

Etude du déplacement de microparticules à la surface de l'eau dans une structure d'ondes de Faraday

Méthodes & Protocoles

1. Liste du matériel

- Cuve carrée (20x20 cm², fixation avec pas de vis) dont le fond a été peint en noir à l'aide d'une bombe aérosol
- Particules de PIV (50-100 µm de diamètre, couleur blanche, hydrophobes, moins denses que l'eau)
- Eprouvette graduée (250 mL)
- Petite spatule métallique

- Pot vibrant (TIRA vib, fixation avec pas de vis)
- Générateur Basses Fréquences (Rigol DG 1022A 25MHz, 100 MSa/s)
- Amplificateur (P-Amp BAA 500)
- Accéléromètre (Brüel & Kjær, Delta Tron Accelerometer Type 4534-B-001)
- Oscilloscope (Agilent Technologies DSO1052B, 50MHz 1GSa/s)

- Lampe plate (EuroshopLED modèle DL-011-AW-01NW Puissance 11W)
- lame semi réfléchissante
- Appareil photo (Nikon D3300 18,55 mm, 50 images par seconde)

- Structure métallique pour le montage expérimental faite avec des MakerBeam
- Tissu opaque pour protéger le montage de la luminosité extérieure

2. Logiciels d'acquisition et de traitement des données

Nous avons utilisé le logiciel DigiCamControl pour pouvoir gérer la prise d'images depuis l'ordinateur. Afin de pouvoir utiliser les vidéos enregistrées sous format Clip Video sur ImageJ, nous les avons converties en format avi avec le logiciel Free MP4 to AVI Converter 1.13. Elles étaient ensuite chargées sur le logiciel ImageJ (64 bits, mémoire maximale : 5472 MB, 4 unités d'exécution) et analysées avec le plugin Mosaic - Particle Tracker 2D. Les données brutes ainsi obtenues étaient sauvegardées sous format csv puis converties au format txt pour être traitées sous MATLAB.

3. Liens

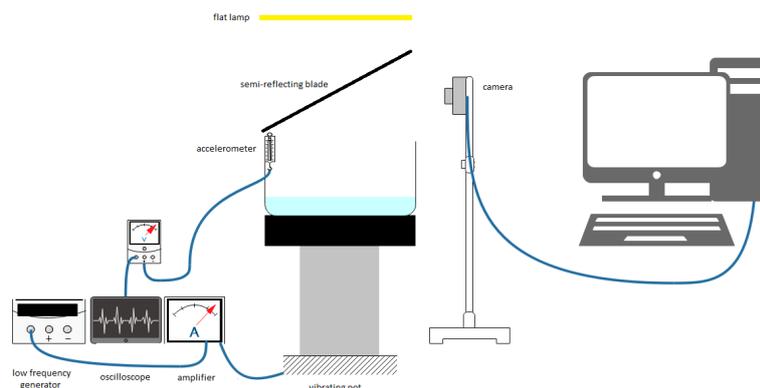
Pour télécharger ImageJ : <https://imagej.nih.gov/ij/download.html>

Pour télécharger le plugin et son mode d'emploi : <http://mosaic.mpi-cbg.de/?q=downloads/imageJ>

Pour télécharger le convertisseur AVI-JPEG :

http://www.01net.com/telecharger/windows/Multimedia/encodeurs_et_decodeurs/fiches/45350.html

4. Montage



5. Objectif

Etude qualitative et quantitative du déplacement de particules dans une structure d'ondes de Faraday à la surface de l'eau dans une cuve. Nous avons décidé d'étudier l'influence relative de la fréquence d'excitation de la cuve et de son accélération verticale pour une même cuve, une même hauteur d'eau et les mêmes particules.

6. Protocole

6.1. Sans particule

Tout d'abord, nous avons souhaité obtenir expérimentalement des informations sur l'arrangement spatial des motifs observés à la surface de l'eau liés aux ondes de Faraday. Sur des enregistrements au format avi de 30 s, nous avons extrait une image sur cents au format JPEG que nous avons recadrée pour ne sélectionner que la surface de l'eau. Nous avons converti ces images en images binaires avec un seuil adaptable selon la vidéo. Puis, nous avons effectué une transformée de Fourier 2D de ces images pour obtenir des informations sur les vecteurs d'onde de surface. Ce travail nous a permis de confronter nos résultats expérimentaux avec la théorie sur les ondes de surface.

6.2. Avec particules

Nous nous sommes ensuite concentrés sur la caractérisation des mouvements des particules. Pour différents paliers de fréquence (50, 60 et 70 Hz), nous avons réalisé plusieurs enregistrements de 3 min environ pour des accélérations verticales de la cuve différentes. Pour des raisons de mémoire, nous devons traiter ces vidéos en 7 séquences de 24 s (1200 images sur ImageJ).

Notre programme MATLAB de traitement des données contient donc une partie permettant de faire la corrélation entre ces différentes séquences vidéo. Il contient aussi deux parties consacrées à deux traitements anti-reflets. L'un reposant sur l'élimination des trajectoires très localisées spatialement ou dont la durée de suivi est très courte. L'autre s'appuyant sur l'élimination des erreurs d'ImageJ sur quelques particules qu'il corrèle avec un reflet.

Notre programme de traitement de ces images avec particules renvoie quatre figures : l'image des trajectoires de toutes les particules sélectionnées après les traitements anti-reflets ; les graphiques de vitesse et d'accélération de ces particules avec des données complémentaires telles que des moyennes et la densité de pics ; deux histogrammes offrant des informations sur les périodes entre deux pics de vitesse pour une même particule ; et enfin la transformée de Fourier 2D de l'image des trajectoires de l'ensemble des particules.

7. Difficultés notables

Les reflets liés à l'éclairage et à l'irrégularité de la surface de l'eau auront été la principale source de difficultés dans notre travail. L'utilisation d'un tissu noir pour couvrir notre montage, d'une lame semi-réfléchissante et les innombrables tentatives de déplacement et de réglage de l'intensité de notre lampe plate n'auront pas permis de supprimer tous les reflets, surtout lorsque l'amplitude des ondes de surface tendait à être élevée.

Ce problème des reflets a entraîné la nécessité de réaliser un compromis dans l'utilisation du plugin Mosaic – Particle Tracker 2D sur ImageJ entre dimension de la surface à analyser et taille des particules à sélectionner en fonction des vidéos pour permettre tant de ne pas saturer la mémoire d'ImageJ que le bon fonctionnement du programme MATLAB derrière avec un temps de calcul acceptable (quelques heures maximum).

8. Notions importantes autour de ce travail

- Il est important de faire la distinction entre la mobilité de la structure des ondes de surface et la mobilité des particules même si les deux semblent être corrélés par des tendances mathématiques que nous avons observées.
- Les brutales variations de vitesse de manière ponctuelle des particules ont tout d'un phénomène d'intermittence.
- Les particules circuleraient selon nous dans les creux de la structure mobile que forme les ondes de surface, et la présence d'un pic de vitesse pour une particule correspondrait à nos yeux à la formation d'une onde juste au niveau de la particule.
- Bien que la surface de l'eau soit irrégulière du fait des ondes de Faraday, nous n'avons pas pris en compte les variations de hauteur des particules qui reste relativement faible (de l'ordre de 1 mm) comparée à leur déplacement dans le plan de la surface de la cuve.

9. Scripts MATLAB

Voir documents PDF ci-joint

```
% Convertir_avi_jpg

clear all
close all

f=rdir('*.avi')

v=VideoReader(f{1});
v.Duration
v.FrameRate
Nb_images=v.Duration*v.FrameRate

for i=1:100:Nb_images;

    % Regler l'horloge
    v.CurrentTime=(i-1)/(v.FrameRate);

    % Lire l'image
    image=readFrame(v);

    % La sauver en jpg avec son numero
    nom_fichier=f{1};
    nom_fichier=nom_fichier(1:end-4)
    imwrite(image,strcat(nom_fichier,'_',num2str(i),'.jpg'),'jpg')
end
```

Published with MATLAB® R2017b

```
% Traitement des images sans particule

clear all
close all

X=rdir('*.jpg');
A=zeros(1,14);

for i=1:14;
    T1=imread(X{i});
    T2=T1(50:1040,720:1100); % Sélection de la zone intéressante
    B=im2bw(T2,0.30);
    C=fft2(B);
    D=abs(fftshift(C));
    figure, imagesc([0,1/0.0187],[0,1/0.0187],D),axis equal
    figure, imagesc(T2)
    figure, imagesc(B)
end
```

Published with MATLAB® R2017b

`% Confrontation des résultats expérimentaux avec la théorie`

```
clear all
close all

lambda=0.0054:0.0001:0.016;
f = sqrt(1/pi^2.*(9.8.*2.*pi./lambda + 0.055.*(pi^3).*8./
(1000.*lambda.^3)).*tanh(2.*pi*0.0125./lambda));
plot(lambda, f)
grid on;
xlabel('\lambda (m)')
ylabel('Fréquence d\'excitation (Hz)')
hold on
Lambda1exp=0.01.*[1.000, 0.935, 0.831, 0.789, 0.748, 0.711, 0.678,
0.647];
Lambda2exp=0.01.*[0.483, 0.4675, 0.44, 0.392, 0.374, 0.356, 0.338,
0.323];
Freqexp=[45, 50, 55, 60, 65, 70, 75, 80];
Lambda1experror=0.005.*[0.11, 0.10, 0.09, 0.08, 0.07, 0.06, 0.05,
0.04];
Lambda2experror=0.01.*[0.0075, 0.0070, 0.0065, 0.0060, 0.0055, 0.0050,
0.0045, 0.0040];
errorbar(Lambda1exp,Freqexp,Lambda1experror,'horizontal')
hold on
errorbar(Lambda2exp,Freqexp,Lambda2experror,'horizontal')
legend('Theory', '\lambda_{1}', '\lambda_{2}');
```

Published with MATLAB® R2017b

```

% Traitement des images avec particules

clear all
close all

X=rdir('*.txt');

Pfull1=load(X{2});
[P1,Ind1] = Initiation(Pfull1);

Pfull2=load(X{3});
[P2,Ind2] = Initiation(Pfull2);

[Cor] = Correlation(P1,P2);

[P1] = Jonction(P1,Ind1,P2,Ind2,Cor,1);

[Ind1,trajmax] = Init(P1);

for k=2:length(X)-1;
    Pfull=load(X{k+1});
    [P2,Ind2] = Initiation(Pfull);

    [Cor] = Correlation(P1,P2);

    [P1] = Jonction(P1,Ind1,P2,Ind2,Cor,k);

    [Ind1,trajmax] = Init(P1);

end

Ind=Ind1;
P=zeros(length(P1),14);
P(:,1)=P1(:,1);
P(:,2)=P1(:,2);
P(:,3)=P1(:,3);
P(:,4)=P1(:,4);

% Vitesse et accélération des particules
for n=1:trajmax;
    for k=(Ind(n)+2):Ind(n+1);
        vx=(P(k,3)-P(k-1,3))/(0.02*(P(k,2)-P(k-1,2)));
        vy=(P(k,4)-P(k-1,4))/(0.02*(P(k,2)-P(k-1,2)));
        v=sqrt(vx^2+vy^2);

        P(k,5)=vx;
        P(k,6)=vy;
        P(k,7)=v;

        if k>=Ind(n)+3;

```

```

        ax=(P(k,5)-P(k-1,5))/(0.02*(P(k,2)-P(k-1,2)));
        ay=(P(k,6)-P(k-1,6))/(0.02*(P(k,2)-P(k-1,2)));
        a=sqrt(ax^2+ay^2);

        P(k,9)=ax;
        P(k,10)=ay;
        P(k,11)=a;
    end
    k;
end

% Elimination des trajectoires incluant des reflets
parametre=8;
par_accel=5000;
testind=zeros(1,length(P));
b=1;
te=0;
for n=1:trajmax;
    j=Ind(n)+3+parametre;
    while j<=Ind(n+1) & te==0;
        M=mean(P(j-parametre:j,11))-max(P(j-parametre:j,11))/
(parametre+1);
        %Le - max permet d'éliminer l'influence d'un pic isolé sur la
        moy
        if M>par_accel;
            te=1;
            testind(1,b)=P(j,1);
            b=b+1;
        end
        j=j+1;
    end
    te=0;
end

testind(testind==0)=[ ];
Long=length(testind);
for q=1:Long;
    h=testind(1,Long+1-q);
    Pdeb=P(1:Ind(h),:);
    Pfin=P(Ind(h+1)+1:length(P),:);
    P=[Pdeb;Pfin];
end

Nbpart=P(length(P),1)-Long;

% Remodélisation de P : réindissage des trajectoires
[Ind,trajmax] = Init(P);

% Travail sur les vitesses et les accélérations
Vmoy=0;
Vmoypic=0;

```

```

Ntot=0;
Ntotpic=0;

Amoy=0;
Amoypic=0;
Ntota=0;
Ntotpica=0;

Duree=0;

for n=1:trajmax;
    i=0; % indice qui permet de savoir si pour une particule, c'est le
    premier pic
    for k=(Ind(n)+2):Ind(n+1);

        Duree=Duree+0.02*(P(k,2)-P(k-1,2));

        if P(k,7)>100;
            P(k,13)=1;
            if i~=0 & (k-i)>1; % car plusieurs points peuvent être en
            haut du pic
                P(k,14)=0.02*(P(k,2)-P(i,2));
            end
            i=k;
            Ntotpic=Ntotpic+1;
            Vmoypic=Vmoypic+P(k,7);
            if k>=Ind(n)+3;
                Ntotpica=Ntotpica+1;
                Amoypic=Amoypic+P(k,11);
            end
        else;
            Ntot=Ntot+1;
            Vmoy=Vmoy+P(k,7);
            if k>=Ind(n)+3;
                Ntota=Ntota+1;
                Amoy=Amoy+P(k,11);
            end
        end

        if k==(Ind(n)+2);
            P(k,8)=0.02*(P(k,2)-P(k-1,2));
        else;
            P(k,8)=P(k-1,8)+0.02*(P(k,2)-P(k-1,2));
        end
    end
end
X=[0 length(X)*1200*0.02];

Vmoy=Vmoy/Ntot;
Vmoypic=Vmoypic/Ntotpic;
Yvmoy=[Vmoy Vmoy];
Yvmoypic=[Vmoypic Vmoypic];
Densitev=Ntotpic/Duree;

```

```

Amoy=Amoy/Ntota;
Amoypic=Amoypic/Ntotpica;
Yamoy=[Amoy Amoy];
Yamoypic=[Amoypic Amoypic];

% Affichage des trajectoires
figure
for n=1:trajmax;
    plot(0.19*P(Ind(n)+1:Ind(n+1),3),0.19*P(Ind(n)+1:Ind(n+1),4));
    title(['Trajectoire des particules   Nbparticules :
',num2str(Nbpart),' Durée totale de suivi (équivalent une
particule) : ',num2str(Duree),'s']);xlabel('x (mm)');ylabel('y (mm)')
    hold on
end
axis equal

% Affichage des vitesses et accélérations
figure
subplot 211
plot(X,0.19*Yvmoy,'r')
hold on
plot(X,0.19*Yvmoypic,'b')
hold on
for n=1:trajmax;
    plot(P(Ind(n)+2:Ind(n+1),8),0.19*P(Ind(n)+2:Ind(n+1),7));
    hold on
end
Vmoymm=0.19*Vmoy;
Vmoypicmm=0.19*Vmoypic;
title(['Vitesse des particules',' Vmoy ',num2str(Vmoymm),'mm/s Vmoypic
',num2str(Vmoypicmm),'mm/s Densité ',num2str(Densitev),'Nbpic/
s']);xlabel('Temps (s)');ylabel('Vitesse (mm/s)')

subplot 212
plot(X,0.19*Yamoy,'r')
hold on
plot(X,0.19*Yamoypic,'b')
hold on
for n=1:trajmax;
    plot(P(Ind(n)+3:Ind(n+1),8),0.19*P(Ind(n)+3:Ind(n+1),11));
    hold on
end
Amoymm=Amoy;
Amoypicmm=Amoypic;
title(['Accélération des particules',' Amoy ',num2str(Amoymm),'mm/
s{2} Amoypic ',num2str(Amoypicmm),'mm/s{2}']);xlabel('Temps
(s)');ylabel('Accélération (mm/s{2})')

% Affichage période entre deux pics de vitesse
figure
Ord=P(:,14);
Ord(Ord==0)=[];

```

```

% Histogramme période entre deux pics
subplot 211
hist(Ord,200);title('Histogramme des périodes');xlabel('Période
(s)');ylabel('Nb de particules')
hold on
A=0.02:0.01:6;
n=length(A);
B=zeros(1,n);
for k=1:n;
    B(1,k)=4/A(1,k);
end
plot(A,B)
legend('Histogramme','y=4/T')

subplot 212
hist(1./Ord,50);title('Histogramme des fréquences');xlabel('Fréquence
(Hz)');ylabel('Nb de particules')

% Fréquence spatiale
L1=round(max(P(:,3)));
L2=round(max(P(:,4)));
Figure=zeros(L1,L2);
for n=1:length(P);
    Figure(round(P(n,3)),round(P(n,4)))=100;
end
Figfft=fft2(Figure);
FigAbs=abs(fftshift(Figfft));
figure, imagesc([0,1/0.0187],[0,1/0.0187],FigAbs),axis equal
title('Transformée de Fourier 2D de l'ensemble des trajectoires des
particules')

```

Published with MATLAB® R2017b

```
% Fonction Correlation
```

```
function [Cor] = Correlation(P1,P2)
```

```
M1=max(P1(:,2));
```

```
trajmax1=P1(length(P1),1);
```

```
Ind1=zeros(3,trajmax1);
```

```
i=1;
```

```
for k=1:length(P1);
```

```
    if P1(k,2)==M1;
```

```
        Ind1(1,i)=P1(k,3);
```

```
        Ind1(2,i)=P1(k,4);
```

```
        Ind1(3,i)=P1(k,1);
```

```
        i=i+1;
```

```
    end
```

```
end
```

```
trajmax2=P2(length(P2),1);
```

```
Ind2=zeros(3,trajmax2);
```

```
i=1;
```

```
for k=1:length(P2);
```

```
    if P2(k,2)==0;
```

```
        Ind2(1,i)=P2(k,3);
```

```
        Ind2(2,i)=P2(k,4);
```

```
        Ind2(3,i)=P2(k,1);
```

```
        i=i+1;
```

```
    end
```

```
end
```

```
L1=length(Ind1);
```

```
L2=length(Ind2);
```

```
m=max(L1,L2);
```

```
Cor=zeros(2,m);
```

```
j=1;
```

```
for a=1:L1;
```

```
    for b=1:L2;
```

```
        if Ind1(1,a)==Ind2(1,b) & Ind1(2,a)==Ind2(2,b);
```

```
            Cor(1,j)=Ind1(3,a);
```

```
            Cor(2,j)=Ind2(3,b);
```

```
            j=j+1;
```

```
        end
```

```
    end
```

```
end
```

Published with MATLAB® R2017b

```

% Fonction Jonction

function [M] = Jonction(P1,Ind1,P2,Ind2,Cor,nb)

trajmax1=P1(length(P1),1);
M1=length(P1);
trajmax2=P2(length(P2),1);
M2=length(P2);

%Paramètres sélection particules _ dist et nbpts
nbpts=300;
dist=50;

P=zeros(M1+M2,5);

traj=1;
n1=1;
i=1;
while n1<=trajmax1;
    d=sqrt((P1(Ind1(n1)+1,3)-P1(Ind1(n1+1),3))^2+(P1(Ind1(n1)+1,4)-
P1(Ind1(n1+1),4))^2);
    C=Cor(1,:);
    C(C==0)=[];
    [test,ind] = Test(C,n1);
    if Ind1(n1+1)-Ind1(n1)>nbpts & d>dist;
        if test==0;
            for k=Ind1(n1)+1:Ind1(n1+1);
                P(i,1)=traj;
                P(i,2)=P1(k,2);
                P(i,3)=P1(k,3);
                P(i,4)=P1(k,4);
                P(i,5)=P1(k,5);
                i=i+1;
            end
        end
        if test==1;
            for k=Ind1(n1)+1:Ind1(n1+1);
                P(i,1)=traj;
                P(i,2)=P1(k,2);
                P(i,3)=P1(k,3);
                P(i,4)=P1(k,4);
                P(i,5)=P1(k,5);
                i=i+1;
            end
            indice=Cor(2,ind);
            %ne pas prendre première valeur (correlation)
            for j=Ind2(indice)+2:Ind2(indice+1);
                P(i,1)=traj;
                P(i,2)=1200*nb+P2(j,2);
                P(i,3)=P2(j,3);
                P(i,4)=P2(j,4);
                P(i,5)=P2(j,5);
            end
        end
    end
    n1=n1+1;
end

```

```

        i=i+1;
    end
    end
    traj=traj+1;
end
n1=n1+1;
end

n2=1;
while n2<=trajmax2;
    d=sqrt((P2(Ind2(n2)+1,3)-P2(Ind2(n2+1),3))^2+(P2(Ind2(n2)+1,4)-
P2(Ind2(n2+1),4))^2);
    [test,ind] = Test(Cor(2,:),n2);
    if Ind2(n2+1)-Ind2(n2)>nbpts & d>dist;
        if test==0;
            for k=Ind2(n2)+1:Ind2(n2+1);
                P(i,1)=traj;
                P(i,2)=1200*nb+P2(k,2);
                P(i,3)=P2(k,3);
                P(i,4)=P2(k,4);
                P(i,5)=P2(k,5);
                i=i+1;
            end
        end
        traj=traj+1;
    end
    n2=n2+1;
end

M=P(1:i-1,:);

```

Published with MATLAB® R2017b

```
% Fonction Test

function [a,indice] = Test(M,valeur)
%Attention M doit être une ligne
L=length(M);
a=0;
indice=0;
k=1;
while k<=L;
    if M(k)==valeur;
        a=1;
        indice=k;
    end
    k=k+1;
end
```

Published with MATLAB® R2017b

```
% Fonction Initiation

function [P,Ind] = Initiation(Pfull)

P=zeros(length(Pfull),5);
P(:,1)=Pfull(:,2);
P(:,2)=Pfull(:,3);
P(:,3)=Pfull(:,4);
P(:,4)=Pfull(:,5);
trajmax=P(length(P),1);

Ind=zeros(1,trajmax+1);
j=1;
i=1;
for j=1:(trajmax-1);
    while P(i,1)==j;
        i=i+1;
    end
    Ind(1,j+1)=i-1;
end
Ind(1,trajmax+1)=length(P);
```

Published with MATLAB® R2017b

`% Fonction Init`

`function [Ind,trajmax] = Init(P)`

`trajmax=length(P);`

`Ind=zeros(1,trajmax+1);`

`j=1;`

`i=1;`

`for j=1:(trajmax-1);`

`while P(i,1)==j;`

`i=i+1;`

`end`

`Ind(1,j+1)=i-1;`

`end`

`Ind(1,trajmax+1)=length(P);`

Published with MATLAB® R2017b