

Protocoles et méthodes

1 Matériels et logiciels

1.1 Matériels

- Profilés en aluminium noir (fente 5mm) avec vis, écrous et équerres adaptés
- Boîte de 100g de bracelets caoutchouc - Office DEPOT - 150 x 1,5 mm
- Mobiles en plexiglas de 5mm d'épaisseur (voir annexes)
- Fils de cuivres
- GBF et câbles électriques
- Vibreur mécanique
- Tiges fileté relié au vibreur
- Cube mécanique creux (10 x 10 mm) pour faire l'angle entre le vibreur et le réseau
- Profilés en aluminium (fente 8mm) avec vis, écrous et coins adaptés
- Caméra Basler
- Câble Ethernet
- Ordinateur

1.2 Logiciel

- Logiciel Basler associé avec la caméra via le câble ethernet
- MatLab
- Image J

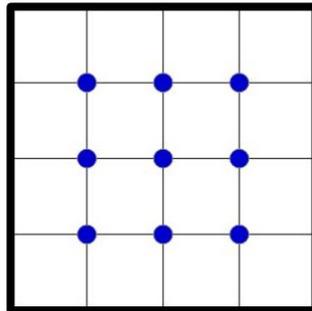
2 Protocoles

2.1 Réseaux

2.1.1 Réseau symétrique

Étapes de fabrication :

- Réalisation du cadre carré (45 x 45 mm) avec les profilés noirs
- Ajout des élastiques aux mobiles et au cadre suivant le schéma suivant

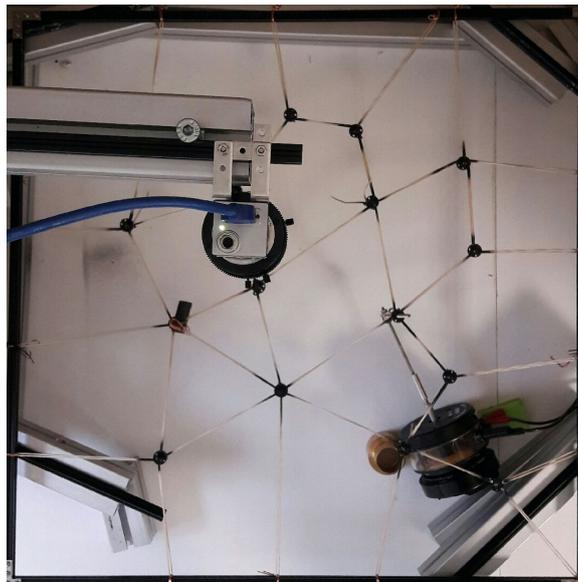


2.1.2 Réseau asymétrique

On réalise les mêmes étapes mais avec un placement aléatoire des mobiles.

2.2 Acquisition des données

Nous avons construit un support pour pouvoir mettre la caméra au-dessus du réseau.



Acquisition des images :

- Envoi d'un pulse sinusoïdale (15-20 Hz, 5 périodes, 10 Vpp) par le GBF vers le vibreur mécanique.

- Acquisition de la position d'un mobile via le logiciel de la caméra (500 images, 100 Hz)
- Extraire le chemin de la séquence d'images, l'intégrer au programme Matlab "detection_final.m" et lancer le programme
- Sélection de la zone d'intérêt dans la fenêtre qui s'affiche
- Affichage de la trajectoire du mobile choisi

2.3 Traitement des données

- Banque de trajectoire :

Pour faire la bibliothèque, on fait une acquisition du signal issu de l'antenne pour chaque position de la masse, et on stock le tout dans une matrice 2D 7x499. Le traitement des signaux est fait plus en aval.

- Corrélation croisée :

La fonction "phase_corr" renvoie la fonction de corrélation de deux signaux d'entrée. Les signaux d'entrée sont d'abord fenêtrés pour n'en garder que la partie utile. On utilise une fenêtre de Hamming. Ils sont ensuite filtrés pour enlever leur composante continue et normalisés, et enfin binarisés. La corrélation est ensuite effectuée, et le résultat est à nouveau filtré.

Le script "Comparaison_signaux" compare les signaux « attempt » aux signaux de la bibliothèque grâce à la fonction "phase_corr". Du bruit gaussien est ajouté artificiellement pour faire des statistiques.

2.4 Simulation

La simulation prend en compte les hypothèses et les caractéristiques du réseau suivantes :

- Les élastiques sont modélisés par des ressorts de raideur k de 17.780 $\text{kg}\cdot\text{s}^{-2}$ et une longueur à vide l_0 de 8 cm.
- La simulation prend en compte de la dissipation sous la forme d'un coefficient de frottement c de 0.0125 $\text{g}\cdot\text{s}^{-1}$.
- La distance entre les mobiles est de 14 cm et les masses de chaque mobile vaut 3.9g. La masse supplémentaire vaut 30g.

Fonction main(taille_reseau,freq,masse,rec,traj) :

- Le paramètre `taille_reseau` permet de choisir la taille du réseau. Les autres fonctions n'ont pas été prévues pour être utilisées avec d'autres valeurs de ce paramètre que 5. Cela correspond à un réseau 3x3.
- Le paramètre `freq` permet de choisir la fréquence de l'excitation.
- Le paramètre `masse` est une matrice donnant la masse de chaque mobile.
- Le paramètre `rec` vaut 0 ou 1. Cela permet d'enregistrer la trajectoire quand il vaut 0.
- Le paramètre `traj` permet de donner une trajectoire à utiliser pour l'excitation. C'est un paramètre optionnel.

Lorsqu'on ne lui donne pas d'arguments, le programme choisit 5 pour `taille_reseau`, une masse uniforme pour chaque mobile (3.9 g), une fréquence de 15Hz et pour `rec`, il prend 0.

Fonction `excitation(t,cst,n)` :

- Le paramètre `t` correspond au temps pour le signal d'excitation.
- Le paramètre `cst` permet de rajouter la position d'équilibre à l'excitation.
- Le paramètre `n` permet de choisir le type d'excitation appliqué au réseau.

Fonction `force_rappel(num_mob,position1,mur)` :

- Le paramètre `num_mob` indique le mobile sur lequel la fonction s'applique.
- Le paramètre `position1` correspond à la position à `t` de `num_mob`.
- Le paramètre `mur` correspond à la position d'origine de tous les points du réseau. Ce paramètre est créé par la fonction `main`.

Fonction `force_frottement(num_mob,vitesse,t)` :

- Le paramètre `num_mob` indique le mobile sur lequel la fonction s'applique.
- Le paramètre `vitesse` correspond à la vitesse à `t` de `num_mob`.
- Le paramètre `t` est un paramètre optionnel permettant de modifier la force de frottement pendant la simulation.

Fonction `point_suivant(force,position,vitesse,num_mob)` :

- Le paramètre `force` correspond à la force à appliquer à `t+1` à `num_mob`.
- Le paramètre `position` correspond à la position à `t` de `num_mob`.
- Le paramètre `vitesse` correspond à la vitesse à `t` de `num_mob`.
- Le paramètre `num_mob` indique le mobile sur lequel la fonction s'applique.

Ces fonctions sont utilisées par la fonction `main`.

3 Annexes

Dessin des mobiles

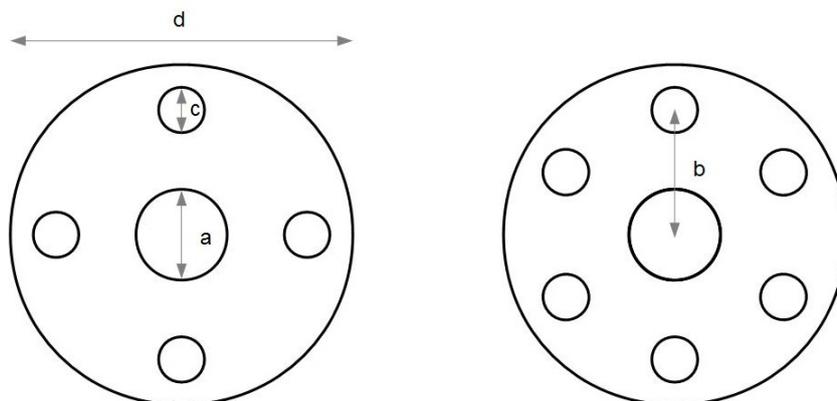


FIGURE 1 – Mobile à 4 trous pour réseau symétrique et à 6 trous pour réseau asymétrique. $a = 4$ mm, $b = 6.5$ mm, $c = 2$ mm, $d = 15$ mm