MAT300 CURVES AND SURFACES

Julia Sánchez Sanz

DigiPen Instiiute of Technology Europe

julia.sanchez@digipen.edu

Spring 2020

Parametric curves and surfaces

Line in 2D

In $\mathbb{R}^2$ a line is expressed in vector form as

$$(x, y) = (p_1, p_2) + \lambda \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}, \qquad \lambda \in \mathbb{R}$$

from vector form we can pass to parametric equation

$$\begin{cases} x = p_1 + \lambda v_1, \\ y = p_2 + \lambda v_2, \end{cases} \quad \lambda \in \mathbb{R}$$

So every point (x, y) in the line can be expressed in terms of the parameter λ as $x(\lambda) = p_1 + \lambda v_1$ and $y(\lambda) = p_2 + \lambda v_2$ for $\lambda \in \mathbb{R}$.

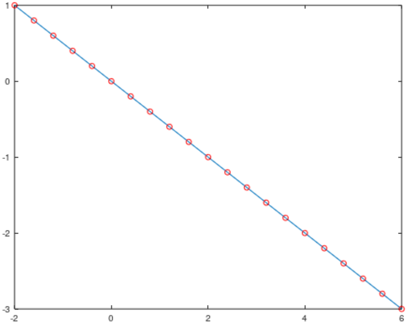
Definition

A line in $\mathbb{R}^2$ can be expressed as the graph of a parametric function $\gamma : \mathbb{R} \to \mathbb{R}^2$ given as

$$\gamma(\lambda) = (p_1 + \lambda v_1, p_2 + \lambda v_2), \qquad \lambda \in \mathbb{R}$$

Example line in 2D

```
%======================================
% MAT300 CURVES AND SURFACES
% DigiPen Bilbao
% Julia Sanchez julia.sanchez@digipen.edu
% 5/6/2019
%
% Plot line in 2d given in parametric
%=======================================

function line2d
  t=[-2:.2:2]; %mesh for parameter t

  x=2+2*t; %parametric x component
  y=-1-t;  % parametric y component

  plot(x,y) % plot line
  hold on
  plot(x,y,'ro') % plot nodes
end
```



Line in 3D

In $\mathbb{R}^3$ the idea is the same.

The parametric equation of a line is

$$\begin{cases} x = p_1 + \lambda v_1, \\ y = p_2 + \lambda v_2, \\ z = p_3 + \lambda v_3, \end{cases} \quad \lambda \in \mathbb{R}$$

So every point (x, y, z) in the line can be expressed in terms of the parameter λ as $x(\lambda) = p_1 + \lambda v_1$, $y(\lambda) = p_2 + \lambda v_2$, $z(\lambda) = p_3 + \lambda v_3$ for $\lambda \in \mathbb{R}$.

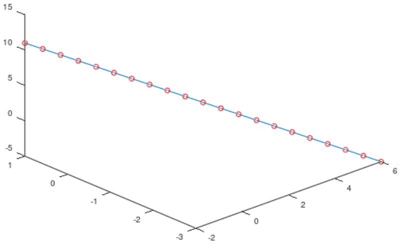
Definition

A line in $\mathbb{R}^3$ can be expressed as the graph of a parametric function $\gamma : \mathbb{R} \to \mathbb{R}^3$ given as

$$\gamma(\lambda) = (p_1 + \lambda v_1, p_2 + \lambda v_2, p_3 + \lambda v_3), \qquad \lambda \in \mathbb{R}$$

Example line in 3D

```matlab
%======================================
% MAT300 CURVES AND SURFACES
% DigiPen Bilbao
% Julia Sanchez julia.sanchez@digipen.edu
% 5/6/2019
%
% Plot line in 3d given in parametric
%=======================================

function line3d
  t=[-2:.2:2]; %mesh for parameter t

  x=2+2*t; %parametric x component
  y=-1-t;  % parametric y component
  z=3-4*t; % parametric z component

  plot3(x,y,z) % plot line
  hold on
  plot3(x,y,z,'ro') % plot nodes
end
```



Plane in 3D

In $\mathbb{R}^3$ a plane is expressed in vector form as

$$(x, y, z) = (p_1, p_2, p_3) + \lambda \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} + \mu \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix}, \qquad \lambda,\ \mu \in \mathbb{R}$$

from vector form we can pass to parametric equation

$$\begin{cases} x = p_1 + \lambda u_1 + \mu v_1, \\ y = p_2 + \lambda u_2 + \mu v_2, \\ z = p_3 + \lambda u_3 + \mu v_3, \end{cases} \quad \lambda,\ \mu \in \mathbb{R}$$

So every point (x, y, z) in the plane can be expressed in terms of the parameters λ and μ.

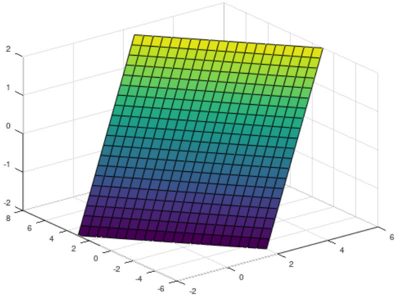
Definition

A plane in $\mathbb{R}^3$ is given as the graph of a function $\gamma : \mathbb{R}^2 \to \mathbb{R}^3$ as

$$\gamma(\lambda, \mu) = (p_1 + \lambda u_1 + \mu v_1, p_2 + \lambda u_2 + \mu v_2, p_3 + \lambda u_3 + \mu v_3), \qquad \lambda,\ \mu \in \mathbb{R}$$

Example plane in 3D

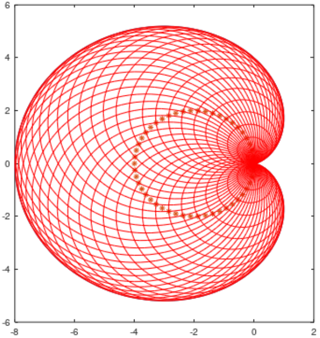
```
%======================================
% MAT300 CURVES AND SURFACES
% DigiPen Bilbao
% Julia Sanchez julia.sanchez@digipen.edu
% 5/6/2019
%
% Plot plane in 3d given in parametric
%=======================================

function plane1
   t=[-2:.2:2]; %mesh for parameter t
   s=[-2:.2:2]; %mesh for parameter s

   [T,S]=meshgrid(t,s); % create grid

   x=2+T+S; %evaluate x-component
   y=1-2.*T+S; %evaluate y-component
   z=S; %evaluate z-component

   surf(x,y,z) % plot plane
end
```

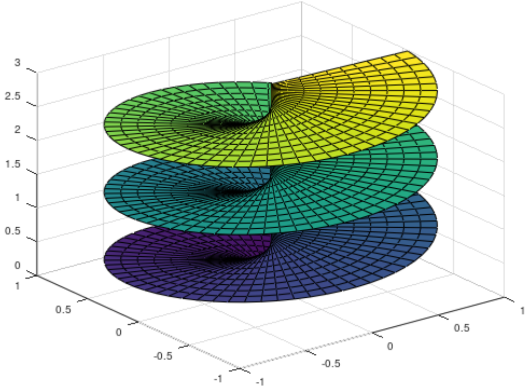


Jump to the nonlinear world

The idea is to extend parametric functions from the linear to the nonlinear case.

Define more complex curves and surfaces.





Parametric curves

Definition

Let t be a parameter $t \in [a, b] \subset \mathbb{R}$.

A parametric curve in 2D is given by a function $\gamma : [a, b] \to \mathbb{R}^2$

$$\gamma(t) = (x(t), y(t)) \tag{1}$$

where $x,\ y : [a, b] \to \mathbb{R}$.

A parametric curve in 3D is given by a function $\gamma : [a, b] \to \mathbb{R}^3$

$$\gamma(t) = (x(t), y(t), z(t)) \tag{2}$$

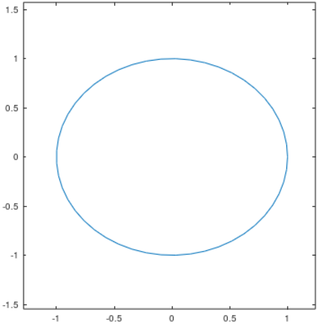
where $x,\ y,\ z : [a, b] \to \mathbb{R}$.

Example: the unit circle

$t \in [0, 2\pi]$

define $\gamma : [0, 2\pi] \to \mathbb{R}^2$ given as $\gamma(t) = (\cos(t), \sin(t))$

graph of the curve:



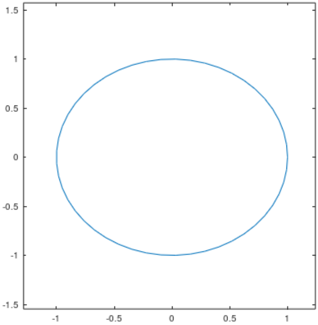

```
%=======================================
% MAT300 CURVES AND SURFACES
% DigiPen Bilbao
% Julia Sanchez julia.sanchez@digipen.edu
% 5/6/2019
%
% Plot graph of a unit circle
%=========================================

function circle1
  t=linspace(0,2*pi,50); %mesh for parameter t

  x=cos(t); %evaluate x-component
  y=sin(t); %evaluate y-component

  plot(x,y) % plot plane
end
```

Example: the other unit circle

$t \in [0, 4\pi]$

define $\hat{\gamma} : [0, 4\pi] \to \mathbb{R}^2$ given as $\hat{\gamma}(t) = \left(\cos\left(\frac{t}{2}\right), \sin\left(\frac{t}{2}\right)\right)$

graph of the curve:




```
%========================================
% MAT300 CURVES AND SURFACES
% DigiPen Bilbao
% Julia Sanchez julia.sanchez@digipen.edu
% 5/6/2019
%
% Plot graph of a unit circle
%=========================================

function circle2
  t=linspace(0,4*pi,50); %mesh for parameter t

  x=cos(t/2); %evaluate x-component
  y=sin(t/2); %evaluate y-component

  plot(x,y) % plot plane
end
```

Parametric curves

Remark: γ and $\hat{\gamma}$ have the same graph but they are different curves.

$\gamma : [0, 2\pi] \to \mathbb{R}^2$ given as $\gamma(t) = (\cos(t), \sin(t))$

$\hat{\gamma} : [0, 4\pi] \to \mathbb{R}^2$ given as $\hat{\gamma}(t) = \left(\cos\left(\frac{t}{2}\right), \sin\left(\frac{t}{2}\right)\right)$

They have different domains, and the expressions of the maps are different.

Definition

The image of $[a, b]$ under γ is the graph of the curve.

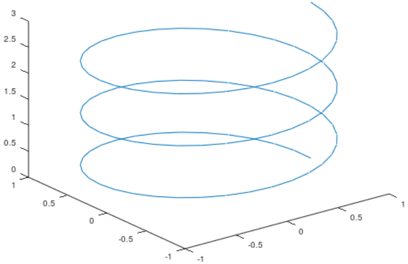
If we think t as a time variable, γ completes the circle faster than $\hat{\gamma}$.

Example 3D curve: helix

$t \in [0, 6\pi]$

define $\gamma : [0, 6\pi] \to \mathbb{R}^3$ given as $\gamma(t) = (\cos(t), \sin(t), \frac{t}{2\pi})$

graph of the curve:




```
%================================================
% MAT300
% Julia Sánchez Sanz
% julia.sanchez@digipen.edu
% 11-1-2018

% Parametric curves using Octave 3d
%================================================

function helix

%obtain the graph of an helix in 3d

t=linspace(0,6*pi,100); %parameter t in [0,6pi]

x=cos(t); % x-component
y=sin(t); % y-component
z=t/(2*pi); % z-component

plot3(x,y,z) % graph of the curve

end
```

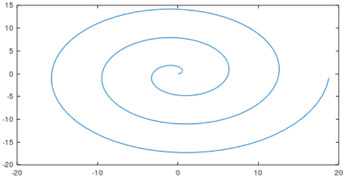
Why parametric curves?

Curves can be defined explicitly, implicitly or parametric.

- Explicitly: $y = f(x)$ for $x \in [a, b]$

Advantages: easy to define, easy to compute.

Disadvantages: few possibilities, no intersections, no go back in the x-axis



not possible!

Why parametric curves?

- Implicitly: through equations $f(x, y) = 0$ for $x \in [a, b]$ and $y \in [c, d]$

  Advantages: we can represent almost every type of curve

  Disadvantages: expensive to obtain graph from computational perspective (curve continuation methods, level curves, problems with singularities)

Why parametric curves?

- Parametric: $\gamma(t) = (x(t), y(t))$ for $t \in [a, b]$.

Advantages:

it is explicit for each component so easy to define and to compute.

They are not proper explicit functions so we can do intersections, and turn back in the x-axis, so we can compute almost every type of curve.

Disadvantages:

for the type of problems that we want to solve, there are no disadvantages.

Parametric surfaces

Definition

Let t and s be parameters $t \in [a, b] \subset \mathbb{R}$ and $s \in [c, d] \subset \mathbb{R}$.

A parametric surface in 3D is given by a function $\gamma : [a, b] \times [c, d] \to \mathbb{R}^3$

$$\gamma(t, s) = (x(t, s), y(t, s), z(t, s)) \tag{3}$$

where $x,\ y,\ z : [a, b] \times [c, d] \to \mathbb{R}$.

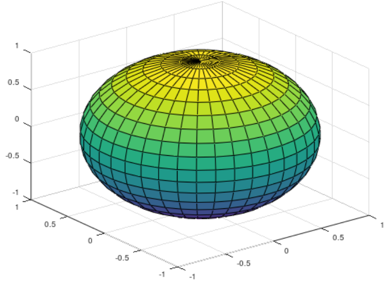
Example: unit sphere

$t \in [0, \pi]$, $s \in [0, 2\pi]$

define $\gamma : [0, \pi] \times [0, 2\pi] \to \mathbb{R}^3$ given as

$$\gamma(t, s) = (\cos(s)\sin(t), \sin(s)\sin(t), \cos(t))$$

graph of the surface:




```
%===== MAT300=================================
%
% Julia Sanchez Sanz
% julia.sanchez@digipen.edu
% 5/6/2019
%
% Parametric surface sphere
%=============================================
function sphere

t = linspace (0,pi,20); % mesh parameter t
s=linspace(0,2*pi,40); % mesh parameter s
[T, S] = meshgrid (t, s); % compute the grid

X=cos(S).*sin(T); % x-component
Y=sin(S).*sin(T); % y-component
Z=cos(T); % z-component

surf(X,Y,Z) % graph of the surface
end
```

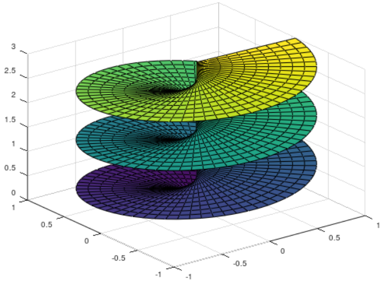
Example: helicoid

$t \in [0, 6\pi]$, $s \in [0, 1]$

define $\gamma : [0, 6\pi] \times [0, 1] \to \mathbb{R}^3$ given as

$$\gamma(t, s) = (s\cos(t), s\sin(t), \tfrac{t}{2\pi})$$

graph of the surface:




```
%==========================================
% MAT300
% Julia Sánchez Sanz
% julia.sanchez@digipen.edu
% 11-1-2018
% Parametric surfaces using Octave 3d
%==========================================

function helicoid
% obtain the graph of an helicoid surface

t=[0:0.1:6*pi]'; % parameter t
s=[0:0.1:1]; % parameter s

[T,S] = meshgrid (t,s); % create the grid

X1=S.*cos(T); % x-component
X2=S.*sin(T); % y-component
X3=T/(2*pi); % z-component

surf(X1,X2,X3) % graph of the curve
end
```

Why parametric surfaces?

- Same reasons than for curves.

We choose polynomials

Given a curve or a surface, which functions do we use for parametrizing them?

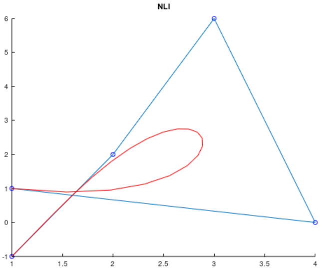
how can we simulate-approximate them?

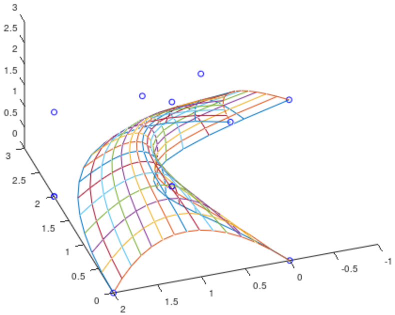
In this course we will use polynomial functions.

Why?

- C^∞ smooth.
- Easy to derive and integrate.
- Given $n+1$ points I can define an interpolant polynomial of degree at most n through them.
- I can use linear algebra of polynomial vector spaces.

Examples:


NLI
6
5
4
3
2
1
0
-1
1
1.5
2
2.5
3
3.5
4


3
2.5
2
1.5
1
0.5
0
3
2.5
2
1.5
1
0.5
0
2
1.5
1
0.5
0
-0.5
-1