Calcul formel

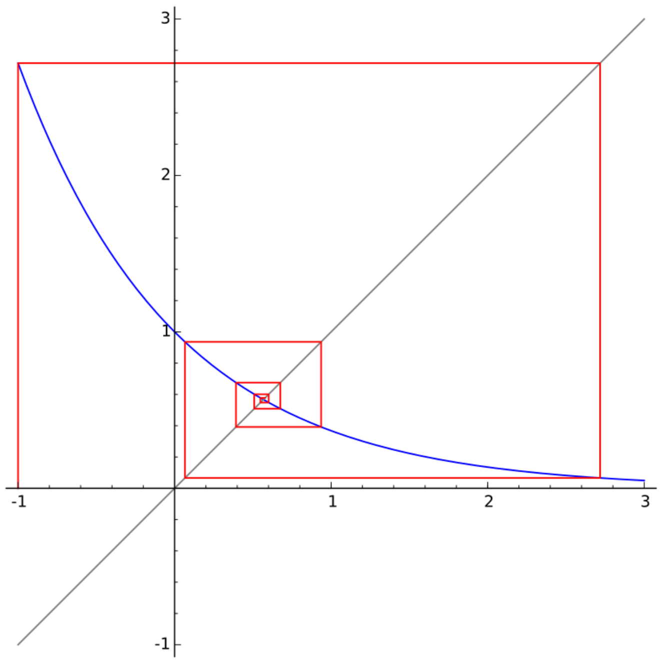
4. Suites récurrentes et visualisation

4.1. Visualiser une suite récurrente

Travaux pratiques 1.
Fixons $a \in \mathbb{R}$. Définissons une suite $(u_n)_{n\in\mathbb{N}}$ par récurrence :

$$u_0 = a \qquad \text{et} \qquad u_{n+1} = \exp(-u_n) \quad \text{pour } n \geqslant 0.$$

1. Calculer les premiers valeurs de la suite pour $a = -1$. Écrire une fonction qui renvoie ces premières valeurs sous la forme d'une liste.
2. Sur un même graphique tracer le graphe de la fonction de récurrence $f(x) = \exp(-x)$, la première bissectrice $(y = x)$ et la trace de la suite récurrente, c'est-à-dire la ligne brisée joignant les points $\big(u_k, f(u_k)\big)$, (u_{k+1}, u_{k+1}) et $\big(u_{k+1}, f(u_{k+1})\big)$.
3. Émettre plusieurs conjectures : la suite ou certaines sous-suites sont-elles croissantes ou décroissantes ? Majorées ? Minorées ? Convergentes ?
4. Prouver vos conjectures.



1. On définit la fonction $f(x) = \exp(-x)$ par la commande `f(x) = exp(-x)`.

Code 1 (*suites-visual.sage (1)*).

```
def liste_suite(f,terme_init,n):
    maliste = []
    x = terme_init
    for k in range(n):
        maliste.append(x)
        x = f(x)
    return maliste
```

Par exemple, la commande `liste_suite(f,-1,4)` calcule, pour $a = -1$, les 4 premiers termes de la suite (u_n) ; on obtient la liste :

```
[-1, e, e^(-e), e^(-e^(-e))]
```

correspondant aux termes : $u_0 = -1$, $u_1 = e$, $u_2 = e^{-e}$ et $u_3 = e^{-e^{-e}}$, où l'on note $e = \exp(1)$.

2. Nous allons maintenant calculer une liste de points. On démarre du point initial $(u_0, 0)$, puis pour chaque rang on calcule deux points : $\big(u_k, f(u_k)\big)$ et $\big(u_{k+1}, u_{k+1}\big)$. Notez que chaque élément de la liste est ici un couple (x, y) de coordonnées.

Code 2 (*suites-visual.sage (2)*).

```
def liste_points(f,terme_init,n):
    u = liste_suite(f,terme_init,n)
    mespoints = [ (u[0],0) ]
    for k in range(n-1):
        mespoints.append( (u[k],u[k+1]) )
        mespoints.append( (u[k+1],u[k+1]) )
    return mespoints
```

Par exemple, la commande `liste_points(f,-1,3)` calcule, pour $a = -1$, le point initial $(-1, 0)$ et les 4 premiers points de la visualisation de la suite (u_n) par une ligne brisée ; on obtient la liste :

```
[(-1, 0), (-1, e), (e, e), (e, e^(-e)), (e^(-e), e^(-e))]
```

et il ne reste plus qu'à tracer le graphe des objets demandés.

Code 3 (*suites-visual.sage (3)*).

```
def dessine_suite(f,terme_init,n):
    mespoints = liste_points(f,terme_init,n)
    G = plot(f,(x,-1,3))       # La fonction
    G = G + plot(x,(x,-1,3))  # La droite (y=x)
    G = G + line(mespoints)   # La suite
    G.show()
```

Par exemple, la figure illustrant ce tp est construite par la commande `dessine_suite(f,-1,10)`.

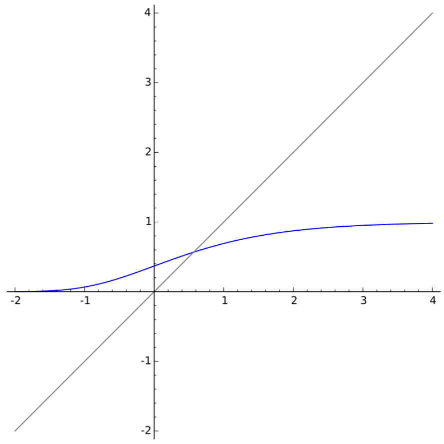
3. (et 4.) Passons à la partie mathématique. Pour simplifier l'étude, nous allons supposer $a = -1$. Donc $u_0 = -1$ et $u_1 = f(u_0) = \exp(1) = e$.
   (a) La fonction f définie par $f(x) = \exp(-x)$ est décroissante. La fonction g définie par $g(x) = f\big(f(x)\big)$ est donc croissante.
   (b) La suite $(u_{2n})_{n\in\mathbb{N}}$ des termes de rangs pairs est croissante. En effet : $u_2 = e^{-e} \geqslant -1 = u_0$, puis $u_4 = f \circ f(u_2) \geqslant f \circ f(u_0) = u_2$ car $f \circ f = g$ est croissante. Par récurrence $u_6 = f \circ f(u_4) \geqslant f \circ f(u_2) = u_4$... La suite (u_{2n}) est croissante.
   (c) Comme $u_2 \geqslant u_0$ et que f est décroissante alors $u_3 \leqslant u_1$. On démontre alors de la même façon que la suite $(u_{2n+1})_{n\in\mathbb{N}}$ des termes de rangs impairs est décroissante.
   (d) Enfin, toujours par la même méthode et en partant de $u_0 \leqslant u_1$, on prouve que $u_{2n} \leqslant u_{2n+1}$.

(e) La situation est donc la suivante :

$$u_0 \leqslant u_2 \leqslant \cdots \leqslant u_{2n} \leqslant \cdots \leqslant u_{2n+1} \leqslant \cdots \leqslant u_3 \leqslant u_1$$

La suite (u_{2n}) est croissante et majorée par u_1, donc elle converge. Notons ℓ sa limite.
La suite (u_{2n+1}) est décroissante et minorée par u_0, donc elle converge. Notons ℓ' sa limite.

(f) Par les théorèmes usuels d'analyse, la suite (u_{2n}) converge vers un point fixe de la fonction $g = f \circ f$, c'est-à-dire une valeur x_0 vérifiant $f \circ f(x_0) = x_0$. Une étude de la fonction g (ou plus précisément de $g(x) - x$) montrerait que g n'a qu'un seul point fixe. Voici le graphe de g.



Cela prouve en particulier que $\ell = \ell'$.

(g) Par ailleurs la fonction f admet un unique point fixe x_0. Mais comme $f(x_0) = x_0$ alors l'égalité $f\big(f(x_0)\big) = f\big(x_0\big) = x_0$ prouve que x_0 est aussi le point fixe de g.

(h) Conclusion : la suite (u_{2n}) et la suite (u_{2n+1}) convergent vers $x_0 = \ell = \ell'$. Ainsi la suite (u_n) converge vers x_0 le point fixe de f.

(i) Il n'y a pas d'expression simple de la solution x_0 de l'équation $f(x) = x$. La commande `solve(f(x)==x,x)` renvoie seulement l'équation. Par contre, on obtient une valeur numérique approchée par l'instruction `find_root(f(x)==x,0,1)`. On trouve $x_0 = 0,56714329041\ldots$

4.2. Listes

Une *liste* est ce qui ressemble le plus à une suite mathématique. Une liste est une suite de nombres, de points... ou même de listes !

- Une liste est présentée entre crochets : `mesprems = [2,3,5,7,11,13]`. On accède aux valeurs par `mesprems[i]` ; `mesprems[0]` vaut 2, `mesprems[1]` vaut 3...
- On parcourt les éléments d'une liste avec `for p in mesprems:`, p vaut alors successivement 2, 3, 5, 7...
- Une première façon de créer une liste est de partir de la liste vide, qui s'écrit `[]`, puis d'ajouter un à un des éléments à l'aide de la méthode `append`. Par exemple `mespoints=[]`, puis `mespoints.append((2,3))`, `mespoints.append((7,-1))`. La liste `mespoints` contient alors deux éléments (c'est-à-dire deux points) `[ (2,3), (7,-1) ]`.

Travaux pratiques 2.
Pour un entier n fixé. Construire :

1. la liste des entiers de 0 à $n - 1$,
2. la liste des entiers premiers strictement inférieurs à n,
3. la liste des $2p + 1$, pour les premiers p strictement inférieurs à n,
4. les 10 premiers éléments de la liste précédente,
5. la liste de $p_i + i$, où $(p_i)_{i \geqslant 0}$ sont les nombres premiers strictement inférieurs à n ($p_0 = 2$, $p_1 = 3$,...),
6. la liste de 1 et 0 selon que le rang $0 \leqslant k < n$ est premier ou non.

Une utilisation intelligente permet un code très court et très lisible. Il faut consulter le manuel pour apprendre à manipuler les listes avec aisance.

1. `entiers = range(n)`
2. `premiers = [k for k in range(n) if is_prime(k)]`
3. `doubles = [2*p+1 for p in premiers]`
4. `debut = doubles[0:10]`
5. `premiersplusrang = [premiers[i]+i for i in range(len(premiers))]`
6. `binaire = [zeroun(k) for k in range(n)]` où `zeroun(k)` est une fonction à définir qui renvoie 1 ou 0 selon que k est premier ou pas.

On peut aussi trier les listes, les fusionner...

4.3. Suites et chaos

Travaux pratiques 3.
On considère la fonction f définie sur $[0,1]$ par

$$f(x) = rx(1-x)$$

où r est un réel compris entre 0 et 4.
Pour r fixé et $u_0 \in [0,1]$ fixé, on définit une suite (u_n) par

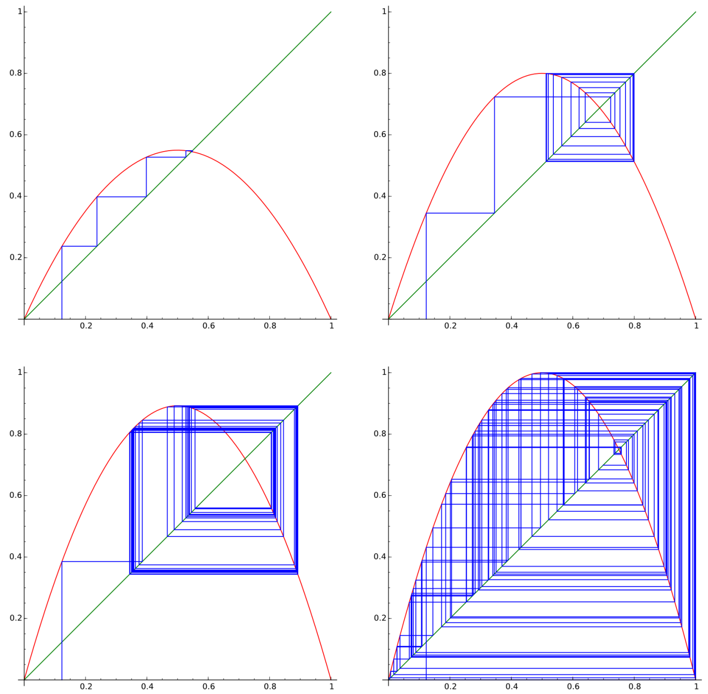
$$u_{n+1} = f(u_n).$$

1. Pour différentes valeurs de r et u_0 choisies, tracer sur un même graphique le graphe de f, la droite d'équation $(y = x)$ et quelques termes de la suite (u_n). Essayez de conjecturer le comportement de la suite pour $0 < r \leqslant 3$, $3 < r \leqslant 1+\sqrt{6}$, $1+\sqrt{6} < r < 4$ et $r = 4$.
2. Pour u_0 fixé et $M < N$ grands (par exemple $M = 100$, $N = 200$), tracer le ***diagramme de bifurcation*** de f, c'est-à-dire l'ensemble des points

$$(u_i, r) \qquad \text{pour} \qquad M \leqslant i \leqslant N \text{ et } 0 \leqslant r \leqslant 4$$

3. (a) Montrer que les points fixes de f sont 0 et $\frac{r-1}{r}$.
   (b) Montrer que le point fixe 0 est répulsif (c'est-à-dire $|f'(0)| > 1$) dès que $r > 1$.
   (c) Montrer que le point fixe $\frac{r-1}{r}$ est attractif (c'est-à-dire $|f'(\frac{r-1}{r})| < 1$) si et seulement si $1 < r < 3$.
4. **Cas** $1 < r \leqslant 3$**. Pour** $0 < u_0 < 1$ **la suite** (u_n) **converge vers** $\frac{r-1}{r}$**.**
   Nous allons prouver ce résultat dans le cas particulier $r = 2$:
   (a) Montrer que $u_{n+1} - \frac{1}{2} = -2\left(u_n - \frac{1}{2}\right)^2$.
   (b) Montrer que si $|u_0 - \frac{1}{2}| < k < \frac{1}{2}$ alors $|u_n - \frac{1}{2}| < \frac{1}{2}(2k)^{2^n}$.
   (c) Conclure.
5. **Cas** $3 < r < 1+\sqrt{6}$**. La suite** (u_n) **possède un cycle attracteur de période** 2**.**
   (a) Déterminer les points fixes ℓ_1 et ℓ_2 de $f \circ f$ qui ne sont pas des points fixes de f.
   (b) Montrer que $f(\ell_1) = \ell_2$ et $f(\ell_2) = \ell_1$.
   (c) À l'aide du graphe de $f \circ f$, vérifier graphiquement sur un exemple, que les suites (u_{2n}) et (u_{2n+1}) sont croissantes ou décroissantes à partir d'un certain rang et convergent, l'une vers ℓ_1, l'autre vers ℓ_2.
6. **Cas** $1+\sqrt{6} < r < 3,5699456\ldots$ **La suite** (u_n) **possède un cycle attracteur de période** $4, 8, 16\ldots$
   Trouver de tels exemples.
7. **À partir de** $r > 3,5699456\ldots$ **la suite** (u_n) **devient chaotique.**

1. Voici le comportement de différentes suites définies par le même $u_0 = 0,123$ et pour r valant successivement $r = 2,2$, $r = 3,2$, $r = 3,57$ et $r = 4$.

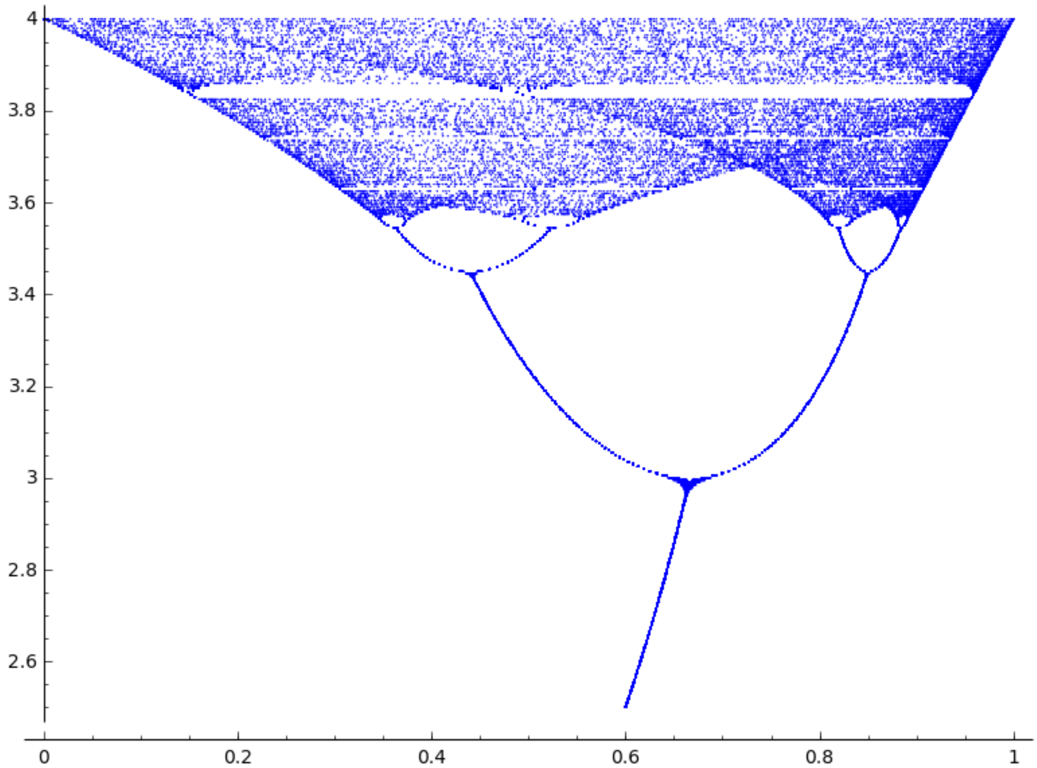


On voit d'abord une limite, puis la suite semble avoir « deux limites », c'est-à-dire deux valeurs d'adhérence, puis 4 valeurs d'adhérence. Pour $r = 4$ la situation semble chaotique.

2. Voici le code et le résultat de l'exécution `bifurcation(f,0.102)`, où `f(x,r) = r*x*(1-x)`.

Code 4 (*suites-chaos.sage (1)*).

```
def bifurcation(F,terme_init):
    Nmin = 100              # On oublie Nmin premiers termes
    Nmax = 200              # On conserve les termes entre Nmin et Nmax
    epsilon = 0.005         # On fait varier r de epsilon à chaque pas
    r = 2.5                 # r initial
    mespoints = []
    while r <= 4.0:
        u = liste_suite(F(x,r),terme_init,Nmax)  # On calcule la suite
        for k in range(Nmin,Nmax):
            mespoints = mespoints + [(u[k],r)]
        r = r + epsilon
    G = points(mespoints)
    G.show()
```



3. Voici le code pour les trois questions.

Code 5 (*suites-chaos.sage (2)*).

```
var('x,r')
f(x,r) = r*x*(1-x)
pts_fixes = solve(f==x,x)       # (a) Points fixes
ff = diff(f,x)                  # Dérivée
ff(x=0)                         # (b) f'(0)
solve(abs(ff((r-1)/r))<1,r) # (c) Condition point attractif
```

(a) Les deux points fixes fournis dans la liste `pts_fixes` sont 0 et $\frac{r-1}{r}$.

(b) On calcule la dérivée `ff`, on l'évalue en $x = 0$, on trouve $f'(0) = r$. Ainsi, si $r > 1$ alors $|f'(0)| > 1$ et le point 0 est répulsif.

(c) Par contre lorsque l'on résout l'inéquation $|f'(\frac{r-1}{r})| < 1$, la machine renvoie les conditions $r > 1$ et $r < 3$. Ainsi, pour $1 < r < 3$, le point fixe $\frac{r-1}{r}$ est attractif.

4. On pose d'abord `r = 2`, alors le point fixe est $\frac{r-1}{r} = \frac{1}{2}$.

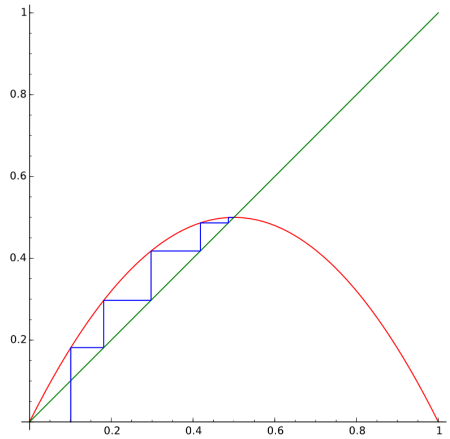
(a) Après simplification `(r*u*(1-u) - 1/2) + 2*(u-1/2)^2` vaut 0. Autrement dit $u_{n+1} - \frac{1}{2} = -2(u_n - \frac{1}{2})^2$, quel que soit u_n.

(b) C'est une simple récurrence :

$$\left|u_{n+1} - \frac{1}{2}\right| = 2\left(u_n - \frac{1}{2}\right)^2 < 2\left(\frac{1}{2}(2k)^{2^n}\right)^2 = \frac{1}{2}(2k)^{2^{n+1}}.$$

(c) Ainsi pour $u_0 \neq 0, 1$, il existe k tel que $|u_0 - \frac{1}{2}| < k < \frac{1}{2}$ et la suite (u_n) tend (très très) vite vers le point fixe $\frac{1}{2}$.

Avec $r = 2$, en quelques itérations, le terme de la suite n'est plus discernable de la limite $\frac{1}{2}$:



5. Voici le code pour les deux premières questions :

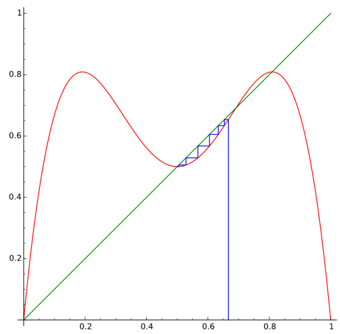
Code 6 (*suites-chaos.sage (3)*).

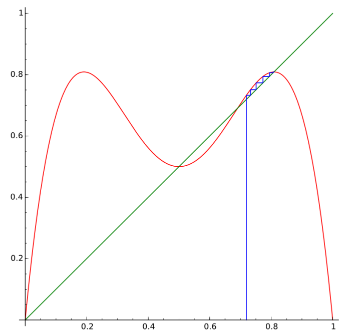
```
var('x,r')
f(x,r) = r*x*(1-x)              # f
g = f(f(x,r),r)                 # g(x) = f(f(x))
pts_doubles = solve(g==x,x)     # (a) Points fixes de g
# Les deux nouveaux points fixes de g :
ell1 = pts_doubles[0].rhs()
ell2 = pts_doubles[1].rhs()
eq = f(ell1,r)-ell2             # (b) l'image de ell1 est-elle ell2 ?
eq.full_simplify()              # Oui !
```

(a) On calcule les points fixes de $g = f \circ f$. Il y en a quatre, mais deux d'entre eux sont les points fixes de f. Les deux nouveaux sont :
$$\ell_1 = \frac{1}{2}\frac{r+1-\sqrt{r^2-2r-3}}{r} \qquad \text{et} \qquad \ell_2 = \frac{1}{2}\frac{r+1+\sqrt{r^2-2r-3}}{r}.$$

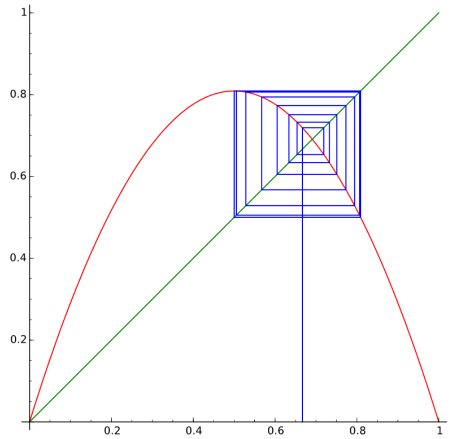
(b) Bien sûr, ℓ_1 et ℓ_2 ne sont pas des points fixes pour f. Par contre, on vérifie que $f(\ell_1) = \ell_2$ et $f(\ell_2) = \ell_1$.

(c) Voici les dessins pour $r = 1+\sqrt{5}$ et $u_0 = \frac{2}{3}$. La suite (u_{2n}), qui est vue comme suite récurrente de fonction g, est décroissante vers ℓ_1 (figure de gauche). La suite (u_{2n+1}), vue comme suite récurrente de fonction g, est croissante vers ℓ_2 (figure de droite).

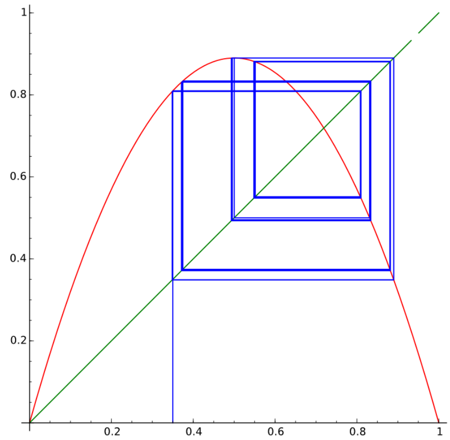




La suite (u_n), comme suite récurrente de fonction f, possède le cycle (ℓ_1, ℓ_2) comme cycle attracteur :



6. Voici un exemple de cycle de longueur 8 ($r = 3,56$, $u_0 = 0,35$) :



7. Le cas $r = 4$ sera étudié dans le tp suivant.

Travaux pratiques 4.
On s'intéresse au cas $r = 4$. On rappelle que

$$f(x) = rx(1-x)$$

et que

$$u_0 \in [0,1] \qquad \text{et} \qquad u_{n+1} = f(u_n) \quad (n \geqslant 0).$$

1. (a) Montrer que $\sin^2(2t) = 4\sin^2 t \cdot (1-\sin^2 t)$.
   (b) Montrer que pour $u_0 \in [0,1]$, il existe un unique $t \in [0, \frac{\pi}{2}]$ tel que $u_0 = \sin^2 t$.
   (c) En déduire $u_n = \sin^2(2^n t)$.
2. Pour $t_k = \frac{2\pi}{2^k+1}$ et $u_0 = \sin^2 t_k$, montrer que la suite (u_n) est périodique de période k.
3. **La suite est instable.**
   (a) Pour $k = 3$, on pose $t_3 = \frac{2\pi}{9}$ et $u_0 = \sin^2 t_3$. Calculer une valeur exacte (puis approchée) de u_n, pour différentes valeurs de n.
   (b) Partant d'une valeur approchée $\tilde{u}_0$ de u_0, que donne la suite des approximations $(\tilde{u}_n)$ définie par la relation de récurrence ?
4. **La suite est partout dense.**
   On va construire $u_0 \in [0,1]$ tel que, tout $x \in [0,1]$ peut être approché d'aussi près que l'on veut par un terme u_n de notre suite :
   $$\forall x \in [0,1] \quad \forall \epsilon > 0 \quad \exists n \in \mathbb{N} \qquad \left|x - u_n\right| < \epsilon.$$
   - Soit σ la constante binaire de Champernowne, formée de la juxtaposition des entiers en écriture binaire à 1, 2, 3,... chiffres 0, 1, 00, 01, 10,... dont l'écriture binaire est $\sigma = 0,0100011011\ldots$
   - Soit $u_0 = \sin^2(\pi\sigma)$.
   - Soit $x \in [0,1[$. Ce réel peut s'écrire $x = \sin^2(\pi t)$ avec $t \in [0,1[$ qui admet une écriture binaire $t = 0, a_1 a_2 \ldots a_p \ldots$.

Pour $\epsilon > 0$ fixé, montrer qu'il existe n tel que $\left|x - u_n\right| < \epsilon$.

1. (a) On pose `eq = sin(2*t)^2 - 4*sin(t)^2*(1-sin(t)^2)` et `eq.full_simplify()` renvoie 0. Donc $\sin^2(2t) = 4\sin^2 t \cdot (1-\sin^2 t)$, pour tout $t \in \mathbb{R}$.

   (b) Soit $h(t) = \sin^2 t$. On définit la fonction `h = sin(t)^2`, sa dérivée `hh = diff(h,t)` dont on cherche les zéros par `solve(hh==0,t)`. La dérivée ne s'annulant qu'en $t = 0$ et $t = \frac{\pi}{2}$, la fonction h' est strictement monotone sur $[0, \frac{\pi}{2}]$. Comme $h(0) = 0$ et $h(\frac{\pi}{2}) = 1$, pour chaque $u_0 \in [0,1]$, il existe un unique $t \in [0, \frac{\pi}{2}]$ tel que $u_0 = h(t)$.

   (c) Pour $u_0 = \sin^2 t$, on a

   $$u_1 = f(u_0) = ru_0(1-u_0) = 4\sin^2 t \cdot (1-\sin^2 t) = \sin^2(2t).$$

   On montre de même par récurrence que $u_n = \sin^2(2^n t)$.

2. Le code suivant calcule $u_k - u_0$. Sage sait calculer que cela vaut 0, pour k prenant la valeur 0, 1, 2,... mais pas lorsque k est une variable. Le calcul à la main est pourtant très simple !

   Code 7 (*suites-chaos.sage (4)*).

```
t = 2*pi/(2^k+1)
u_0 = sin(t)^2
u_k = sin(2^k*t)^2
zero = u_k-u_0
print(zero(k=0))
print(zero(k=1))
print(zero(k=2))
print(zero(k=3))
zero.simplify_trig()
```

3. **La suite est instable.**

   (a) Pour $k = 3$ et $t_3 = \frac{2\pi}{9}$, la suite (u_k) est périodique de période 3. Il suffit donc de calculer u_0, u_1, u_2. Par exemple :

   $$u_{3p} = u_0 = \sin^2\left(\frac{2\pi}{9}\right) \simeq 0{,}413\,175\,911\,1\ldots$$

   (b) Partons de la valeur approchée de u_0 avec 10 décimales exactes : $\tilde{u}_0 = 0{,}413\,175\,911\,1$ et calculons les premiers termes de la suite. On en extrait :

   $$\begin{aligned}
   \tilde{u}_0 &\simeq 0{,}413\,175\,911\,100\\
   \tilde{u}_3 &\simeq 0{,}413\,175\,911\,698\\
   \tilde{u}_6 &\simeq 0{,}413\,175\,906\,908\\
   \tilde{u}_9 &\simeq 0{,}413\,175\,945\,232\\
   \tilde{u}_{12} &\simeq 0{,}413\,175\,638\,640\\
   \tilde{u}_{15} &\simeq 0{,}413\,178\,091\,373\\
   \tilde{u}_{18} &\simeq 0{,}413\,158\,469\,568\\
   \tilde{u}_{21} &\simeq 0{,}413\,315\,447\,870\\
   \tilde{u}_{24} &\simeq 0{,}412\,059\,869\,464\\
   \tilde{u}_{27} &\simeq 0{,}422\,119\,825\,829\\
   \tilde{u}_{30} &\simeq 0{,}342\,897\,499\,745\\
   \tilde{u}_{33} &\simeq 0{,}916\,955\,513\,784\\
   \tilde{u}_{36} &\simeq 0{,}517\,632\,311\,613\\
   \tilde{u}_{39} &\simeq 0{,}019\,774\,022\,431
   \end{aligned}$$

   Si l'approximation de départ et tous les calculs étaient exacts, on devrait obtenir u_0 à chaque ligne. On constate que l'erreur augmente terme après terme, et après $\tilde{u}_{30}$, le comportement de $\tilde{u}_n$ n'a plus rien à voir avec u_n. L'explication vient de la formule $u_n = \sin^2(2^n t)$, une erreur sur t, même infime au départ, devient grande par multiplication par 2^n.

4. **La suite est partout dense.**

La construction est assez jolie mais délicate. (Il ne faut pas avoir peur de l'écriture en base 2 qui n'est pas ici le point clé. Vous pouvez penser en base 10 pour mieux comprendre.)

Fixons $\epsilon > 0$. Soit un entier p tel que $\frac{\pi}{2^p} < \epsilon$. L'écriture binaire de t étant $t = 0,a_1a_2\dots a_p\dots$ la séquence $a_1a_2\dots a_p$ apparaît quelque part dans la constante de Champernowne. Plus précisément, il existe un entier n tel que $2^n\sigma = \dots b_2b_1b_0,a_1a_2\dots a_p\dots$ (on a fait apparaître la séquence juste après la virgule par décalage de la virgule).

On va pouvoir oublier la partie entière $m = \dots b_2b_1b_0 \in \mathbb{N}$ et on note $\tilde{\sigma} = 0,a_1a_2\dots a_p\dots$ la partie fractionnaire de σ, de sorte que $2^n\sigma = m + \tilde{\sigma}$.

$$\begin{aligned}
|x-u_n| &= \left|\sin^2(\pi t)-\sin^2(2^n\pi\sigma)\right| \\
&= \left|\sin^2(\pi t)-\sin^2(2^n\pi m+\pi\tilde{\sigma})\right| \\
&= \left|\sin^2(\pi t)-\sin^2(\pi\tilde{\sigma})\right| \qquad \text{car } \sin^2 \text{ est } \pi\text{-périodique} \\
&\leqslant |\pi t-\pi\tilde{\sigma}| \qquad \text{par le théorème des accroissements finis} \\
&\leqslant \pi\frac{1}{2^p} \qquad \text{car } t \text{ et } \tilde{\sigma} \text{ ont les mêmes } p \text{ premières décimales} \\
&< \epsilon
\end{aligned}$$

Conclusion : n'importe quel x est approché d'aussi près que l'on veut par la suite.

Pour en savoir plus :

- *La suite logistique et le chaos*, Daniel Perrin.
- *Étude d'une suite récurrente*, Jean-Michel Ferrard.