

Exemple de signal de clavier telephonique

Analyse de fréquences

Question 1

durée d'un signal de clavier: $d = 0.2$ s

fréquence d'échantillonnage: $F_{ech} = 44100$ Hz

$FRAME_SIZE = d * F_{ech} = 0.2 * 44100 = 8820$

résolution fréquentielle: **res_freq** = $F_{ech} / FRAME_SIZE = 5$ Hz

Question 2

Pour voir tout le signal, il suffit de mettre la `FRAME_SIZE` au nombre d'échantillons total du signal, ici **158760** (donné par la variable `SIZE`).

En visualisant tout le signal, on observe **7 pics**. On en déduit que des **fréquences apparaissent plusieurs fois**.

Question 3

Au minimum, on voit **2 pics**.

Au maximum, on voit **4 pics**.

Cela signifie que l'analyse de plusieurs touches se chevauchent. Il faudrait analyser par tranche de **0.1 s** au lieu de 0.2 s, afin d'être sûr d'analyser 1 touche à la fois.

On en déduit: **HOP_SIZE** = $F_{ech} * durée = 44100 * 0.1 = 4410$.

Question 5

On remarque qu'il y a, pour chaque frame, un nombre de maxima locaux de l'ordre de **10^3** . Or, on en cherche seulement 2, puisqu'on veut 2 fréquences. Afin de remédier à cela, il suffit de récupérer les 2 maxima locaux avec les plus grandes amplitudes.

Question 6

Précision: $freq_prec = res_freq / 2 = 2.5$ Hz.

On peut améliorer la précision à l'aide d'une fenêtre de **Hann** ainsi que l'**interpolation parabolique**.

Question 7

Après avoir observé les fréquences pour chaque touche, on peut désormais stocker ces valeurs. Pour des raisons de praticité et éviter la redondance, utilisons un tableau 4x3, contenant les touches (`c_tab`), ainsi que 2 tableaux contenant les fréquences (`line` et `col`).

Question 8

Même sans avoir optimisé l'affichage, ni redéfini `FRAME_SIZE` et `HOP_SIZE`, on peut deviner, à partir de la sortie:

```
00555555666888444666555000000
```

Que le numéro est le suivant:

```
0556846500
```

Notamment en remarquant que si un numéro est répété plus de fois que les autres, il apparaît sûrement 2 fois.

Détection de touches

En affichant les énergies à chaque frame, on remarque que celles qui sont utiles sont supérieurs à $*0.005$. On peut donc par exemple définir le seuil à cette valeur.

Analyse de numéros

durée minimale de pression d'une touche: $d = 0.065$ s $FRAME_SIZE = d * F_{ech} = 0.065 * 44100 = 2866.5$

Arrondissons donc à **2867**.

Voici ce que l'on obtient:

- A: 0556846500
- B: 227772888666
- C: 0556340548
- D: 0556846500
- E: 05Element not in list

En utilisant cette valeur pour `FRAME_SIZE` et `HOP_SIZE`, on parvient à analyser tous les sons, sauf `telE.wav`.

En affichant toutes les fréquences trouvées, on observe qu'elles sont pour la plupart à ± 1 Hz d'écart d'une fréquence attendue. Cependant le plus grand écart observé est de 12 Hz. Une solution, bien que pas forcément pérenne, serait d'autoriser un écart (fonction `nearest_freq_index`).

E: 0540006000