I)Travail préliminaire :

Dans la partie précédente, nous avons utilisé la fonction 'trapz' pour calculer l'intégrale. Cette méthode n'est pas précis. Donc, la partie suivante nous allons calculer l'intégrale avec la méthode de Simpson. La fonction se défini comme la suivante :

```
function I=Simpson(f,a,b,step) % f:fonction // a,b: bornes
x1=[a:step:b];
y1=f(x1);
n1=length(x1);
x2=[a+step/2:step:b-step/2];
y2=f(x2);
I=step*((2*sum(y1)-y1(1)-y1(n1))/6+2*sum(y2)/3);
```

Nous testons notre fonction sur des intégrales qu'on connaît.

Par exemple ; $\int_0^{\infty} x^n e^{-x} = n!$

On définit la fonction myfun de maniéré suivante :

```
function y=myfun(x,n,a)
y=(x.^n).*exp(-x);
```

On remarque plus on augmente 'step' , plus notre intégrale est precise.

On modifie la fonction myfun pour pour s'assure que la fonction correctement :

$$\int_0^{\infty} x^n e^{-ax} = \frac{n!}{a^{n+1}}$$

II) Équation de chaleur :

Une des applications de séries des Fourier qu'on a vu en cours, c'était la calcul d'équation de chaleur en connaissant que l'état initial. On tente d'écrire une programme qui détermine le profil de température d'un barre de métal aux moments différents en connaissant son profil initial.

On prend la suivante comme profil initial :

$$u(x,0) = f(x) = \begin{cases} T_0 + \dfrac{2(T_1 - T_0)}{L} & \text{pour } 0 < x < L/2 \\ 2T_1 - T_2 + \dfrac{2(T_2 - T1)}{L} & \text{pour } L/2 < x < L \end{cases}$$

et définira le profil du maniéré suivante :

$$u(x,t) = \sum_{n=1}^{+\infty} b_n(t).\sin\left(\frac{n.\pi.x}{L}\right) \quad (1)$$

On cherche $\forall n \in \mathbb{N}, \forall t, b_n(t)$.

On connaît que $\forall n \in \mathbb{N}, b_n(0) = \frac{2}{L} \int_0^L f(x).\sin(\frac{n.\pi.x}{L})$ (2).

Nous traduisons ceci en langage MATLAB de maniere suivante :

```
b0=zeros(1,N); %b0 correspond a bn(t=0) pour des valeur de n different

for n=1:N
   b0(1,n)=Simpson(@(x)f1(x,T0,T1,L,n),0,L/2,step);
   b0(1,n)=b0(1,n)+Simpson(@(x)f2(x,T1,T2,L,n),L/2,L,step);
end
```

Note:f1 correspond à f sur $0 < x < \frac{L}{2}$ et f2 à f sur $\frac{L}{2} < x < L$.

Utilisant une équation différentielle, on trouve que :

$$\forall n \in \mathbb{N}, b_n(t) = b_n(0).e^{\frac{-a n^2 \pi^2}{L^2} t} \quad (3)$$

qui nous donne la suivante :

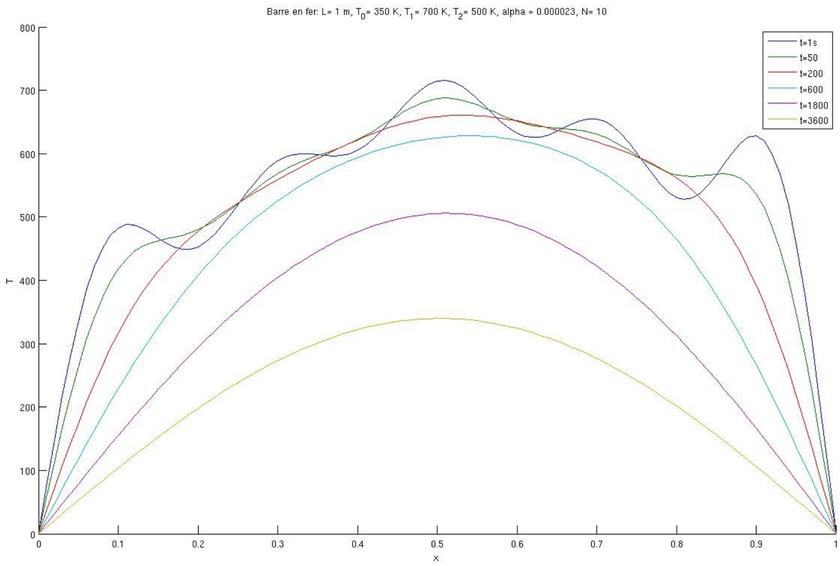
```
t=[1,50,200,600,1800,3600]; %Les temps que nous chercherons
b=zeros(length(t),N); %Lines correspond to x, coloums to T.

%Calcul des coefficient de Fourier pour t differents
for n=1:N
   b(:,n)=b0(n)*exp((-(n*pi)^2)*alpha*t');
end
```

Puis, utilisant (1), on ecrit :

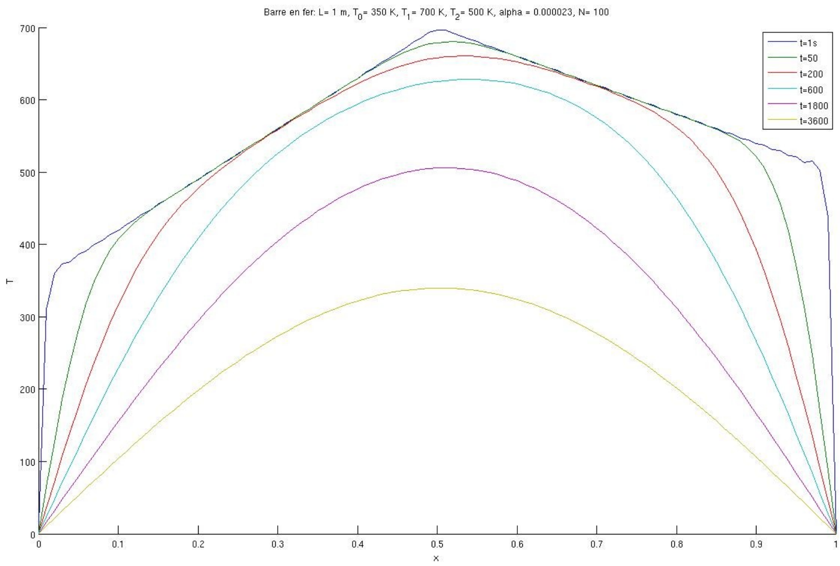
```
x=[0:0.01:L]; %les absissces

u=zeros(length(t),length(x)); %Les temperatures pour les t differents

for n=1:N
   u=u + b(:,n)*sin(n*pi*x);
end
```

Ce code nous donne avec un nombre d'harmoniques bases:

En



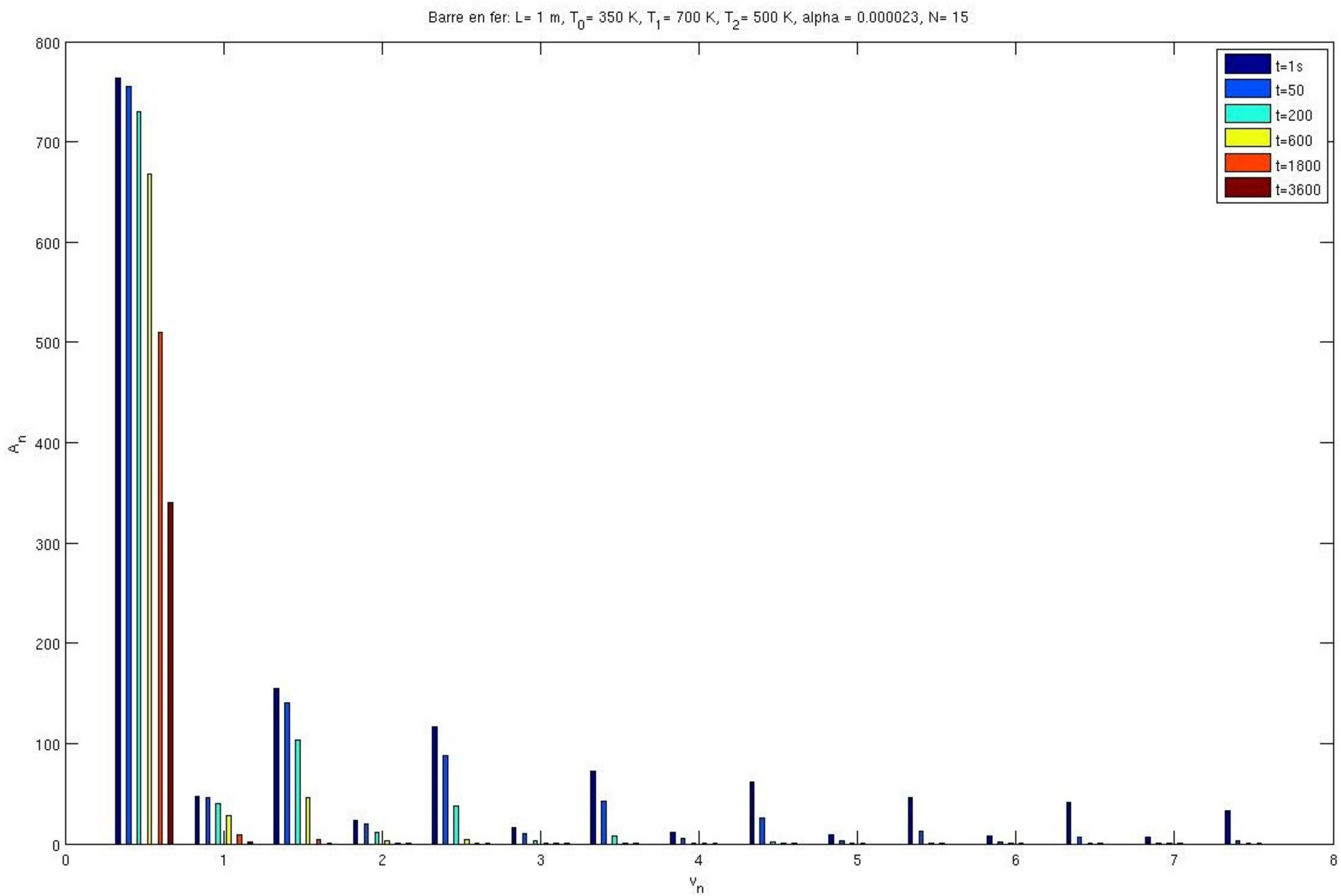
augmentant N le nombre d'harmonique, on obtient un courbe plus précis pour t=1s et t=50s mais il y a peu de différence pour les temps plus avancé:



Ceci peut s'expliquer par en regardant quelques premières harmoniques (voir annexe n°1).

On remarque les amplitudes de t=1s et t=50s décroit moins vite que celles du t=3600s. Donc, on peut dire que les coefficients de fréquence plus ont moins en moins d'importance lors que le temps croit.

Annexe n°1 :


Barre en fer: L= 1 m, T0= 350 K, T1= 700 K, T2= 500 K, alpha = 0.000023, N= 15
An
vn
0
100
200
300
400
500
600
700
800
1
2
3
4
5
6
7
8
t=1s
t=50
t=200
t=600
t=1800
t=3600