

(%o15) $[c^2 > 4 m r]$

$$(\%o16) \ y = \%k1 e^{\frac{\left(\sqrt{\frac{c^2}{m^2} - \frac{4r}{m} - \frac{c}{m}}\right) x}{2}} + \%k2 e^{\frac{\left(-\sqrt{\frac{c^2}{m^2} - \frac{4r}{m} - \frac{c}{m}}\right) x}{2}} + \frac{1}{r}$$

Pour delta nul

```
(%i17) forget(facts());
      assume(equal(c**2-4*m*r,0));
      ode2(equation, y, x);
```

(%o17) $[[c^2 > 4 m r]]$

(%o18) $[equal(c^2, 4 m r)]$

$$(\%o19) \ y = (\%k2 x + \%k1) e^{-\frac{c x}{2 m}} + \frac{4 m}{c^2}$$

Pour delta négatif

```
(%i20) forget(facts());
      assume(c**2-4*m*r<0);
      ode2(equation, y, x);
```

(%o20) $[[equal(c^2, 4 m r)]]$

(%o21) $[4 m r > c^2]$

$$(\%o22) \ y = e^{-\frac{c x}{2 m}} \left(\%k1 \sin \left(\frac{\sqrt{\frac{4r}{m} - \frac{c^2}{m^2}} x}{2} \right) + \%k2 \cos \left(\frac{\sqrt{\frac{4r}{m} - \frac{c^2}{m^2}} x}{2} \right) \right) + \frac{1}{r}$$

Chapitre 15

Séries de Fourier

Pour le paragraphe « Utilisation du package `piecewise` », il faut utiliser une bibliothèque, pour la récupérer, cliquer deux fois sur le lien ci-contre : lien [Piecewise](#).

Enregistrez ce fichier dans le répertoire de votre choix et n'oubliez pas de donner le chemin complet lors de l'utilisation :

```
load("C:/chemin-complet/pw.mac");
```

15.1 Définition de la fonction périodique

On suppose que l'on se donne une fonction périodique de période T , définie sur \mathbb{R} et qu'une seule expression est donnée sur l'intervalle $[a; b]$ de longueur T . Pour les fonctions définies par morceaux sur l'intervalle $[a; b]$, le lecteur est renvoyé à la section sur le package `piecewise`. La notation g désignera la fonction définie sur $[a; b]$. La notation f désignera la fonction définie sur \mathbb{R} .

```
(%i1) a:-%pi;  
      b:%pi;  
      T:b-a;  
      %omega:2*%pi/T;  
      g(t):=t;  
      f(t):=g(t-(b-a)*floor((t-a)/(b-a)));
```

```
(%o1) - pi
```

```
(%o2) pi
```

```
(%o3) 2 pi
```

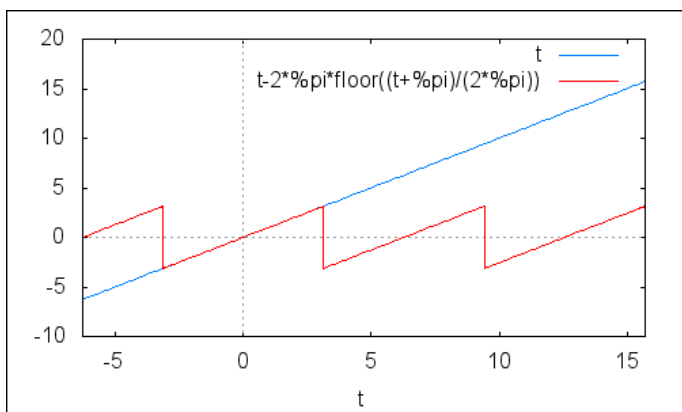
```
(%o4) 1
```

```
(%o5) g(t) := t
```

```
(%o6) f(t) := g(t - (b - a) floor((t - a) / (b - a)))
```

15.2 Courbe de la fonction étudiée

```
(%i7) wxplot2d([g(t), f(t)], [t,a-b,3*b-2*a])$
```



```
(%t7)
```

15.3 Calcul des coefficients de Fourier

15.3.1 Calcul de la moyenne sur une période : a_0

```
(%i8) a0:1/T*integrate(g(t),t,a,b);
```

```
(%o8) 0
```

15.3.2 Calcul des coefficients a_n et b_n

```
(%i9) (declare(n,integer), n>0);
```

```
(%o9) n > 0
```

```
(%i10) an:2/T*integrate(g(t)*cos(n*%omega*t),t,a,b);
define(a(n),an);
```

```
(%o10) 0
```

```
(%o11) a(n) := 0
```

```
(%i12) bn:2/T*integrate(g(t)*sin(n*%omega*t),t,a,b);
define(b(n),bn);
```

```
(%o12) - 2(-1)^n
n
```

```
(%o13) b(n) := - 2(-1)^n
n
```

N désigne le nombre de termes affichés. La fonction `makelist` permet de créer la liste des entiers de 1 à N et `map` permet de calculer les images par la fonction a (ou b) des nombres de la liste précédente.

```
(%i14) N:4$
      map('a, makelist(i,i,1,N));
      map('b, makelist(i,i,1,N));
```

```
(%o15) [0,0,0,0]
```

```
(%o16) [2, -1, 2/3, -1/2]
```

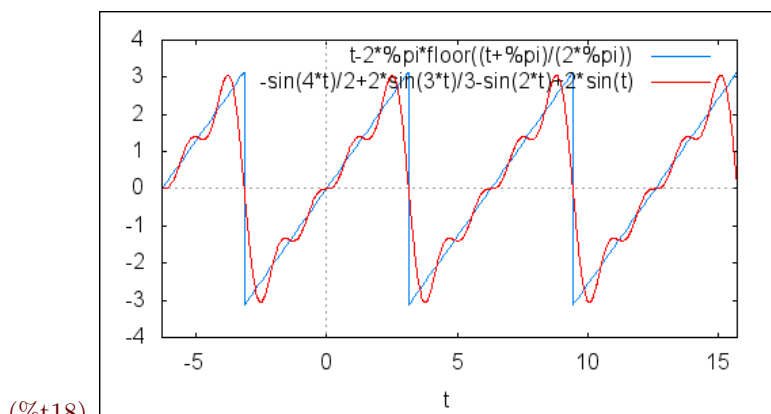
15.4 Somme de Fourier partielle

On désigne par N le nombre de termes de la somme de série de Fourier partielle :

```
(%i17) Sf(t,N):=a0+sum(a(n)*cos(n*%omega*t)+b(n)*sin(n*%omega*t), n, 1, N);
```

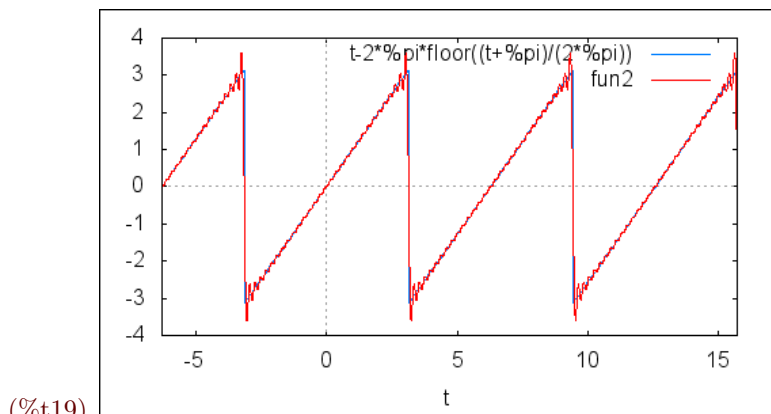
```
(%o17) Sf(t,N) := a0 + \sum_{n=1}^N a(n) \cos(n\omega t) + b(n) \sin(n\omega t)
```

```
(%i18) wxplot2d([f(t),Sf(t,4)], [t,a-b,3*b-2*a])$
```



```
(%t18)
```

```
(%i19) wxplot2d([f(t),Sf(t,30)], [t,a-b,3*b-2*a])$
```



```
(%t19)
```

15.5 Spectre

Le spectre de f est donné par la suite (A_n) , où $A_n = \sqrt{a_n^2 + b_n^2}$, pour $n \geq 1$.

```
(%i20) define(A(n), sqrt(a(n)^2+b(n)^2));
```

```
(%o20) A(n) :=  $\frac{2}{|n|}$ 
```

15.6 Valeur efficace et formule de Parseval

La valeur efficace est donnée par : $f_{\text{eff}}^2 = \frac{1}{T} \int_a^{T+a} f^2(t) dt$.

La formule de Parseval est : $f_{\text{eff}}^2 = a_0^2 + \frac{1}{2} \sum_{n=1}^{+\infty} (a_n^2 + b_n^2)$.

La valeur efficace est approchée par : $P(N) = a_0^2 + \frac{1}{2} \sum_{n=1}^N (a_n^2 + b_n^2)$.

```
(%i21) P(N):=a0^2+1/2*sum(a(n)^2+b(n)^2, n, 1, N);
```

```
(%o21) P(N) :=  $a_0^2 + \frac{1}{2} \sum_{n=1}^N a(n)^2 + b(n)^2$ 
```

```
(%i22) N:10;
```

```
for n:1 thru N step 1 do print("P(",n,")=", float(P(n)), " ; ");
```

```
(%o22) 10P(1) = 2.0; P(2) = 2.5; P(3) = 2.7222222222222222; P(4) = 2.8472222222222222; P(5) =
2.9272222222222222; P(6) = 2.9827777777777778; P(7) = 3.02359410430839; P(8) = 3.05484410430839; P(9) =
3.079535462333082; P(10) = 3.099535462333082;
```

```
(%o23) done
```

```
(%i24) Peff:float(1/T*integrate(f(t)**2,t,a,b));
```

```
(%o24) 3.289868133696452
```

Erreur relative

```
(%i25) for n:1 thru N step 1 do print("err(",n,")=", float((Peff-P(n))/Peff), " ; ");
```

```
err(1) = 0.39207289814597; err(2) = 0.24009112268247; err(3) = 0.17254366692091; err(4) =
0.13454822305503; err(5) = 0.11023113898087; err(6) = 0.093344275040481; err(7) = 0.080937599492439; err(8) =
0.07143873852597; err(9) = 0.063933465663575; err(10) = 0.057854194645034;
```

```
(%o25) done
```

15.7 Utilisation du package piecewise

Package à utiliser pour les fonctions définie par morceaux sur une période.

```
(%i26) kill(all);
```

```
(%o0) done
```

```
(%i1) load("maxima-BTS_V_1/pw.mac");
```

```
(%o1) maxima - BTS_V_1/pw.mac
```

On suppose que la fonction g est donnée sur les intervalles $\left[a_{-1}; b_{-1}\right]$, \dots , $\left[a_r; b_r\right]$ par les expressions $g_{-1}(x)$, \dots , $g_r(x)$. Dans Maxima et avec la bibliothèque pw, on peut définir la fonction g directement par :

```
define(g(t), piecewise([a_{-1}, g_{-1}(t), b_{-1}, a_{-2}, g_{-2}(t), b_{-2}, \dots, a_r, g_r(t), b_r], t));
```

```
(%i2) a:0;
      b:4*%pi;
      T:b-a;
      %omega:2*%pi/T;
      define(g(t), piecewise([a, %pi-t, %pi, 0, b/2, 1, b], t));
      f(t):=g(t-(b-a)*floor((t-a)/(b-a)));
      wxplot2d([f(t)], [t,a-b,3*b-2*a], [y,-0.5,3.5])$
```

```
(%o2) 0
```

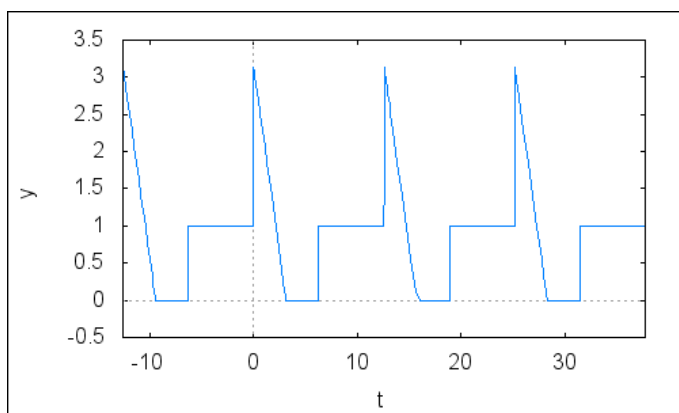
```
(%o3) 4π
```

```
(%o4) 4π
```

```
(%o5) 1/2
```

```
(%o6) g(t) := (π - t) (signum(t) - signum(t - π)) / 2 + (signum(t - 2π) - signum(t - 4π)) / 2
```

```
(%o7) f(t) := g(t - (b - a) floor((t - a) / (b - a)))
```



```
(%t8)
```

La fonction `pwint` est la fonction intégrate utilisable sur les fonctions définies par `pw`, c'est l'intégrale d'une fonction définie par morceaux.

```
(%i9) a0:radcan(1/T*pwint(g(t), t, a, b));
```

```
(%o9) (π + 4) / 8
```

```
(%i10) declare(n, integer);
       assume(n>0);
```

```
(%o10) done
```

```
(%o11) [n > 0]
```

La fonction `radcan`, sert à simplifier l'expression obtenue (simplification canonique).

```
(%i12) an:radcan(2/T*pwint(g(t)*cos(n*%omega*t), t, a, b));
       define(a(n), an);
```

```
(%o12) - (2 cos(πn/2) - 2) / (π n^2)
```

$$(\%o13) \ a(n) := -\frac{2 \cos\left(\frac{\pi n}{2}\right) - 2}{\pi n^2}$$

```
(%i14) bn:radcan(2/T*pwint(g(t)*sin(n*%omega*t), t, a, b));
define(b(n),bn);
```

$$(\%o14) \ -\frac{2 \sin\left(\frac{\pi n}{2}\right) - n(-1)^n + (1 - \pi) n}{\pi n^2}$$

$$(\%o15) \ b(n) := -\frac{2 \sin\left(\frac{\pi n}{2}\right) - n(-1)^n + (1 - \pi) n}{\pi n^2}$$

```
(%i16) N:4$
map('a, makelist(i,i,1,N));
map('b, makelist(i,i,1,N));
```

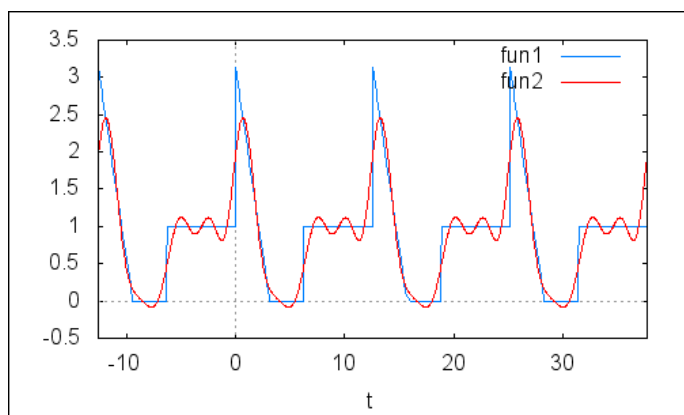
$$(\%o17) \ \left[\frac{2}{\pi}, \frac{1}{\pi}, \frac{2}{9\pi}, 0\right]$$

$$(\%o18) \ \left[-\frac{4 - \pi}{\pi}, -\frac{2(1 - \pi) - 2}{4\pi}, -\frac{3(1 - \pi) + 1}{9\pi}, -\frac{4(1 - \pi) - 4}{16\pi}\right]$$

```
(%i19) Sf(t,N):=a0+sum(a(n)*cos(n*%omega*t)+b(n)*sin(n*%omega*t), n, 1, N);
```

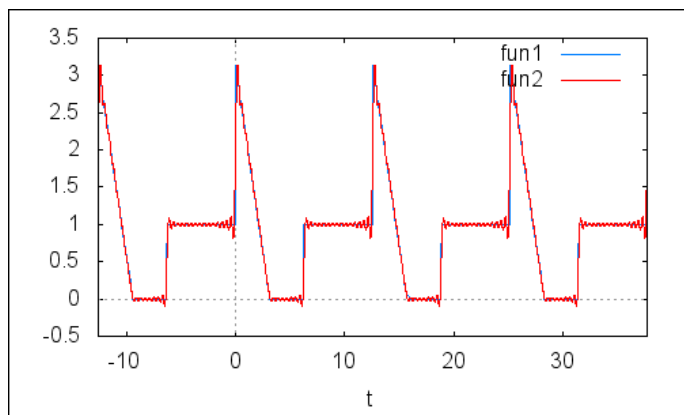
$$(\%o19) \ Sf(t, N) := a_0 + \sum_{n=1}^N a(n) \cos(n\omega t) + b(n) \sin(n\omega t)$$

```
(%i20) wxplot2d([f(t),Sf(t,5)], [t,a-b,3*b-2*a])$
```



```
(%t20)
```

```
(%i21) wxplot2d([f(t),Sf(t,30)], [t,a-b,3*b-2*a])$
```



```
(%t21)
```


Chapitre 16

Transformée de Laplace

16.1 Définition de l'échelon unité

Échelon unité : $v(t) = 0$ si $t < 0$ et 1 si $t > 0$;

Dans Maxima c'est la fonction u suivante qui est définie : $u(t) = 0$ si $t \leq 0$ et 1 si $t > 0$;

Dans la pratique en mathématiques, c'est la fonction v que l'on utilise mais avec maxima c'est avec la fonction u que cela fonctionne correctement !

```
(%i1) u(t):=unit_step(t);  
      u(-1);  
      u(0);  
      u(1);
```

```
(%o1) u(t) := unit_step(t)
```

```
(%o2) 0
```

```
(%o3) 0
```

```
(%o4) 1
```

La fonction v peut être définie de la façon suivante :

```
(%i5) v(t):=unit_step(t)+kron_delta(t,0);  
      v(-1);  
      v(0);  
      v(1);
```

```
(%o5) v(t) := unit_step(t) + kron_delta(t,0)
```

```
(%o6) 0
```

```
(%o7) 1
```

```
(%o8) 1
```

Les calculs sur la transformée de Laplace ne changent pas quel que soit le choix d'une des deux définitions précédentes.

16.2 Calculs de transformées de Laplace

Menu/Calculs/Transformée de Laplace...

(%i9) `laplace(u(t), t, p);`

$$(\%o9) \frac{1}{p}$$

(%i10) `laplace(t*u(t), t, p);`

$$(\%o10) \frac{1}{p^2}$$

Ci-dessous, $\Gamma(n+1)$ désigne $n!$.

(%i11) `laplace(t^n*u(t), t, p);`

Isn + 1 positive, negative, or zero? positive;

$$(\%o11) \gamma(n+1) p^{-n-1}$$

Retard $u(t-1)$

(%i12) `laplace(u(t-tau), t, p);`

$$(\%o12) \frac{e^{-p\tau}}{p}$$

Exponentielle

(%i13) `laplace(exp(-a*t), t, p);`

$$(\%o13) \frac{1}{p+a}$$

Fonctions trigonométriques

(%i14) `laplace(cos(omega*t)*u(t), t, p);`

$$(\%o14) \frac{p}{p^2 + \omega^2}$$

(%i15) `laplace(sin(omega*t)*u(t), t, p);`

$$(\%o15) \frac{\omega}{p^2 + \omega^2}$$

Propriétés : les deux premières ne sont pas reconnues...

(%i16) `laplace(f(alpha*t)*u(alpha*t), t, p);`

$$(\%o16) \text{laplace}(f(\alpha t) \text{unit_step}(\alpha t), t, p)$$

(%i17) `laplace(f(t-tau), t, p);`

$$(\%o17) \text{laplace}(f(t - \tau), t, p)$$

(%i18) `laplace(exp(-a*t)*f(t), t, p);`

Isp + a positive, negative, or zero? positive;

$$(\%o18) \text{laplace}(f(t), t, p)|_{p=p+a}$$

(%i19) `laplace(diff(f(t), t), t, p);`

$$(\%o19) p \text{laplace}(f(t), t, p) - f(0)$$

(%i20) `laplace(diff(f(t), t, 2), t, p);`

$$(%o20) - \frac{d}{dt} f(t) \Big|_{t=0} + p^2 \text{laplace}(f(t), t, p) - f(0) p$$

(%i21) `laplace(-t*f(t), t, p);`

$$(%o21) \frac{d}{dp} \text{laplace}(f(t), t, p)$$

(%i22) `laplace(integrate(f(s), s, 0, t), t, p);`

$$(%o22) \frac{\text{laplace}(f(t), t, p)}{p}$$

16.3 Calculs de transformées inverse de Laplace

Menu/Calculs/Transformée inverse de Laplace... Attention, il faut multiplier les résultats par $u(t)$.

(%i23) `ilt(1/p, p, t);`

$$(%o23) 1$$

Si le dénominateur est du second degré avec un discriminant strictement positif :

(%i24) `ilt((p+1)/(p^2+4*p-1), p, t);`

$$(%o24) e^{-2t} \left(\cosh(\sqrt{5}t) - \frac{\sinh(\sqrt{5}t)}{\sqrt{5}} \right)$$

Des fonctions trigonométriques hyperboliques ! Pour éviter cela, on utilise `exponentialize` :

(%i25) `exponentialize(ilt((p+1)/(p^2+4*p-1), p, t));`

$$(%o25) e^{-2t} \left(\frac{e^{\sqrt{5}t} + e^{-\sqrt{5}t}}{2} - \frac{e^{\sqrt{5}t} - e^{-\sqrt{5}t}}{2\sqrt{5}} \right)$$

On peut aussi développer !

(%i26) `expand(exponentialize(ilt((p+1)/(p^2+4*p-1), p, t)));`

$$(%o26) - \frac{e^{\sqrt{5}t-2t}}{2\sqrt{5}} + \frac{e^{\sqrt{5}t-2t}}{2} + \frac{e^{-\sqrt{5}t-2t}}{2\sqrt{5}} + \frac{e^{-\sqrt{5}t-2t}}{2}$$

Si le dénominateur est du second degré avec un discriminant nul :

(%i27) `ilt((p+1)/(p^2+2*p+1), p, t);`

$$(%o27) e^{-t}$$

Rien à modifier ! Si le dénominateur est du second degré avec un discriminant strictement négatif :

(%i28) `ilt((p+1)/(p^2+p+1), p, t);`

$$(%o28) e^{-\frac{t}{2}} \left(\frac{\sin\left(\frac{\sqrt{3}t}{2}\right)}{\sqrt{3}} + \cos\left(\frac{\sqrt{3}t}{2}\right) \right)$$

Attention : "ilt" (inverse laplace transform) ne fonctionne que pour les fonctions rationnelles pour lesquelles le dénominateur est factorisable en produits de facteurs linéaires et quadratiques (dont on sait calculer la valeur exacte des racines). Cela pose des problèmes si l'on veut travailler avec des retards. Il faut dans ce cas utiliser un autre logiciel ou trouver une autre solution !

16.4 Résolutions d'équations différentielles avec la transformée de Laplace

– Résolution directe : Menu/Equations/Résoudre une équation différentielle avec Laplace...

```
(%i29) desolve(['diff(y(t), t, 2) + 5*diff(y(t), t) + 4*y(t) = u(t)], [y(t)]);
```

$$(\%o29) \quad y(t) = -\frac{e^{-4t} \left(4 \left(\frac{d}{dt} y(t) \Big|_{t=0} \right) + 4y(0) - 1 \right)}{12} + \frac{e^{-t} \left(\frac{d}{dt} y(t) \Big|_{t=0} + 4y(0) - 1 \right)}{3} + \frac{1}{4}$$

Contrairement à ode2 les paramètres restants sont les valeurs de $y(0)$ et $y'(0)$ et non les deux réels quelconques notés %k1 et %k2 dans maxima. Pour faciliter la lecture des résultats on pose les conditions initiales $f(0) = y0$ et $f'(0) = yp0$:

```
(%i30) atvalue(y(t), t=0, y0);
       atvalue('diff(y(t), t), t= 0, yp0);
```

```
(%o30) y0
```

```
(%o31) yp0
```

On affiche le changement :

```
(%i32) '>%i19;
```

```
(%o32) p laplace(f(t), t, p) - f(0)
```

On développe :

```
(%i33) expand(%o22);
```

$$(\%o33) \quad \frac{\text{laplace}(f(t), t, p)}{p}$$

Avec les conditions initiales : $y(0)=1$ et $y'(0)=2$:

```
(%i34) subst(1, y0, %o23);
```

```
(%o34) 1
```

```
(%i35) subst(2, yp0, %o24);
```

$$(\%o35) \quad e^{-2t} \left(\cosh(\sqrt{5}t) - \frac{\sinh(\sqrt{5}t)}{\sqrt{5}} \right)$$

– Par étapes :

```
(%i36) equation: 'diff(y(t), t, 2) + 5*diff(y(t), t) + 4*y(t) = u(t);
```

$$(\%o36) \quad \frac{d^2}{dt^2} y(t) + 5 \left(\frac{d}{dt} y(t) \right) + 4y(t) = \text{unit_step}(t)$$

```
(%i37) equationtransformee:laplace(equation,t,p);
```

```
(%o37) -yp0-py0+5(p laplace(y(t),t,p) - y0)+p^2 laplace(y(t),t,p)+4 laplace(y(t),t,p) = 1/p
```

On impose des conditions initiales :

```
(%i38) atvalue(y(t), t=0, 1);
       atvalue('diff(y(t), t), t= 0, 2);
       equationtransformee:laplace(equation,t,p);
```

```
(%o38) 1
```

```
(%o39) 2
```

```
(%o40) 5(p laplace(y(t),t,p) - 1) + p^2 laplace(y(t),t,p) + 4 laplace(y(t),t,p) - p - 2 = 1/p
```

On résout l'équation linéaire :

```
(%i41) equationsolution: solve(equationtransformee, 'laplace(y(t), t, p));
```

```
(%o41) [laplace(y(t),t,p) = (p^2 + 7p + 1) / (p^3 + 5p^2 + 4p)]
```

On donne directement l'original. Si map et lambda paraissent obscurs, on peut faire un copier coller !

```
(%i42) map( lambda( [eq], ilt(eq, p, t)), equationsolution);
```

```
(%o42) [y(t) = 5e^-t / 3 - 11e^-4t / 12 + 1/4]
```

On peut aussi décomposer le calcul en passant par la décomposition en éléments simples.

```
(%i43) map( lambda( [expression], partfrac(expression, p)), equationsolution);
```

```
(%o43) [laplace(y(t),t,p) = -11 / (12(p+4)) + 5 / (3(p+1)) + 1/(4p)]
```

```
(%i44) map( lambda( [expression], ilt(expression, p, t)), %);
```

```
(%o44) [y(t) = 5e^-t / 3 - 11e^-4t / 12 + 1/4]
```

L'ordre des termes a été modifié !

Chapitre 17

Transformée en z

Pour ce chapitre, il faut utiliser deux bibliothèques, pour les récupérer, cliquer deux fois sur les liens ci-contre : lien 1 [z-transform](#) et lien 2 [supplement-z-transform](#).

Enregistrez ces fichiers dans le répertoire de votre choix et n'oubliez pas de donner le chemin complet lors de l'utilisation :

```
load("C:/chemin-complet/z-transform.mac");  
load("C:/chemin-complet/supplement-z-transform.mac");
```

17.1 Introduction

Définition des fonctions `tz` (transformée en z), `tiz` (transformée inverse en z) `tiz` retourne une liste, les premiers termes sont donnés lorsqu'ils sont nuls, le dernier terme de la liste (disons d'ordre r dans la liste) donne l'expression du signal discret en fonction de n pour n plus grand ou égal à r . On peut obtenir ce terme générique directement avec la fonction `tizgen`. En utilisant la fonction `plotliste`, on peut représenter le signal discret.

```
(%i1) load("z-transform.mac")$  
      load("supplement-z-transform.mac")$
```

17.2 Transformée en z

Des exemples :

Dirac (impulsion unité discrète) et sa transformée

```
(%i3) dirac(n):=kron_delta(n,0)$  
      tz(dirac(n));
```

```
(%o4) 1
```

Dirac retardé de k et sa transformée

```
(%i5) d(n,k):=kron_delta(n,k)$  
      tz(d(n,k));
```

```
(%o6)  $\frac{1}{z^k}$ 
```

Échelon unité discret

```
(%i7) u(n):=unit_step(n)$
      tz(u(n));
```

$$(\%o8) \frac{z}{z-1}$$

Rampe unité causale discrète : n

```
(%i9) tz(n*u(n));
      factor(ev(''(tz(n*u(n))),diff));
      tz(n);
```

$$(\%o9) -z \left(\frac{d}{dz} \frac{z}{z-1} \right)$$

$$(\%o10) \frac{z}{(z-1)^2}$$

$$(\%o11) \frac{z}{(z-1)^2}$$

Carré

```
(%i12) tz(n^2*u(n));
      factor(ev(''(tz(n^2*u(n))),diff));
      tz(n^2);
```

$$(\%o12) -z \left(\frac{d}{dz} \left(-z \left(\frac{d}{dz} \frac{z}{z-1} \right) \right) \right)$$

$$(\%o13) \frac{z(z+1)}{(z-1)^3}$$

$$(\%o14) \frac{z(z+1)}{(z-1)^3}$$

Exponentiel

```
(%i15) tz(a^n*u(n));
      factor(ev(''(tz(a^n*u(n))),nouns));
      tz(a^n);
```

$$(\%o15) \text{substitute} \left(\frac{z}{a}, z, \frac{z}{z-1} \right)$$

$$(\%o16) \frac{z}{z-a}$$

$$(\%o17) \frac{z}{z-a}$$

```
(%i18) tz(b^(n-1)*u(n-1));
```

$$(\%o18) \frac{1}{z-b}$$

```
(%i19) tz(exp(a*n));
```

$$(\%o19) \frac{z}{z-e^a}$$

Trigonométrie


```
(%i20) tz(sin(a*n));
      tz(cos(a*n));
```

$$(\%o20) \frac{\sin(a) z}{z^2 - 2 \cos(a) z + 1}$$

$$(\%o21) \frac{z (z - \cos(a))}{z^2 - 2 \cos(a) z + 1}$$

```
(%i22) factor(ev(''(tz(b^n*cos(a*n))), nouns));
      factor(ev(''(tz(b^n*sin(a*n))), nouns));
```

$$(\%o22) \frac{z (z - \cos(a) b)}{z^2 - 2 \cos(a) b z + b^2}$$

$$(\%o23) \frac{\sin(a) b z}{z^2 - 2 \cos(a) b z + b^2}$$

Des propriétés Linéarité

```
(%i24) tz(lambda*x(n)+mu*y(n));
```

$$(\%o24) z_transform(x(n), n, z) \lambda + \mu z_transform(y(n), n, z)$$

Retard

```
(%i25) tz(x(n-2)*u(n-2));
      y(n):=n^2*u(n);
      factor(ev(''(tz(y(n-2))), diff));
```

$$(\%o25) \frac{z_transform(x(n), n, z)}{z^2}$$

$$(\%o26) y(n) := n^2 u(n)$$

$$(\%o27) \frac{z + 1}{(z - 1)^3 z}$$

L'avance ne fonctionne pas encore!

```
(%i28) tz(x(n+3)*unit_step(n+3));
      y(n):=n^2*u(n);
      ev(''(tz(y(n+2))), diff);
      tz(expand((n+2)^2));
```

$$(\%o28) z_transform(x(n), n, z) z^3$$

$$(\%o29) y(n) := n^2 u(n)$$

$$(\%o30) -z^3 \left(-z \left(\frac{2z}{(z-1)^3} - \frac{2}{(z-1)^2} \right) + \frac{z}{(z-1)^2} - \frac{1}{z-1} \right)$$

$$(\%o31) \frac{z(z+1)}{(z-1)^3} + \frac{4z}{z-1} + \frac{4z}{(z-1)^2}$$

Multiplication par a^n

```
(%i32) tz(a^n*x(n));
      y(n):=n^2*u(n);
      tz(a^n*y(n));
      factor(ev(''(tz(a^n*y(n))), nouns));
```

$$(\%o32) substitute\left(\frac{z}{a}, z, z_transform(x(n), n, z)\right)$$

(%o33) $y(n) := n^2 u(n)$

(%o34) `substitute` $\left(\frac{z}{a}, z, -z \left(\frac{d}{dz} \left(-z \left(\frac{d}{dz} \frac{z}{z-1}\right)\right)\right)\right)$

(%o35) $\frac{az(z+a)}{(z-a)^3}$

17.3 Transformée en z inverse

Le premier terme donne la valeur du signal discret pour $n=0$. Le dernier terme est l'expression du terme général du signal.

(%i36) `tiz(z/(z-1));`

(%o36) [1, 1]

(%i37) `tiz(z/(z-1)^2);`

(%o37) [0, n]

(%i38) `tiz((z*(z+1))/(z-1)^3);`

(%o38) [0, n²]

(%i39) `tiz(z/(z-b));`

(%o39) [1, bⁿ]

(%i40) `tiz(z/(z-%e^a));`

(%o40) [1, e^{an}]

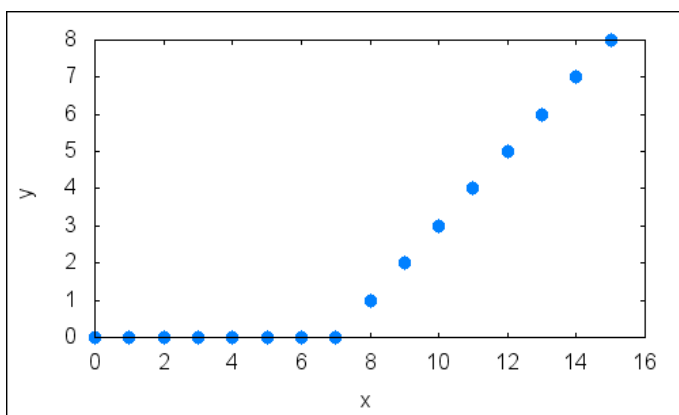
(%i41) `tiz(z^(6)/(z-1)^2);`

(%o41) [∞ , n + 5]

(%i42) `define(f(z),z^(-6)/(z-1)^2);
signal:tiz(f(z));
plotliste(signal,15);`

(%o42) $f(z) := \frac{1}{(z-1)^2 z^6}$

(%o43) [0, 0, 0, 0, 0, 0, 0, n - 7]



(%t44)

(%o44)

17.4 Transformée en z à partir d'une liste

```
(%i45) tzliste([1.1,1.2,1.3,1.4,1.5*n]);
```

$$(\%045) \frac{1.5 \left(\frac{4z}{z-1} + \frac{z}{(z-1)^2} \right)}{z^4} + \frac{1.2}{z} + \frac{1.3}{z^2} + \frac{1.4}{z^3} + 1.1$$

```
(%i46) tzliste([0,0,n-2]);
```

$$(\%046) \frac{1}{(z-1)^2 z}$$

```
(%i47) factor(tzliste([n+1]));
```

$$(\%047) \frac{z^2}{(z-1)^2}$$

17.5 Exemples d'utilisations

```
(%i48) tz(n^2+3*n+2);
```

$$(\%048) \frac{z(z+1)}{(z-1)^3} + \frac{2z}{z-1} + \frac{3z}{(z-1)^2}$$

```
(%i49) factor(ev(''(tz(n*2^n)),diff, nouns));
```

$$(\%049) \frac{2z}{(z-2)^2}$$

```
(%i50) factor(tz(n-3));
factor(tzliste([0, 0, 0, n-3]));
```

$$(\%050) - \frac{z(3z-4)}{(z-1)^2}$$

$$(\%051) \frac{1}{(z-1)^2 z^2}$$

```
(%i52) factor(tz(expand((n+2)^2)));
factor(z^3*(z+1)/(z-1)^3-z);
```

$$(\%052) \frac{z(4z^2-3z+1)}{(z-1)^3}$$

$$(\%053) \frac{z(4z^2-3z+1)}{(z-1)^3}$$

```
(%i54) tiz(2*z/((z-1)*(z-3)));
```

$$(\%054) [0, 3^n - 1]$$

```
(%i55) tiz(3/(z-2));
```

$$(\%055) [0, 3 \cdot 2^{n-1}]$$

```
(%i56) residue(3/(z*(z-2)),z,2);
```

$$(\%056) \frac{3}{2}$$

```
(%i57) tiz(z/(z-1));
```

```
(%o57) [1, 1]
```

```
(%i58) tiz(1/(z-1));
```

```
(%o58) [0, 1]
```

```
(%i59) tiz(1/(z^5*(z-1)));
```

```
(%o59) [0, 0, 0, 0, 0, 1]
```

```
(%i60) residue(1/(z*(z-1)),z,1);
```

```
(%o60) 1
```

```
(%i61) tiz((2*z^2)/(11*z^2-21*z+10));
```

```
(%o61) [ $\frac{2}{11}$ ,  $2 - 2 \cdot 11^{-n-1} 10^{n+1}$ ]
```

17.6 Résolution d'équations récurrentes avec la transformée en z

```
(%i62) kill(x, y);
```

```
(%o62) done
```

```
(%i63) equation : 11*y(n)-10*y(n-1)*unit_step(n-1) = 2*unit_step(n);
```

```
(%o63) 11 y (n) - 10 unit_step (n - 1) y (n - 1) = 2 unit_step (n)
```

```
(%i64) equationtransformee:tz(equation);
```

```
(%o64) 11 z_transform (y (n) , n, z) -  $\frac{10 z\_transform (y (n) , n, z)}{z} = \frac{2 z}{z - 1}$ 
```

```
(%i65) solve(equationtransformee, 'tz(y(n)))[1];
```

```
(%o65) z_transform (y (n) , n, z) =  $\frac{2 z^2}{11 z^2 - 21 z + 10}$ 
```

```
(%i66) tiz(rhs(%));
```

```
(%o66) [ $\frac{2}{11}$ ,  $2 - 2 \cdot 11^{-n-1} 10^{n+1}$ ]
```