Computing on Surfaces with Chebfun2

Rodrigo Platte

ASU | School of Mathematical & Statistical Sciences
Arizona State University

SIAM Annual Meeting, July 2013
San Diego, CA
Non-rectangular domains by change of variables

% Define parameters on a rectangle
r = chebfun2(@(r,t) r,[1/2 2 0 2*pi]);
t = chebfun2(@(r,t) t,[1/2 2 0 2*pi]);

% Change variables
[x,y] = pol2cart(t, r+1*cos(5*t));

% Define a function on non-rectangular domain
f = cos(5*x.*y) + y;
surf([x;y;f]), view(0,90), axis equal tight off

See example by Alex Townsend [here](#).
PARAMETRIC SURFACES

\[ u = \text{chebfun2}(\{u,v\} u, [-1 1 0 2*pi]); \]
\[ v = \text{chebfun2}(\{u,v\} v, [-1 1 0 2*pi]); \]

\[ x = u.*\cos(v); \]
\[ y = u.*\sin(v); \]
\[ z = u; \]
\[ \text{surf}(x,y,z), \text{axis equal} \]

\[ x = \sqrt{1/4+u.^2}.*\cos(v); \]
\[ y = \sqrt{1/4+u.^2}.*\sin(v); \]
\[ z = u; \]
\[ \text{surf}(x,y,z), \text{axis equal} \]
Parametric surfaces

t = chebfun2(@(t,p) t, [0 2*pi -pi/2 pi/2]);
p = chebfun2(@(t,p) p, [0 2*pi -pi/2 pi/2]);

[x,y,z] = sph2cart(t,p,0*t+1);
surf(x,y,z), axis equal

d = 0.05*sin(20*x);
[x,y,z] = sph2cart(t,p,1+d);
surf(x,y,z), axis equal
PARAMETRIC SURFACES

\[ u = \text{chebfun2}(\@u,v, [-1 1 0 2*pi]) ; \]
\[ v = \text{chebfun2}(\@u,v, [-1 1 0 2*pi]) ; \]

\[ x = u.*\cos(v); \]
\[ y = u.*\sin(v); \]
\[ z = u; \]
\[ \text{surf}(x,y,z), \text{axis equal} \]

\[ x = \sqrt{1/4+u.^2}.*\cos(v); \]
\[ y = \sqrt{1/4+u.^2}.*\sin(v); \]
\[ z = u; \]
\[ \text{surf}(x,y,z), \text{axis equal} \]
Function representation on surfaces

\[ f = \sin(10 \cdot z^2); \]
\[ \text{surf}(x, y, z, f), \text{ axis } \text{equal} \]
Normal vector field

- `chebfun2v` can be used to represent vector fields on surfaces.
- Let \( R(u, v) \) be the position vector on a surface parameterized by \( u \) and \( v \). The vectors \( R_u \) and \( R_v \) are tangent to the surface.
- The vector \( n = R_u \times R_v \) is normal to the surface.

Möbius strip:

```matlab
dom = [0 2*pi -1 1];
u = chebfun2(@(u,v) u, dom);
v = chebfun2(@(u,v) v, dom);

x = (1+0.5*v.*cos(u/2)).*cos(u);
y = (1+0.5*v.*cos(u/2)).*sin(u);
z = 0.5*v.*sin(u/2);
r = [x;y;z];  % chebfun2v
n = normal(r);  % chebfun2v

surf(x,y,z), view(-63,78)
camlight, hold on,
quiver3(x,y,z,n,2,'k')
axis tight, hold off
```
NORMAL vector field - MöBIUS strip
\begin{verbatim}
% The Klein bottle

u = chebfun2(@(u,v) u, [0 pi 0 2*pi]);
v = chebfun2(@(u,v) v, [0 pi 0 2*pi]);

x = -(2/15)*cos(u).*(3*cos(v)-30*sin(u)+90*cos(u).^4.*sin(u)-60*cos(u).^6.*sin(u)+5*cos(u).*cos(v).*sin(u));
y = -(1/15)*sin(u).*(3*cos(v)-3*cos(u).^2.*cos(v)-48*cos(u).^4.*cos(v)-48*cos(u).^6.*cos(v)-60*sin(u)+5*cos(u).*cos(v).*sin(u)-5*cos(u).^3.*cos(v).*sin(u)-80*cos(u).^5.*cos(v).*sin(u)+80*cos(u).^7.*cos(v).*sin(u));
z = (2/15)*(3+5*cos(u).*sin(u)).*sin(v);

surf(x,y,z,'FaceAlpha',.5), camlight left, axis tight equal off
\end{verbatim}
The Klein bottle

```
hold on
quiver3(x, y, z, -normal([x; y; z]), 2, 'k')
hold off
```
THE GAUSS MAP

dom = [-1 1 0 2*pi];
u = chebfun2(@(u,v) u, dom);
v = chebfun2(@(u,v) v, dom);
x = sqrt(1/4+u.^2).*cos(v);
y = sqrt(1/4+u.^2).*sin(v);
z = u;
surf(x,y,z), axis equal

N = -normal([x;y;z], 'unit');
surf(N(1),N(2),N(3),z),
axis image
THE GAUSS MAP
The Gauss map
Surface integrals

Surface area:

\[ A = \int \int_{S} dS = \int_{a}^{b} \int_{c}^{d} \left| \frac{\partial R}{\partial u} (u, v) \times \frac{\partial R}{\partial v} (u, v) \right| \, du \, dv = \int_{a}^{b} \int_{c}^{d} \left| N(u, v) \right| \, du \, dv \]

```plaintext
dom = [0 2*pi 0 2*pi];
u = chebfun2(@(u,v) u, dom);
v = chebfun2(@(u,v) v, dom);
x = (3+cos(v)).*cos(u);
y = (3+cos(v)).*sin(u);
z = sin(v);
n = normal([x;y;z]);
A = integral2(sqrt(n'*n))
Exact = 4*pi^2*3*1
```

\[ A = 1.184352528130722e+02 \]

Exact = 1.184352528130723e+02
Surface integrals:

Surface area:

\[ A = \iint_S dS = \int_a^b \int_c^d \left| \frac{\partial R}{\partial u}(u, v) \times \frac{\partial R}{\partial v}(u, v) \right| \, du \, dv = \int_a^b \int_c^d |N(u, v)| \, du \, dv \]

Remark: computation may fail if \( \left| \frac{\partial R}{\partial u}(u, v) \times \frac{\partial R}{\partial v}(u, v) \right| \approx 0 \).
THE VOLUME OF A HEART AND THE DIVERGENCE THEOREM

\[ \iiint_V (\nabla \cdot F) \, dV = \iint_S (F \cdot n) \, dS. \]

Let \( F(x, y, z) = [0, 0, z]^T \) and notice that \( \nabla \cdot F = 1 \), then

\[ V = \iint_S (F \cdot n) \, dS \]
THE VOLUME OF A HEART AND THE DIVERGENCE THEOREM

\[
\begin{align*}
  u &= \text{chebfun2(@(v,u) u,[0 1 0 4*pi])}; \\
  v &= \text{chebfun2(@(v,u) v,[0 1 0 4*pi])}; \\
  x &= \sin(pi*v) \cdot \cos(u/2); \\
  y &= 0.7 \cdot \sin(pi*v) \cdot \sin(u/2); \\
  z &= (v-1) \cdot (\cos(u)+\cos(2*u))/(\cos(u)^2); \\
  N &= \text{normal([x;y;z])}; \\
  F &= [0*z;0*z;z]; \\
  \text{Vol} &= \text{integral2(F' \cdot N)}
\end{align*}
\]

\[
\text{Vol} = 2.199114857512853
\]
grad, div, Lap, curl, ...
The first fundamental form is the inner product on the tangent plane. Let $x = a_1 R_u + b_1 R_v$ and $y = a_2 R_u + b_2 R_v$,

$$I(x, y) = \langle x, y \rangle = \langle a_1 R_u + b_1 R_v, a_2 R_u + b_2 R_v \rangle$$

$$= \begin{bmatrix} a_2 & b_2 \end{bmatrix} \begin{bmatrix} \langle R_u, R_u \rangle & \langle R_u, R_v \rangle \\ \langle R_u, R_v \rangle & \langle R_v, R_v \rangle \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix}$$

$$= \begin{bmatrix} a_2 & b_2 \end{bmatrix} \begin{bmatrix} E & F \\ F & G \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix}.$$
THE SURFACE GRADIENT

\[
\begin{pmatrix}
  f_u \\
  f_v
\end{pmatrix}
= 
\begin{pmatrix}
  x_u & y_u & z_u \\
  x_v & y_v & z_v
\end{pmatrix}
\begin{pmatrix}
  \tilde{f}_x \\
  \tilde{f}_y \\
  \tilde{f}_z
\end{pmatrix}
\]

We also know that the gradient of \( f \) must be tangent to the surface, i.e.,

\[ n \cdot \nabla f = 0. \]

We now let \( \nabla_{sf} = \alpha r_u + \beta r_v \) and rewrite the system in terms of \( \alpha \) and \( \beta \),

\[
\begin{pmatrix}
  f_u \\
  f_v
\end{pmatrix}
= 
\begin{pmatrix}
  R_u^T \\
  R_v^T
\end{pmatrix}
(\alpha R_u + \beta R_v)
= 
\begin{pmatrix}
  E & F \\
  F & G
\end{pmatrix}
\begin{pmatrix}
  \alpha \\
  \beta
\end{pmatrix},
\]

where \( E = \langle R_u, R_u \rangle, F = \langle R_u, R_v \rangle, \) and \( G = \langle r_v, r_v \rangle \) are the coefficients of the first fundamental form.
The surface gradient

Solving the system gives

\[
\begin{pmatrix}
\alpha \\
\beta \\
\end{pmatrix} = \frac{1}{EG - F^2} \begin{pmatrix}
G & -F \\
-F & E \\
\end{pmatrix} \begin{pmatrix}
f_u \\
f_v \\
\end{pmatrix},
\]

and

\[
\nabla_{sf} = \frac{1}{EG - F^2} \left[ (Gf_u - Ff_v)R_u + (-Ff_u + Ef_v)R_v \right],
\]

or

\[
\nabla_{sf} = \frac{1}{EG - F^2} \left[ (GR_u - FR_v)f_u + (ER_v - FR_u)f_v \right].
\]
function G = grads(f, r)

% GRADS Surface gradient of a Chebfun2.
% GRADS(F,R) computes the gradient of the Chebfun2 F over the
% parametric surface defined by the position vector R (chebfun2v).

% Tangent vectors
ru = diffx(r);
rv = diffy(r);

% Coeffs of the first fundamental form
E = ru.'*ru;
F = ru.'*rv;
G = rv.'*rv;
D = E.*G - F.^2;

% Surface gradient
r1 = (G*ru - F*rv)/D;
r2 = (E*rv - F*ru)/D;
G = diffx(f).*r1 + diffy(f).*r2;
The surface gradient on a torus:

```
u = chebfun2(@(u,v) u, [0 2*pi 0 2*pi]);
v = chebfun2(@(u,v) v, [0 2*pi 0 2*pi]);
x = (3+cos(v)).*cos(u);
y = (3+cos(v)).*sin(u);
z = sin(v);

f = sin(2*u).*sin(v);
G = grads(f,[x;y;z]);
surf(x,y,z,f), axis off equal, hold on
quiver3(x,y,z,G,2,'k','linewidth',2)
hold off
```
Divergence

Let $F = [P, Q, R]^T$ be tangent to $S$.

$$\nabla \cdot F = P_x + Q_y + R_x$$

$$\nabla_S \cdot F = (\nabla_S P)_1 + (\nabla_S Q)_2 + (\nabla_S R)_3$$

```matlab
n = normal([x;y;z], 'unit');
V = cross(n, G);
surf(x,y,z,f), axis off equal, hold on
quiver3(x,y,z,V,2,'k','linewidth',2)

minandmax2(divs(V,[x;y;z]))
```

```
ans =
   0
   0
```
\[ \Delta_{sf} = \nabla_s \cdot \nabla_{sf} \]

Lapf = divs(G, [x; y; z]);
surf(x, y, z, Lapf), axis off equal
Gaussian and mean curvature

Code and examples by Courtney Page-Bottorff
The principal curvatures at $p$, denoted $\kappa_1$ and $\kappa_2$, are the maximum and minimum values of the curvatures of the different normal planes at point $p$. 

Image from Wikipedia, Principle Curvature
Computation of Gaussian and mean Curvature

If $S$ is a regular surface in $\mathbb{R}^3$, then the Mean and Gaussian Curvatures can be defined by

Gaussian Curvature $(\kappa_1 \kappa_2)$

$$K = \frac{LN - M^2}{EG - F^2}$$

Mean Curvature $(\frac{\kappa_1 + \kappa_2}{2})$

$$H = \frac{EN - 2FM + GL}{2(EG - F^2)}$$

- $E$, $F$, and $G$ are the coefficients of the first fundamental form
- $L$, $M$, and $N$ are the coefficients of the second fundamental form
Implementation using chebfun2

```
function gc = gcurvature(S)
    Su  = diffx(S);
    Sv  = diffy(S);
    Suu = diffx(Su);
    Suv = diffy(Su);
    Svv = diffy(Sv);
    E   = dot(Su,Su);
    F   = dot(Su,Sv);
    G   = dot(Sv,Sv);
    m   = cross(Su,Sv);
    n   = m./sqrt(dot(m,m));
    L   = dot(Suu,n);
    M   = dot(Suv,n);
    N   = dot(Svv,n);
    gc  = ((L.^N-M.^2).../.((E.*G)-F.^2));
end
```

```
function mc = mcurvature(S)
    Su  = diffx(S);
    Sv  = diffy(S);
    Suu = diffx(Su);
    Suv = diffy(Su);
    Svv = diffy(Sv);
    E   = dot(Su,Su);
    F   = dot(Su,Sv);
    G   = dot(Sv,Sv);
    m   = cross(Su,Sv);
    n   = m./sqrt(dot(m,m));
    L   = dot(Suu,n);
    M   = dot(Suv,n);
    N   = dot(Svv,n);
    mc  = ((E.*N)+(G.*L)-(2.*F.*M)).../.(2.*((E.*G)-F.^2));
end
```
Test problems

gc = gcurvature(r);
mc = mcurvature(r);
format long
gc(0,0.1)
mc(0,0.1)

ans =
0.999999999999963
ans =
0.999999999999979
SURFACES WITH NON-CONSTANT CURVATURES

\[ S = \text{chebfun2v}(\theta(u,v)u, \theta(u,v)v, \theta(u,v)u^3 - 3u*v^2); \]
S = chebfun2v(@(u,v) (3+cos(v)).*cos(u), @(u,v) (3+cos(v)).*sin(u), @(u,v) sin(v), [0 2*pi 0 2*pi]);
Mean curvature flow is an example of geometric flow of surfaces. It can be used to model droplets of oil on water, evolution of soap films, etc.

1 A problem suggested by Colin B. Macdonald
work in progress ...
work in progress ...
Thank you! In particular, thanks to Alex Townsend.