
PSE - Protocoles et méthodes - Flexion et fracture de la banquise

Montage expérimental complet

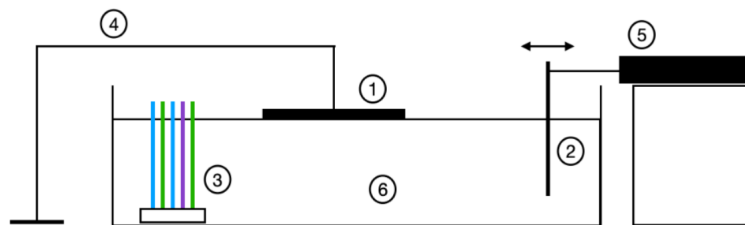


FIGURE 1 – Schéma du montage expérimental

Le montage expérimental est composé d'une cuve en plexiglas de dimension intérieure ($L = 1500$ mm, $l = 130$ mm et $H = 150$ mm) remplie d'eau colorée (6) avec du bleu de méthylène. L'eau colorée va permettre un meilleur traitement des images (cf partie "Enregistrement de film"). Trois sous-parties du montage se distinguent : un montage de génération de vagues composé d'un piston (5) relié à une plaque en plexiglas (2), d'une plaque servant de banquise (1) positionnée entre deux plaques métalliques maintenues par un serre-joint relié à une potence (4), un système d'atténuation des vagues réfléchies (3). Chacunes de ces sous-parties sont expliquées en détail dans la suite de ce document.

Réalisation des plaques (simulation de la banquise)

Pour fabriquer les plaques, nous utilisons un mélange de paraffine et d'acide stéarique, les deux constituants de la cire à bougie. Dans un premier temps, nous déposons une masse m_1 d'acide stéarique et une masse m_2 de paraffine dans un cristallisateur que nous faisons fondre par la suite dans un bain marie réalisé à l'aide d'un autre cristallisateur déposé sur une plaque chauffante dont on contrôle la température (FIGURE 2-A). Nous coulons ensuite ce mélange dans un plat rempli d'eau chaude en veillant à ce que le mélange recouvre la totalité du plat, le mélange peut être étalé avec une spatule métallique si besoin. Nous utilisons un plat cannelé (FIGURE 2-B) de dimension : $L = 300$ mm, $l = 230$ mm, $H = 50$ mm). Nous laissons ensuite refroidir à température ambiante le mélange jusqu'à solidification complète de la plaque en veillant à ne pas bouger le plat pour obtenir une plaque la plus fine possible : cette dernière étape prend environ une heure. La dernière étape consiste à découper la plaque aux dimensions requises (130 mm x 220 mm) à l'aide d'un fer à souder.

Nous utilisons $m_1 = 17$ g et $m_2 = 40$ g pour obtenir une plaque dont ces propriétés mécaniques se rapprochent de celle de la glace. Pour déterminer ces masses, nous avons réalisé 3 échantillons dont nous faisons varier les proportions paraffine/acide stéarique (80/20, 0/100, 70/30).

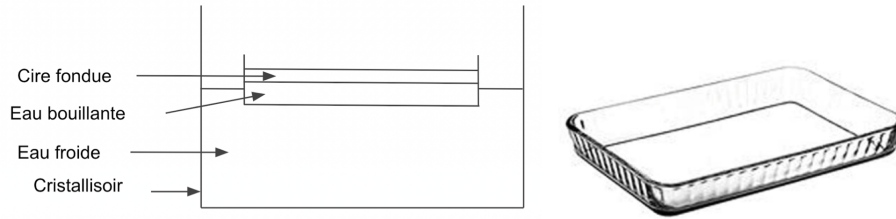


FIGURE 2 – A) Bain marie utilisé (à gauche) b) Plat cannelé (à droite)

Mesure du module d'Young des plaques

La mesure du module d'Young des plaques est réalisée en utilisant la méthode de flexion faible des poutres : la plaque est encastree à l'un de ces côtés et différentes masses sont positionnées sur l'autre de ces côtés. L'encastrement est réalisé en plaçant la plaque entre 2 pièces métalliques maintenues avec un petit serre joint accroché à une potence.



FIGURE 3 – Déformation d'une plaque - méthode de flexion faible des poutres

Pour chaque plaque de dimension L , nous positionnons à une distance x connue différentes masses connues et nous prenons une photographie, avec un appareil photo sur pied, de la plaque déformée. Sur *ImageJ*, nous procédons ensuite à l'analyse des photos : pour chacune d'entre elles, nous mesurons la distance de la déformée (distance entre la position d'équilibre de la plaque et sa position lorsqu'elle est déformée, à l'abscisse x). Nous en déduisons alors la valeur du module d'Young selon la relation : $\delta = \frac{mg}{2EI} (Lx^2 - \frac{x^3}{3})$ soit $E = \frac{mg}{2I\delta} (Lx^2 - \frac{x^3}{3})$ avec I le moment quadratique de la plaque $I = \frac{bh^3}{12}$ où b est la largeur de la plaque et h son épaisseur.

Mise en place du système de génération de vagues

Les vagues sont générées à l'aide d'une plaque en plexiglas ($L = 160$ mm, $l = 125$ mm et $e = 8$ mm) mise en mouvement à l'aide d'un piston. Le piston est fixé sur une plaque en plexiglas et la hauteur du dispositif est ajustée par un support élévateur sur lequel la plaque en plexiglas est posée. Le piston est un "electric cylinders" de série LZB (FIGURE 4) contrôlé par une carte Arduino. Un amplificateur est placé entre la carte Arduino et le piston. L'amplitude et la fréquence sont contrôlées et sont modifiées à l'aide du programme de la FIGURE 5.



FIGURE 4 – Piston "electric cylinders" de série LZB

```

#include <Controllino.h>

// the setup function runs once when you press reset (CONTROLLINO RST button) or connect the CONTROLLINO to the PC
void setup() {
  // initialize all used digital output pins as outputs
  pinMode(CONTROLLINO_D0, OUTPUT);
  pinMode(CONTROLLINO_D1, OUTPUT); // note that we are using CONTROLLINO aliases for the digital outputs
  pinMode(CONTROLLINO_D2, OUTPUT);
  pinMode(CONTROLLINO_D3, OUTPUT); // the alias is always like CONTROLLINO_
  pinMode(CONTROLLINO_D4, OUTPUT); // and the digital output label as you can see at the CONTROLLINO device
  pinMode(CONTROLLINO_D5, OUTPUT); // next to the digital output screw terminal
  pinMode(CONTROLLINO_D6, OUTPUT);
  pinMode(CONTROLLINO_D7, OUTPUT);
  // digitalWrite(CONTROLLINO_D3, HIGH);
  // digitalWrite(CONTROLLINO_D6, LOW);
  // delay(100);
  // digitalWrite(CONTROLLINO_D3, LOW);
  // delay(10);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(CONTROLLINO_D3, HIGH);
  digitalWrite(CONTROLLINO_D6, LOW);
  delay(150);
  digitalWrite(CONTROLLINO_D3, LOW); // by using CONTROLLINO aliases instead of Arduino pin numbers
  delay(10); // you ensure sketch portability between all CONTROLLINO variants
  digitalWrite(CONTROLLINO_D3, HIGH);
  digitalWrite(CONTROLLINO_D6, HIGH);
  delay(150);
  digitalWrite(CONTROLLINO_D3, LOW);
  delay(10);
  // while(1);
}
}

```

FIGURE 5 – Programme de contrôle de la carte Arduino

Dispositif d'atténuation des vagues réfléchies

Pour s'affranchir des ondes réfléchies en bout de cuve, nous avons développé un système d'atténuation (FIGURE 6-A) inspiré d'une mangrove marine que nous plaçons en bout de cuve. Pour cela, nous fabriquons des tubes constitués d'un mélange de polymère (aussi appelé "pâte à dentiste"), ces tubes sont réalisés en coulant le polymère (liquide) dans une paille de hauteur 10 cm puis solidification pendant une trentaine de minutes. Ces tubes sont ensuite placés dans une plaque en plexiglas leur permettant de rester droit. La plaque de plexiglas est une carrée de côté 125 mm sur laquelle des trous sont percés selon la FIGURE 6-B.

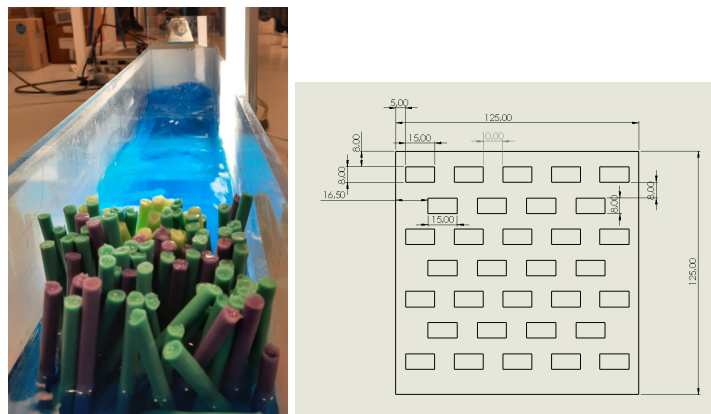


FIGURE 6 – A) Système d'atténuation des vagues réfléchies (à gauche) B) Schéma de la plaque de plexiglas à la base du dispositif

Enregistrement de film

Les données brutes récoltées lors de nos expériences sont des vidéos enregistrées à l'aide d'une caméra enregistrant les vidéos en couleur et munie d'un objectif de focale 8 mm permettant d'observer un grand champ tout en étant proche de la cuve. Nous veillons à mettre un panneau LED derrière la cuve dans le but d'obtenir un éclairage uniforme nécessaire pour traiter la vidéo par la suite.

Mesure des caractéristiques des vagues

La mesure que nous allons détailler va permettre de déterminer la hauteur et la longueur d'ondes des vagues dans la cuve en fonction de la fréquence de la commande sinusoïdale envoyée au piston. L'amplitude étant fixée à la valeur minimale de l'amplificateur ($A = 7.8V$). Pour cette mesure, nous mettons le montage (système de génération de vagues + système d'atténuation des vagues réfléchies) en place et nous enregistrons une série de vidéos pour des fréquences d'excitation différentes. Les vidéos sont ensuite traitées à l'aide du programme *Matlab* de la (FIGURE 7) .

```
v = VideoReader('video1.avi');
for i=1:49
    frame = read(v,i);
    subplot(7,7,i)
    imshow(frame)
end

frameinit=read(v,3);
framevag=read(v,31);
figure()
imshow(frameinit)
figure()
imshow(framevag)
im_binaryinit=im2bw(frameinit,0.9);
figure()
imshow(im_binaryinit)
im_binary=im2bw(framevag,0.9);
figure()
imshow(im_binary)
L=length(im_binary)
H=length(im_binary(:,1));
echellepex=980-411 %pixels
echellereel=34 %mm
Hreel=echellepex*H/echellepex;
%co=im_binary(:,col);
h_lum=[];

for j=30:L-20
    for i=H:-1:4
        a=i;
        if im_binary(i,j)==0
            if im_binary(i-1,j)==0
                if im_binary(i-3,j)==0
                    h_lum(j)=(H-i)*echellereel/echellepex;
                end
            end
        end
    end
end

h_lum
figure()
plot(h_lum)
h_max=(H-min(h_lum))*echellereel/echellepex
L2=length(im_binaryinit)
H2=length(im_binaryinit(:,1));
echellepex=27 %pixels
echellereel=10 %mm
Hreel=echellepex*H/echellepex;
%co=im_binary(:,col);
h_lum2=[];
for j=20:L-20
    for i=H2:-1:4
        a=i;
        if im_binaryinit(i,j)==0
            if im_binaryinit(i-1,j)==0
                if im_binaryinit(i-3,j)==0
                    h_lum2(j)=(H2-i)*echellereel/echellepex;
                end
            end
        end
    end
end
h_lum2;
figure()
plot(h_lum2)
h_max2=(H2-min(h_lum2))*echellereel/echellepex
```

FIGURE 7 – Programme *Matlab*

Mesure de la flexion de la plaque

Le montage complet est mis en place et nous excitons la plaque via des vagues dont la fréquence est contrôlée. Cette mesure va permettre de connaître la fréquence d'oscillation de la plaque et de l'amplitude de déformation en fonction de la fréquence des vagues. Nous traitons les vidéos à l'aide du programme *Matlab* suivant élaboré à partir du programme précédent et optimisé pour traiter l'ensemble des images de la vidéo. La sortie de ce programme est une courbe donnant l'évolution de la déformation tout au long de la vidéo.

```

%% echelle
% framesize=imread('echelle.bmp');
% framesize_crop=imcrop(framesize,1.0e+03 * [0.18 0.5 1.4400 0.5280]);
% figure()
% imshow(framesize_crop)

y=40.1;
x=9.9;
L_m=26;
H_m=108;
echelle_x=x/L_m;
echelle_y=y/H_m;

%% initialisation
frameinit=imread('echelle1.bmp');
% imshow(frameinit)
% figure()
frameinitcrop=imcrop(frameinit,1.0e+03 * [0.18 0.4635 1.4400 0.5280]);
% imshow(frameinitcrop)
% figure()
im_binaryinit=im2bw(frameinitcrop,0.7);
imshow(im_binaryinit)

L=length(im_binaryinit);
H=length(im_binaryinit(:,1));
%co=im_binary(:,col);

h_lum=[];
for j=1:L
    for i=H:-1:4
        if im_binaryinit(i,j)==0
            if im_binaryinit(i-1,j)==0
                if im_binaryinit(i-3,j)==0
                    h_lum(j)=(H-i)*echelle_y;
                end
            end
        end
    end
end

%% traitement des images
% framevag=imread('ref2.bmp');
v = VideoReader('1.avi');

N=200;
d=[];
for k=1:N
    framevag=read(v,k);
    framevagcrop=imcrop(framevag,1.0e+03 * [0.18 0.4635 1.4400 0.5280]);
    % figure()
    % imshow(framevagcrop)

    im_binary=im2bw(framevagcrop,0.695);
    % figure()
    % imshow(im_binary)

    h_lum=[];
    for j=1:L
        for i=H:-1:4
            if im_binary(i,j)==0
                if im_binary(i-1,j)==0
                    if im_binary(i-3,j)==0
                        h_lum(j)=(H-i);
                    end
                end
            end
        end
    end
end
d(k)=h_lum(410);
end

% h_lum;
figure()
plot(d)
% h_max=(H-min(h_lum))*echelle_reel/echelle_pex
% L2=length(im_binaryinit);
% H2=length(im_binaryinit(:,1));
% %co=im_binary(:,col);

```

FIGURE 8 – Programme *Matlab*