Initiation au logiciel Matlab

- Faire des simulations numériques
- Représenter graphiquement les résultats
- Sera utilisé en cycle préparatoire (projets) et en cycle ingénieur
- Attention, Matlab n’est pas un logiciel libre : le logiciel libre Scilab est très voisin

Calendrier des séances de TP

Quand ?

- Groupe 1 les mardis 7 et 14, 9h à 12h
- Groupe 2 les mercredis 8 et 15, 14h à 17h
- Groupe 3 les lundis 6 et 13, 9h à 12h

Où ?

A la Maison de l’Ingénieur, salle SUN (rez-de-chaussée)

Matlab comme calculette graphique

```
>> x=0 : pi/50 : 2*pi ;   plot(sin(3 * x),sin(2 * x) / 2)
```

* pour la multiplication, / pour la division

() pour évaluer une fonction

: sert à définir des vecteurs-lignes dont les éléments sont uniformément espacés

; empêche l'affichage du résultat

plot trace une ligne brisée

>> x+sin(x), >> x+2

sont acceptés

>> x*x

>> x^3

>> 2^x

>> 1/x

Erreurs !

Matrices

Dans Matlab, uniquement des tableaux rectangulaires de chiffres appelés matrices.

>> A=[-2 1 0]

retourne une matrice à une ligne=vecteur-ligne

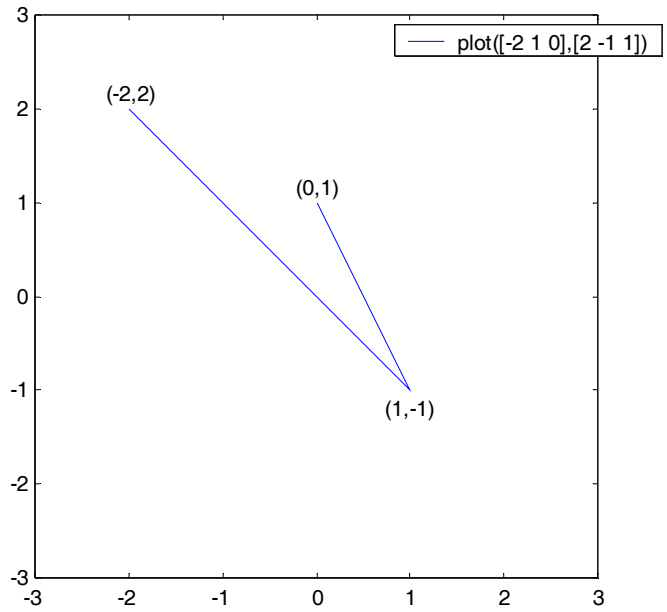
>> B=[1;4;3]

retourne une matrice à une colonne= vecteur-colonne

>> C=[-2 1 0;2 5 6]

retourne une matrice à 2 lignes et 3 colonnes.

plot prend deux vecteurs-lignes x et y de même longueur et retourne une figure, ligne brisée joignant les points de coordonnées [x_i y_i]



Opérations sur les matrices

Les fonctions préprogrammées comme sin, cos, tan, exp, log, abs, sqrt s'appliquent à chaque coefficient d'une matrice.

>> x.*x >> x.^3 >> 2.^x >> 1./x

effectuent aussi les opérations souhaitées élément par élément.

>> a:b:c

retourne les termes de la progression arithmétique de raison b, de valeur initiale a, qui sont inférieurs ou égaux à c.

>> a:b

est une abréviation de a:1:b.

>> x(2) >> x(end) >> x(2:4)

pour récupérer des éléments d'un vecteur (ligne ou colonne).

>> x'

est la matrice transposée (échange lignes et colonnes).

>> zeros(1,4) >> ones(3,1)

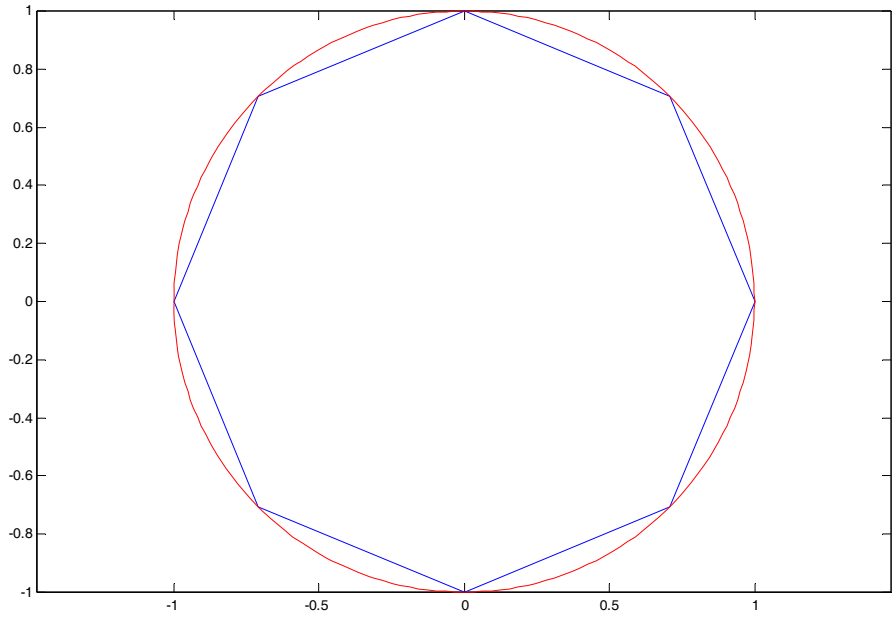
sont des matrices pleines de 0 ou de 1.

Exemple : demi-périmètre d'un polygone régulier à 2^n côtés

$u_n = 2^n \sin(\pi/(2^n))$, $n>0$

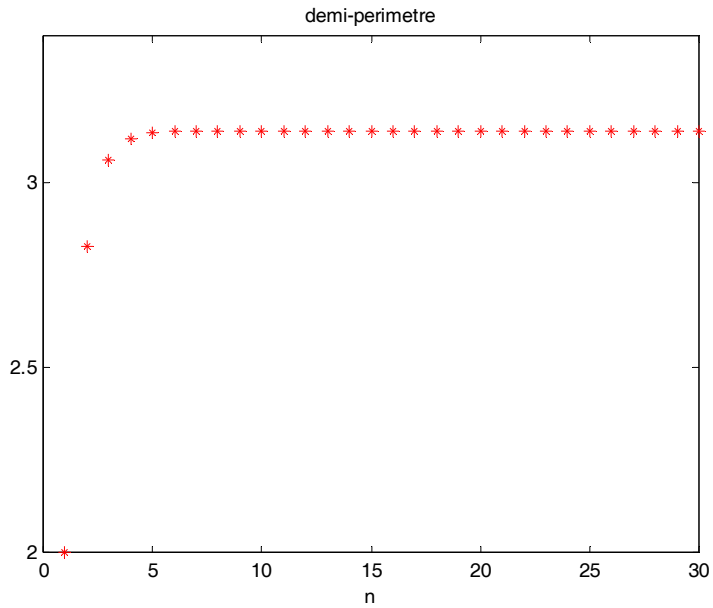
Pour construire un vecteur qui contient les 100 premiers termes de la suite,

```
>> n=1:30;u=(2.^n).*sin(pi./2.^n)
>> format long e
>> u-pi
>> u(27:end)
```



Représentation graphique d'une suite

>> plot(n,u,'r*');xlabel('n');title('demi-périmètre…')
retourne



Pour davantage d'options graphiques, interroger help plot

Créer de nouvelles fonctions

```
>> f=inline('1./(t.^2 +1)')
>> f(0:5)
```

Autre moyen, mieux adapté à des fonctions plus compliquées (sous-programmes), on lance un éditeur

```
>> edit
```

qui ouvre un fichier dans lequel on tape

```
function z=mafonction(t)
z= 1./(t.^2 +1) ;
```

et on sauvegarde le fichier sous le nom mafonction.m (obligatoire)

```
>>mafonction(3)
```

Attention, mafonction ne passe pas toujours à travers les vecteurs.

Branchement conditionnel

Exemple : redéfinir la fonction valeur absolue

```
% valeur absolue faite maison

function y=absolue(x)
if x>=0
y=x;
else
y=-x;
end
```

Attention: à la différence de la fonction préprogrammée abs, cette fonction ne passe pas à travers les vecteurs.

Autres conditions ==, <, >, <=, >=, or ~= (interroger help if).

Suites récurrentes

On définit une suite u_n par la valeur de u_0 et une relation de récurrence de la forme $u_{n+1}=f(u_n)$.

Exemple: $u_{n+1}=au_n+b$.

Alors, $u_n=c+a^n(u_0-c)$ où c est la solution de $c=ac+b$.

$a=-1$, $b=2$, $u_0=2$ donne $u_n=1+(-1)^n$.

```
>> n=0:10;u=1+(-1).^n
```

En général, pas de formule simple.

Exemple: $u_{n+1}=1+(1/u_n)$.

Alors $u_1=1+(1/u_o)$, $u_2=1+(1/u_1)=1+1/1+1/u_o$…

Facile à programmer par une boucle.

Boucle

```
% suite u_n+1 =f(u_n)

function u=marecurrence(n,uo)
u=zeroes(1,n);
u(1)=f(uo);
for i=1:n-1
   u(i+1)=f(u(i));
end
u
```

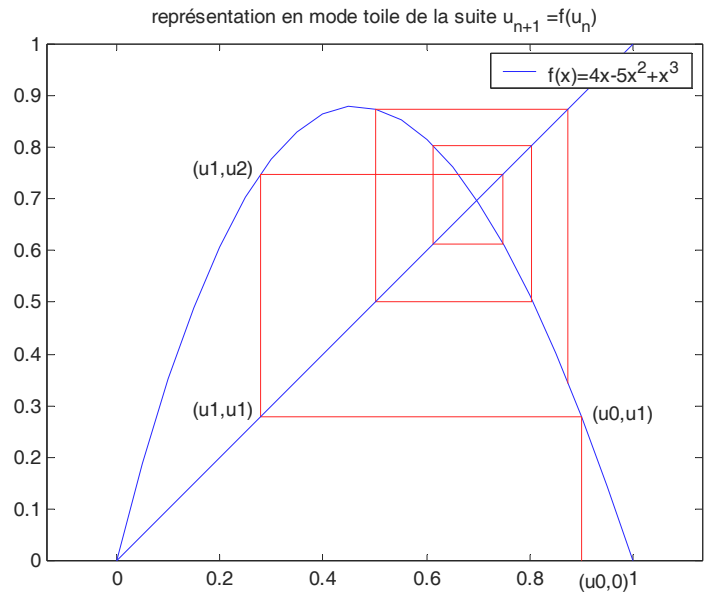
Attention : la boucle retourne le vecteur $[u_1\ u_2\ \ldots\ u_n]$ sans u_0
Un vecteur est toujours indexé à partir de 1.

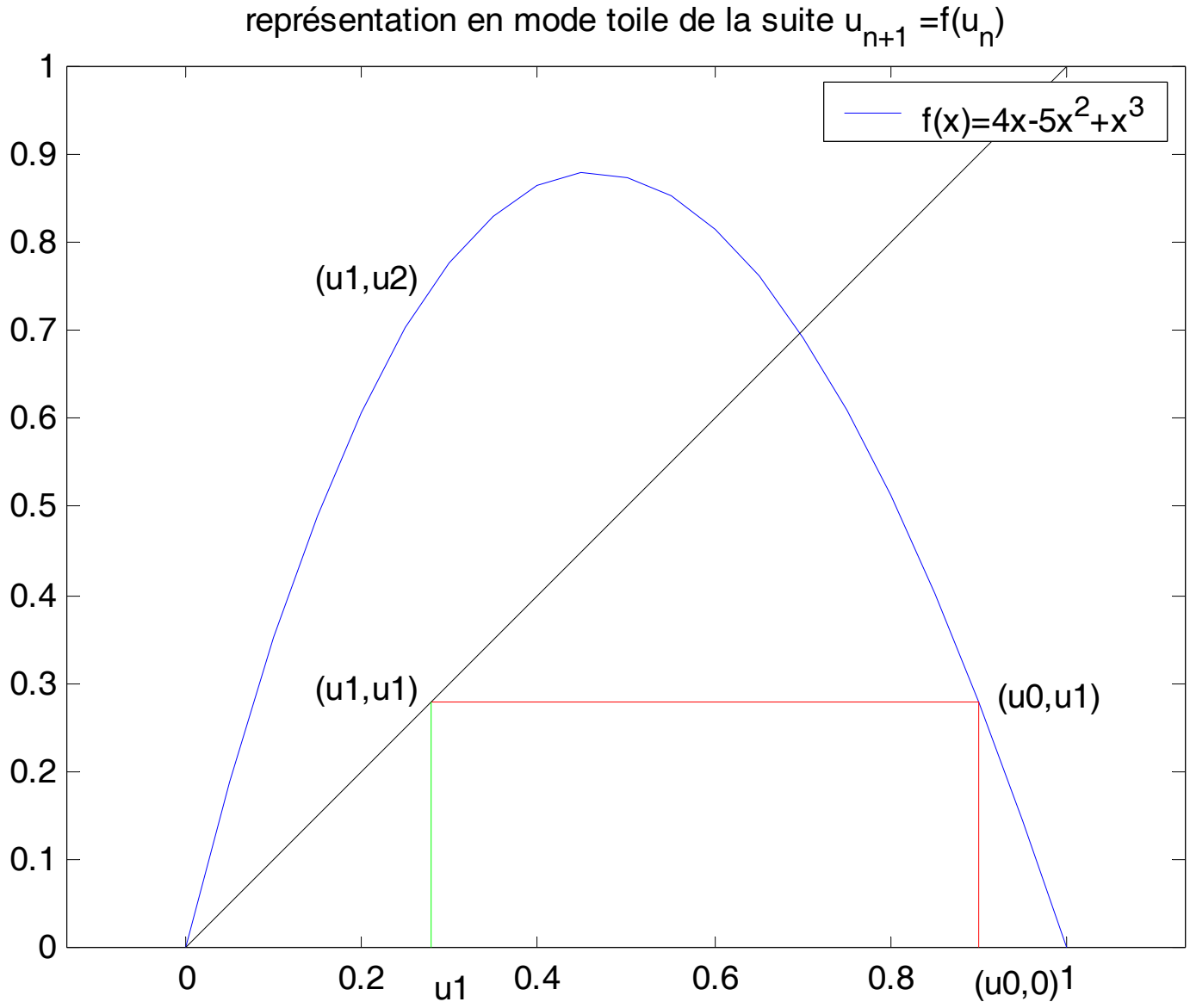
Représentation en mode toile

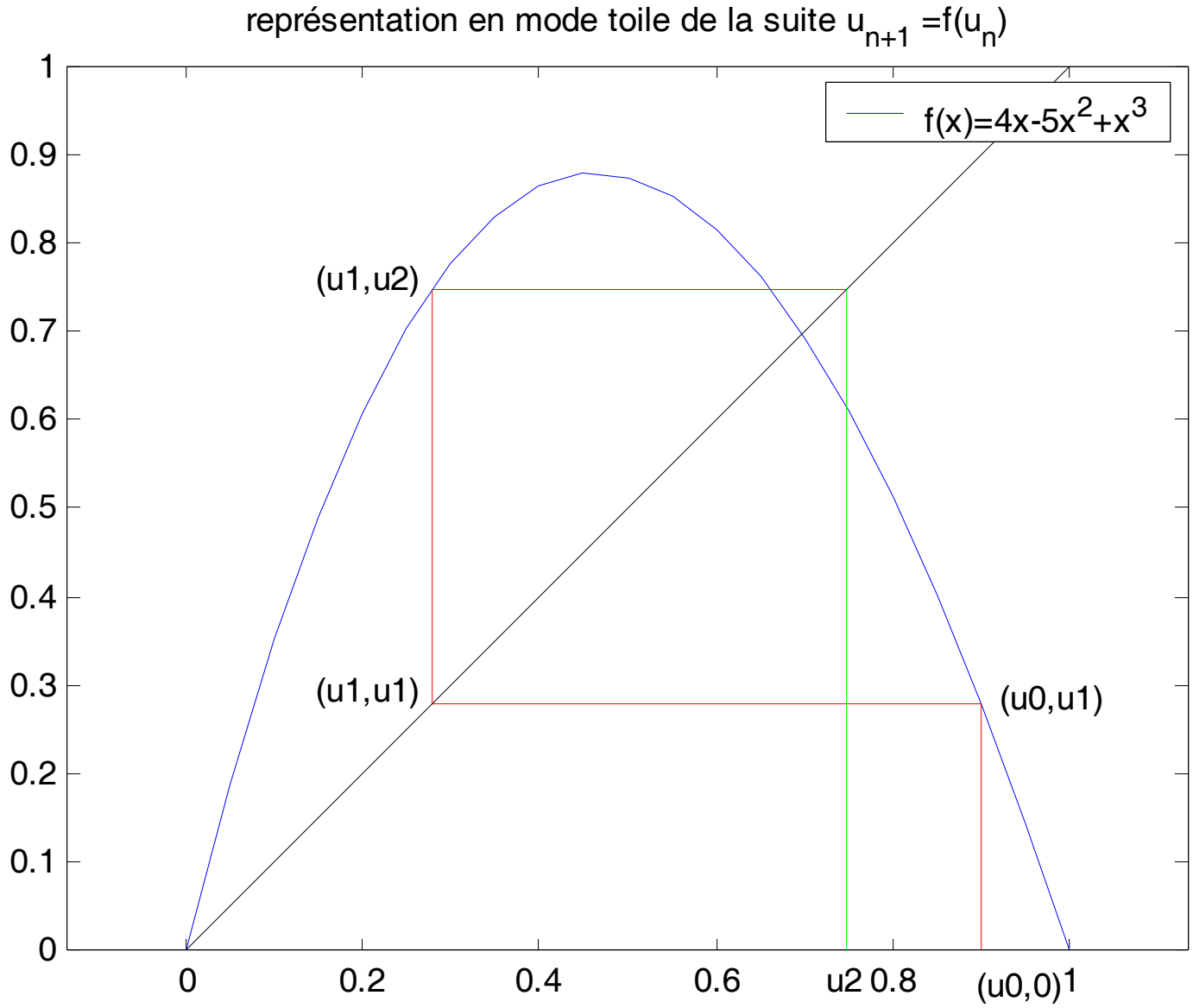
On voit mieux si une suite est convergente, bascule entre deux valeurs, etc… à l'aide de la représentation en mode toile.

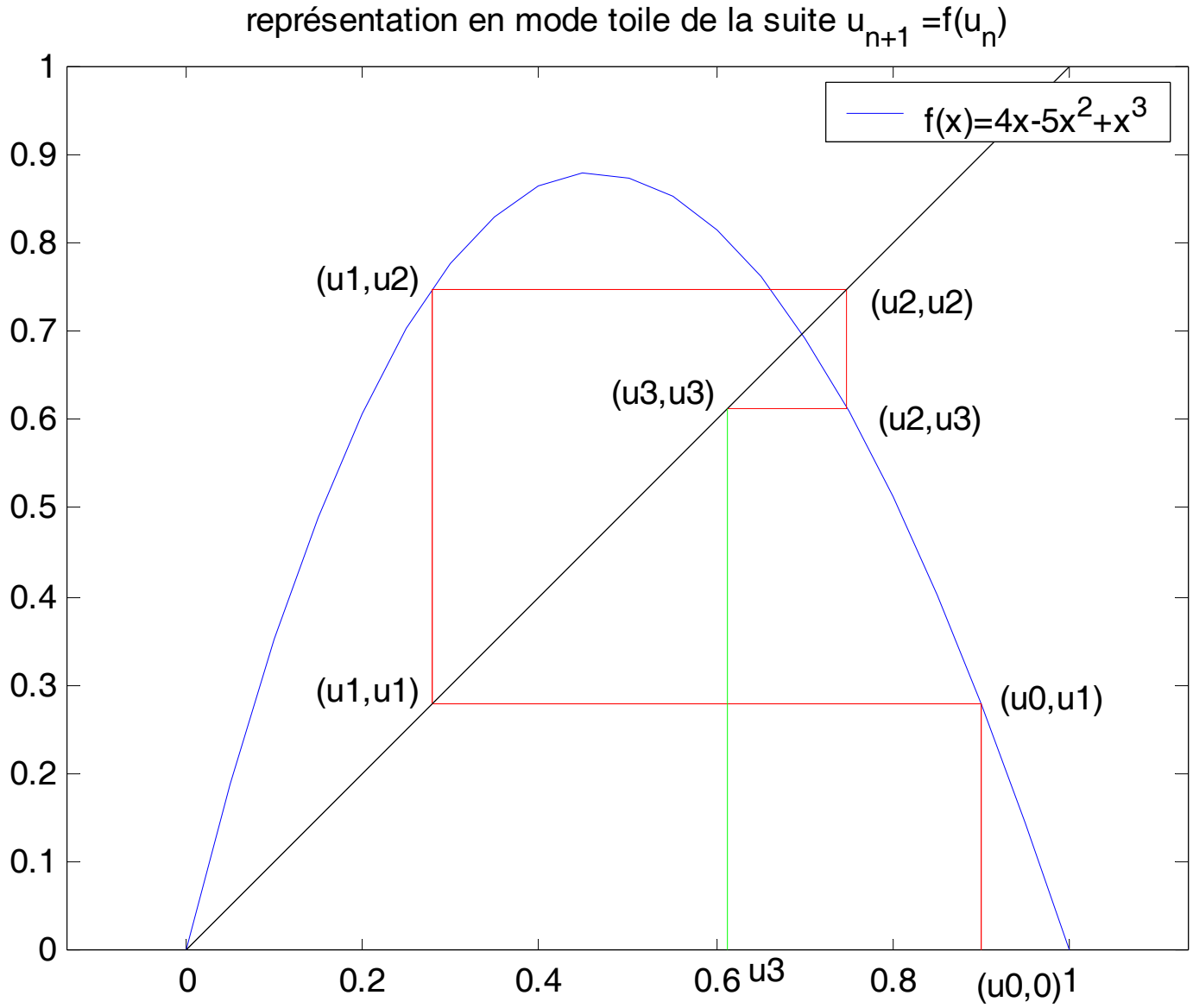
On place successivement les points $[u_0\ 0]$, $[u_0\ u_1]$, $[u_1\ u_1]$, $[u_1\ u_2]$, $[u_2\ u_2]$, $[u_2\ u_3]$,…

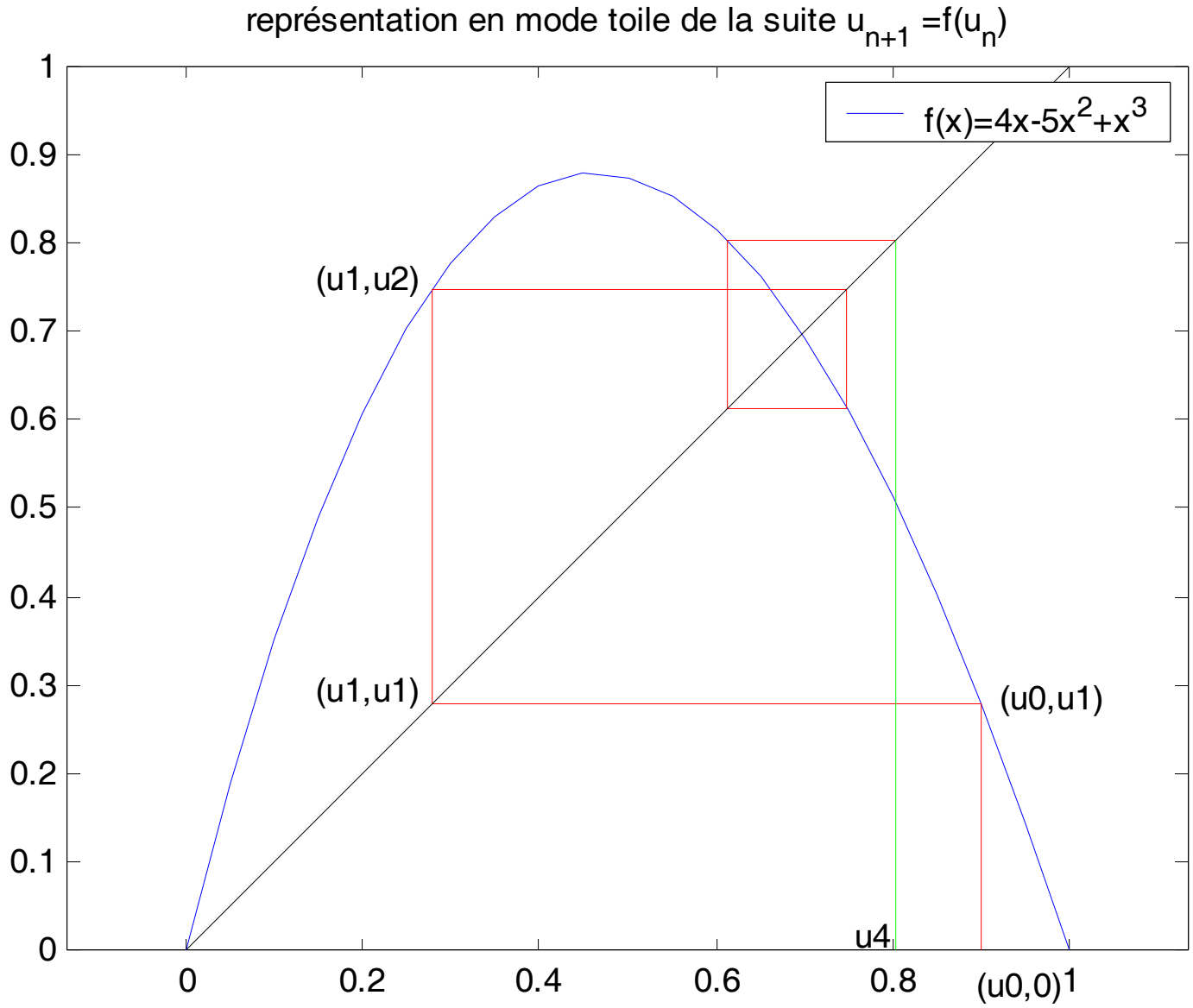
On saute de la courbe à la première bissectrice.

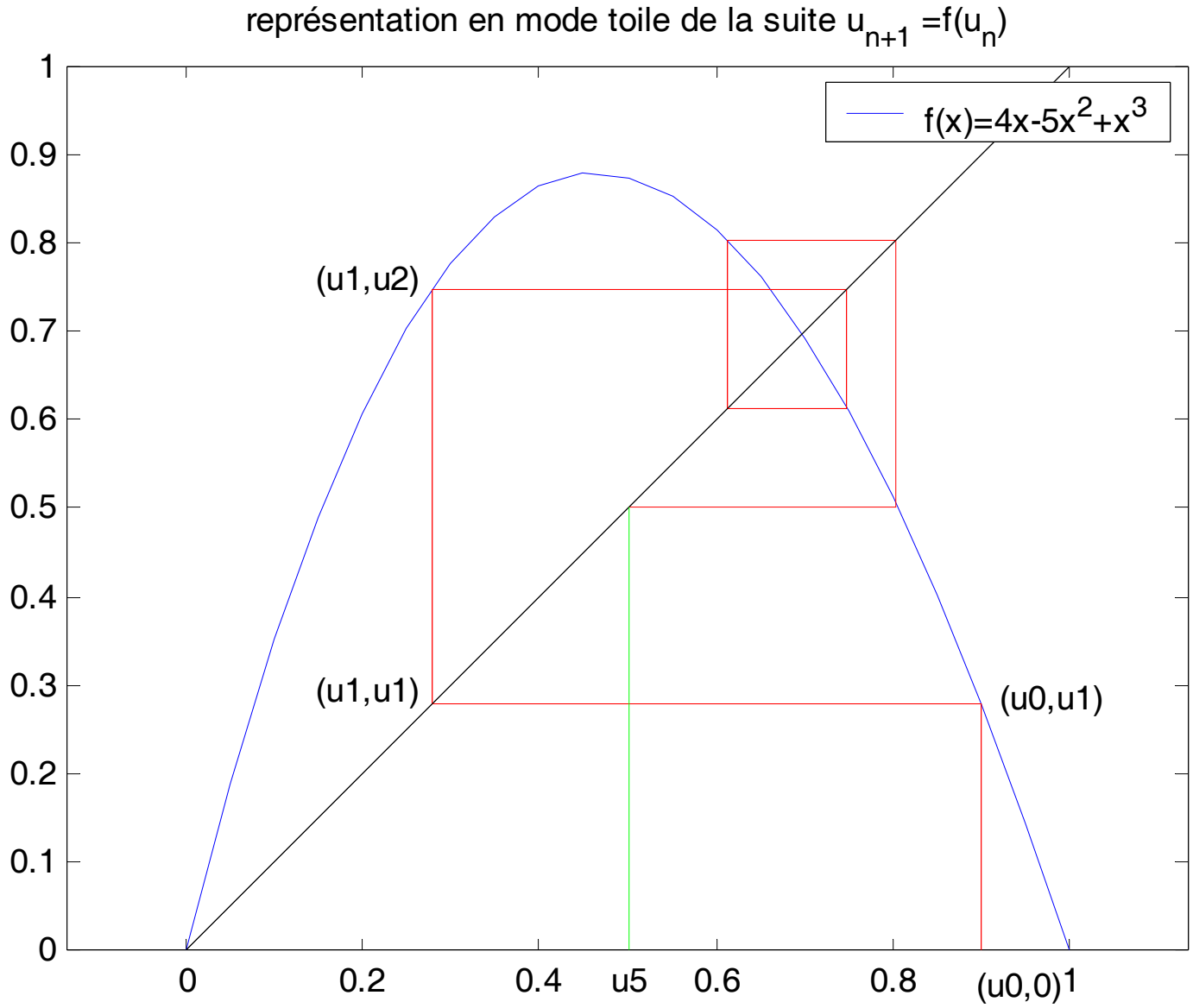


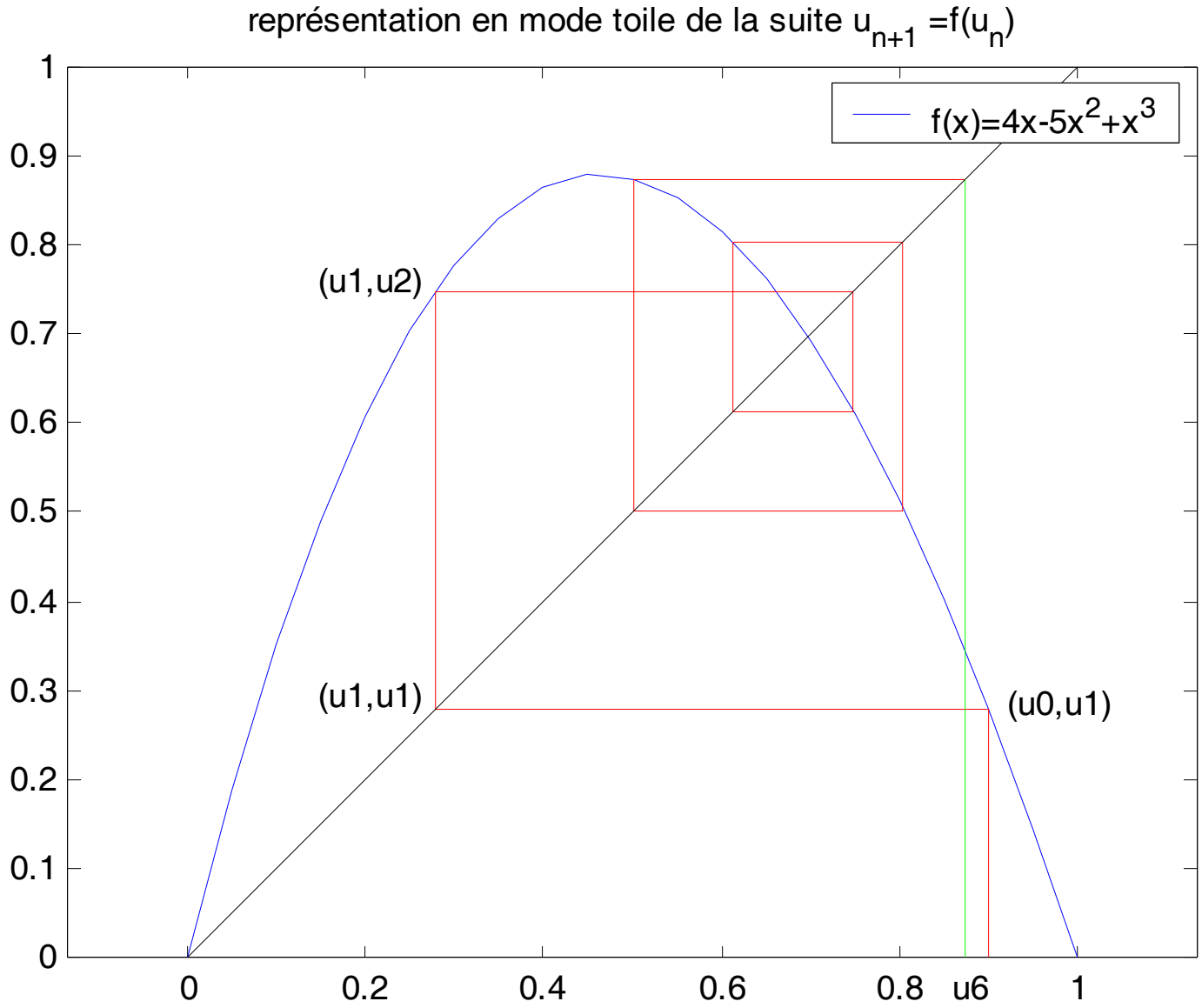

représentation en mode toile de la suite $u_{n+1} = f(u_n)$
f(x)=4x-5x^2+x^3
(u1,u2)
(u1,u1)
(u0,u1)
(u0,0)
u1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

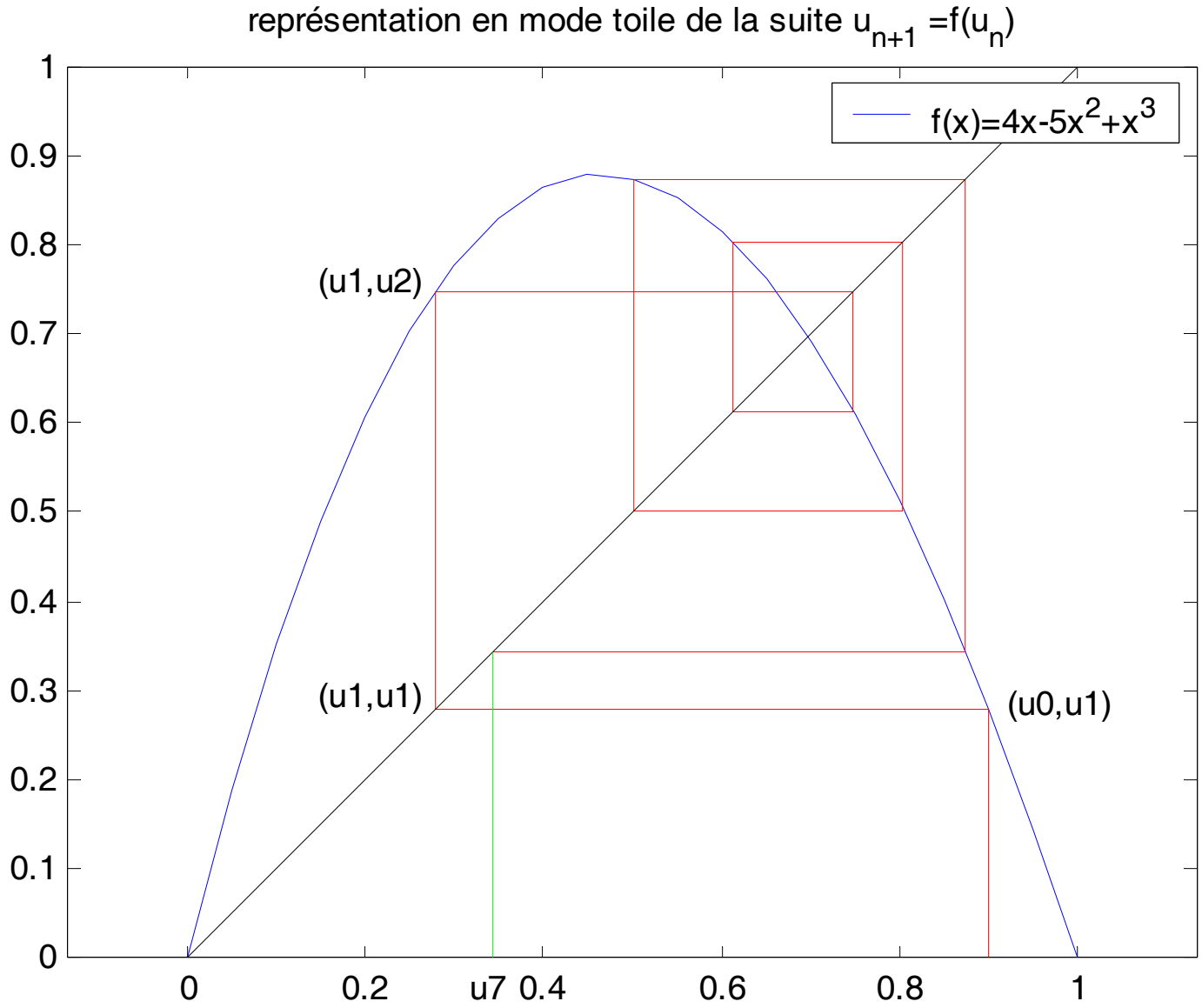

représentation en mode toile de la suite $u_{n+1} = f(u_n)$
f(x)=4x-5x^2+x^3
(u1,u2)
(u1,u1)
(u0,u1)
(u0,0)
u2
0
0.2
0.4
0.6
0.8
1
0.9
0.7
0.5
0.3
0.1

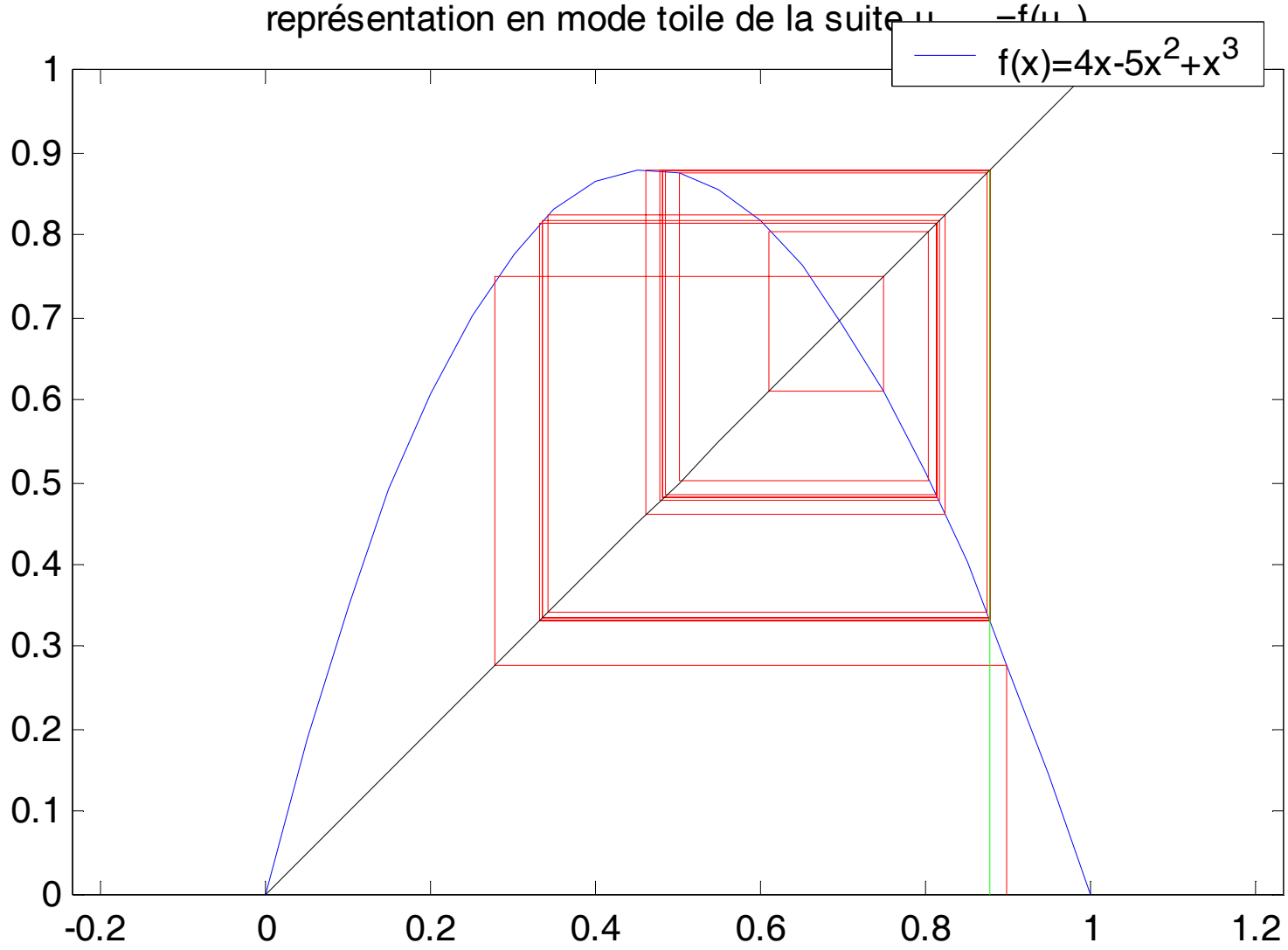

représentation en mode toile de la suite $u_{n+1} = f(u_n)$
f(x)=4x-5x^2+x^3
(u1,u2)
(u2,u2)
(u3,u3)
(u2,u3)
(u1,u1)
(u0,u1)
(u0,0)
u3
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.2
0.4
0.6
0.8
1


représentation en mode toile de la suite $u_{n+1} = f(u_n)$
f(x)=4x-5x^2+x^3
(u1,u2)
(u1,u1)
(u0,u1)
u4
(u0,0)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


représentation en mode toile de la suite $u_{n+1} = f(u_n)$
f(x)=4x-5x^2+x^3
(u1,u2)
(u1,u1)
(u0,u1)
(u0,0)
u5
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


représentation en mode toile de la suite $u_{n+1} = f(u_n)$
f(x)=4x-5x^2+x^3
(u1,u2)
(u1,u1)
(u0,u1)
u6
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.2
0.4
0.6
0.8
1


représentation en mode toile de la suite $u_{n+1} = f(u_n)$
f(x)=4x-5x^2+x^3
(u1,u2)
(u1,u1)
(u0,u1)
u7
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.2
0.4
0.6
0.8
1


représentation en mode toile de la suite u
f(x)=4x-5x2+x3
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-0.2
0
0.2
0.4
0.6
0.8
1
1.2

Film

>> toile_film(0.9,30)

Interroger help movie