

Outils, modélisation et simulation en calcul numérique – Corrigé série 3

5 avril 2005

Exercice 1.

i) Soit l'équation différentielle ordinaire de second ordre à coefficients constants

$$f^{(2)}(x) + af^{(1)}(x) + bf(x) = h(x). \quad (1)$$

Sa solution générale est alors donnée par :

– cas 1 : $a^2 - 4b > 0$

$$f(x) = c_1 e^{\lambda_+ x} + c_2 e^{\lambda_- x} + \int_0^x ds \frac{e^{\lambda_+(x-s)} - e^{\lambda_-(x-s)}}{\lambda_+ - \lambda_-} h(s), \quad (2)$$

$$\lambda_{\pm} = \frac{-a \pm \sqrt{a^2 - 4b}}{2}. \quad (3)$$

– cas 2 : $a^2 - 4b = 0$

$$f(x) = e^{-ax/2} (c_1 + c_2 x) + \int_0^x ds e^{-a(x-s)/2} (x-s) h(s). \quad (4)$$

– cas 3 : $a^2 - 4b < 0$

$$f(x) = e^{-ax/2} [c_1 \cos(\lambda x) + c_2 \sin(\lambda x)] + \frac{1}{\lambda} \int_0^x ds e^{-a(x-s)/2} \sin[\lambda(x-s)] h(s), \quad (5)$$

$$\lambda = \frac{\sqrt{4b - a^2}}{2}. \quad (6)$$

Notre situation correspond au troisième cas, et donc

$$x(t) = e^{-\varepsilon t/2} [A \cos(\omega t) + B \sin(\omega t)], \quad \omega = \omega(\varepsilon) = \sqrt{1 - \varepsilon^2/4}. \quad (7)$$

On constate bien l'existence de deux échelles de temps distinctes :

- l'échelle de temps τ_0 de la période d'oscillation sans amortissement notable. La période de l'oscillation est $\sim 2\pi/\omega(\varepsilon)$, et avec un développement de Taylor au premier ordre en ε pour $\omega(\varepsilon)$ on obtient $\tau_0 \sim 2\pi + \mathcal{O}(\varepsilon^2) \sim \mathcal{O}(1)$.
- l'échelle de temps τ_1 de l'amortissement de l'oscillateur : $\tau_1 \sim \mathcal{O}(1/\varepsilon)$.

On constate que plus l'amortissement ε est petit, plus les échelles de temps τ_0 et τ_1 sont séparées, c'est-à-dire $\tau_1 \gg \tau_0$.

ii) Soit une solution de la forme

$$x(t, \varepsilon) = \sum_{k \geq 0} \varepsilon^k x_k(t). \quad (8)$$

– Ordre 0 : on obtient l'équation

$$\ddot{x}_0(t) + x_0(t) = 0, \quad (9)$$

dont la solution est

$$x_0(t) = A_0 \sin t + B_0 \cos t. \quad (10)$$

– Ordre 1 : on obtient l'équation

$$\ddot{x}_1(t) + x_1(t) = -\dot{x}_0(t). \quad (11)$$

La solution correspond au troisième cas, et est donnée par

$$\begin{aligned} x_1(t) &= A_1 \sin t + B_1 \cos t - \int_0^t ds \sin(t-s) \dot{x}_0(s) \\ &= A_1 \sin t + B_1 \cos t - \int_0^t ds \sin(t-s) [A_0 \cos s - B_0 \sin s]. \end{aligned} \quad (12)$$

Ceci s'intègre en faisant usage des identités

$$\sin x \sin y = \frac{1}{2} [-\cos(x+y) + \cos(x-y)], \quad (13)$$

$$\sin x \cos y = \frac{1}{2} [\sin(x+y) + \sin(x-y)], \quad (14)$$

et on obtient

$$x_1(t) = A_1 \sin t + B_1 \cos t - \frac{t}{2} (A_0 \sin t + B_0 \cos t) + \frac{B_0}{2} \sin t. \quad (15)$$

– Ordre $k \geq 1$: on obtient l'équation

$$\ddot{x}_k(t) + x_k(t) = -\dot{x}_{k-1}(t), \quad (16)$$

dont la solution est

$$x_k(t) = A_k \sin t + B_k \cos t - \int_0^t ds \sin(t-s) \dot{x}_{k-1}(s). \quad (17)$$

En exprimant $\dot{x}_{k-1}(s)$ en fonction de $\dot{x}_0(s)$ dans le membre de droite de cette expression on obtient

$$\begin{aligned} x_k(t) &= A_k \sin t + B_k \cos t \\ &+ \sum_{j=1}^k (-1)^j \left[\prod_{l=1}^j \int_0^{s_{k-l+1}} ds_{k-l} \sin(t-s_{k-l}) \right] (A_{k-j} \cos s_{k-j} - B_{k-j} \sin s_{k-j}), \end{aligned} \quad (18)$$

où on a noté $s_k \equiv t$. On observe donc qu'à l'ordre k apparaîtra un terme proportionnel à $\varepsilon^k t^k$. Ainsi, la solution diverge pour toute valeur non nulle de ε , ce qui n'est pas le comportement physique attendu. La méthode de Hilbert fait donc apparaître des divergences séculaires.

iii) Le développement de Chapman-Enskog permet d'exploiter explicitement la séparation des échelles de temps $\tau_1 \gg \tau_0$ en éliminant les divergences séculaires. Soit $x = x(\tau_0, \tau_1)$, $\tau_0 = t$, $\tau_1 = \varepsilon t$, ainsi que $\partial_t = \partial_{\tau_0} + \varepsilon \partial_{\tau_1}$, et $x(\tau_0, \tau_1) = x_0(\tau_0, \tau_1) + \varepsilon x_1(\tau_0, \tau_1)$. Pour simplifier les écritures, notons $x_k = x_k(t, \tau)$ et $\partial_t \rightarrow \partial_t + \varepsilon \partial_\tau$. On obtient ainsi

$$\dot{x}_k = (\partial_t + \varepsilon \partial_\tau) x_k, \quad (19)$$

$$\ddot{x}_k = (\partial_t^2 + 2\varepsilon \partial_t \partial_\tau + \varepsilon^2 \partial_\tau^2) x_k. \quad (20)$$

On insère ces relations dans l'Eq. (1) de la donnée pour obtenir

$$\begin{aligned} &(\partial_t^2 + 2\varepsilon \partial_t \partial_\tau + \varepsilon^2 \partial_\tau^2) [x_0(t, \tau) + \varepsilon x_1(t, \tau)] \\ &+ \varepsilon (\partial_t + \varepsilon \partial_\tau) [x_0(t, \tau) + \varepsilon x_1(t, \tau)] + x_0(t, \tau) + \varepsilon x_1(t, \tau) = 0. \end{aligned} \quad (21)$$

On collecte ensuite les termes de différents ordre en ε .

– Ordre 0 : on obtient l'équation

$$\partial_t^2 x_0(t, \tau) + x_0(t, \tau) = 0, \quad (22)$$

d'où la solution

$$x_0(t, \tau) = A_0(\tau) \sin t + B_0(\tau) \cos t. \quad (23)$$

Les fonctions $A_0(\tau)$ et $B_0(\tau)$ sont inconnues et peuvent être déterminées grâce à l'ordre 1.

– Ordre 1 : on obtient l'équation

$$\partial_t^2 x_1(t, \tau) + x_1(t, \tau) + 2[\partial_\tau A_0(\tau) \cos t - \partial_\tau B_0(\tau) \sin t] + A_0(\tau) \cos t - B_0(\tau) \sin t = 0. \quad (24)$$

Nous avons vu de la méthode de Hilbert que les divergences séculaires sont issues des termes en $\sin t$ et $\cos t$ à l'ordre ε^k , $k \geq 1$. Il faut donc éliminer ces contributions par un choix approprié de $A_0(\tau)$ et $B_0(\tau)$. Ainsi

$$2\partial_\tau A_0(\tau) + A_0(\tau) = 0, \quad (25)$$

$$2\partial_\tau B_0(\tau) + B_0(\tau) = 0, \quad (26)$$

$$(27)$$

ce qui s'intègre pour donner

$$\begin{pmatrix} A_0 \\ B_0 \end{pmatrix}(\tau) = \begin{pmatrix} \tilde{A} \\ \tilde{B} \end{pmatrix} \exp(-\tau/2), \quad (28)$$

où on a noté \tilde{A} et \tilde{B} les conditions initiales. La solution à l'ordre 1 s'écrit donc

$$x_1(t, \tau) = A_1(\tau) \sin t + B_1(\tau) \cos t, \quad (29)$$

où $A_1(\tau)$ et $B_1(\tau)$ sont déterminés de façon à éliminer les divergences séculaires des ordres supérieurs.

La solution à l'ordre 0 s'écrit donc

$$x_0(t, \tau) = \tilde{A} \exp(-\tau/2) \sin t + \tilde{B} \exp(-\tau/2) \cos t \quad (30)$$

$$= \tilde{A} \exp(-\varepsilon t/2) \sin t + \tilde{B} \exp(-\varepsilon t/2) \cos t. \quad (31)$$

Il s'agit de l'approximation de plus bas ordre dans le développement de Taylor $\omega(\varepsilon) = 1 + \mathcal{O}(\varepsilon^2)$ de la solution exacte (7), qui fait apparaître (et utilise) de façon explicite l'existence de deux échelles de temps distinctes.

Notons que pour alléger les écritures nous n'avons considéré que les deux premiers ordres dans le développement de Chapman-Enskog. La solution complète requiert le développement complet $x(\tau_0, \tau_1, \tau_2, \dots) = \sum_{k \geq 0} \varepsilon^k x_k(\tau_0, \tau_1, \tau_2, \dots)$.

Le développement de Chapman-Enskog est tel que lorsque la dynamique se déroule sur l'échelle de temps j , c'est-à-dire quand $\tau \sim \varepsilon^{-j}$, les produits $\varepsilon^k \tau$ tendent vers l'infini pour $k < j$ et vers zéro pour $k > j$. Ainsi en fixant la variable $\tau_j = \varepsilon^j \tau$ on étudie à la limite $\varepsilon \rightarrow 0$ la dynamique sur l'échelle de temps $\tau \sim \varepsilon^{-j}$. Sur les autres échelles l'évolution est alors ou bien accomplie ($k < j$) ou bien n'a pas encore commencé ($k > j$).

Référence : pour plus d'explications sur le développement de Chapman-Enskog, voir : J. Piasecki, *Echelles de temps en théorie cinétique* (Presses polytechniques universitaires romandes, Lausanne, 1997). Cote : 235 PIA.

Exercice 2.

i) Programme Matlab :

```
clear;
MAXIT=5000; %maximum number of time steps
POINTS=100; %number of t_i
N=200; %number of independent runs for average
p=0.5; %probability of going to the right
savedata=MAXIT/POINTS;
walk=zeros(3,POINTS+1);
for i=1:N
    k=1;
    position=0;
    for j=2:MAXIT
        randomnumber=rand(1,1);
        if (randomnumber < p)
            position = position + 1;
        else
            position = position - 1;
        end
        if (mod(j,savedata)==0)
            k=k+1;
            walk(3,k)=walk(3,k)+position;
        end
    end
end
end
for i=1:POINTS+1
    walk(1,i)=i*savedata;
    walk(2,i)=walk(3,i)/N;
    walk(3,i)=N*walk(2,i)^2-walk(2,i)^2;
end
plot(walk(1,:),walk(2,:),'kx-'); %<x>
pause;
plot(walk(1,:),walk(3,:),'kx-'); %<x^2>-<x>^2
```

Programme Fortran 90 :

```
module fonctions
implicit none
integer, parameter :: maxit=10000000 !maximum number of time steps
integer, parameter :: points=2000 !number of t_i
integer, parameter :: N=10000 !number of independent runs for average
real, parameter :: p=0.5 !probability of going to the right
character(*), parameter :: filename = 's3.txt'
contains

FUNCTION ran2(idum) !Random generator. Initialize idum with negative integer
INTEGER idum,IM1,IM2,IMM1,IA1,IA2,IQ1,IQ2,IR1,IR2,NTAB,NDIV
```

```

REAL :: ran2,AM,EPS,RNMX
PARAMETER (IM1=2147483563,IM2=2147483399,AM=1./IM1,IMM1=IM1-1,IA1=40014,&
           &IA2=40692,IQ1=53668,IQ2=52774,IR1=12211,IR2=3791,NTAB=32,&
           &NDIV=1+IMM1/NTAB,EPS=1.2e-7,RNMX=1.-EPS)
INTEGER :: idum2,j,k,iv(NTAB),iy
SAVE iv,iy,idum2
DATA idum2/123456789/, iv/NTAB*0/, iy/0/
if(idum.le.0)then
  idum=max(-idum,1)
  idum2=idum
  do j=NTAB+8,1,-1
    k=idum/IQ1
    idum=IA1*(idum-k*IQ1)-k*IR1
    if(idum.lt.0)idum=idum+IM1
    if(j.le.NTAB)iv(j)=idum
  enddo
  iy=iv(1)
endif
k=idum/IQ1
idum=IA1*(idum-k*IQ1)-k*IR1
if(idum.lt.0)idum=idum+IM1
k=idum2/IQ2
idum2=IA2*(idum2-k*IQ2)-k*IR2
if(idum2.lt.0)idum2=idum2+IM2
j=1+iy/NDIV
iy=iv(j)-idum2
iv(j)=idum
if(iy.lt.1)iy=iy+IMM1
ran2=min(AM*iy,RNMX)
return
end function ran2
end module fonctions
!-----
program s3
use fonctions
implicit none
double precision, dimension(0:points) :: walk !to avoid integer overflows
integer :: i,j,k,seed,position,savedata
seed = -14731
savedata=maxit/points !speed up: out of the loop
walk=0.d0
do i=1,N
  k=0
  position=0
  do j=2,maxit
    if (ran2(seed) < p) then
      position = position + 1
    else

```

```

        position = position - 1
    end if
    if (mod(j,savedata)==0) then
        k=k+1
        walk(k)=walk(k)+dble(position)
    end if
end do
end do
open(unit=1,file=filename)
do i=0,points
    write(1,*) i*savedata,real(walk(i))/real(N),&
        &real(walk(i))**2/real(N)-(real(walk(i))/real(N))**2
end do
close(1)
end program s3

```

Pour la marche aléatoire symétrique ($p = 1/2$), les simulations suggèrent que la position moyenne est approximativement nulle, tandis que les fluctuations de position sont de l'ordre $t^{1/2}$, caractéristique d'une dynamique diffusive (voir Fig. 1). La statistique doit être très grande ($N \rightarrow \infty$ réalisations) pour diminuer l'ampleur des fluctuations (un langage de programmation plus rapide que Matlab s'impose dans ce cas). Néanmoins, les fluctuations restent grandes et il n'est pas aisé de conclure sur le caractère diffusif du mouvement. Il est alors utile de recourir à une étude analytique pour permettre de tirer des conclusions sur la dynamique (cf. point iii).

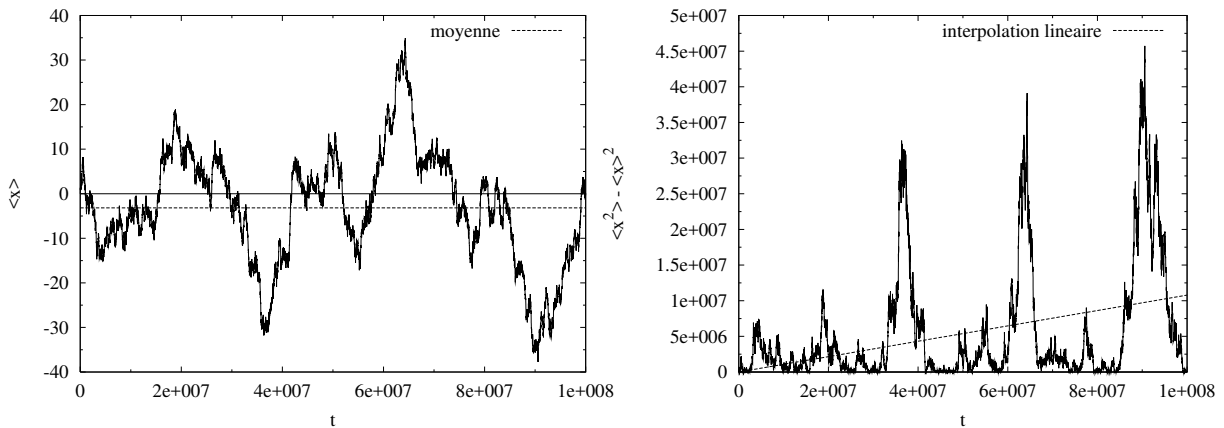


FIG. 1 – Marche aléatoire unidimensionnelle symétrique ($p = 0.5$, $N = 5 \times 10^4$ réalisations).

- ii) Par contre, pour une marche aléatoire asymétrique ($p \neq 0.5$) le mouvement est biaisé. Une composante balistique apparaît, c'est-à-dire que la position moyenne est proportionnelle au temps : $\langle x(t) \rangle \propto t$. Nous constatons de la Fig. 2 que pour $p = 0.501$ la pente est comme prévu $m = 0.02$. En effet, si $p = 0.5 + x$, $x \in [-0.5, 0.5]$, alors la pente sera donnée par $m = 2x$.

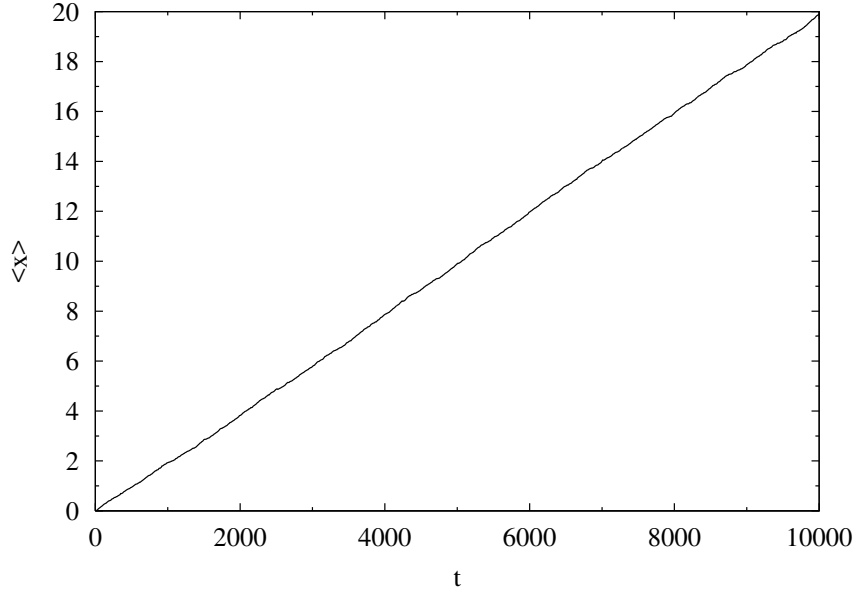


FIG. 2 – Marche aléatoire unidimensionnelle asymétrique pour $p = 0.501$.

iii)

- a) On vérifie facilement que la forme donnée de la solution pour $P(x, t)$ satisfait bien à l'équation de diffusion.
- b) Pour des raisons de symétrie le premier moment est nul (domaine d'intégration symétrique, et fonction antisymétrique)

$$\langle x \rangle = \int_{\mathbb{R}} dx x P(x, t) = 0, \quad (32)$$

ce qui signifie que la particule reste à l'origine en moyenne. On a par contre

$$(\Delta x)^2 = \langle x^2 \rangle - \langle x \rangle^2 = \langle x^2 \rangle = \int_{\mathbb{R}} dx x^2 P(x, t) = 2Dt, \quad (33)$$

où on a fait usage d'un cas particulier de la relation générale

$$\int_{\mathbb{R}^d} d\mathbf{x} |\mathbf{x}|^n e^{-\alpha \mathbf{x}^2} = \frac{\pi^{d/2}}{\alpha^{(d+n)/2}} \frac{\Gamma[(d+n)/2]}{\Gamma(d/2)}, \quad (34)$$

avec $\text{Re}(\alpha) > 0$, $\mathbf{x} \in \mathbb{R}^d$, $n \in \mathbb{N}$, et Γ la fonction gamma de Euler. Le mouvement est donc bien diffusif, et les fluctuations autour de la position moyenne croissent en fonction du temps selon $t^{1/2}$. Ce résultat s'obtient donc très facilement de façon analytique, alors que le temps nécessaire pour l'obtenir à partir de simulations numériques est rédhibitoire.