

ÉQUATIONS DIFFÉRENTIELLES RETARDÉES

Une équation différentielle retardée (EDR), appelée en anglais delayed differential equation (DDE), a pour forme générale :

$$y'(t) = f(t, y(t), y(t - T(t)))$$

L'inconnue est la fonction $y : t \rightarrow y(t)$, dont la dérivée dépend fonctionnellement du temps, t , de la valeur prise par y en t , ainsi que la valeur prise par y à la date $t - T(t)$. On appelle la fonction T le délai. Si le délai est une fonction constante strictement positive, on a $T(t) = T$ et on est en présence d'une EDR à délai constant. L'interprétation économique d'une EDR est la suivante : les variations de l'état d'un système dynamique continu à une date t sont fonction de l'état du système à cette date et du retard - souvent inévitable - avec lequel un agent de contrôle prend des décisions (l'état du système à la date t conduit l'agent à prendre une décision à la date ultérieure $t + T$).

Dans la classe des EDR à délai constant, on distingue les EDR linéaires :

$$y'(t) = -ay(t) + by(t - T) \text{ pour } t \geq t_0$$

où a et b sont des réels.

Une équation de Frisch-Holme est une EDR linéaire à laquelle est adjointe la condition initiale, dite historique, $\forall t \in [t_0 - T, t_0]$, $y(t) = h(t)$, h étant une fonction continue connue. Le problème consiste à déterminer sur $[t_0; +\infty[$ une fonction y au moins de classe C^1 vérifiant l'EDR linéaire et qui coïncide avec la fonction historique h sur l'intervalle $[t_0 - T, t_0]$. Il se trouve que de nombreux modèles économiques mettent à jour des équations du type Frisch-Holme et c'est pourquoi ce chapitre leur est entièrement consacré.

Le théorème de Bellman et Cooke assure l'existence et l'unicité de la solution d'une équation de Frisch-Holme. Dans son ouvrage récent "Time-delay Systems with Applications to Economic Dynamics and Control" (Lambert Academic Publishing, 2011), André Keller détaille quatre méthodes pour exprimer cette solution :

1. la méthode des pas (ou résolution séquentielle) qui est la plus intuitive.
2. la méthode basée sur la fonction Lambert W qui est la plus utilisée en pratique par les économistes parce qu'elle reprend leurs préoccupations majeures : recherche des équilibres, stabilité des équilibres, analyse de bifurcations quand la nature de l'équilibre dépend d'un paramètre.
3. la méthode basée sur la transformée de Laplace, qui provient des sciences de l'ingénieur et que les économistes utilisent peu, voire jamais.
4. la méthode basée sur la transformation différentielle qui est la plus récente et provient, elle aussi, des sciences de l'ingénieur.

André Keller double systématiquement les considérations théoriques par des applications pratiques dans lesquelles il utilise les logiciels de calcul symbolique Mathematica et MATLAB. Il est vrai que Maple ne propose pas de telles facilités, ce qui ne veut pas dire, bien entendu, que ce logiciel est inefficace dans le domaine des EDR. Ce chapitre a précisément pour but de montrer que les équations de Frisch-Holme sont solubles avec Maple. Comme la fonction **LambertW** est disponible sous forme d'instruction, on a choisi de s'attaquer à la méthode des pas en mettant au point des procédures.

La première section propose deux manières de résoudre une EDR de Frisch-Holme, une orientée vers la résolution exacte, l'autre vers une résolution approchée. Les sections 2 et 3 sont des applications au modèle de la construction navale étudié en 1931 par Jan Tinbergen et au modèle célèbre du cycle des

▼ Résolution de l'équation de Frisch-Holme par la méthode des pas

Dans leur article "The Characteristic Solutions of Mixed Difference and Differential Equations" (Econometrica, avril 1935), R. Frisch et H. Holme ont étudié essentiellement l'EDR linéaire avec condition initiale qui porte maintenant leur nom :

$$(FH) \quad \begin{cases} y'(t) = -a y(t) - b y(t - T) \\ \forall t \in [d - T; d], y(t) = h(t) \end{cases}$$

où T est le paramètre (réel strictement positif) de retard, les coefficients a et b sont des réels et la fonction historique h est continue et connue sur $[d - T, d]$.

Le principe de la résolution de (FH) par la méthode des pas est simple. La fonction-solution est continue par morceaux et coïncide sur chaque morceau de longueur T avec la solution de l'EDO avec condition initiale fournie par la valeur prise par la solution obtenue au pas précédent à la date finale de l'intervalle de l'intervalle précédent.

Ainsi, sur l'intervalle $[d - T, d]$, la solution coïncide avec la fonction historique h connue et on note ce morceau de trajectoire y_0 .

Sur l'intervalle $[d, d + T]$, la solution coïncide avec la solution de l'EDO suivante :

$$y'(t) = -a y(t) - b y_0(t - T) \text{ avec la condition initiale } y(d) = y_0(d)$$

et on la note y_1 .

Sur l'intervalle $[d + T, d + 2T]$, la solution coïncide avec la solution de l'EDO suivante :

$$y'(t) = -a y(t) - b y_1(t - T) \text{ avec la condition initiale } y(T) = y_1(T)$$

et on la note y_2 .

Tous les morceaux de la solution de l'EDR sont calculés de proche en proche suivant le même principe.

Deux types d'algorithmes automatisant la résolution de (FH) sont possibles avec l'instruction **dsolve**. La première procédure, appelée **fhmpas**, repose sur une résolution exacte des EDO successives. Elle devient lente dès que le nombre d'itérations devient grand et doit être alors remplacée par des procédures de résolution approchée, appelées **fhmpasn** et **fhmpasndp**, utilisant les ressources de l'option **numeric** dans **dsolve**.

▼ Procédure de résolution exacte **fhmpas**

La procédure **fhmpas** (pour Frisch Holmes Méthode des PAS) a 6 arguments : **T** est le retard, **a** et **b** sont les coefficients de l'EDR linéaire, **h** est la fonction historique, **d** est la borne droite de l'intervalle sur lequel la fonction historique est donnée et **N** est le nombre d'intervalles de longueur T où on cherche par itération les trajectoires-solutions.

Le programme commence par l'initialisation de la fonction historique, identifiée à la fonction **h** et qu'on renomme **y[0]**.

La boucle permet de résoudre l'EDO avec condition initiale et de stocker la solution sous le nom **y[i]**. On en déduit la solution sur le segment $[d + (i - 1)T, d + iT]$ qu'on note **morc[i]**.

L'option **remember** force le stockage des solutions au fur et à mesure de leur apparition.

La sortie est une fonction définie par morceaux.

```
> restart;
```

```

fhmpas:=proc(T,a,b,h,d,N)
option remember;
local y,eqd,morc,i,sol;
y[0]:=unapply(h(t),t);
morc[0]:=-T<=t and t<=0,y[0](t);
for i from 1 to N do
eqd[i]:=diff(y[i](t),t)=-a*y[i](t)-b*y[i-1](t-T);#equation de
Frisch-Holme
y[i]:=unapply(rhs(dsolve({eqd[i],y[i]((i-1)*T)=y[i-1]((i-1)*
T)},y[i](t))),t);#résolution de l'EDO
morc[i]:=(i-1)*T<=t and t<=i*T,y[i](t);#intervalle et
solution sur l'intervalle à l'étape i
end do;
sol:=unapply(piecewise(seq(morc[i],i=0..N)),t);#expression de
la solution comme fonction par morceaux
end proc;

```

fhmpas := **proc**(*T, a, b, h, d, N*) (1.1.1)

```

option remember;
local y, eqd, morc, i, sol;
y[0] := unapply(h(t), t);
morc[0] := - T <= t and t <= 0, y[0](t);
for i to N do
    eqd[i] := diff(y[i](t), t) = - a * y[i](t) - b * y[i - 1](t - T);
    y[i] := unapply(rhs(dsolve( {eqd[i], y[i]((i - 1) * T) = y[i - 1]((i - 1)
    * T) }, y[i](t) )), t);
    morc[i] := (i - 1) * T <= t and t <= i * T, y[i](t)
end do;
sol := unapply(piecewise(seq(morc[i], i = 0 .. N) ), t)
end proc

```

On teste cette procédure sur 3 exemples canoniques et on va en profiter pour montrer comment obtenir des représentations graphiques, cartésiennes et paramétriques, de la solution.

Exemple 1 : Résoudre $y'(t) = -y(t) + y(t-1)$ avec $y(t) = e^t$ pour $-1 \leq t \leq 0$.

Il s'agit d'une équation (FH) où $T=1$, $a=1$, $b=-1$, $h : t \rightarrow \exp(t)$, $d=0$ et on choisit $N=4$.

```
> eqrsol:=fhmpas(1,1,-1,t->exp(t),0,4);
```

$$\text{eqrsol} := t \rightarrow \text{piecewise} \left(-1 \leq t \text{ and } t \leq 0, e^t, 0 \leq t \text{ and } t \leq 1, \left(\frac{1}{2} e^{2t-1} + 1 - \frac{1}{2} e^{-1} \right) e^{-t}, 1 \leq t \text{ and } t \leq 2, \left(\frac{1}{4} e^{2t-2} + e t - \frac{1}{2} t - \frac{1}{2} e + \frac{5}{4} \right) \right.$$

(1.1.2)

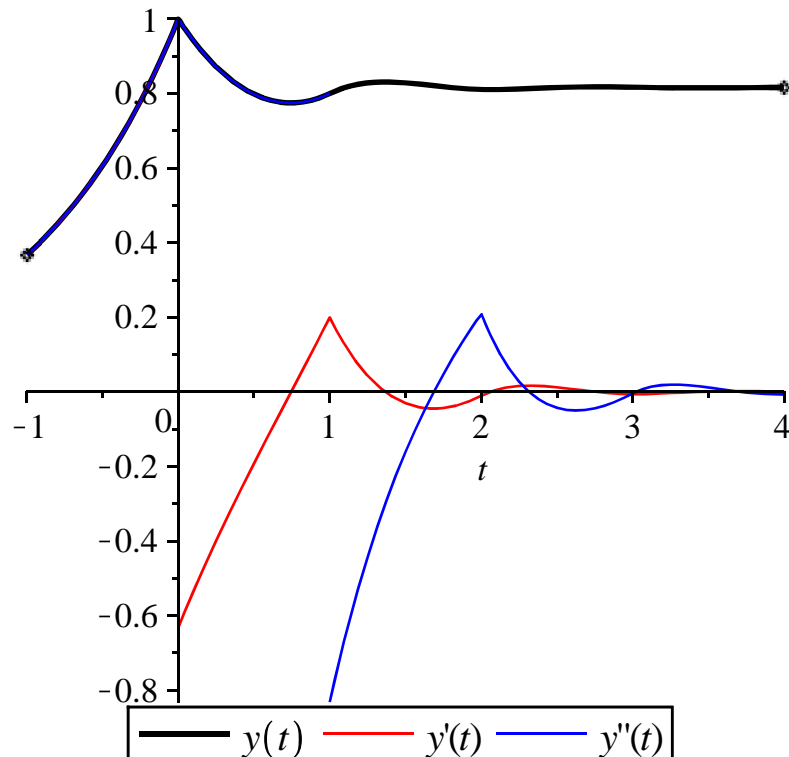
$$\begin{aligned}
& -\frac{1}{2} e^{-1}) e^{-t}, 2 \leq t \text{ and } t \leq 3, \left(\frac{1}{8} e^{2t-3} + \frac{1}{2} e^2 t^2 - \frac{3}{2} e^2 t - \frac{1}{4} e t^2 + \frac{7}{4} e t \right. \\
& \left. - \frac{1}{2} t + \frac{5}{4} e^2 - \frac{9}{8} e + \frac{5}{4} - \frac{1}{2} e^{-1} \right) e^{-t}, 3 \leq t \text{ and } t \leq 4, -\frac{1}{e^{-3}} \left(e^{-t} \left(\right. \right. \\
& \left. \left. -\frac{1}{8} (e^3 + 10 e^2 + 15 e - 2 - 4 e^{-1}) e^{-3} + \frac{1}{48} (-69 e^2 + 144 e^3 + 144 e \right. \right. \\
& \left. \left. - 72) e^{-3} \right) \right) + \frac{1}{48} (3 e^{2t-4} + 8 e^3 t^3 - 60 e^3 t^2 + 156 e^3 t - 4 e^2 t^3 + 54 e^2 t^2 \\
& \left. - 150 e^2 t - 12 e t^2 + 84 e t - 24 t) e^{-t} \right)
\end{aligned}$$

La représentation cartésienne de la solution et de ses dérivées première (vitesse) et seconde (accélération) donne la figure suivante :

```

> plot([eqrsol(t),diff(eqrsol(t),t),diff(eqrsol(t),t,t)],t=-1.
.4,discont=true,color=[black,red,blue],thickness=[2,1,1],
legend=[typeset(y(t)),typeset(y,'"',"(",t,")"),typeset(y,
"'','"(",t,")")]);

```

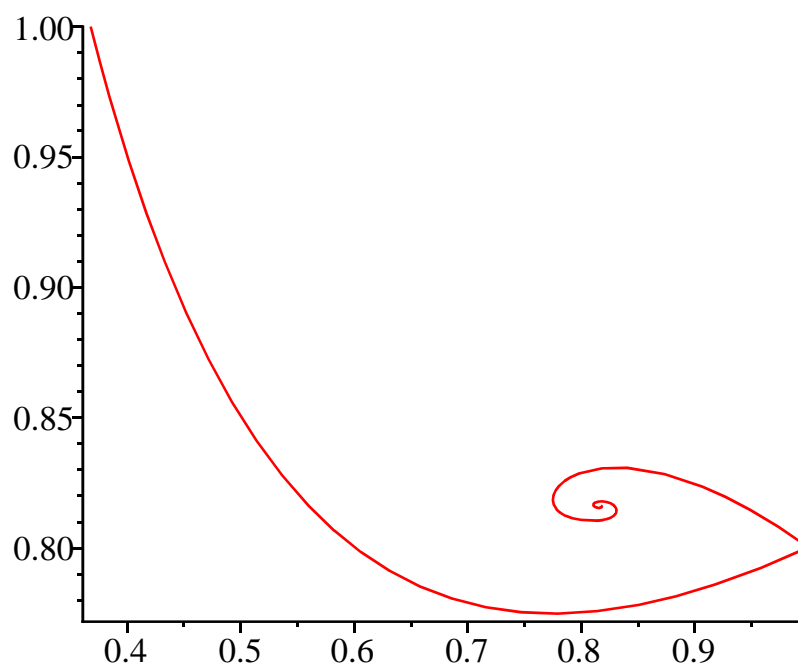


Le diagramme des phases correspond à la représentation paramétrique dans le plan $(O, y(t-1), y(t))$:

```

> plot([eqrsol(t-1),eqrsol(t),t=0..4],discont=true);

```



Exemple 2 : Résoudre $y'(t) = -y(t) + y(t-1)$ avec $y(t) = -4t$ pour $-1 \leq t \leq 0$.
On a maintenant $T=1$, $a=1$, $b=-1$, $h : t \rightarrow -4t$, $d=0$ et on choisit encore $N=4$.

> eqrsol:=fhmpas(1,1,-1,t->-4*t,0,4);

eqrsol := t → piecewise $\left(-1 \leq t \text{ and } t \leq 0, -4t, 0 \leq t \text{ and } t \leq 1, -4t + 8 - 8e^{-t}, 1 \right)$ **(1.1.3)**

$$\begin{aligned} & \leq t \text{ and } t \leq 2, -4t + 16 - 8te^{-t+1} - 8e^{-t}, 2 \leq t \text{ and } t \leq 3, \left(-8et - 4e^t t \right. \\ & \left. + 24e^t - 4e^2 t^2 + 8e^2 t - \frac{8(-1 + 2e^{-1} + e^{-2} - 2e^{-2}e + 2e^{-2}e^2)}{e^{-2}} \right) e^{-t}, 3 \leq t \\ & \text{and } t \leq 4, \left(-4e^2 t^2 + 8e^2 t - 4e^t t + 32e^t - \frac{4}{3}e^3 t^3 + 8e^3 t^2 - 20e^3 t - 8et \right. \\ & \left. - \frac{8(-2e^{-2}e - 2e^3 e^{-2} + 2e^{-2}e^2 - 1 + 2e^{-1} + e^{-2})}{e^{-2}} \right) e^{-t} \end{aligned}$$

Représentation graphique de la trajectoire, de sa vitesse et de son accélération :

> plot([eqrsol(t),diff(eqrsol(t),t),diff(eqrsol(t),t,t)],t=-1.4,discont=true,color=[black,red,blue],thickness=[2,1,1],legend=[typeset(y(t)),typeset(y,"',"(",t,")"),typeset(y,"''',"(",t,")")]);

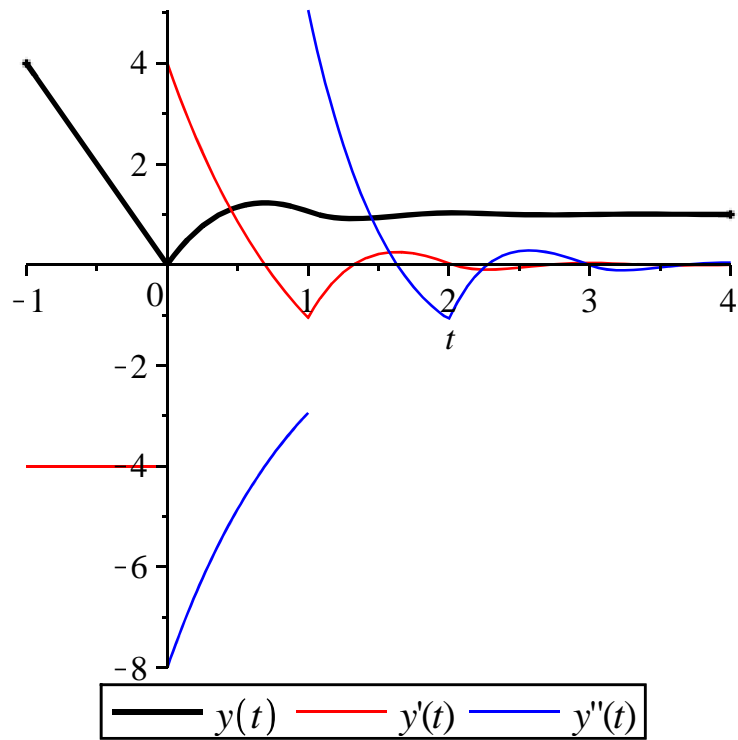
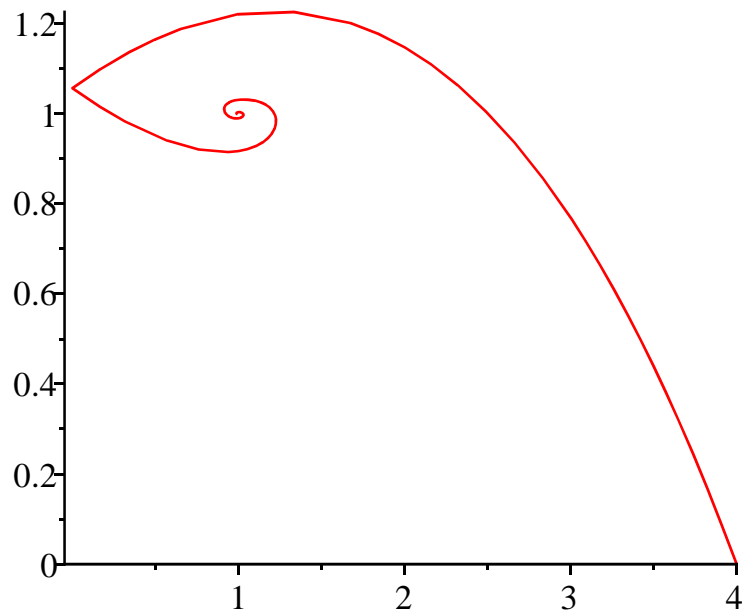


Diagramme des phases

```
> plot([eqlsol(t-1), eqlsol(t), t=0..4],discont=true);
```



Exemple 3 : Résoudre $y'(t) = -y\left(t - \frac{\pi}{2}\right)$ avec $y(t) = \cos(t)$ pour $-\frac{\pi}{2} \leq t \leq 0$.

On a $T = \frac{\pi}{2}$, $a = 0$, $b = 1$, $h : t \rightarrow \cos(t)$, $d = 0$ et on choisit $N = 6$.

```
> eqlsol:=fhmpas(Pi/2,0,1,t->cos(t),0,6);
```

(1.1.4)

$$\begin{aligned}
 \text{eqrsol} := t \rightarrow \text{piecewise} & \left(-\frac{1}{2} \pi \leq t \text{ and } t \leq 0, \cos(t), 0 \leq t \text{ and } t \leq \frac{1}{2} \pi, \cos(t), \frac{1}{2} \pi \right. \\
 & \leq t \text{ and } t \leq \pi, \cos(t), \pi \leq t \text{ and } t \leq \frac{3}{2} \pi, \cos(t), \frac{3}{2} \pi \leq t \text{ and } t \leq 2 \pi, \cos(t), \\
 & \left. 2 \pi \leq t \text{ and } t \leq \frac{5}{2} \pi, \cos(t), \frac{5}{2} \pi \leq t \text{ and } t \leq 3 \pi, \cos(t) \right)
 \end{aligned}
 \tag{1.1.4}$$

Représentation graphique de la solution, de sa vitesse et de son accélération :

```

> plot([eqrsol(t),diff(eqrsol(t),t),diff(eqrsol(t),t,t)],t=-
Pi/2..3*Pi,discont=true,color=[black,red,blue],thickness=[2,
1,1],legend=[typeset(y(t)),typeset(y,'"',"(",t,")"),typeset
(y,'"',"(",t,")")]);

```

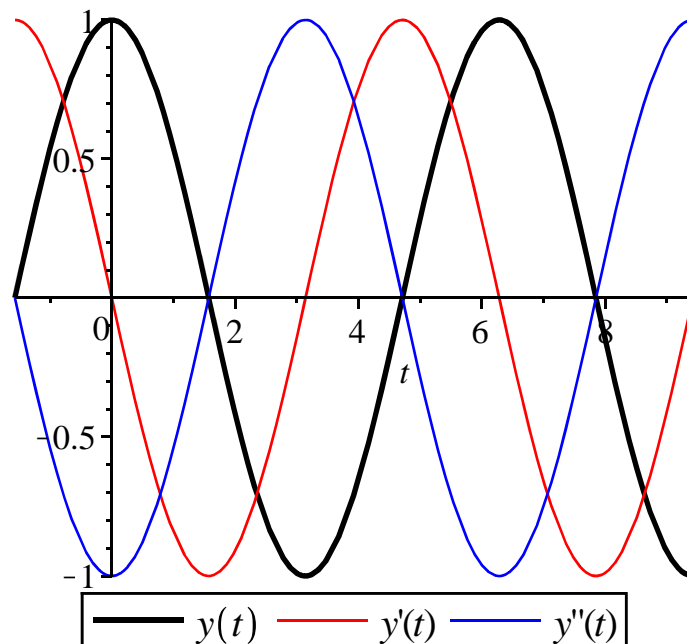
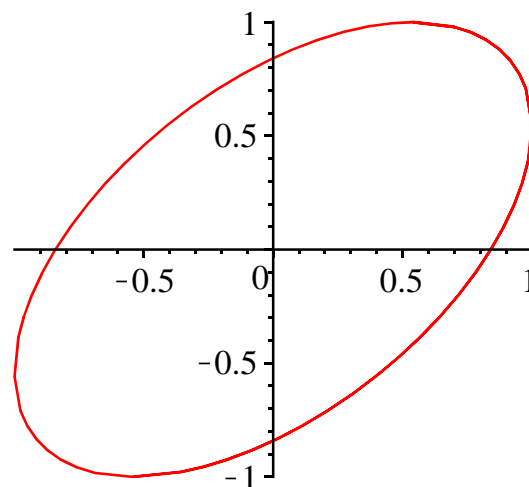


Diagramme des phases :

```

> plot([eqrsol(t-1),eqrsol(t),t=0..3*Pi],discont=true);

```



Procédures de résolution approchée **fhmpasn** et **fhmpasndp**

Bien que fruste, la procédure **fhmpas** est très gourmande en calculs. Il est pratiquement impossible d'obtenir la solution en un temps raisonnable sur une dizaine de périodes. Ce défaut se corrige si on se contente d'une résolution approchée des équations différentielles au moyen de l'option **numeric** de l'instruction **dsolve**. Les programmes suivants, appelés **fhmpasn** (pour Frisch Holme Méthode des PAS Numeric) et **fhmpasndp** (pour Frisch Holme Méthode des PAS Numeric Diagramme des Phases), sont construits pour donner respectivement la trajectoire-solution et le diagramme des phases.

Représentation graphique de la trajectoire-solution

A partir du moment où on privilégie une résolution numérique et qu'on ne s'intéresse qu'à la visualisation de la trajectoire, il n'est pas utile de connaître la forme exacte de la solution puisque Maple renvoie une procédure non détaillée quand on invoque **dsolve** avec l'option **numeric**. C'est pourquoi le programme **fhmpasn** est construit pour fournir une représentation graphique cartésienne de la trajectoire-solution.

Ses paramètres sont les mêmes que ceux de **fhmpas**, soit le retard T , les coefficients de l'EDR linéaire a et b , la fonction historique h , la borne droite d et le numéro d'ordre de la solution N .

La sortie est un ensemble (set) de graphes, chacun correspondant à un morceau de la trajectoire-solution.

L'algorithme commence par initialiser la solution et l'ensemble des graphes. La boucle **for** pose d'abord l'équation différentielle ordinaire à résoudre et sa condition initiale puis demande une résolution numérique, signalée par l'option **numeric** dans **dsolve**.

L'instruction **dsolve** comprend trois autres options : **range** pour se concentrer sur l'intervalle concerné par le pas et éviter ainsi la recherche hasardeuse de solution sur tout \mathbb{R} ; **known** pour indiquer à Maple que $y(t - T)$ se réfère à la solution calculée au pas précédent; **output** prépare l'extraction de la solution avec **subs** à la ligne suivante. A chaque pas, l'ensemble des graphes est augmenté d'une unité par réunion de l'ensemble obtenu au pas précédent avec l'ensemble constitué par le nouveau graphique. Après exécution de tous les rounds, la variable **graf** contient tous les morceaux de la solution.

```
> fhmpasn:=proc(T,a,b,h,d,N) option remember;
  local i,sol,soly,eqd,cinit,graf;
  soly[0]:=unapply(h(t),t);#initialisation de la solution
  graf:={plot(soly[0],d-T..d)}:#initialisation de l'ensemble
  des graphes
  for i from 1 to N do
    eqd[i]:=diff(y(t),t)=-a*y(t)-b*soly[i-1](t-T);#équation de
    Frisch-Holme
    cinit[i]:=y(d+(i-1)*T)=soly[i-1](d+(i-1)*T);#condition
    initiale
    sol[i]:=dsolve({eqd[i],cinit[i]},numeric,range=d+(i-1)*T..
    d+i*T,known=soly[i-1],output=listprocedure);#résolution
    numérique de l'EDO avec options appropriées au problème
    soly[i]:=subs(sol[i],y(t));#récupération de la solution
    graf:=graf union {plot(soly[i],d+(i-1)*T..d+i*T)}:#ajout
```



```

d'un graphe dans l'ensemble des graphes
end do;
graf;#récupération de l'ensemble des graphes
end proc;
fhmpasn := proc(T, a, b, h, d, N)
option remember;
local i, sol, soly, eqd, cinit, graf;
soly[0] := unapply(h(t), t);
graf := {plot(soly[0], d - T..d)};
for i to N do
    eqd[i] := diff(y(t), t) = -a*y(t) - b*soly[i-1](t - T);
    cinit[i] := y(d + (i - 1)*T) = soly[i-1](d + (i - 1)*T);
    sol[i] := dsolve({eqd[i], cinit[i]}, numeric, range = d + (i - 1)*T..d + i
        *T, known = soly[i-1], output = listprocedure);
    soly[i] := subs(sol[i], y(t));
    graf := union(graf, {plot(soly[i], d + (i - 1)*T..d + i*T)});
end do;
graf
end proc

```

(1.2.1.1)

Pour visualiser la trajectoire solution, il faut d'abord charger le paquetage **plots** (puisqu'on ne peut pas insérer une instruction **with** dans une procédure Maple) puis exécuter le programme avec **display**.

Reprenons l'exemple de la section précédente :

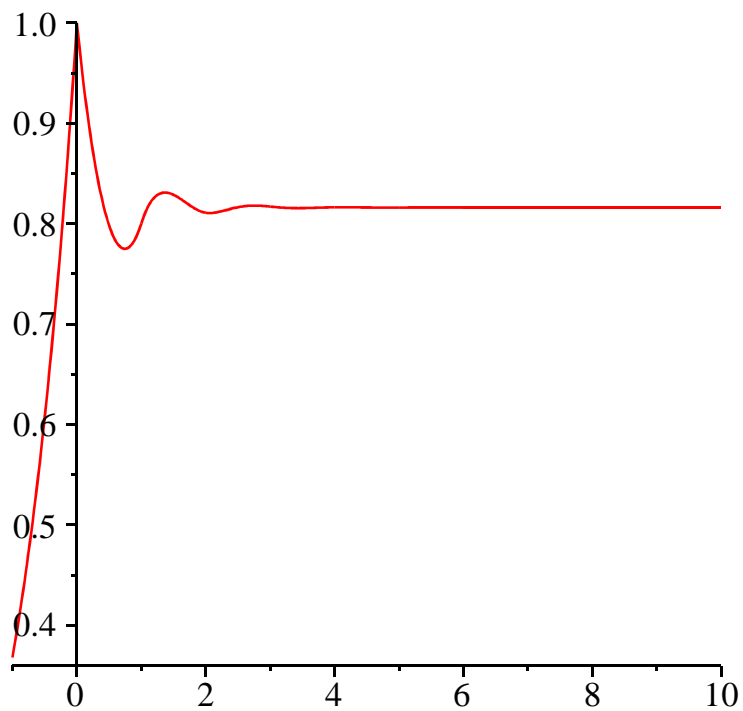
$$y'(t) = -y(t) + y(t-1) \text{ avec } y(t) = e^t \text{ pour } -1 \leq t \leq 0$$

L'application de la procédure **fhmpasn** donne :

```

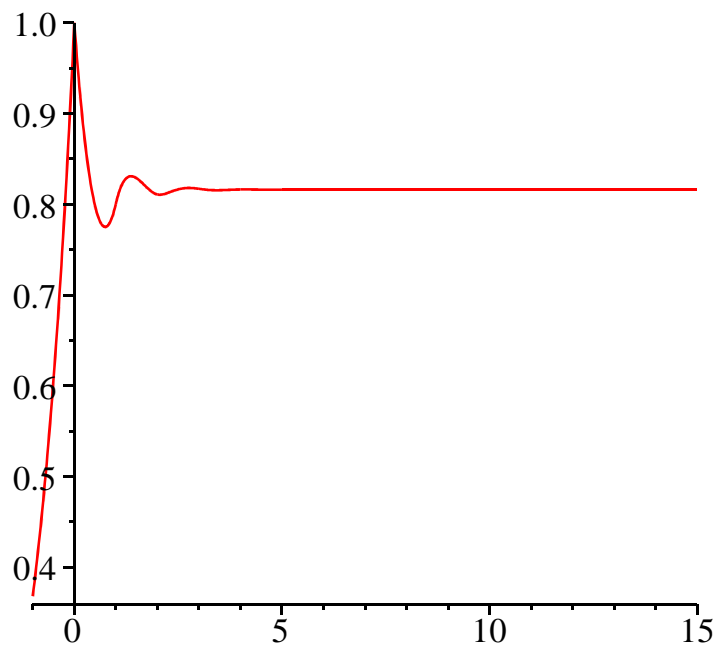
> with(plots):#chargement du paquetage plots
display(fhmpasn(1,1,-1,t->exp(t),0,10));#visualisation de
la trajectoire-solution

```



Le test suivant montre que la procédure **fhmpasn** est plus rapide/performante que **fhmpas** dès que N est "grand", c'est à dire en pratique $N > 9$. Pour l'EDR ci-dessus, la procédure approchée paramétrée avec 15 boucles fournit une sortie graphique en 11 secondes environ.

```
> st := time[real]():#initialisation du chronomètre
display(fhmpasn(1,1,-1,t->exp(t),0,15));#appel à la
procédure de résolution approchée
time[real]() - st;#calcul du "temps de travail" réel
effectué par Maple
```

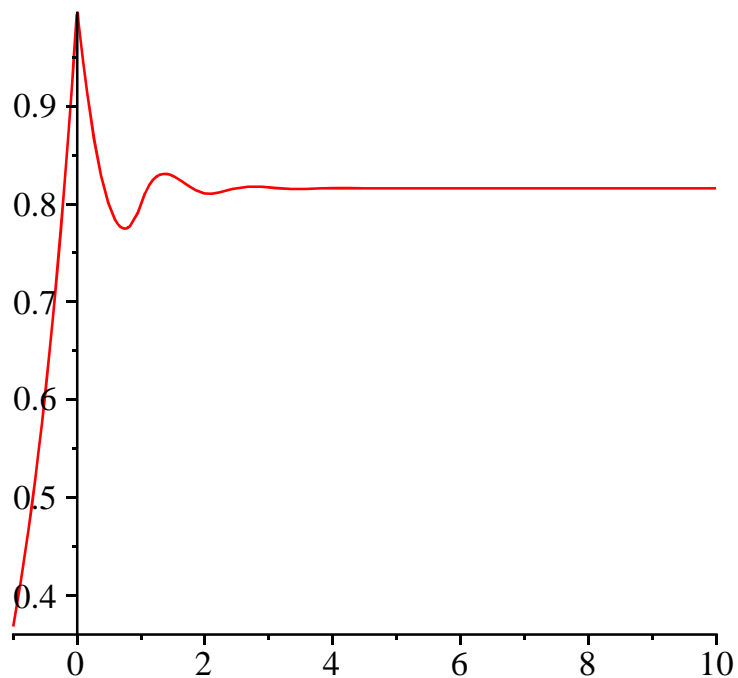


11.012

(1.2.1.2)

Avec la procédure exacte, paramétrée avec 10 boucles, la sortie est disponible en près de 65 secondes. Il est déconseillé de fixer le nombre de boucles à 15, à moins d'être extrêmement patient ou masochiste...

```
> st := time[real]():
plot(fhmpas(1,1,-1,t->exp(t),0,10),-1..10);#appel à la
procédure de résolution exacte
time[real]() - st;
```



65.614

(1.2.1.3)

On note au passage que le résultat obtenu par la procédure approchée est très proche de celui obtenu par résolution exacte. C'est bon signe sur la fiabilité de cette méthode.

Représentation graphique du diagramme des phases

Pour obtenir une représentation graphique du diagramme des phases à partir d'une résolution approchée, il faut procéder en deux temps.

En premier lieu, il faut être en mesure de savoir calculer la solution sur n'importe quel intervalle, y compris la portion historique. La procédure `fhmpasndp` renvoie la solution approchée sur l'intervalle $[d + (N - 1)T, d + NT]$. Ses paramètres sont les mêmes que ceux de `fhmpasn`, soit le retard T , les coefficients de l'EDR linéaire a et b , la fonction historique h , la borne droite d et le numéro d'ordre de la solution N . La condition `if..then..else` distingue le cas où $N = 0$ où la solution coïncide avec la fonction historique et le cas $N \geq 1$ où il faut résoudre numériquement une EDO avec `dsolve` et ses options adéquates pour résoudre l'EDR (`numeric, range` et `known`) et récupérer la solution en y (`output`).

```
> fhmpasndp:=proc(T,a,b,h,d,N)
  option remember;
  local i,sol,soly,eqd,cinit;
  soly[0]:=h;
  if N=0 then h;#solution correspondant à la fonction
  historique
  else
    for i from 1 to N do
      eqd[i]:=diff(y(t),t)=-a*y(t)-b*soly[i-1](t-T);
      #équation de Frisch-Holme
      cinit[i]:=y(d+(i-1)*T)=soly[i-1](d+(i-1)*T);
      #condition initiale
      sol[i]:=dsolve({eqd[i],cinit[i]},numeric,range=d+
      (i-1)*T..d+i*T,known=soly[i-1],output=listprocedure);
      #résolution numérique de l'EDO avec options appropriées au
      problème
      soly[i]:=subs(sol[i],y(t));#récupération de la
      solution
    end do;
  end if;
end proc;
fhmpasndp := proc(T, a, b, h, d, N) (1.2.2.1)
  option remember;
  local i, sol, soly, eqd, cinit;
  soly[0] := h;
  if N = 0 then
    h
  else
    for i to N do
      eqd[i] := diff(y(t), t) = -a*y(t) - b*soly[i - 1](t - T);
```

```

cinit[i] := y(d + (i - 1) * T) = soly[i - 1](d + (i - 1) * T);
sol[i] := dsolve( {eqd[i], cinit[i]}, numeric, range = d + (i - 1) * T..d
+ i * T, known = soly[i - 1], output = listprocedure);
soly[i] := subs(sol[i], y(t))

```

end do

end if

end proc

En second lieu, il faut une autre procédure, qu'on nommera simplement **F**, pour calculer automatiquement la valeur prise par la fonction-solution en n'importe quel point du plan. Notons E la fonction partie entière d'un nombre instruction **floor** en Maple. Pour un délai donné T , on sait que la date t se situe dans l'intervalle

$\left[d + \left(E\left(\frac{t}{T} \right) \right) T, d + \left(E\left(\frac{t}{T} \right) + 1 \right) T \right]$ correspondant au $\left(E\left(\frac{t}{T} \right) + 1 \right)$ -ième morceau

(par exemple, si $d = 0$, $T = 2$ et $t = 3, 5$, $y(3, 5)$ se calcule par la solution approchée dans l'intervalle $[2, 4]$, qui se trouve dans le deuxième morceau). En paramétrant la procédure

fhmpasndp avec $N = \left(E\left(\frac{t}{T} \right) + 1 \right)$, on connaît la valeur prise par la solution en n'importe

quelle date t .

```

> F:=proc(t)
  local en, fen;
  en:=floor(t/T);
  fen:=fhmpasndp(T, a, b, h, d, en+1);
  fen(t);
end proc;

```

$F := \text{proc}(t)$

local en, fen;

en := floor(t/T); fen := fhmpasndp(T, a, b, h, d, en + 1); fen(t)

(1.2.2.2)

end proc

En prenant la précaution de retarder l'évaluation de cette procédure, on peut facilement tracer le diagramme des phases avec **plot**.

Reprenons notre exemple vedette :

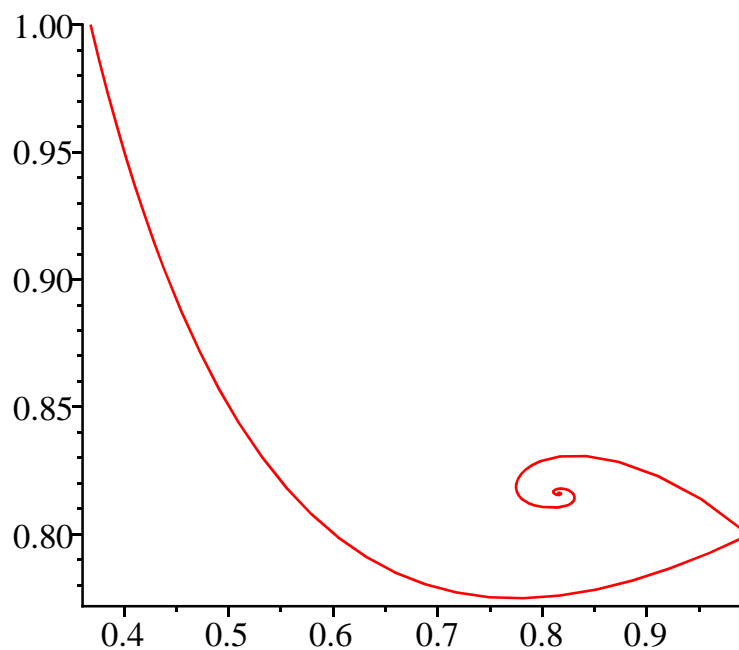
$$y'(t) = -y(t) + y(t-1) \text{ avec } y(t) = e^t \text{ pour } -1 \leq t \leq 0$$

L'application de la méthode **fhmpasndp+F** donne :

```

> T, a, b, h, d := 1, 1, -1, t->exp(t), 0;
  plot(['F(t-1)', 'F(t)', t=0..8]);
      T, a, b, h, d := 1, 1, -1, t->e^t, 0

```



Application économique 1. Le modèle de construction navale de J. Tinbergen

Dans la construction navale, le temps moyen de construction d'un navire est une constante T . La différence entre le tonnage effectif et le tonnage "moyen" est notée $y(t)$. L'accroissement de la flotte en t s'explique par une réaction linéaire à la différence constatée à la date $t - T$, de sorte que $y'(t) = -b y(t - T)$ avec $y(t) = h(t)$ quand $t \in [-T, 0]$. Un écart positif se traduit par une réduction de mise en chantier, et inversement. En pratique, on considère que b est un paramètre compris entre $\frac{1}{2}$ et 1 et que le délai de construction est supérieur à 1 (Tinbergen propose T compris entre 1 et 2 ans).

Il s'agit de résoudre et représenter graphiquement $y'(t) = -b y(t - T)$ avec $y(t) = h(t)$ quand $t \in [-T, 0]$. Conformément aux indications de Tinbergen, on prendra $T = 1.5$ et $b = 0.75$. La fonction "histoire" h est du type exponentielle : $h(t) = e^{0.01 \cdot t}$.

La résolution exacte fait appel à la procédure **fhmpas**, avec laquelle on obtient une représentation cartésienne et un diagramme des phases pour N "petit" (on prendra $N = 5$).

> **eqrsol:=fhmpas(1.5,0,0.75,t->exp(0.01*t),0,5);#appel à la procédure exacte avec 5 boucles.**

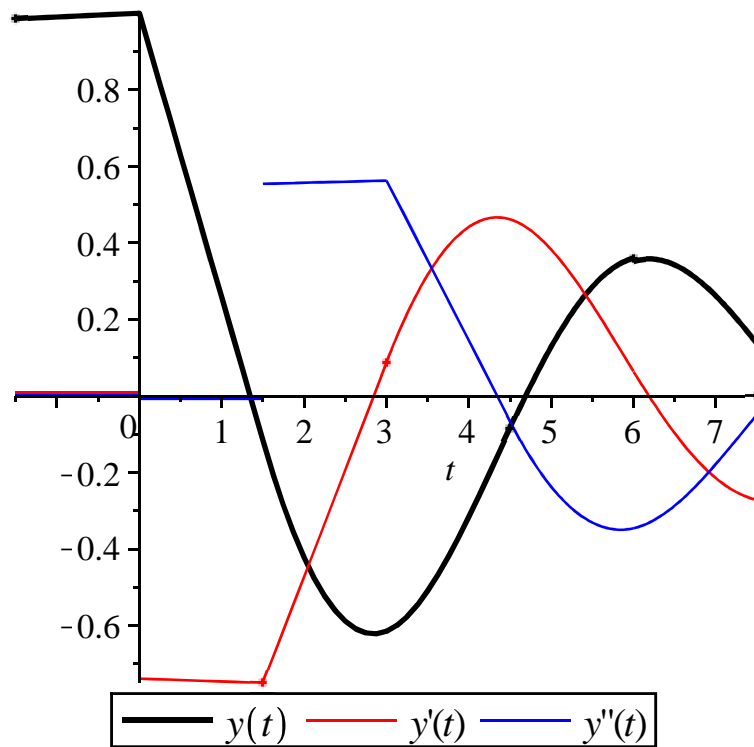
$$\begin{aligned}
 \text{eqrsol} := t \rightarrow \text{piecewise} & \left(-1.5 \leq t \text{ and } t \leq 0, e^{0.01t}, 0. \leq t \text{ and } t \leq 1.5, -75 e^{\frac{1}{100}t - \frac{3}{200}} \right. & (2.1) \\
 & + 1 + 75 e^{-\frac{3}{200}}, 1.5 \leq t \text{ and } t \leq 3.0, 5625 e^{\frac{1}{100}t - \frac{3}{100}} - \frac{3}{4}t - \frac{225}{4}t e^{-\frac{3}{200}} \\
 & \left. - \frac{583}{8} - \frac{43725}{8} e^{-\frac{3}{200}}, 3.0 \leq t \text{ and } t \leq 4.5, -421875 e^{\frac{1}{100}t - \frac{9}{200}} + \frac{9}{32}t^2 \right)
 \end{aligned}$$

$$\begin{aligned}
& + \frac{675}{32} t^2 e^{-\frac{3}{200}} + \frac{861}{16} t + \frac{64575}{16} t e^{-\frac{3}{200}} + \frac{172349}{32} + \frac{12926175}{32} e^{-\frac{3}{200}}, 4.5 \leq t \\
\text{and } t \leq 6.0, & 31640625 e^{\frac{1}{100} t - \frac{3}{50}} - \frac{9}{128} t^3 - \frac{5085}{256} t^2 - \frac{675}{128} t^3 e^{-\frac{3}{200}} \\
& - \frac{381375}{256} t^2 e^{-\frac{3}{200}} - \frac{2037435}{512} t - \frac{95504765637}{320000} t e^{-\frac{3}{200}} - \frac{254672294383}{640000} \\
& - \frac{19100422077977}{640000} e^{-\frac{3}{200}}, 6.0 \leq t \text{ and } t \leq 7.5, -2373046875 e^{\frac{1}{100} t - \frac{3}{40}} \\
& + \frac{27}{2048} t^4 + \frac{5864062501}{1200000000} t^3 + \frac{2352150000169}{1600000000} t^2 + \frac{2025}{2048} t^4 e^{-\frac{3}{200}} \\
& + \frac{3518437499}{9600000} t^3 e^{-\frac{3}{200}} + \frac{1411290000431}{12800000} t^2 e^{-\frac{3}{200}} + \frac{470401040653759}{1600000000} t \\
& + \frac{282240624337841}{12800000} t e^{-\frac{3}{200}} + \frac{23520074548285537}{800000000} \\
& + \frac{14112044730536863}{6400000} e^{-\frac{3}{200}} \Big)
\end{aligned}$$

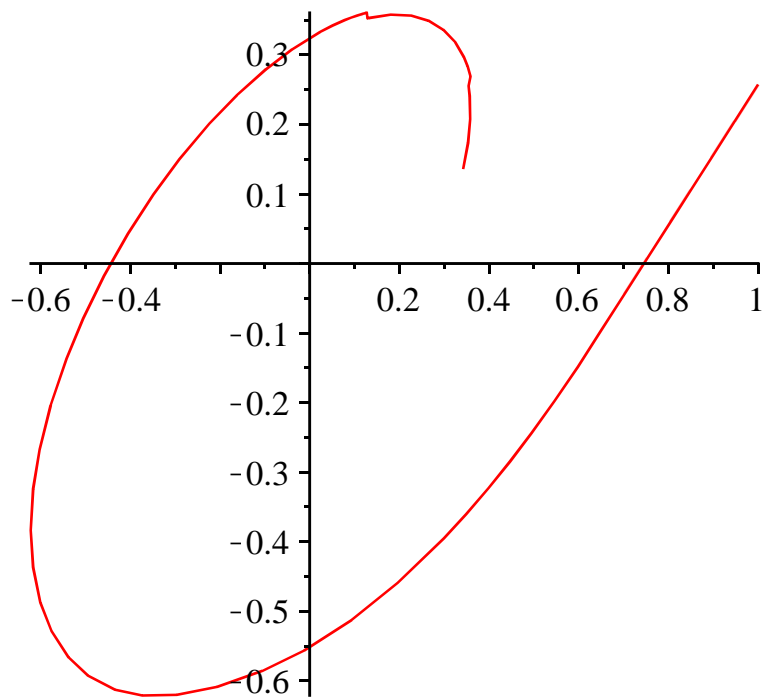
```

> plot([eqrsol(t),diff(eqrsol(t),t),diff(eqrsol(t),t,t)],t=-1.5.
.7.5,discont=true,color=[black,red,blue],thickness=[2,1,1],
legend=[typeset(y(t)),typeset(y,"","(",t,")"),typeset(y,"'",
(",t,")")]);#représentation cartésienne de la solution et de ses
dérivées première et seconde

```

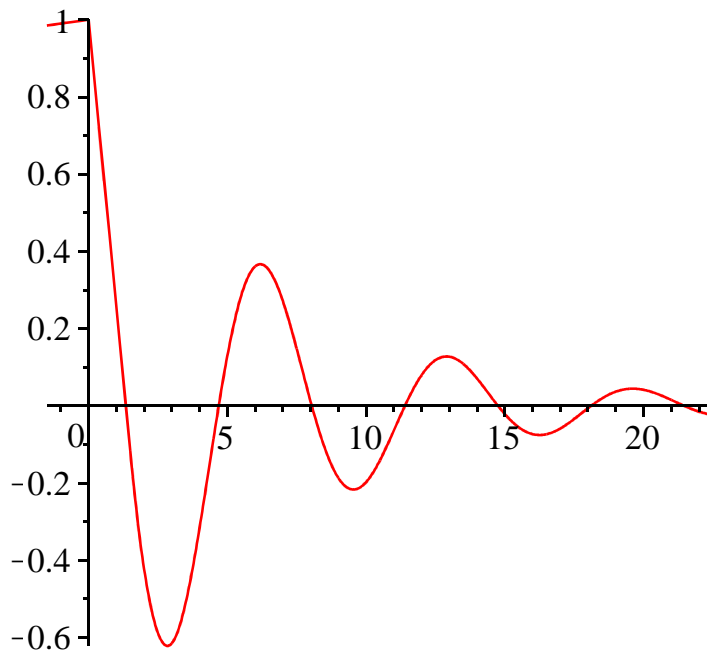


```
> plot([eqrsol(t-1),eqrsol(t),t=1..7.5]);#diagramme des phases
```

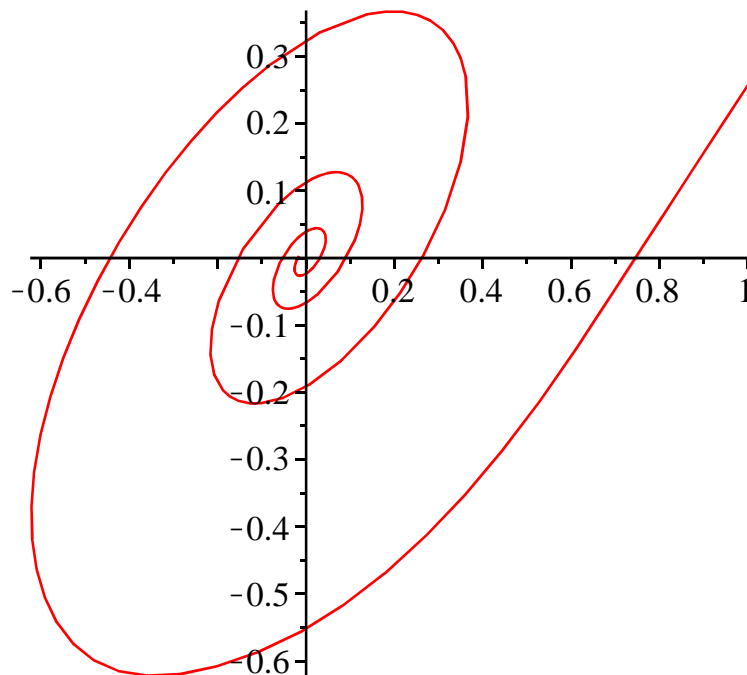


La nature cyclique amortie de la trajectoire-solution est mieux mise en valeur avec les procédures approchées.

```
> display(fhmpasn(1.5,0,0.75,t->exp(0.01*t),0,15));
#représentation graphique de la trajectoire-solution en
appelant fhmpasn
```

```
> T,a,b,h,d:=1.5,0,0.75,t->exp(0.01*t),0;
plot(['F(t-1)', 'F(t)', t=1..25]);#diagramme des phases
T, a, b, h, d := 1.5, 0, 0.75, t -> e0.01t, 0
```



Application économique 2. Le modèle du cycle de M. Kalecki

Cette section est basée sur l'article "A Macrodynamics Theory of the Business Cycle", publié en 1935 dans la revue *Econometrica*. Vu la complexité de son contenu économique, le modèle doit être présenté avant d'en explorer quelques propriétés formelles.

Présentation du modèle

Dans une économie fermée sans Etat, l'équilibre sur le marché des produits équivaut à l'égalité du revenu national, équivalent à l'offre agrégée, et de la demande globale, somme de la demande de consommation et de la demande d'investissement :

$$(1) \quad Y(t) = C(t) + I(t)$$

La consommation est proportionnelle au revenu et est influencée par une politique de stabilisation notée u :

$$(2) \quad C(t) = cY(t) - u(t) \text{ avec } c \in]0, 1[$$

Le point crucial du modèle de Kalecki est le traitement de l'investissement. Si $K(t)$ est le stock de capital à la date t , l'accumulation du capital en tant que formation brute de capital au sens de la comptabilité nationale est $K'(t)$. Il s'agit de livraisons de matériels qui ont été commandées dans le passé et qui ont exigé un délai de fabrication. En notant B les commandes de capital et en admettant que le délai de fabrication soit constant et égal à T , on a :

$$(3) \quad K'(t) = B(t - T)$$

Les firmes évaluent leurs besoins en biens d'équipement selon le principe de l'accélérateur simple. Si v est l'inverse de la productivité du capital de la fonction de production macroéconomique à facteurs complémentaires et si λ exprime la sensibilité des firmes à l'écart entre le stock de capital désiré, $vY(t)$, et le stock de capital effectif, $K(t)$, l'investissement désiré à la date t vaut :

$$(4) \quad B(t) = \lambda (vY(t) - K(t)) \text{ avec } \lambda \in]0, 1[$$

La demande d'investissement sur les marchés des produits correspond à la moyenne des sommes d'investissements désirés sur une période de durée T . En effet, à la date t , les équipements commandés avant la date $t - T$ ont été livrés, les équipements nécessaires après la date t ne sont pas encore connus, seuls les investissements désirés sur la période $[t - T, t]$ génèrent des dépenses qu'on suppose uniformes sur la durée de fabrication. On a alors :

$$(5) \quad I(t) = \frac{1}{T} \int_{t-T}^t B(\tau) d\tau$$

Concentrons-nous sur l'accumulation du capital.

{(1)+(2)} donne $I(t) = (1 - c)Y(t) + u(t)$

(4) implique $Y(t) = \frac{1}{v} \left[\frac{B(t)}{\lambda} + K(t) \right]$, d'où la relation :

$$(6) \quad I(t) = \frac{1 - c}{v} \left[\frac{B(t)}{\lambda} + K(t) \right] + u(t)$$

Or il y a des liens entre I , B et K .

D'après (5), $I(t) = \frac{1}{T} \int_{t-T}^t B(\tau) d\tau$. Mais, d'après (3), on a $B(\tau) = K'(\tau + T)$, d'où

$$I(t) = \frac{1}{T} \int_{t-T}^t K'(\tau + T) d\tau.$$

Effectuons le changement de variable $s = \tau + T$, qui implique $ds = d\tau$. Quand τ parcourt $[t - T, t]$,

s parcourt $[t, t + T]$. On a alors $I(t) = \frac{1}{T} \int_t^{t+T} K'(s) ds = \frac{1}{T} [K(t + T) - K(t)]$.

La relation (6) s'écrit alors :

$$\frac{1}{T} [K(t + T) - K(t)] = \frac{1 - c}{v} \left[\frac{B(t)}{\lambda} + K(t) \right] + u(t)$$

soit encore, en isolant $B(t)$:

$$B(t) = \frac{\lambda v}{(1-c)T} K(t+T) - \lambda \left[1 + \frac{v}{(1-c)T} \right] K(t) - \frac{\lambda v}{1-c} u(t)$$

A la date $t-T$, on aura :

$$B(t-T) = \frac{\lambda v}{(1-c)T} K(t) - \lambda \left[1 + \frac{v}{(1-c)T} \right] K(t-T) - \frac{\lambda v}{1-c} u(t-T)$$

soit encore, en tenant compte de (3) :

$$K'(t) = \frac{\lambda v}{(1-c)T} K(t) - \lambda \left[1 + \frac{v}{(1-c)T} \right] K(t-T) - \frac{\lambda v}{1-c} u(t-T)$$

En posant $a = \frac{\lambda v}{(1-c)T}$ et $b = \lambda \left[1 + \frac{v}{(1-c)T} \right]$, l'équation d'accumulation du capital s'écrit :

$$K'(t) = a K(t) - b K(t-T) - a T u(t-T)$$

Par la suite, on restera plus proche du modèle initial de Kalecki en prenant $u(t) = 0$.

Exploration des propriétés du modèle de Kalecki

L'équation d'accumulation du capital est $K'(t) = a K(t) - b K(t-T)$ pour $t \geq 0$ où

$$a = \frac{\lambda v}{(1-c)T} \text{ et } b = \lambda \left[1 + \frac{v}{(1-c)T} \right].$$

R. D. G Allen mène une étude complète avec la méthode basée sur la fonction de Lambert en posant que la solution générale est de la forme $K(t) = C e^{\lambda t}$. On propose ici une résolution alternative, basée sur la méthode des pas et donc sur les procédures exposées dans la première section. L'étude est menée en fixant $c = 0.5$, $v = 0.6$ et $\lambda = 0.3$. La fonction historique est $h : t \rightarrow 1$.

```
> c,nu,lambda:=0.5,0.6,0.3;#assignation des paramètres
économiques
T,d,h:='T',0,t->1;#assignation des paramètres
d'initialisation et désassignation de T
af:=unapply(-lambda*nu/((1-c)*T),T);
bf:=unapply(lambda*(1+nu/((1-c)*T)),t);
c, v, λ := 0.5, 0.6, 0.3
T, d, h := T, 0, t → 1
af := T → -  $\frac{0.3600000000}{T}$ 
bf := t → 0.3 +  $\frac{0.3600000000}{T}$  (3.2.1)
```

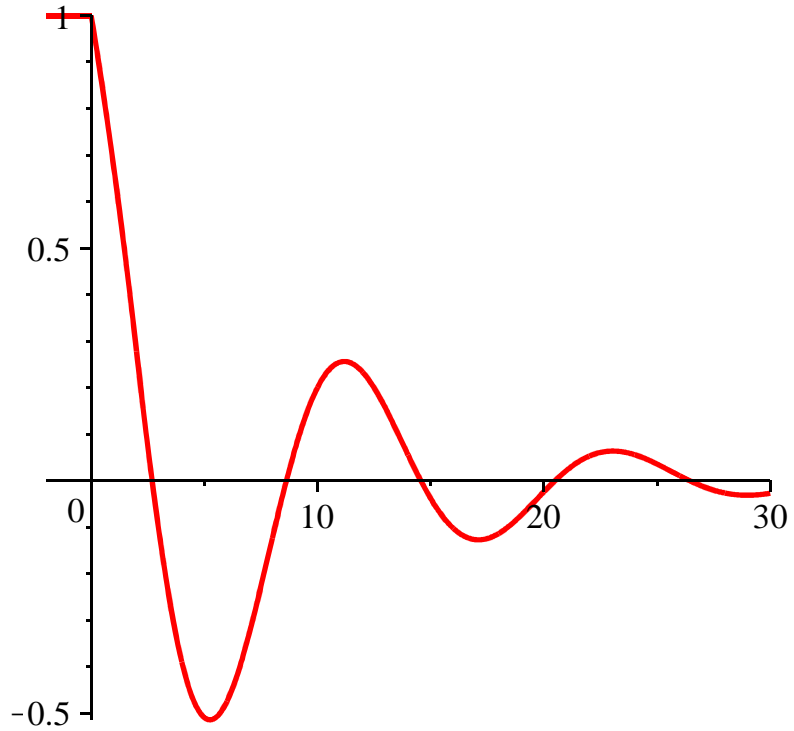
Les paramètres a et b dépendent uniquement du délai T et on va montrer empiriquement que les propriétés qualitatives de la dynamique en dépendent cruciallement, de sorte que $T=3.3$ va être une valeur de bifurcation pour la dynamique.

Cas $T=2$ Oscillations amorties.

On veut l'allure de la trajectoire sur un nombre conséquent de périodes. L'utilisation de la procédure **fhmpasn** est retenue de préférence à **fhmpas** pour ses performances de vitesse.

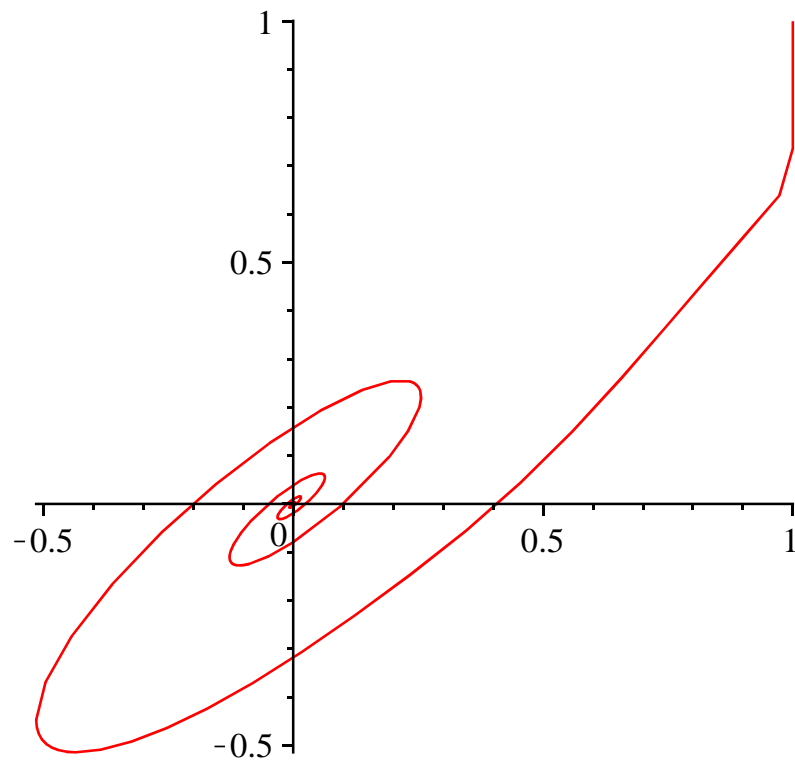
```
> T:=2;a,b,h,d:=af(T),bf(T),t->1,0;
display(fhmpasn(T,a,b,t->1,0,15),thickness=2);
#trajectoire-solution par appel à la procédure fhmpasn
T:=2
```

$a, b, h, d := -0.1800000000, 0.4800000000, t \rightarrow 1, 0$



Le stock de capital converge en oscillant vers une position d'équilibre, ce que confirme le diagramme des phases :

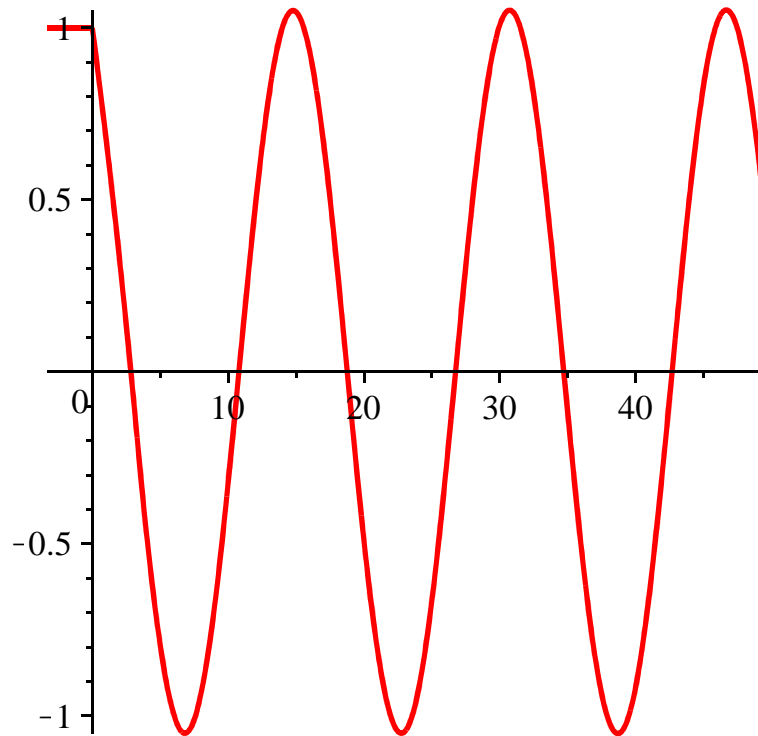
```
> plot(['F(t-1)', 'F(t)', t=0..50]);#construction du diagramme  
des phases par la méthode fhmpasndp+F
```



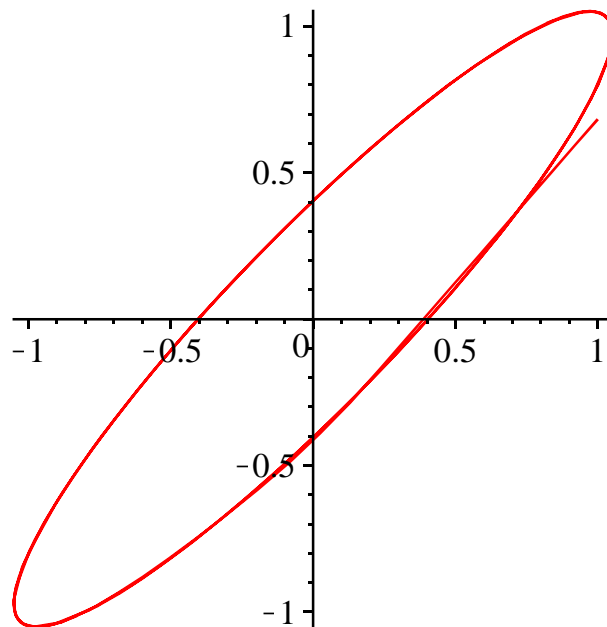
▼ **Cas $T=3.3$. Oscillations perpétuelles.**

Quand le délai est $T=3.3$, on voit apparaître un phénomène de cycle-limite.

```
> T:=3.3;a,b,h,d:=af(T),bf(T),t->1,0;  
display(fhmpasn(T,a,b,t->1,0,15),thickness=2);#trajectoire  
T:=3.3  
a,b,h,d := -0.1090909091, 0.4090909091, t->1, 0
```



```
> plot(['F(t-1)', 'F(t)', t=1..50]);#diagramme des phases
```

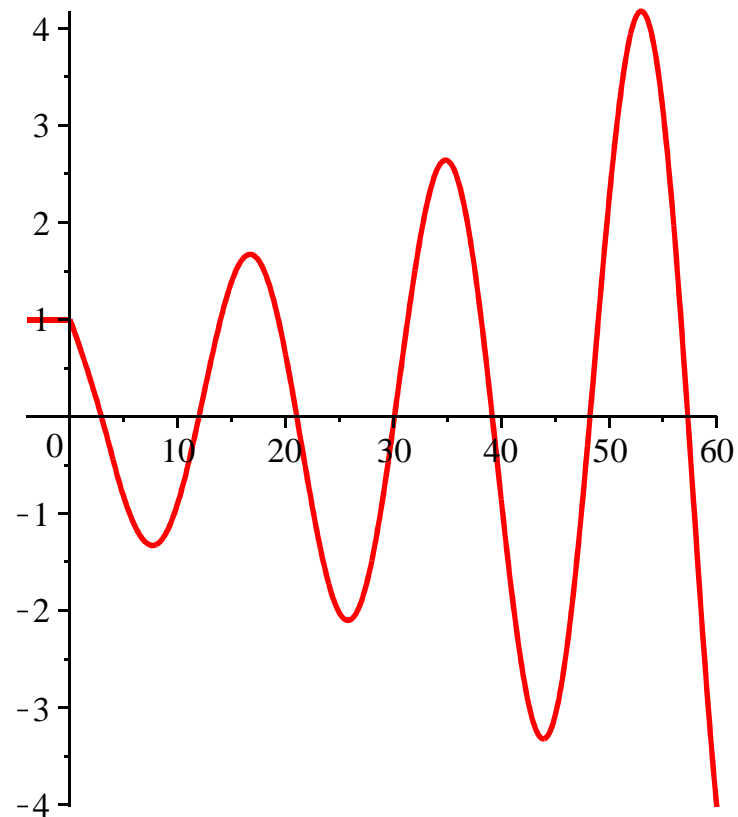


▼ **Cas $T=4$. Oscillations explosives.**

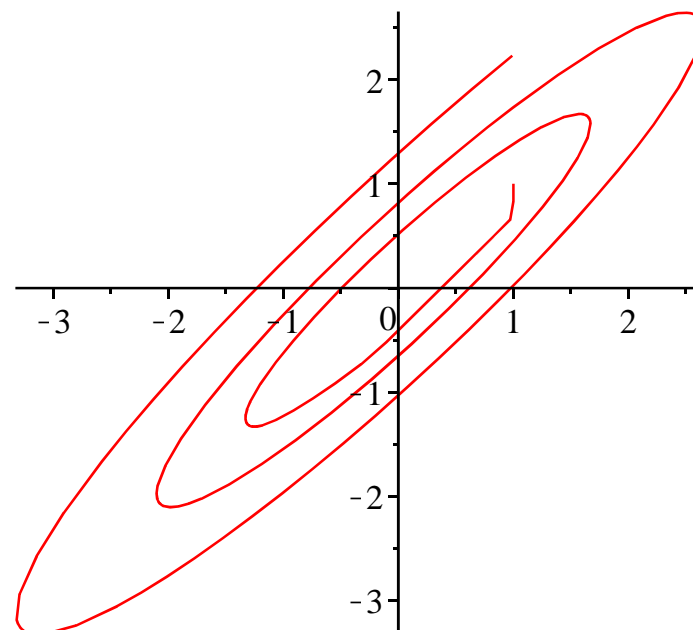
Pour $T=4$, la trajectoire du capital présente des oscillations explosives.

```
> T:=4;a,b,h,d:=af(T),bf(T),t->1,0;  
display(fhmpasn(T,a,b,t->1,0,15),thickness=2);#trajectoire  
T:=4
```

```
a,b,h,d := -0.09000000000, 0.3900000000, t->1, 0
```



```
> plot(['F(t-1)', 'F(t)', t=0..50]);#diagramme des phases
```



Bibliographie

- ALLEN R.D.G, *Théorie macroéconomique*, Armand Colin, collection U, 1969.
- FRISCH R. et H. HOLME, "The Characteristic Solutions of Mixed Difference and Differential Equations", *Econometrica*, avril 1935.
- KALECKI M., "A Macrodynamic Theory of Business Cycles", *Econometrica*, juillet 1935.
- KELLER A., *Time-delay Systems with Applications to Economic Dynamics and Control*, Lambert Academic Publishing, 2011.
- TINBERGEN J. "A Shipbuilding Cycle?", Jan Tinbergen – Selected Papers ([1931], 1959).