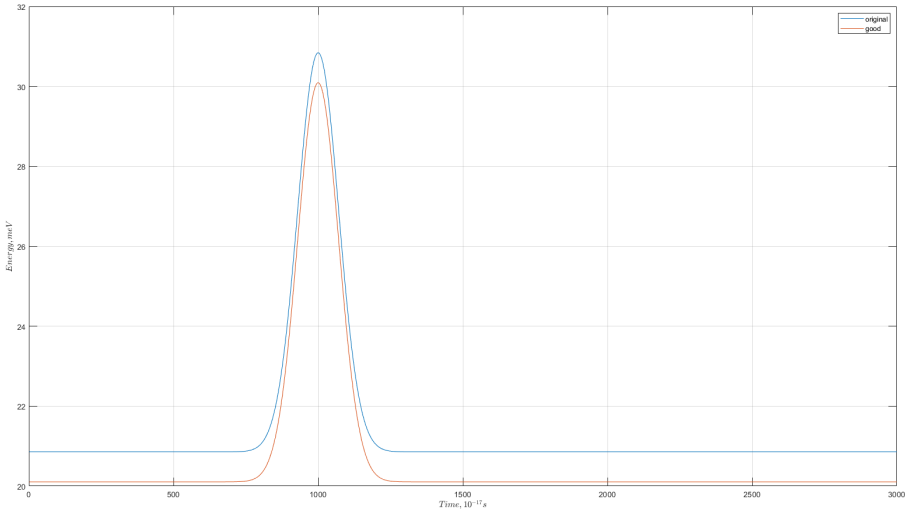
```
close all
clear
clc
%
fig=figure('Name','Method of variations','NumberTitle','off');
fig.Units='normalized';
fig.Position=[0 0 1 1];

%defining constants
hbar=1.0546e-34;
m0=9.1e-31;
e=1.6e-19;

%forming a task
L=1e-8;                          %width of structure
Np=100;                          %amount of steps
x=linspace(0,L, Np);             %creating a 'x-axis'
dx=x(2)-x(1);    dy=dx;          %definig a primive step
dt=1e-17;
y=x';                            %creating a 'y-axis'
[X,Y]=meshgrid(x,y);             %creating a 2d-sapce
koef=-hbar^2/(2*m0*12*(dx^2));
koefx=hbar*dt/(2*m0*(dx^2));
koeft=dt/hbar;

%defining potential feild and perubation
U0=0*e*ones(Np);
Upetr=0.01*e*heaviside(-x+L/4).*heaviside(-y+L/4);

%defining wave functions for quantumm well
px=@(n,x)sqrt(2/L).*sin(pi*n*x/L);
py=@(m,y)sqrt(2/L).*sin(pi*m*y/L);
p=@(n,m,x,y)px(n,x).*py(m,y);
E=@(n,m)pi^2*hbar^2/(2*m0*L^2)*(n^2+m^2);

%set n=1 and m=2 as it is degenerated w/ n=2 and m=1
Psi1=px(1,x);
Psi2=py(2,y)';

X1re=real(Psi1);
X1im=imag(Psi1);
X2re=real(Psi2);
X2im=imag(Psi2);
m=3000;
Engood=zeros(1,m);
En=zeros(1,m);

for Time=1:m
    if(Time>500 && Time<1500)
        U=U0+Upetr(1,:)*exp(-(Time-1000)^2/100^2);
%         U=U0+Upetr(1,:);
    else
```

```
        U=U0;
    end

    for n=2:Np-1
       X1re(n)=X1re(n)-koefx*(X1im(n-1)-2*X1im(n)+X1im(n
+1))+koeft*U(n)*X1im(n);
    end
    for n=2:Np-1
       X1im(n)=X1im(n)+koefx*(X1re(n-1)-2*X1re(n)+X1re(n+1))-
koeft*U(n)*X1re(n);
    end

    for n=2:Np-1
       X2re(n)=X2re(n)-koefx*(X2im(n-1)-2*X2im(n)+X2im(n
+1))+koeft*U(n)*X2im(n);
    end
    for n=2:Np-1
       X2im(n)=X2im(n)+koefx*(X2re(n-1)-2*X2re(n)+X2re(n+1))-
koeft*U(n)*X2re(n);
    end

    X1=X1re+1i*X1im; X1=X1/sqrt(X1*X1');
    X2=X2re+1i*X2im; X2=X2/sqrt(X2*X2');

    G=zeros(Np+4);
    G(3:Np+2,3:Np+2)=X1.*(X2.');
    dE=zeros(Np);
    for i=3:Np-2
        for j=3:Np-2
            Difx=-G(i,j-2)+16*G(i,j-1)-30*G(i,j)+16*G(i,j+1)-G(i,j+2);
            Dify=-G(i-1,j)+16*G(i-1,j)-30*G(i,j)+16*G(i+1,j)-G(i+2,j);
            dE(i-2,j-2)=conj(G(i,j))*(koef*(Difx+Dify)+U(n).*G(i,j));
        end
    end
    E=real(sum(sum(dE)));
    En(Time)=E;

    G=zeros(Np+4);
    G(3:Np+2,3:Np+2)=(X1.*(X2.')+X2.*(X1.'))/sqrt(2);
    dE=zeros(Np);
    for i=3:Np-2
        for j=3:Np-2
            Difx=-G(i,j-2)+16*G(i,j-1)-30*G(i,j)+16*G(i,j+1)-G(i,j+2);
            Dify=-G(i-1,j)+16*G(i-1,j)-30*G(i,j)+16*G(i+1,j)-G(i+2,j);
            dE(i-2,j-2)=conj(G(i,j))*(koef*(Difx+Dify)+U(n).*G(i,j));
        end
    end
    E=real(sum(sum(dE)));
    Engood(Time)=E;
end
T=1:m;
plot(T,En/e*1e3);
grid on
hold on
```

```
xlabel('$Time,10^{-17}s$','Interpreter','latex');
ylabel('$Energy,meV$','Interpreter','latex');
plot(T,Engood/e*1e3);
legend("original","good")
```



Published with MATLAB® R2020b