
Table of Contents

init	1
parametres i	1
resolution i	2
graphique i	3
parametres ii	4
resolutionflag ii	4

init

```
clear
```

parametres i

Équation différentielle: $d^2 u/dx^2=g(x)$ sur $x=(a,b)$ Conditions aux limites générales:

```
L=0.3; %m ; Épaisseur du mur
k=0.85;%W/(m*K); La conductivité thermique de la brique
si=5.67e-8; %constante de Stefan-Boltzmann
h=20;%

%TL=841.90; NONONONONON
To=293; %oK
% Condition radiative à x=0 (face externe du mur): -k*dT/dx=-si*(T^4-
To^4)
% !!! Condition radiative est implementée SEULEMENT sur la face
externe du mur !!!
c1=-k;
c2=0;
c3=-si*To^4;
% Condition de Neumann à x=L (face interne du mur): dT/dx=0
d1=-k;
d2=0;
d3=-si*To^4;

N=100;
dx=L/N; % Pas de discrétisation
x=(0:dx:L)';
T=To*ones(size(x)); % Approximation initiale

% Source volumique de chaleur q[W/m^3] d'épaisseur dL
% La source est intégrée dans la partie intérieure du mur
dL=0.05;
q=5*10^5;% W/m^3;
S=q*exp(-(x-L)/dL).^2);

% figure(1);plot(x,T,'r');
% xlabel('x [m]');ylabel('T [K]');hold
% boucle de convergence
ci=0;
```

```

Err=[];
tol=1e-12;
flag=1;

```

resolution i

```

while (flag==1)
    ci=ci+1;

    M=diag(-2*ones(1,N+1),0)+diag(ones(1,N),-1)+diag(ones(1,N),1);

    % condition x=0
    M(1,1)=-3*c1;%2*c2*dx-3*c1;
    M(1,2)=4*c1;
    M(1,3)=-c1;
    % -k*dT/dx=-si*(T^4-To^4)
    %c1=-k;
    %c2=0;
    %c3=-si*To^4;

    %d1=-k;
    %d2=0;
    %d3=si*TL^4;
    % condition x=L
    M(N+1,N+1)=-3*d1;%+2*d2*dx;
    M(N+1,N)=4*d1;
    M(N+1,N-1)=-d1;

    % !!! Condition radiative est implementée SEULEMENT sur la face
    externe du mur !!!
    b=(S/k)*dx^2;
    %condition x=0
    b(1)=-2*h*dx*(To-T(1))+2*dx*(T(1)^4*si+c3);%%-2*dx^2*(To-
T(1))+2*dx*(T(1)^4*si+c3);
    %conditon x=L

    b(N+1)=-2*h*dx*(To-T(N+1))+2*dx*(T(N+1)^4*si+d3);%-2*dx^2*(To-T(N
+1))+2*dx*(T(N+1)^4*si+d3);%=2*dx*(T(N+1)^4*si+d3);%-2*dx^2*(To-T(N
+1))+2*dx*(T(N+1)^4*si+d3);%-2*dx*(T(N+1)^4*si+d3);%2*dx^2*(To-T(N
+1))-2*dx*(T(N+1)^4*si+d3);%2*d3*dx;

    F=M*T+b;
    Err=[Err sum(abs(F))/(N+1)];
    % display(['Étape=', num2str(ci), ' ; Err='
num2str(Err(end))])
    if (Err(end)<tol)
        break ; % metre flag ==0 ? jai la fleme
    end
    % pause
    % % pas sur de l'utiliter

    % !!! Condition radiative est implementée SEULEMENT sur la face
    externe du mur !!!

```

```

J=M;
J(1,1)=M(1,1)+T(1)^3*8*dx*si+2*h*dx; %probablement la mon erreure
mais je compren pas
J(N+1,N+1)=M(N+1,N+1)+T(N+1)^3*8*dx*si+2*h*dx;

dT=-J\F;
T=T+dT;
% figure(1)
% plot(x,T,'b')

end
% hold

```

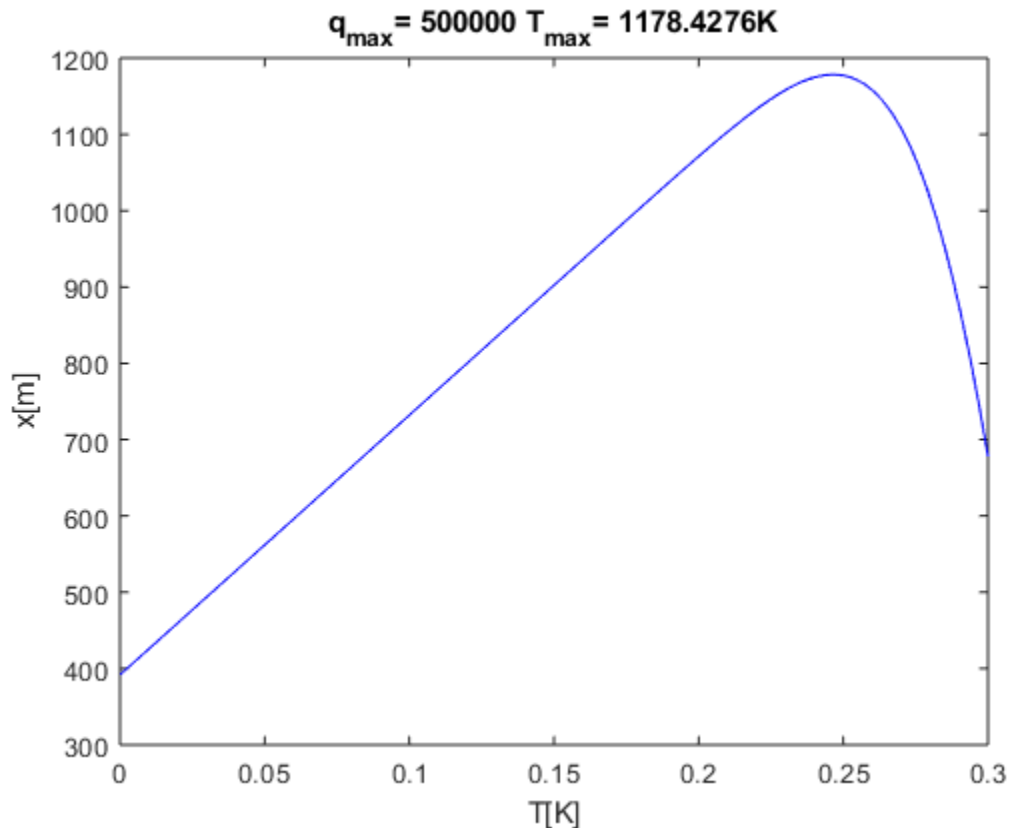
graphique i

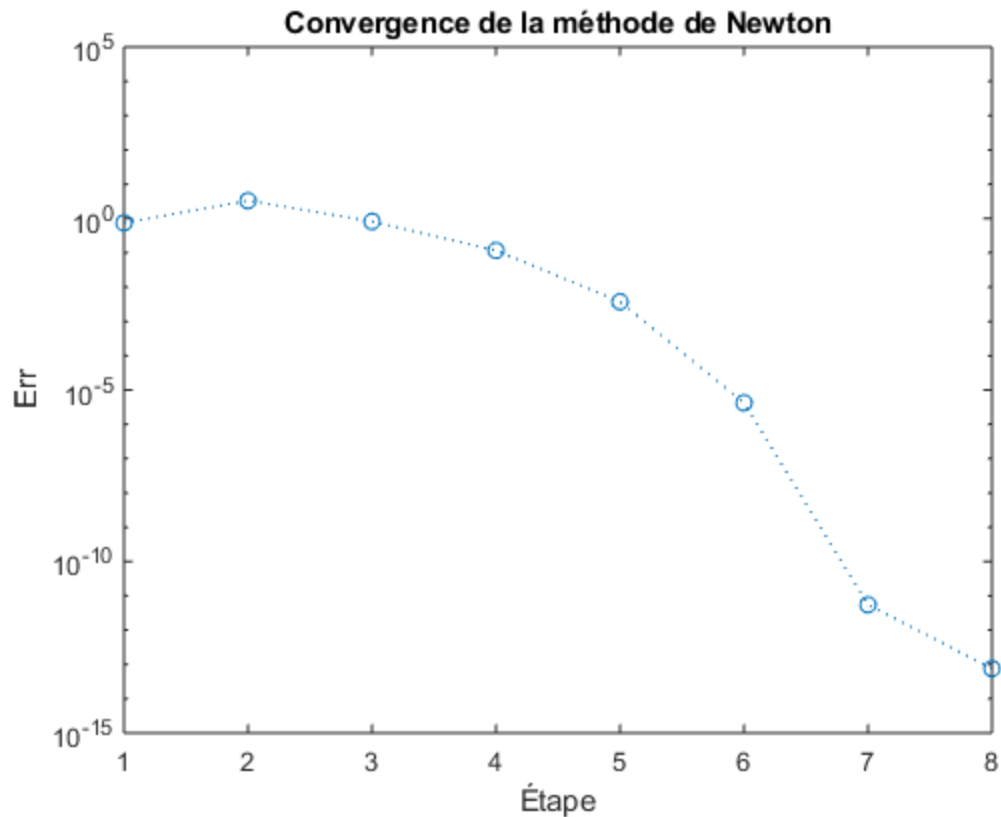
```

Tmax=max(T);
figure(2)
plot(x,T,'b')
xlabel('T[K]');ylabel('x[m]');title(['q_{max}= ',num2str(q) ' T_{max}= ',num2str(Tmax),'K']);

figure(3);semilogy((1:ci),Err,'o');
xlabel('Étape');ylabel('Err'); title('Convergence de la méthode de Newton')

```





parametrres ii

```
Tmm=2250;%T max mur
q=[10^(5),10^(7)];
```

```
ci=0;
Err=[];
tol=1e-12;
flag=1;
```

resolutionflag ii

```
flag2=1;
i=1;
clc
q=[10^(5),10^(7)];
[approx , err_abs] = bissec('tempFunction',q(1),q(2),45,tol);
[Tmax,Err,ci,T] = findTmax(approx(end));
q = approx(end);

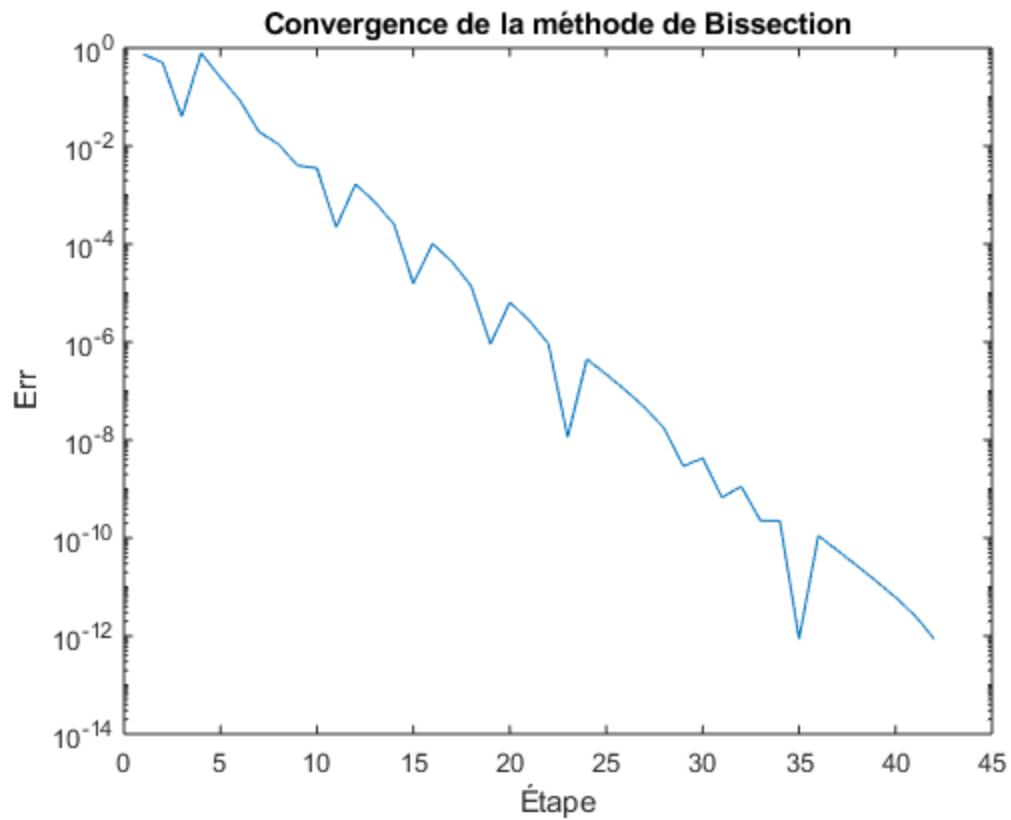
figure(4);semilogy(err_abs./approx)
xlabel('Étape');ylabel('Err'); title('Convergence de la méthode de
Bissection')
figure(5)
```

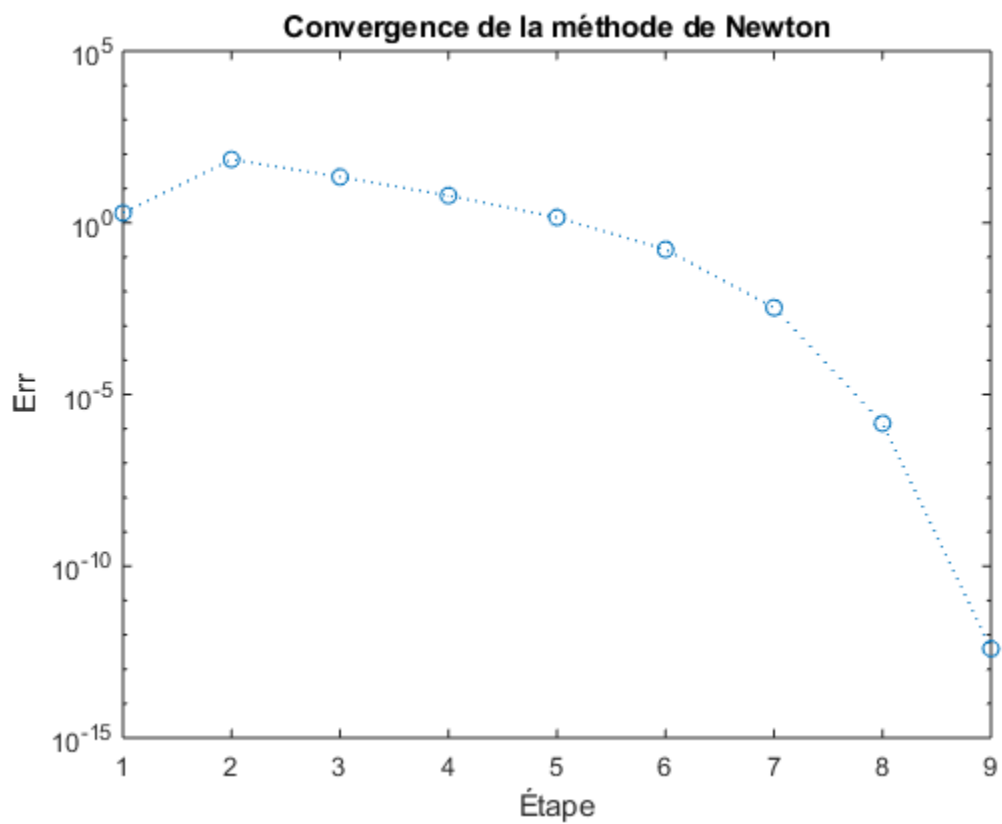
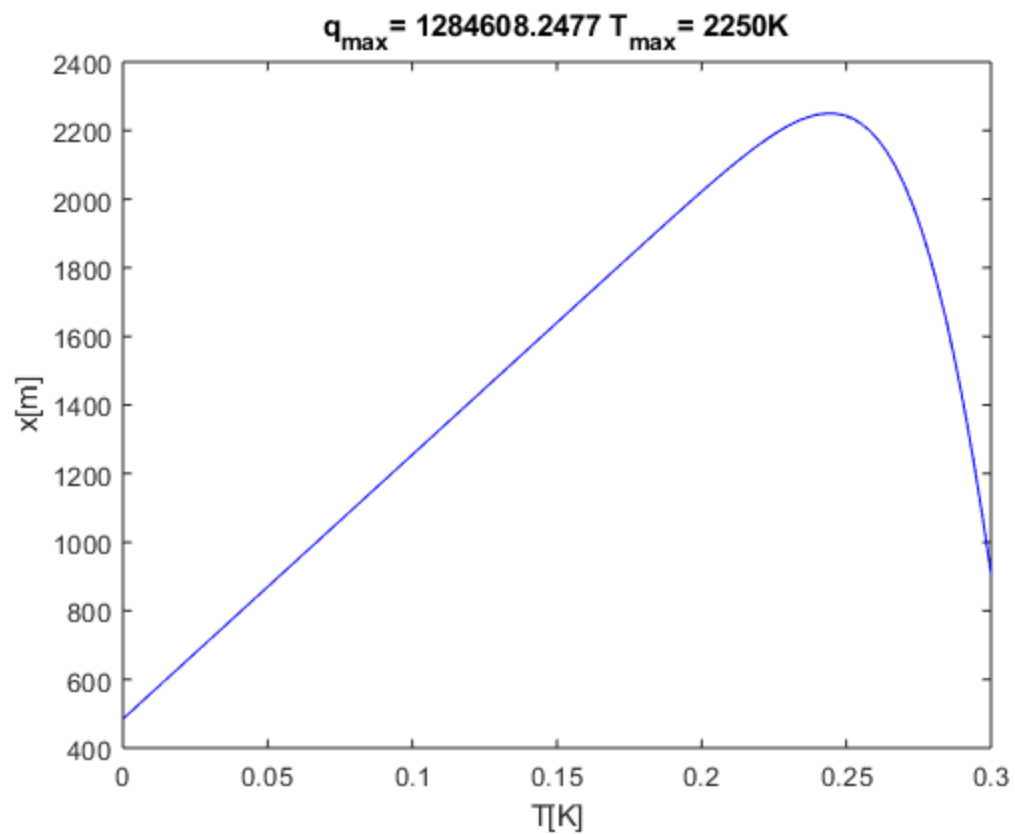
```

plot(x,T,'b')
xlabel('T[K]');ylabel('x[m]');title(['q_{max}= ',num2str(q) ' T_{max}= ',num2str(Tmax),'K']);

figure(6);semilogy(1:ci),Err,':o');
xlabel('Étape');ylabel('Err'); title('Convergence de la méthode de
Newton')

```





Published with MATLAB® R2017b