Bjarki Geir Benediktsson, Haukur Óskar Þorgeirsson, Matthías Páll Gissurarson

# TÖLULEG GREINING HEIMAVERKEFNI 2 8. MARS 2013

## Töluleg Greining Heimaverkefni 2

Bjarki Geir Benediktsson, Haukur Óskar Þorgeirsson, Matthías Páll Gissurarson Kennari: Máni Maríus Viðarsson

8. mars 2013

### Inngangur

## 1 Feristeikning með splæsibrúun

#### 1.1 Teikning með aflestri af skjá

```
% Notar fallid splaesi.
       close all
       clear all
       % Setjum mynd upp
       a=-1; b=1; c=-1; d=1;
       axis([a b c d])
       % Lesum inn punkta
       hold on
10
11
       hnappur=1;
       x = []; y = [];
^{12}
13
       while hnappur==1
            [xtmp,ytmp,hnappur]=ginput(1);
15
            if hnappur==1
                x = [x, xtmp];
17
                y = [y, ytmp];
18
                plot(x,y, 'o')
19
            end
20
21
       end
       x=[x, x(1)];
22
23
       y = [y, y(1)];
       % Stikum ferilinn svo vid lendum ekki i veseni
24
25
       % ef x-hnitin eru ekki i staerdarrod
       n = length(x); t = 1:n; tt=linspace(1,n,100);
26
27
       % Reiknum og teiknum
       xx = splaesi(t, x, 4, 0, 0, tt);
29
       yy = splaesi(t, y, 4, 0, 0, tt);
30
31
       plot(xx,yy)
32
```

#### 1.2 Teikning á lokuðum ferlum með aflestri af skjá

#### 1.2.1 Teikning á þvinguðum ferlum með aflestri af skjá

```
% Notar fallid splaesi.
2
       close all
       clear all
3
       % Setjum mynd upp
       a=-1; b=1; c=-1; d=1;
6
       axis([a b c d])
       % Lesum inn punkta
       hold on
10
11
       hnappur=1;
       x = []; y = [];
12
13
       while hnappur==1
            [xtmp,ytmp,hnappur]=ginput(1);
15
            if hnappur==1
16
                x = [x, xtmp];
17
                y = [y, ytmp];
18
19
                plot(x,y,'o')
            end
20
21
22
       df1x = (x(1)-x(end-1));
23
       df1y = (y(1)-y(end-1));
24
       df2x = (x(end) - x(end-2));

df2y = (y(end) - y(end-2));
25
       plot([x(1) x(end-1)], [y(1) y(end-1)], 'r')
27
       plot([x(end-2) x(end)], [y(end-2) y(end)], 'r')
28
       x = x(1:end-2);
29
       y = y(1:end-2);
30
31
       % Stikum ferilinn svo vid lendum ekki i veseni
32
       % ef x-hnitin eru ekki i staerdarrod
       n = length(x); t = 1:n; tt=linspace(1,n,100);
34
35
       % Reiknum og teiknum
36
       %Tharf ad reikna ut c0 og cn
37
       xx = splaesi(t, x, 2, df1x, df2x, tt);
       yy = splaesi(t, y, 2, df1y, df2y, tt);
39
40
       plot(xx,yy)
41
```

#### 1.3 Ferilteikning með Bezier-splæsibrúun

```
% Notar fallid splaesi.
       close all
2
       clear all
3
       % Setjum mynd upp
6
       a=-1; b=1; c=-1; d=1;
       axis([a b c d])
       % Lesum inn punkta
       hold on
10
       hnappur=1;
11
12
       x = []; y = [];
       n=4;
13
       t=0:1;
       tt=linspace(0,1,100);
15
       xxx = []; %Hehe
yyy = [];
16
17
18
       while hnappur==1
19
            [xtmp,ytmp,hnappur]=ginput(1);
20
            if hnappur==1
```

```
x = [x, xtmp];
22
23
                y = [y, ytmp];
                plot(x,y,'o')
24
            end
            if hnappur=1&length(x) == 4
26
                df1x = (x(2)-x(1));
27
                df1y = (y(2)-y(1));
28
                df2x = (x(4) - x(3));
29
                df2y = (y(4) - y(3));
                plot([x(2) x(1)], [y(2) y(1)], 'r')
31
                plot([x(4) x(3)], [y(4) y(3)], 'r')
32
33
                xx = baz(x,tt);
                xxx = [xxx xx];
34
                yy = baz(y,tt);
35
36
                yyy = [yyy yy];
37
                plot(xx,yy)
38
                x = [x(4) (2*x(4)-x(3))];
39
40
                y = [y(4) (2*y(4)-y(3))];
            end
41
42
       end
43
44
45
       close all
        a=-1; b=1; c=-1; d=1;
46
47
        axis([a b c d])
48
        % Lesum inn punkta
49
       hold on
50
       plot(xxx,yyy)
```

Fallið baz þjónar sama tilgangi fyrir Bezier brúunina og splaesi gerði fyrir splæsibrúunina það tekur við lista af fjórum hnitum  $(x_i)_{i=0}^3$  og lista  $(t_i)_{i=1}^n$  af gildum á [0,1] og skilar lista af gildum  $(r(t_i))_{i=1}^n$  þar sem r

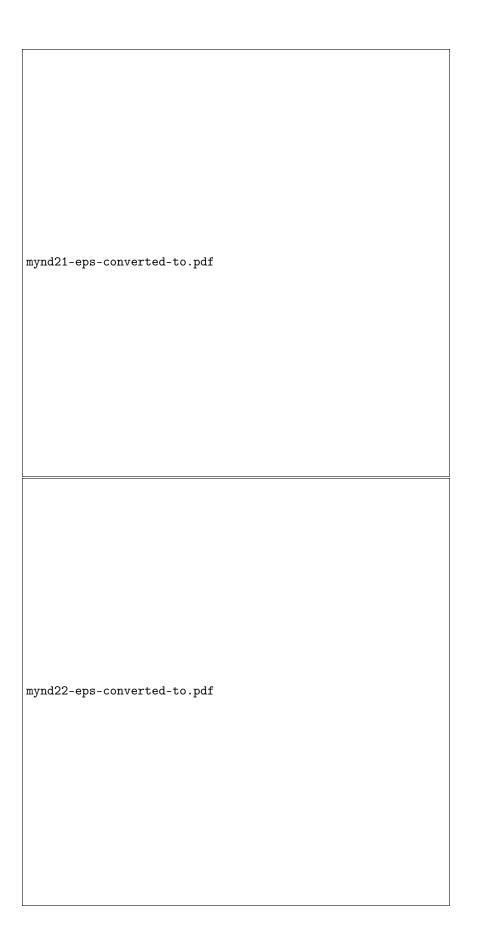
$$r(t) = (1-t)^3 x_0 + 3(1-t)^2 t x_1 + 3(1-t)t^2 x_2 + t^3 x_3$$

nú fæst að með því að gefa baz y-hnit í stað x-hnita sem inntak skilar það lista af s(t) gildum þar sem

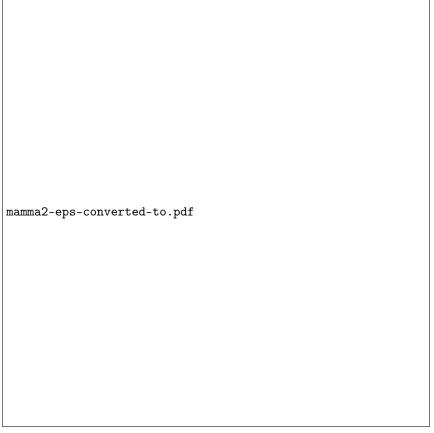
$$s(t) = (1-t)^3 y_0 + 3(1-t)^2 t y_1 + 3(1-t)t^2 y_2 + t^3 y_3$$

en það eru þau gildi sem við þurfum til að geta framkvæmt Bezier brúunina

```
1 function yy = baz(y, tt)
2    for i=1:length(tt)
3         t=tt(i);
4         yy(i)=(1-t)^3*y(1)+3*(1-t)^2*t*y(2)+3*(1-t)*t^2*y(3)+t^3*y(4);
5     end
6 end
```



mamma1-eps-converted-to.pdf					



hér mætti leyfa notenda að rjúfa ferilin meðþað er aðvera með punkta sem ekki væri teiknað á milli Auk þess væri mjög sniðugt að geta hreyft punktana eftir aðþeir hafa verið settir inn til aðlaga ferilin til

## 2 Nálgun á afleiðum, heildu, stiglum og Hessefylkjum

### 2.1 Almenn útgiskun

[input encoding = utf8] extrapolation.m

#### 2.2 Richardson

```
1  function r = R(f,a,h)
2     r = (f(a+h) + f(a-h))/(2*h);
3  end
4
5     >>[X,mat1,mat2]=extrapolation(@(x)sin(x), 0, 0.5, 100, 1e-10)
6
7     X = 1
8
9  mat1 = 2.6201e-012
10
11  mat2 = 1.0235e-014
```

```
1  >> [X,mat1,mat2]=richardson(@(x)sin(x), 0, 0.5, 100, 1e-10)
2
3  X = 1
4
```

```
5 mat1 = 2.6201e-012
6
7 mat2 = 1.0235e-014
```

```
1  function r = R(f,a,h)
2  r = (f(a+h) + f(a-h) - 2*f(a))/(h*h);
3  end
4
5  >> [X,mat1,mat2] = extrapolation(@(x)sin(x), 0, 0.5, 100, 1e-10)
6
7  X = 0
8
9  mat1 = 0
10
11  mat2 = 0
```

## 2.3 Romberg

## 2.4 Nálgun á stiglum

## 2.5 Nálgun á Hessefylkjum

Að ský	rsluni unnu :	 	