

APPROXIMATE DEVELOPMENTS FOR SURFACES OF REVOLUTION

Mădălina Roxana Buneci, *University Constantin Brâncuși of Târgu-Jiu, ROMÂNIA*

Abstract. The purpose of this paper is provide a set of Maple procedures to construct approximate developments of a general surface of revolution generalizing the well-known gore method for sphere.

Keywords: pattern development; approximate model; surfaces of revolution; Runge phenomenon.

1. INTRODUCTION

Pattern development is the procedure of creating a 2D model (pattern) from a 3D object. A development (pattern) is the unfolded or unrolled flat figure of the surface of a 3D-object. When the pattern is cut, it can be rolled or folded back into the original 3D- object. A pattern may be an exact model (true development) or an approximate model. In this paper we consider surfaces of revolution and we provide Maple procedures for obtain approximate developments (patterns). More precisely, we consider a surface generated by rotating a two-dimensional curve about the z-axis (vertical axis): we do not necessary consider a complete rotation. We assume that the curve is specified parametrically by

$$\begin{cases} x=x(t) \\ z=z(t) \end{cases} \quad t \in [a, b]$$

and that $x(t) \geq 0$ for all $t \in [a, b]$. Then the standard parameterization of this surface of revolution is

$$z = z(t) \quad \begin{cases} x = x(t)\cos(\theta) \\ y = x(t)\sin(\theta), \end{cases} \quad t \in [a, b], \theta \in [\theta_1, \theta_2]$$

The sphere, or more general the spheroid (treated in [2]), the paraboloid, the torus are just some examples fitting in this setting.

2. MAPLE PROCEDURES FOR APPROXIMATE DEVELOPMENT OF SURFACES OF REVOLUTION

We consider a set of parameters $\{t_1, t_2, \dots, t_m\} \subset [a, b]$ such that

$$a=t_1 < t_2 < \dots < t_{m-1} < t_m = b.$$

Surface of revolution is divided into a number of n equal sections of $|\theta_2 - \theta_1|/n$ intervals by vertical planes and in m divisions obtained by the intersections with the horizontal planes: $z=z(t_1), z= z(t_2), \dots, z(t_m)$ (as shown in Figure 2.1). Each section is an archshaped sector $D_1D_2 \dots D_mE_1E_2 \dots E_m$. For each $i \in \{1, 2, \dots, m\}$, $|D_iE_i|$ = length of the arc B_iC_i . Hence $|D_iE_i| = x(t_i) |\theta_2 - \theta_1|/n$. The arch shaped sector $D_1D_2 \dots D_mE_1E_2 \dots E_m$ has a symmetry axis that must be overlapped to the curve resulting by cutting the surface of revolution to a vertical plane. The point $A_i(x(t_i)\cos(\theta_0), x(t_i)\sin(\theta_0), z(t_i))$ belong to that curve for every $i \in \{1, 2, \dots, m\}$ and

appropriate $\theta_0 \in [\theta_1, \theta_2]$. If we denote by A'_1, A'_2, \dots, A'_m the corresponding points on the symmetry axis, then $|A'_1 A'_i| =$ length of the curve γ_i specified parametrically by

$$\begin{cases} x=x(t) \cos(\theta_0) \\ z=z(t) \sin(\theta_0) \end{cases} \quad t \in [t_1, t_2]$$

for every $i \in \{1, 2, \dots, m\}$. Thus $|A'_1 A'_i| = \int_{\gamma_i} 1 d\sigma$ and if the curve is of class C^1 , then

$$|A'_1 A'_i| = \int_{t_1}^{t_i} \sqrt{x'(t)^2 + z'(t)^2} dt$$

for every $i \in \{1, 2, \dots, m\}$. The arch shaped sector is obtained joining the points D_1, D_2, \dots, D_m (and symmetrically, E_1, E_2, \dots, E_m) by a smooth curve. n arch shaped sectors (gores) are needed to form the original surface of revolution.

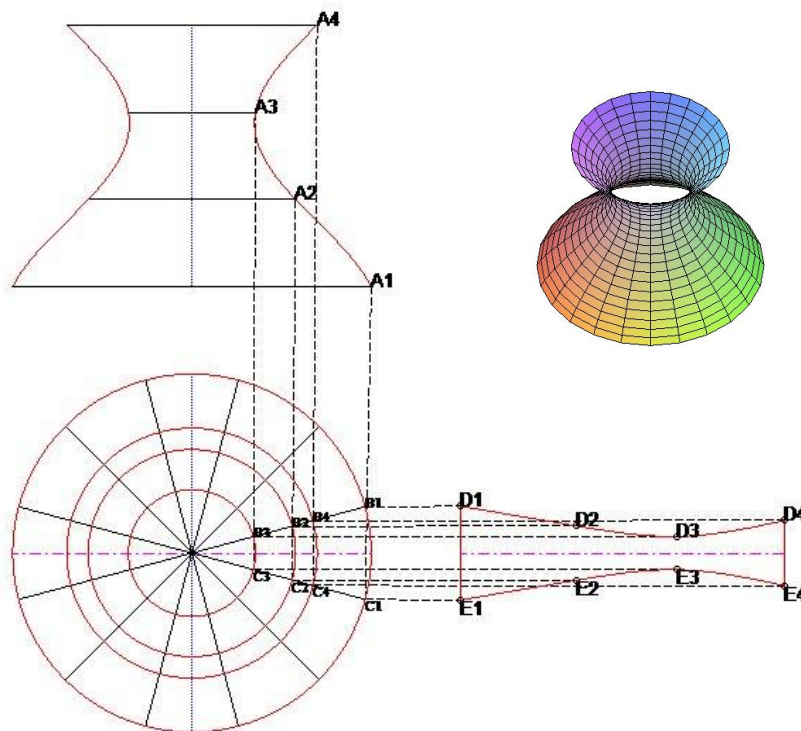


Fig. 2.1

The Maple procedure

```
>param_equad:=proc(a,b,m)
  RETURN([seq(a+(b-a)/m*i,i=0..m)])
```

end proc;

returns a list $[t_1, t_2, \dots, t_{m+1}]$ representing an equidistant division ($a=t_1 < t_2 < \dots < t_{m+1}=b$) of the interval $[a, b]$.

In the following we consider the following example:

```

>theta1:=0; theta2:=2*Pi:
> a:=Pi/6;b:=3*Pi/2:
> xt:=proc(t) RETURN(cos(t)+2) end proc:
> zt:=proc(t) RETURN(t) end proc:
> param_equid(a,b,3);

```

$$\left[\frac{\pi}{6}, \frac{11\pi}{18}, \frac{19\pi}{18}, \frac{3\pi}{2} \right]$$

The next procedure points_gore returns the list $[|D_1E_1|/2, |D_2E_2|/2, \dots, |D_mE_m|/2]$:

```

>points_gore:=proc(x,parameter,theta1,theta2,n)
local i,m,points;
m:=nops(parameter); points:=[seq(1,i=1..m)];
for i from 1 to m do points[i]:=x(parameter[i])*abs(theta2-theta1)/(2*n) end do;
RETURN(points)
end proc;

```

```

> map(eval,points_gore(xt,param_equid(a,b,3),theta1,theta2,12));

```

$$\left[\frac{\left(\frac{\sqrt{3}}{2} + 2\right)\pi}{12}, \frac{1}{12} \left(-\cos\left(\frac{7\pi}{18}\right) + 2\right)\pi, \frac{1}{12} \left(-\cos\left(\frac{\pi}{18}\right) + 2\right)\pi, \frac{\pi}{6} \right]$$

The below procedures curve_length, respectively curve_length_n return the length of the curve specified parametrically by

$$\begin{cases} x=x(t) \\ z=z(t) \end{cases} \quad t \in [a, b]$$

using symbolic computation, respectively numeric computation.

```

>curve_length:=proc(x,z,a,b)
local l;
l:=int((diff(x(t),t)^2+diff(z(t),t)^2)^(1/2),t=a..b);
RETURN(l)
end proc;

```

```

>curve_length_n:=proc(x,z,a,b)
local l;
l:=int((diff(x(t),t)^2+diff(z(t),t)^2)^(1/2),t=a..b, numeric=true);
RETURN(l)
end proc;
(x and z are functions or procedures).

```

If $A_i, i \in \{1, 2, \dots, m\}$ are the points used for horizontally sections of the surface and if A'_1, A'_2, \dots, A'_m are the corresponding points on the symmetry axis of the arch shaped sector, then the next procedure computes $[|A'_1A'_1|, |A'_1A'_2|, \dots, |A'_1A'_m|]$

```

>points_symmetry_axis:=proc(x,z,parameter)
local i,points,m,a,b;

```

```

m:=nops(parameter);a:=min(parameter); b:=max(parameter);
points:=[seq(1,i=1..m)];points[1]:=0;
  for i from 1 to m-1 do
points[i+1]:=points[i]+eval(curve_length(x,z,parameter[i],parameter[i+1]))
end do;
RETURN(points)
end proc;

```

As we mention before the arch shaped sector (**gore**) is obtained joining the points D_1, D_2, \dots, D_m (and symmetrically, E_1, E_2, \dots, E_m) by a smooth curve. This curve joining the points D_1, D_2, \dots, D_m can be seen as a smooth function f with the property that $f(A'_i)=D_i$ for all $i \in \{1, 2, \dots, m\}$. We can use polynomial interpolation to construct f , but if m increases, the Runge phenomenon could appear (as will be shown in the last section). The problem can be avoided by using spline curves (see for instance [3]) which are piecewise polynomials. The procedure `approximation_gore` returns the approximation of f using spline functions (of degree d).

```

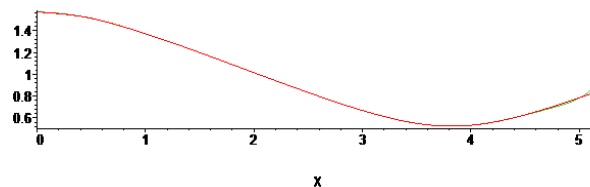
>approximation_gore:=proc(x,z,parameter,theta1,theta2,n,d,X)
local points,values,i,m;
  m:=nops(parameter);
points:=points_symmetry_axis(x,z,parameter);
values:=points_gore(x,parameter,theta1,theta2,n);
RETURN(CurveFitting:-Spline([seq([map(evalf,points[i]),values[i]),i=1..m]),X,degree=d));
end proc;

```

```

>plot(approximation_gore(xt,zt,param_equid(a,b,3),theta1,theta2,12,3,X),X=0..curve_length(
xt,zt,a,b),scaling=constrained);

```



The next procedure plots one of the n arch shaped sectors (**gores**):

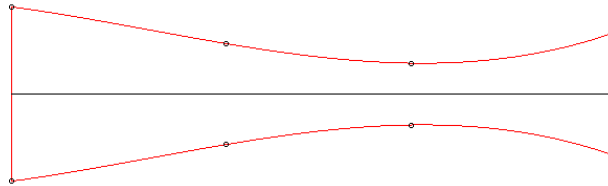
```

>draw_gore:=proc(x,z,parameter,theta1,theta2,n,d)
local d1,d2,d3i,d3f,d4,d5,pointsx,pointsy,len,m;
m:=nops(parameter); len:=eval(curve_length(x,z,parameter[1],parameter[m]));
pointsx:=points_symmetry_axis(x,z,parameter);
pointsy:=points_gore(x,parameter,theta1,theta2,n);
d1:=plot([approximation_gore(x,z,parameter,theta1,theta2,n,d,X),-
approximation_gore(x,z,parameter,theta1,theta2,n,d,X)],X=0..len,color=red,axes=None);
d2:=pointplot([[0,0],[len,0]], connect=true);
d3i:=pointplot([[pointsx[1],pointsy[1]],[pointsx[1],-pointsy[1]]], connect=true,color=red);
d3f:=pointplot([[pointsx[m],pointsy[m]],[pointsx[m],-pointsy[m]]], connect=true,color=red);
d4:=seq(pointplot([pointsx[i],pointsy[i]],symbol=CIRCLE,axes=None),i=1..m);
d5:=seq(pointplot([pointsx[i],-pointsy[i]],symbol=CIRCLE,axes=None),i=1..m);

```

```
display(d1,d2,d3i,d3f,d4,d5, scaling=constrained);
end;
```

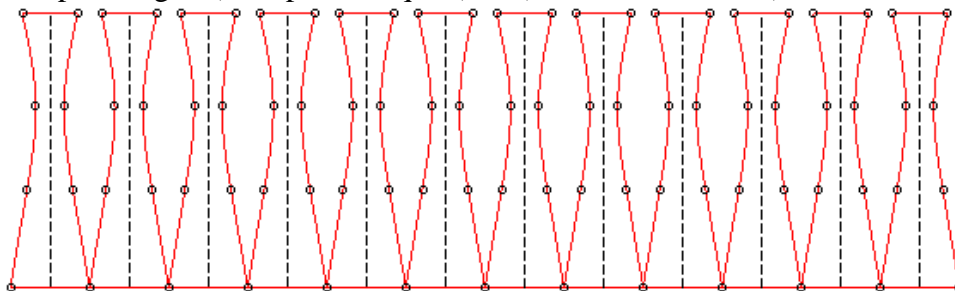
```
> draw_gore(xt,zt,param_equid(a,b,3),theta1,theta2,12,3);
```



The following procedure provides the approximate development of the surface of revolution:

```
>draw_development_gore:=proc(x,z,parameter,theta1,theta2,n,d)
local approx,d1,d1s,d2,d3i,d3f,d4,d4s,pointsx,pointsy,len,maxg,m;
m:=nops(parameter); len:=eval(curve_length(x,z,parameter[1],parameter[m]));
pointsx:=points_symmetry_axis(x,z,parameter);
pointsy:=points_gore(x,parameter,theta1,theta2,n);
approx:=approximation_gore(x,z,parameter,theta1,theta2,n,d,X);maxg:=max(pointsy);
d1:=seq(plot([approx+j*2*maxg,X,X=0..len],color=red,axes=None),j=0..n-1);d1s:=seq(plot([-
approx+j*2*maxg,X,X=0..len],color=red,axes=None),j=0..n-1);
d2:=seq(pointplot([[j*2*maxg,0],[j*2*maxg,len]], connect=true,linestyle=dash),j=0..n-1);
d3i:=seq(pointplot([[pointsy[1]+j*2*maxg,pointsx[1]],[-pointsy[1]+j*2*maxg,pointsx[1]]],
connect=true,color=red),j=0..n-1);d3f:=seq(pointplot([[pointsy[m]+j*2*maxg,pointsx[m]],[-
pointsy[m]+j*2*maxg,pointsx[m]]], connect=true,color=red),j=0..n-1);
d4:=seq(seq(pointplot([pointsy[i]+j*2*maxg,pointsx[i]],symbol=CIRCLE,axes=None),i=1..m)
,j=0..n-1); d4s:=seq(seq(pointplot([-
pointsy[i]+j*2*maxg,pointsx[i]],symbol=CIRCLE,axes=None),i=1..m),j=0..n-1);
display(d1,d1s,d2,d3i,d3f,d4,d4s,scaling=constrained);
end proc;
```

```
>draw_development_gore(xt,zt,param_equid(a,b,3),theta1,theta2,12,3);
```



3. RUNGE PHENOMENON

Runge's phenomenon reveals the error that can appear when constructing a polynomial interpolant of high degree. If $d \in \mathbb{N}$, f is a function of class C^{d+1} on an interval $[a,b]$, $a \leq x_0 < x_1 < \dots < x_d \leq b$ and P_d is the polynomial function of degree d such that $P_d(x_i) = f(x_i)$ for all

$i \in \{0, 1, \dots, d\}$, then for all $x \in [a, b]$, we have:

$$|f(x) - P_d(x)| \leq \frac{\sup_{t \in [a, b]} |f^{(d+1)}(t)|}{(d+1)!} |(x - x_0)(x - x_1) \dots (x - x_d)|.$$

(see for instance, [1, p. 140]). Thus if the magnitude of the derivatives of the function f grows quickly when d increases, then magnitude of the oscillations at the edges of an interval $[a, b]$ that occurs when using polynomial interpolation with a polynomial of degree d also increases.

In particular this can occur when we try to approximate the curve joining the points D_1, D_2, \dots, D_m (the points that define a gore) by an interpolating polynomial of degree $m-1$ (with m big enough). Let us consider the following example:

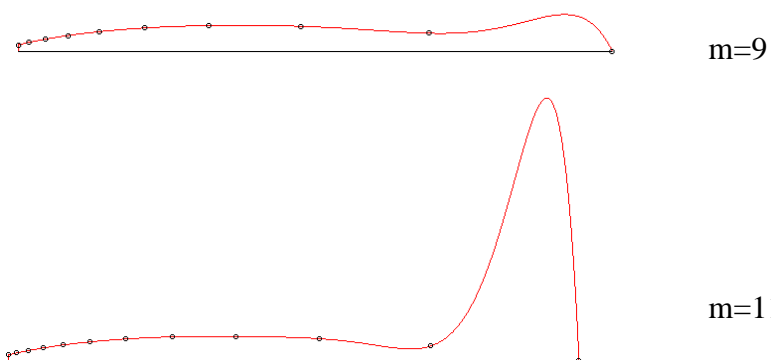
```
>a:=0:b:=Pi/2:theta1:=0:theta2:=Pi:
>xt:=proc(t) RETURN(cos(t)*exp(2*t)) end proc:
>zt:=proc(t) RETURN(sin(t)*exp(2*t)) end proc:
>with(plots):plot3d([xt(t)*cos(theta),xt(t)*sin(theta),zt(t)],t=a..b,theta=theta1..theta2,scaling=
constrained);
```



The following procedure approximate the curve joining the points D_1, D_2, \dots, D_m (the points that define a gore) by an interpolating polynomial (PLagrange is the procedure defined in [1, p. 146]; we can use instead it the Maple command PolynomialInterpolation from CurveFitting package):

```
>approximation_gore_Lagrange:=proc(x,z,parameter,theta1,theta2,n,X)
local points,values,i,m;
m:=nops(parameter);
points:=points_symmetry_axis(x,z,parameter);values:=points_gore(x,parameter,theta1,theta2,
n);
RETURN(PLagrange(points,values,X)) end proc;
```

We illustrate the result of polynomial interpolation for an equidistant division ($a=t_1 < t_2 < \dots < t_{m+1} = b$) of the interval $[a, b]$ (used for parameterization) for $m=9$ and $m=11$:



BIBLIOGRAPHY

- [1] M. Buneci, *Metode Numerice – aspecte teoretice și practice*, Editura Academica Brâncuși, 2009.
- [2] L. Luca, M. Buneci, C. Rădulescu and A. Stăncioiu, Study on a method for determining as accurately as possible the “development” of an ellipsoid, *Annals of the “Constantin Brâncuși” University of Târgu-Jiu. Engineering Series. No. 1(2015)*, 42-53.
- [3] V. Rovenski, *Geometry of Curves and Surfaces with MAPLE*, Springer Science & Business Media, 2013.