BURGEVIN Valentin
DURET Guillaume

GR IMA

# TP3 Géométrie différentielle :

2018-2019

Note : nous n’avons pas réussi à totalement faire marcher le code de ce TP ce qui nous as valu beaucoup de temps de débogage en vain ce qui explique le fait que ce compte rendu n’a qu’un minimum de contenu. De plus excusez-nous pour le retard pour rendre celui-ci.

Le but de TP est de calculer et d’étudier les différentes courbures pour différentes surfaces paramétriques.

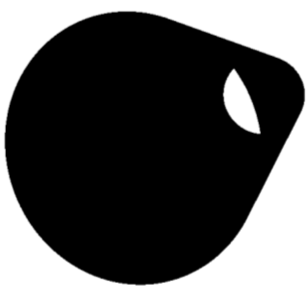
En effet il y a par exemple :

- Les courbures de Gauss : Ks
- Les courbures moyennes : Hs
- Les courbures principales : lambda1 ou lambda2

Les différentes formes testées sont :

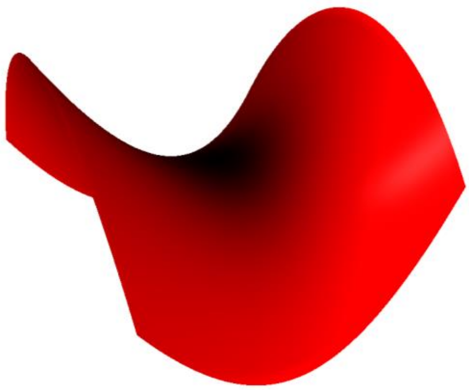
- La sphère
- Le cylindre
- Le paraboloïde hyperbolique
- La caténoïde

Malheureusement après plusieurs longues heures nous n’avons pas réussi à afficher correctement les courbures des différentes formes, en effet nous n’arrivons pas à repérer l’erreur ce qui nous donne des résultats qui n’ont pas de sens pour certaines figures



Pour le cylindre on obtient une courbure uniforme car celui n'a pas différentes pentes ce qui peut être cohérent.

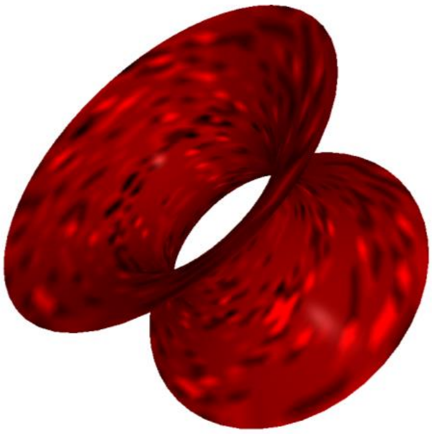
Pour le paraboloïde hyperbolique nous obtenons le résultat suivant :



Pour cette forme nous obtenons un résultat cohérent car cette forme possède des courbes plus resserrées au centre plutôt que sur les bords où c'est plutôt homogène.

Pour la sphère on a des calculs qui ont pour résultat « nan » ce qui donne ces taches noires sur la sphère…

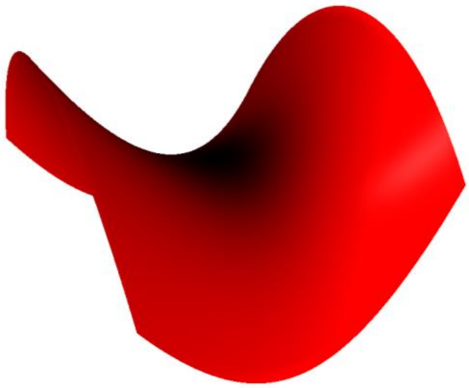
On obtient un résultat semblable pour la caténoïde :



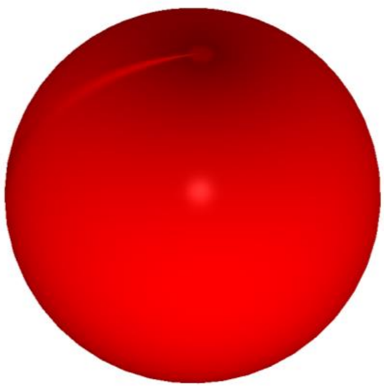
N’ayant pas de résultat très concluant on s’est quand même orienté sue le calcul de dérivée partiel informatiquement en utilisant l’approximation de la rerivé par son taux d’accroissement :

```
float const dxu = ((x_catenoide(ku+1,kv) - x_catenoide(ku,kv))/pas_u);
float const dyu = ((y_catenoide(ku+1,kv) - y_catenoide(ku,kv))/pas_u);
float const dzu = ((z_catenoide(ku+1,kv) - z_catenoide(ku,kv))/pas_u);

vec3 Su=vec3(dxu,dyu,dzu);

float const dxv = ((x_catenoide(ku,kv+1) - x_catenoide(ku,kv))/pas_u);
float const dyv = ((y_catenoide(ku,kv+1) - y_catenoide(ku,kv))/pas_u);
float const dzv = ((z_catenoide(ku,kv+1) - z_catenoide(ku,kv))/pas_u);

vec3 Sv=vec3(dxv,dyv,dzv);

float const dxuu = (x_catenoide(ku+1,kv) - 2*x_catenoide(ku,kv) + x_catenoide(ku-1,kv)) / (pas_u*pas_u);
float const dyuu = (y_catenoide(ku+1,kv) - 2*y_catenoide(ku,kv) + y_catenoide(ku-1,kv)) / (pas_u*pas_u);
float const dzuu = (z_catenoide(ku+1,kv) - 2*z_catenoide(ku,kv) + z_catenoide(ku-1,kv)) / (pas_u*pas_u);

vec3 Suu=vec3(dxuu,dyuu,dzuu);

float const dxuv = ((((x_catenoide(ku+1,kv+1) - x_catenoide(ku,kv+1)) / (pas_u)) - ((x_catenoide(ku+1,kv) - x_catenoide(ku,kv)) / ((pas_u)))) / (pas_v));
float const dyuv = ((((y_catenoide(ku+1,kv+1) - y_catenoide(ku,kv+1)) / (pas_u)) - ((y_catenoide(ku+1,kv) - y_catenoide(ku,kv)) / ((pas_u)))) / (pas_v));
float const dzuv = ((((z_catenoide(ku+1,kv+1) - z_catenoide(ku,kv+1)) / (pas_u)) - ((z_catenoide(ku+1,kv) - z_catenoide(ku,kv)) / ((pas_u)))) / (pas_v));

vec3 Suv=vec3(dxuv,dyuv,dzuv);

float const dxvv = (x_catenoide(ku,kv+1) - 2* x_catenoide(ku,kv) + x_catenoide(ku,kv-1)) / (pas_v*pas_v);
float const dyvv = (y_catenoide(ku,kv+1) - 2* y_catenoide(ku,kv) + y_catenoide(ku,kv-1)) / (pas_v*pas_v);
float const dzvv = (z_catenoide(ku,kv+1) - 2* z_catenoide(ku,kv) + z_catenoide(ku,kv-1)) / (pas_v*pas_v);

vec3 Svv=vec3(dxvv,dyvv,dzvv);

float const dxvu = ((((x_catenoide(ku+1,kv+1) - x_catenoide(ku+1,kv)) / (pas_v)) - ((x_catenoide(ku,kv+1) - x_catenoide(ku,kv)) / ((pas_v)))) / (pas_u));
float const dyvu = ((((y_catenoide(ku+1,kv+1) - y_catenoide(ku+1,kv)) / (pas_v)) - ((y_catenoide(ku,kv+1) - y_catenoide(ku,kv)) / ((pas_v)))) / (pas_u));
float const dzvu = ((((z_catenoide(ku+1,kv+1) - z_catenoide(ku+1,kv)) / (pas_v)) - ((z_catenoide(ku,kv+1) - z_catenoide(ku,kv)) / ((pas_v)))) / (pas_u));
```

Nous obtenons cette fois-ci un résultat identique qu’avec les formules explicites pour la forme du paraboloïde à l’approximation près. (Pour Nu = Nv =50) on a environ une différence de 10% des valeurs entre les deux méthodes qui ne se remarque pas visuellement.



Nous faisons la même chose pour la sphère et nous obtenons la figure suivante :



Nous avons un résultat meilleur qu'avec les formules explicites. On peut remarquer que tout comme le cylindre, les courbures sont plutôt homogènes car la sphère décrit par définition un cercle et dans des courbures constantes.

Pour la caténoïde le résultat peut être cohérent, en effet celle-ci étant une figure symétrique elle possède des courbure inversé ce qui visuellement donne l'effet noir(courbure négative) et effet rouge (courbure positive).