9E-6	3.5E-4

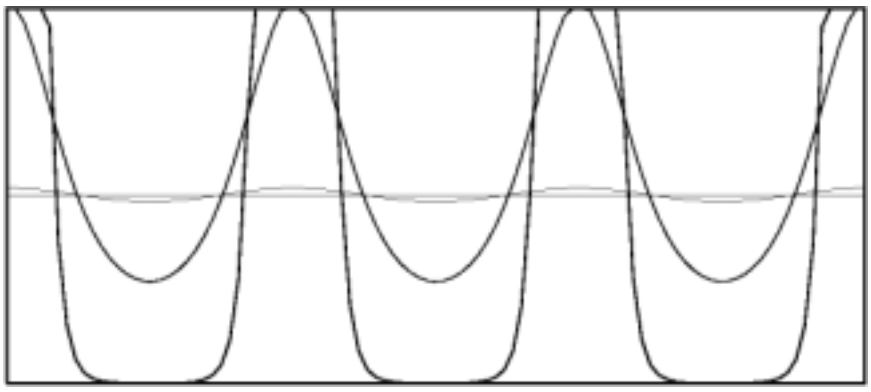

Figure 2: Curvature plots of E_3 as a function of σ. The thickness of the curves is inversely proportional to the value of σ, which was $0.1, 0.2, 0.3, 0.4, 0.5, 0.6$, and 0.7. Curvature magnitudes larger than 2.0 are cut off at 2.0. The center line shows curvature magnitude 1.

are cut off at 2 in this figure. The centerline in the figure shows curvature magnitude 1. It is interesting to note that the curvature error has a sinusoidal characteristic. The error amplitude decreases as σ approaches 0.7 but the error period remains equal to 1/3 under all values of σ.

4 RaG Spheres

Given a set of scattered points in 3-D $\{\mathbf{V}_i : i = 1, \ldots, N\}$ with associated nodes $\{(u_i, v_i) : i = 1, \ldots, N\}$, the RaG surface approximating the points is given by [8, 9]

$$\mathbf{P}(u, v) = \sum_{i=1}^{N} W_i \mathbf{V}_i g_i(u, v), \tag{20}$$

where W_i is the weight of $\mathbf{V}_i$ and $g_i(u, v)$ is the ith basis function of the surface defined by

$$g_i(u, v) = \frac{G_i(u, v)}{\sum_{j=1}^{N} G_j(u, v)}, \qquad i = 1, \ldots, N, \tag{21}$$

and $G_i(u,v)$ is a 2-D Gaussian of height 1 centered at (u_i, v_i):

$$G_i(u,v) = \exp\{-\frac{(u-u_i)^2 + (v-v_i)^2}{2\sigma^2}\}. \tag{22}$$

Note that the formula for a RaG surface as shown in formula (20) contains a single sum as opposed to a double sum used in NURBS and rational Bézier surfaces. This implies that in order to obtain a RaG surface, a regular grid of control points is not necessary and a scattered set of points is sufficient.

The standard deviation of Gaussians determines the smoothness of the obtained surface. By increasing σ, a smoother surface is obtained, and by decreasing σ, a surface that more closely follows the points is obtained.

Formulas (20)–(22) describe an open surface. To obtain a half-closed surface, like a cylinder that is closed along u but open in v direction, formula (22) should be replaced with

$$G_i(u,v) = \sum_{k=-\infty}^{\infty} \exp\{-\frac{(u-u_i+k)^2 + (v-v_i)^2}{2\sigma^2}\}. \tag{23}$$

This is because a 2-D Gaussian extends to infinity in all directions and along the closed side it makes infinite loops. In practice, the infinity can be replaced with a small number depending on the accuracy of the computation as discussed in RaG curves. To represent a surface that is closed in both u and v directions, like a torus, formula (22) should be replaced with

$$G_i(u,v) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \exp\{-\frac{(u-u_i+k)^2 + (v-v_i+l)^2}{2\sigma^2}\}. \tag{24}$$

Again, in practice, k and l are small numbers between 1 and 10.

The RaG surface in formula (20) approximates a given set of points. For the surface to interpolate the points, we should let $\mathbf{P}(u_i, v_i) = \mathbf{V}_i$ and determine the new control points of the surface $\{\mathbf{v}_i : i = 1, \ldots, N\}$ by solving the following system of linear equations:

$$\mathbf{V}_i = \sum_{j=1}^{N} W_j \mathbf{v}_j g_j(u_i, v_i), \qquad i = 1, \ldots, N. \tag{25}$$

A cylinder is obtained when a half-closed RaG surface interpolates six points that represent the vertices of two parallel equilateral triangles as shown in Fig. 3. The surface produces a circular cross-section for a fixed value of v and a line for a fixed value of u. To obtain a cylinder whose axis is normal to its base, the triangles should be equal and parallel to each other. In addition, they should be normal to the lines connecting corresponding triangle vertices. By changing the standard deviation of Gaussians, the surface will change shape between the cylinder and the polyhedron obtained by connecting corresponding vertices of the triangles. A cylinder is obtained from the control points in Table 4 when σ is 0.5. Table

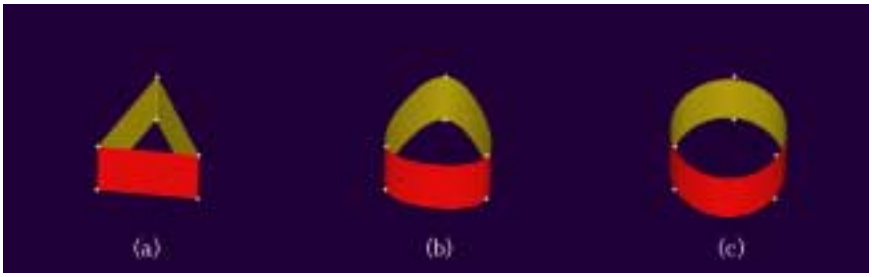

Figure 3: The half-closed RaG surface interpolating the vertices of two parallel equilateral triangles falls between the polyhedron obtained by connecting the triangle vertices and the cylinder that interpolates the polyhedron. (a) $\sigma = 0.01$. (b) $\sigma = 0.2$. (c) $\sigma = 0.5$.

Table 4: The control points and nodes of the half-closed RaG surfaces shown in Fig. 3.

i	X_i	Y_i	Z_i	u_i	v_i
1	1.0	0.0	0.0	0.0	0.0
2	-0.5	$\frac{\sqrt{3}}{2}$	0.0	1/3	0.0
3	-0.5	$-\frac{\sqrt{3}}{2}$	0.0	2/3	0.0
4	1.0	0.0	1.0	0.0	1.0
5	-0.5	$\frac{\sqrt{3}}{2}$	1.0	1/3	1.0
6	-0.5	$-\frac{\sqrt{3}}{2}$	1.0	2/3	1.0

4 shows the control points and the nodes of the surface generating the cylinder shown in Fig. 3. All weights are equal to 1.

A torus is obtained from a RaG surface interpolating 9 points that represent the vertices of three equilateral triangles centered at the vertices of a larger equilateral triangle as shown in Fig. 4. The three triangles ensure that the cross-sections of the surface along parameter u are circles, while corresponding triangle vertices forming an equilateral triangle ensure that cross-sections of the surface along parameter v are circles also, thus creating a torus. By increasing σ, the surface will change shape from the polyhedron that is formed by connecting corresponding vertices of the triangles to the torus that interpolates the polyhedron (see Fig. 4). Table 5 shows the control points and the nodes of the surfaces depicted in Fig. 4. All weights are equal to 1.

When the equilateral triangles defining the torus approach each other and their centers coincide with the center of the larger triangle, a sphere is obtained. Assuming parameter u traces circular cross-sections normal to the axis of the torus and parameter v traces circular cross-sections parallel to the axis of the torus, when the torus transforms to a sphere, the sphere will actually be drawn twice when varying parameters u and v from 0 to 1. In order

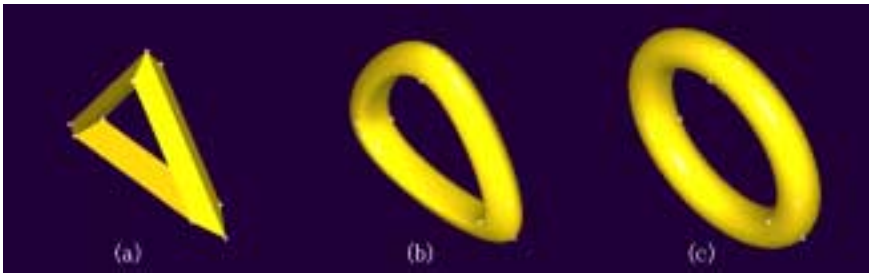

Figure 4: The closed RaG surface interpolating the vertices of three equal equilateral triangles that are centered at the vertices of a larger equilateral triangle falls between the polyhedron that is formed by connecting corresponding triangle vertices and the torus that interpolates the polyhedron vertices. (a) $\sigma = 0.01$. (b) $\sigma = 0.18$. (c) $\sigma = 0.35$.

Table 5: The control points and nodes of the closed RaG surfaces shown in Fig. 4.

i	X_i	Y_i	Z_i	u_i	v_i
1	0.75	0.0	0.0	0.0	0.0
2	0.375	0.0	0.2166	1/3	0.0
3	0.375	0.0	-0.2166	2/3	0.0
4	-0.375	0.6496	0.0	0.0	1/3
5	-0.1875	0.3248	0.2166	1/3	1/3
6	-0.1875	0.3248	-0.2165	2/3	1/3
7	-0.375	-0.6496	0.0	0.0	2/3
8	-0.1875	-0.3248	0.2166	1/3	2/3
9	-0.1875	-0.3248	-0.2165	2/3	2/3

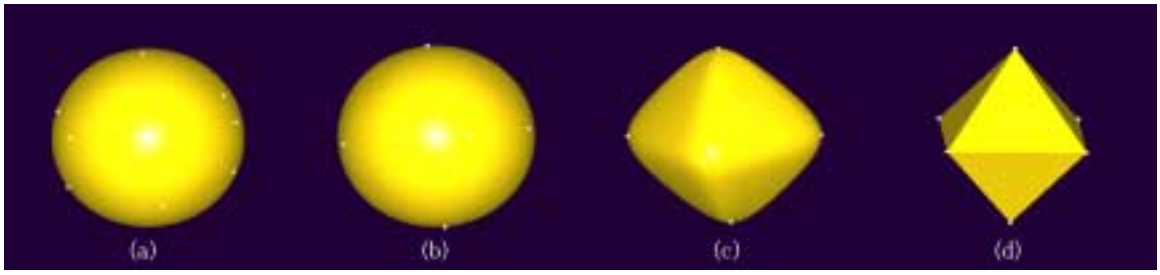

Figure 5: (a) A sphere obtained from a closed RaG surface interpolating the nine points shown in Table 6 with $\sigma = 0.5$. (b)–(d) A RaG surface interpolating the sixteen points shown in Table 7 with σ equal to 0.35, 0.12, and 0.01, respectively.

Table 6: The control points and nodes of the closed RaG surface shown in Fig. 5a.

i	X_i	Y_i	Z_i	u_i	v_i
1	0.5	0.0	0.0	0.0	0.0
2	-0.25	0.0	0.4330	0.0	1/3
3	-0.25	0.0	-0.4330	0.0	2/3
4	-0.25	0.4330	0.0	1/3	0.0
5	0.125	-0.2166	0.4330	1/3	1/3
6	0.125	-0.2166	-0.4330	1/3	2/3
7	-0.25	-0.4330	0.0	2/3	0.0
8	0.125	0.2166	0.4330	2/3	1/3
9	0.125	0.2166	-0.4330	2/3	2/3

to draw the sphere only once, parameter v should be varied from 0 to 1, but parameter u should be varied only from 0.25 to 0.75. Figure 5a shows a sphere obtained in this manner. The control points and the nodes of the sphere are shown in Table 6.

A more intuitive sphere is obtained by starting with four squares that are centered at the vertices of a larger square. As the smaller squares approach each other and their centers coincide with the center of the larger square, the torus will transform to a sphere. The sphere obtained in this manner is shown in Fig. 5b. The control points and nodes of this sphere are shown in Table 7. Note that the sphere that interpolates the vertices of an octahedron uses the top and bottom vertices four times, each with a unique node, while using the vertices along the hemisphere two times, again, each with a unique node. Overall, parameter spacing between the nodes is uniform and all weights are equal to 1. When the standard deviation of Gaussians is very small, the surface resembles an octahedron. As the standard deviation of Gaussians is increased, the surface changes shape to a sphere as shown in Figs. 5b–c.

Next, we will show the accuracy of the closed RaG surface representing a unit sphere

Table 7: The control points and nodes of the RaG surfaces shown in Figs. 5b–d.

i	X_i	Y_i	Z_i	u_i	v_i
1	0.5	0.0	0.0	0.0	0.0
2	0.0	0.0	0.5	0.0	0.25
3	-0.5	0.0	0.0	0.0	0.5
4	0.0	0.0	-0.5	0.0	0.75
5	0.0	0.5	0.0	0.25	0.0
6	0.0	0.0	0.5	0.25	0.25
7	0.0	-0.5	0.0	0.25	0.5
8	0.0	0.0	-0.5	0.25	0.75
9	-0.5	0.0	0.0	0.5	0.0
10	0.0	0.0	0.5	0.5	0.25
11	0.5	0.0	0.0	0.5	0.5
12	0.0	0.0	-0.5	0.5	0.75
13	0.0	-0.5	0.0	0.75	0.0
14	0.0	0.0	0.5	0.75	0.25
15	0.0	0.5	0.0	0.75	0.5
16	0.0	0.0	-0.5	0.75	0.75

with the 16 control points shown in Table 7 as a function of σ. The approximation errors are shown in Table 8. E_1 shows the maximum radial error between the exact unit sphere and its approximation by a RaG surface with 16 control points, and E_2 shows the maximum tangential error between the unit sphere and a RaG surface for different values of u and v. To measure the errors, points from the sphere and the RaG surface were sampled uniformly at a grid of 30×30 and distances between corresponding sample points were determined.

The error characteristics of the RaG sphere are similar to those of the RaG circle. As the standard deviation of Gaussians is increased, error first decreases up to a point and then increases. At a particular value of σ where the smoothness of the surface is most

Table 8: Maximum radial and tangential errors between a unit sphere and its approximation obtained from a 16-point RaG surface when measured at a grid of 30×30 uniformly spaced samples as a function of σ.

σ	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
E_1	2.1E-2	4.9E-3	3.4E-4	9.8E-6	1.0E-7	9.7E-10	2.7E-7	3.9E-5	3.6E-4
E_2	2.5E-2	3.7E-3	3.2E-4	7.3E-6	7.5E-8	7.1E-10	1.7E-7	3.7E-4	4.4E-4

similar to that of the sphere, lowest error is obtained. It is not known whether a RaG surface can reproduce a sphere exactly at some σ, and this could not be experimentally verified because of the inherent computational inaccuracies involved in $\exp(), \sin(), \cos()$, and sqrt(). With the RaG formulation, not only a full sphere can be generated highly accurately, spherical segments can be generated with a high degree of accuracy also because of very small tangential errors in approximation.

From these experiments, it is interesting to note that the optimal σ for a unit sphere is smaller than that for a unit circle. This can be explained from the signal processing point of view. Considering the curve and surface fitting processes as smoothing operations, a control point in curve fitting is subject to smoothing in one direction, while a control point in surface fitting is subject to smoothing in two directions. Since a 2-D Gaussian of standard deviation σ can be written as the product of two 1-D Gaussians of standard deviation σ in u and v directions, smoothing the control points in 2-D with a Gaussian of standard deviation σ is equivalent to smoothing the control points in 1-D along u and v with Gaussians of standard deviation $\sigma\sqrt{2}$ [12]. Therefore, if the optimal σ for a circle obtained from 3 points is 0.7, it is anticipated that the optimal σ for a sphere obtained from 9 points be $0.7/\sqrt{2}$ or 0.49. The optimal σ obtained for a sphere from 9 points was 0.5, which is very close to 0.49 estimated here. The optimal σ for a sphere constructed from 16 points should be $1/\sqrt{2}$ of optimal σ for a circle obtained from 4 points. The optimal σ generating a circle depends on the spacing between the given points. When 3 point are used, the period for $X(u)$ or $Y(u)$ is 1/3, but when 4 points are used, the period reduces to 1/4. σ is proportional to the period of $X(u)$ or $Y(u)$. Therefore, the optimal σ for creating a circle from 4 points is $0.7 \times \frac{3}{4} = 0.51$, where 0.7 is the optimal σ for creating a circle from 3 points. Therefore, the optimal σ for a sphere obtained from 16 points is $0.51/\sqrt{2}$ or 0.36, which is very close to 0.35, experimentally obtained in Table 8.

Experiments also show that the approximation accuracy for a unit sphere is lower than that for a unit circle. This is believed to be due to the use of a larger number of terms in the computations, each involving exp() and sqrt() when computing E_1 and additionally involving sin() and cos() when computing E_2. Inherent errors in these functions have resulted in a lower approximation accuracy for the sphere when compared to that for the circle.

5 Concluding Remarks

Due to the global nature of RaG curves and surfaces, creation of 2-D and 3-D shapes is computationally more expensive than creation of similar shapes with local formulations such as NURBS. Implementation of RaG curves and surfaces is, however, much easier than im-

plementation of NURBS curves and surfaces because a single RaG curve or surface can represent a very complex 2-D or 3-D shape. If the nodes of a RaG curve or surface are uniformly spaced, the Gaussians used in RaG formulations can be precomputed and saved in a table. The curve or surface can then be efficiently generated through table look-ups.

It was shown that a full circle as well as circular arcs of desired subtending angles can be generated with a single RaG curve, and a full sphere as well as spherical segments can be generated with a single RaG surface. Circles and spheres described by RaG curves and surfaces are more intuitive and easier to remember and design than NURBS and rational Bézier representations. A circle is obtained when a closed RaG curve interpolates the vertices of an equilateral triangle with unique and uniformly spaced nodes, and a sphere is obtained when a closed RaG surface interpolates the vertices of an octahedron with equilateral triangular faces, again using unique and uniformly spaced nodes. In NURBS and rational Bézier formulations for circles and spheres, the knots are not unique (some are used multiple times). Moreover, circles and spheres generated by RaG curves and surfaces are infinitely differentiable, thus they are C^{∞}. Circles and spheres generated by NURBS and rational Bézier curves and surfaces are only C^1.

Experimental results show that with regular computational tools on a PC, a RaG circle can be generated with an accuracy of up to 12 decimal places, and a RaG sphere can be generated with an accuracy of up to 10 decimal places. Although it was not proven, it is believed that there exists a smoothness parameter where a RaG curve reproduces a circle exactly and a smoothness parameter exists where a RaG surface generates a sphere exactly. This is currently an open problem that is left for future research.

References

[1] J. F. Blinn, How many ways can you draw a circle? *Computer Graphics and Applications,* Aug. 1987, pp. 39–44.

[2] W. Böhm, G. Farin, and J. Kahmann, A survey of curve and surface methods in CAGD, *Computer Aided Geometric Design,* vol. 1, 1984, pp. 1–60.

[3] T. Dokken, M. Dæhlen, T. Lyche, and K. Mørken, Good approximation of circles by curvature-continuous Bézier curves, *Computer Aided Geometric Design,* vol. 7, 1990, pp. 33–41.

[4] E. F. Eisele, Best approximation of some parametric surfaces by Bézier surfaces, *Mathematics of Surfaces V: Design and Application of Curves and Surfaces,* R. B. Fisher (ed.), Clarendon Press, 1994, pp. 155–168.

[5] I. D. Faux and M. J. Pratt, *Computational Geometry for Design and Manufacture,* Ellis Horwood Ltd., W. Sussex, UK, 1979, p. 134.

[6] G. Farin, B. Piper, and A. J. Worsey, The octant of a sphere as a non-degenerate triangular Bézier patch, *Computer Aided Geometric Design,* vol. 4, 1987, pp. 329–332.

[7] M. Goldapp, Approximation of circular arcs by cubic polynomials, *Computer Aided Geometric Design,* vol. 8, 1991, pp. 227–238.

[8] A. Goshtasby, Design and recovery of 2-D and 3-D shapes using rational Gaussian curves and surfaces, *Int. J. Computer Vision,* vol. 10, no. 3, 1993, pp. 233–256.

[9] A. Goshtasby, Geometric modeling using rational Gaussian curves and surfaces, *Computer-Aided Design,* vol. 27, no. 5, 1995, pp. 363–375.

[10] G. Herron, Smooth closed surfaces with discrete triangular interpolants, *Computer Aided Geometric Design,* vol. 2, 1985, pp. 297–306.

[11] B. K. P. Horn, Extended Gaussian images, *Proceedings of the IEEE*, vol. 72, no. 12, 1984, pp. 1671–1686.

[12] R. Jain, R. Kasturi, and B. G. Schunck, *Machine Vision,* McGraw-Hill, 1995, p. 130.

[13] A. L. Klosterman, R. H. Ard, and J. W. Klahs, A geometric modeling program for the system designer, *Proc. Conf. CAD/CAM Technology in Mechanical Engineering,* MIT, 1982.

[14] E. T. Y. Lee, The rational Bézier representation for conics, *Geometric Modeling: Algorithms and New Trends,* G. E. Farin (ed.), SIAM, 1987, pp. 3–19.

[15] K. Mørken, Best approximation of circle segments by quadratic Bézier curves, *Curves and Surfaces,* P. J. Laurent, A. Le Méhauté, and L. L. Schumaker (eds.), Academic Press, 1991, pp. 331–336.

[16] M. E. Mortenson, *Geometric Modeling,* 2nd Edition, John Wiley & Sons, 1997.

[17] L. Piegl, The sphere as a rational Bézier surface, *Computer Aided Geometric Design,* vol. 3, 1986, pp. 45–52.

[18] L. Piegl, Infinite control points–A method for representing surfaces of revolution using boundary data, *IEEE Computer Graphics and Applications,* March 1987, pp. 45–55.

[19] L. Piegl, Less data for shapes, *IEEE Computer Graphics and Applications,* Aug. 1987, pp. 48–50.

[20] L. Piegl and W. Tiller, A Menagerie of rational B-spline circles, *IEEE Computer Graphics and Applications,* Sept. 1989, pp. 48–56.

[21] L. Piegl, Algorithms for computing conic splines, *ASCE J. Computing in Civil Engineering,* vol. 4, no. 3, 1990, pp. 180–198.

[22] L. Piegl, On NURBS: A survey, *IEEE Computer Graphics and Applications,* Jan. 1991, pp. 51–71.

[23] W. Tiller, Rational B-spline for curve and surface representation, *IEEE Computer Graphics and Applications,* Sept. 1983, pp. 61–69.