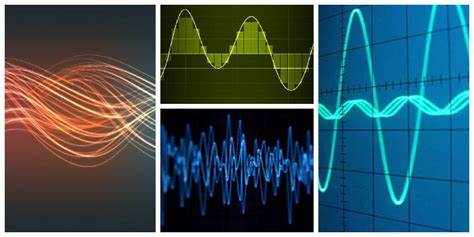
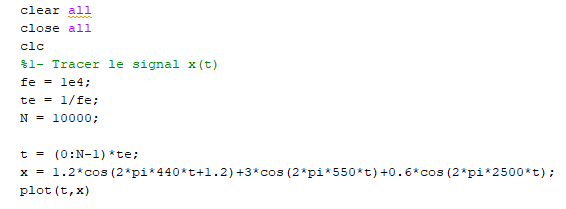
|  |
| --- |
| Rapport Tps Traitement de  Signal |
|  |
| 20 janvier  DJIGO Souleymane  Supervisée par : AMMOUR Alae |

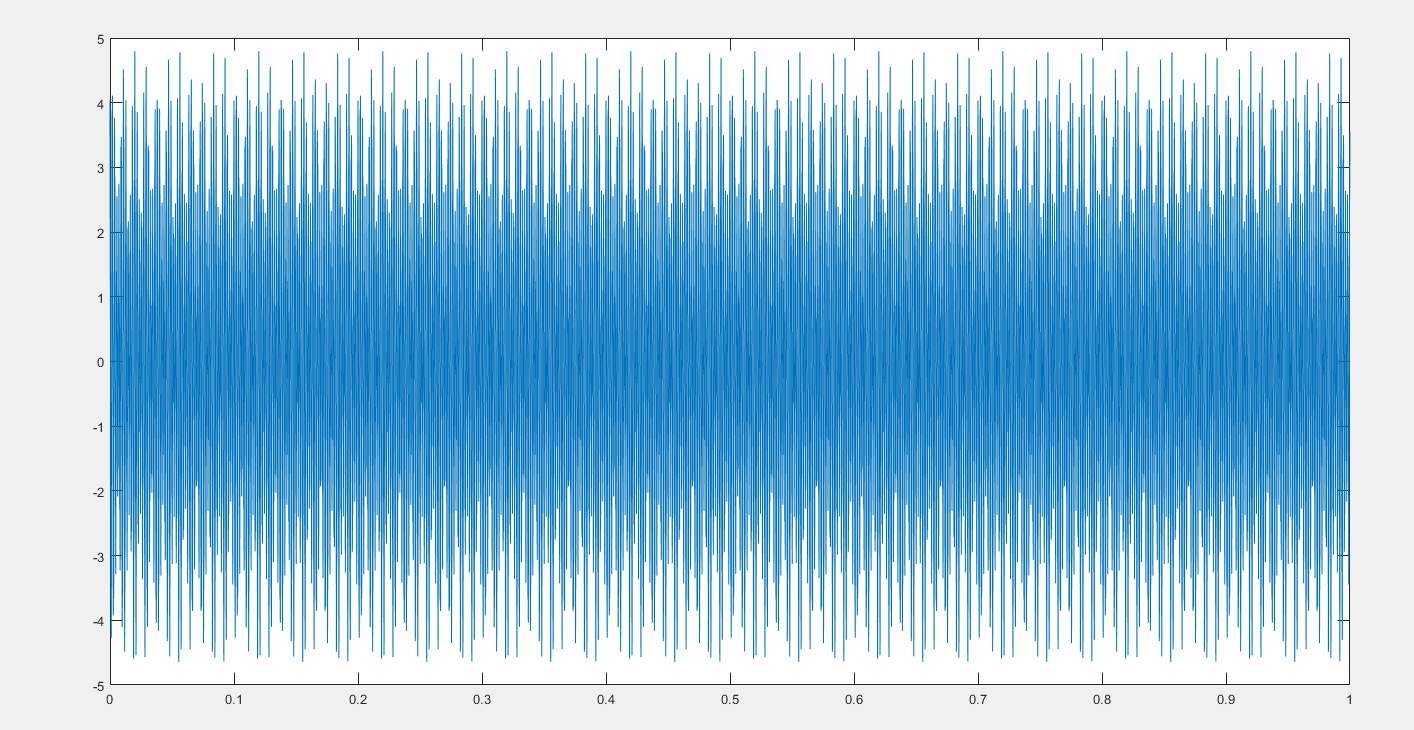


# PLAN

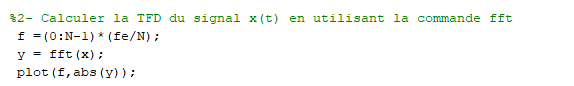
* TP1 : Analyse spectrale d’un signal Transformée de Fourier discrète
* Objectifs
* Représentation temporelle et fréquentielle
* Analyse fréquentielle du chant du rorqual bleu
* TP2 : Jeux de mots Synthèse et analyse spectrale d’une gamme de musique
* Objectifs
* Jeux de mots
* Synthèse et analyse spectrale d’une gamme de musique
* TP3 : Traitement d’un signal ECG
* Objectifs
* Suppression du bruit provoqué par les mouvements du corps
* Suppression des interférences des lignes électriques 50Hz
* Amélioration du rapport signal sur bruit
* Identification de la fréquence cardiaque avec la fonction d’autocorrélation
* TP4 : Filtrage Analogique
* Objectifs
* Filtrage et diagramme de Bode
* Dé-bruitage d'un signal sonore
* TP1 : Analyse spectrale d’un signal Transformée de Fourier discrète
* Objectifs
* Représentation de signaux et applications de la transformée de Fourier discrète (TFD) sous Matlab.
* Evaluation de l’intérêt du passage du domaine temporel au domaine fréquentiel dans l’analyse et l’interprétation des signaux physiques réels.
* Représentation temporelle et fréquentielle

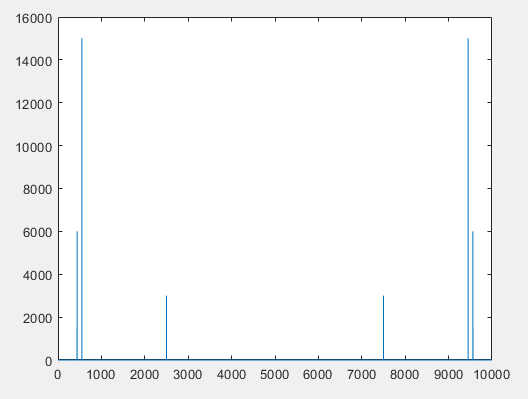
1. Tracer le signal x(t). Fréquence d’échantillonnage : fe = 10000Hz, Intervalle : Nombre d’échantillons : N = 5000. Pour approximer la TF continue d’un signal x(t), représenté suivant un pas Te, on utilise les deux commandes : fft et fftshif.

****

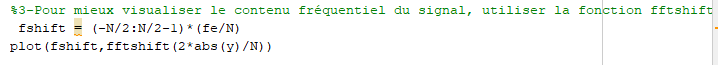
****

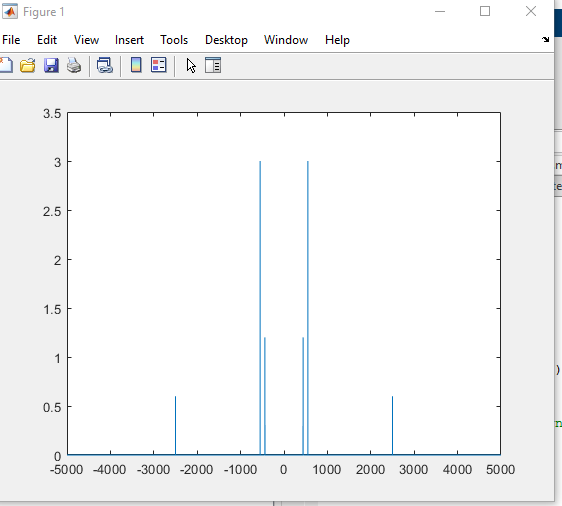
1. Calculer la TFD du signal x(t) en utilisant la commande fft, puis tracer son spectre en amplitude après avoir créé le vecteur f qui correspond à l'échantillonnage du signal dans l'espace fréquentiel. Utiliser la commande abs pour afficher le spectre d’amplitude.

****

****

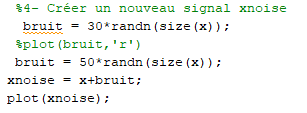
1. Pour mieux visualiser le contenu fréquentiel du signal, utiliser la fonction fftshift, qui effectue un décalage circulaire centré sur zéro du spectre en amplitude obtenu par la commande fft.

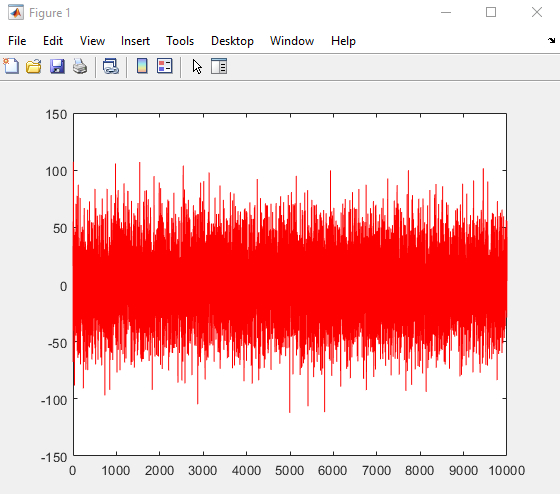


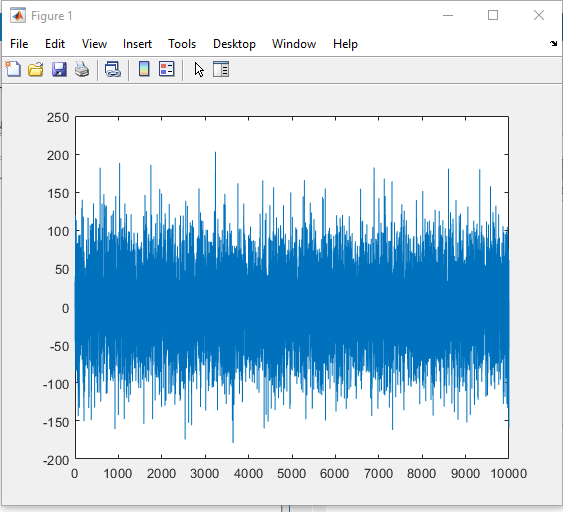


Un bruit correspond à tout phénomène perturbateur gênant la transmission ou l'interprétation d'un signal. Dans les applications scientifiques, les signaux sont souvent corrompus par du bruit aléatoire, modifiant ainsi leurs composantes fréquentielles. La TFD peut traiter le bruit aléatoire et révéler les fréquences qui y correspond.

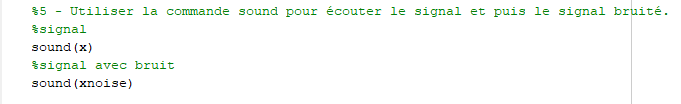
1. Créer un nouveau signal xnoise, en introduisant un bruit blanc gaussien dans le signal d’origine x(t), puis visualisez-le. Utiliser la commande randn pour générer ce bruit. Il est à noter qu’un bruit blanc est une réalisation d'un processus aléatoire dans lequel la densité spectrale de puissance est la même pour toutes les fréquences de la bande passante. Ce bruit suit une loi normale de moyenne 0 et d’écart type 1.



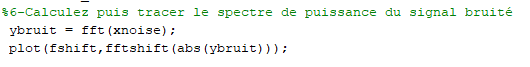


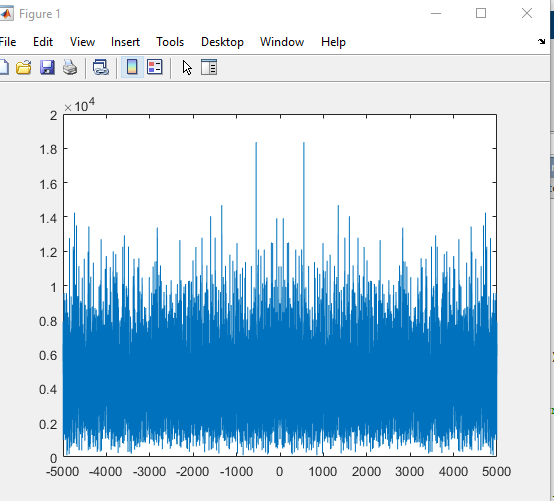


1. Utiliser la commande sound pour écouter le signal et puis le signal bruité.

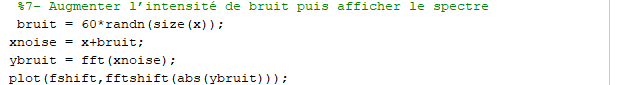


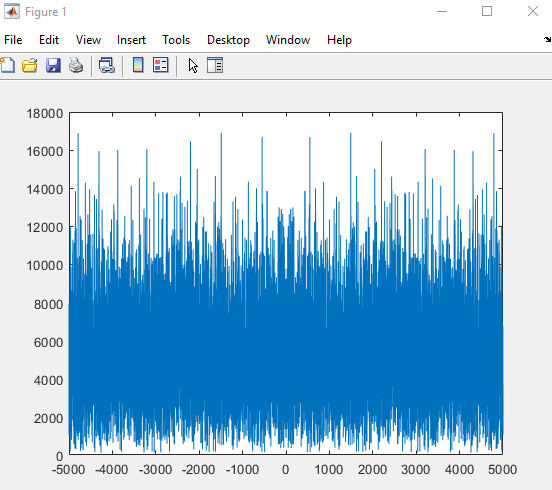
1. Calculez puis tracer le spectre de puissance du signal bruité centré à la fréquence zéro.





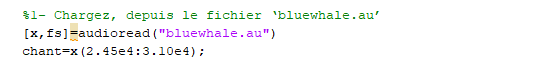
1. Augmenter l’intensité de bruit puis afficher le spectre. Interpréter le résultat obtenu.





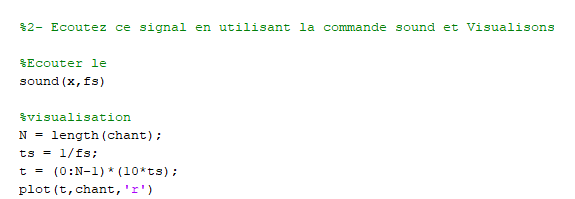
* Analyse fréquentielle du chant du rorqual bleu

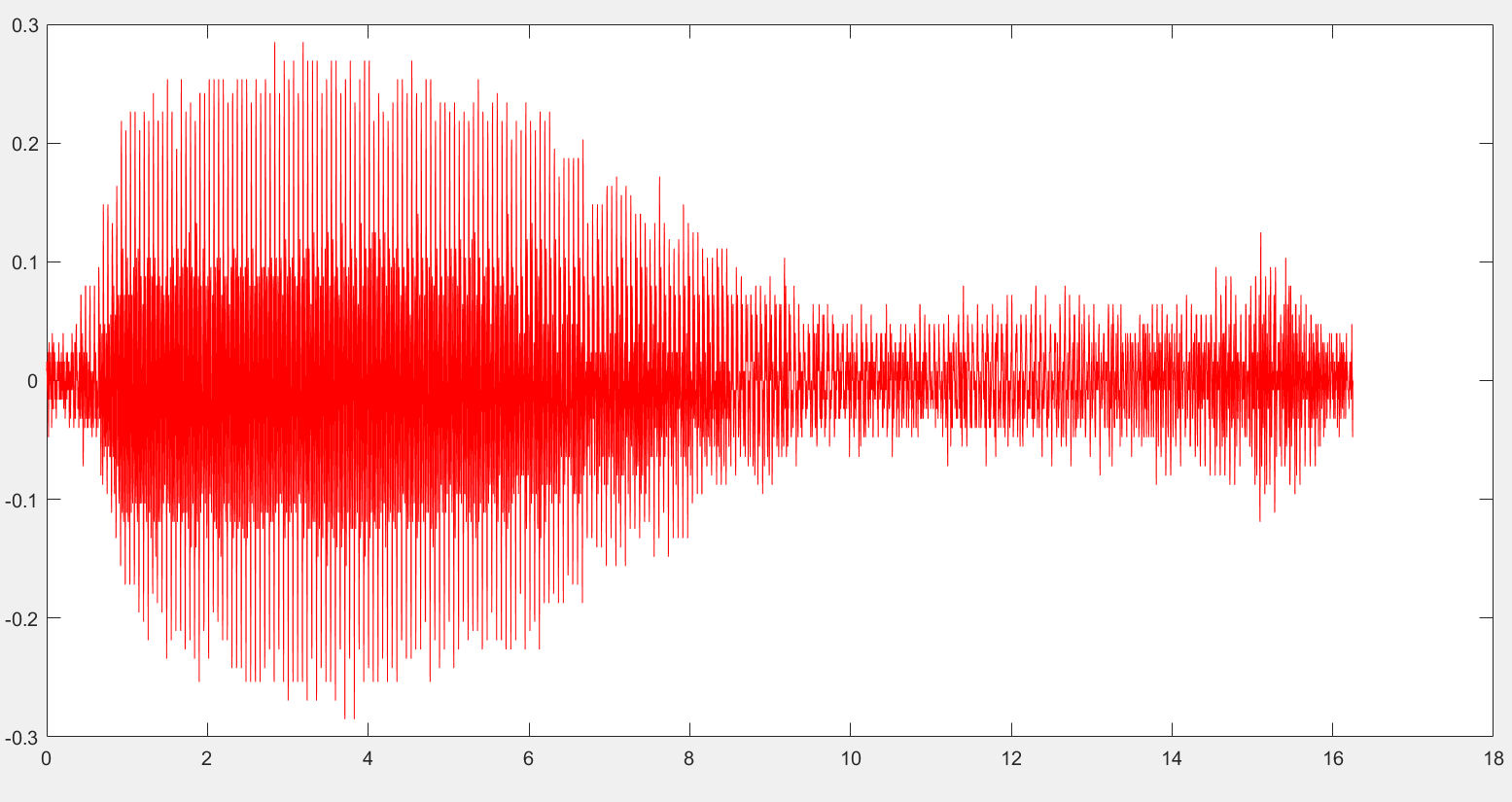
1. Chargez, depuis le fichier ‘bluewhale.au’, le sous-ensemble de données qui correspond au chant du rorqual bleu du Pacifique. En effet, les appels de rorqual bleu sont des sons à basse fréquence, ils sont à peine audibles pour les humains. Utiliser la commande audioread pour lire le fichier. Le son à récupérer correspond aux indices allant de 2.45e4 à 3.10e4.



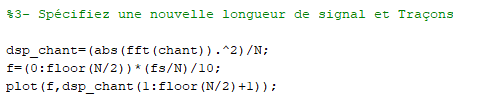
1. Ecoutez ce signal en utilisant la commande sound, puis visualisez-le.

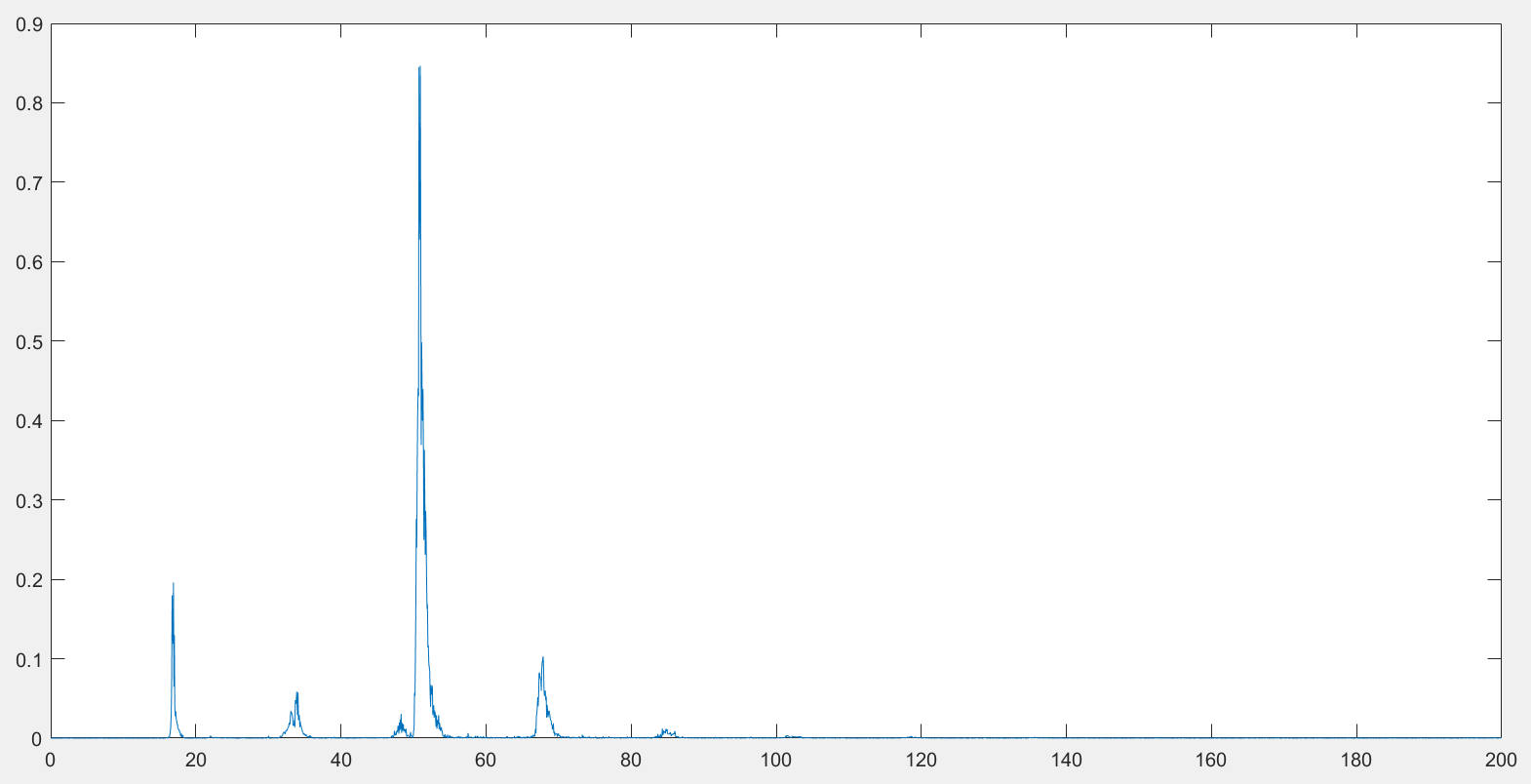
La TFD peut être utilisée pour identifier les composantes fréquentielles de ce signal audio. Dans certaines applications qui traitent de grandes quantités de données avec fft, il est courant de redimensionner l'entrée de sorte que le nombre d'échantillons soit une puissance de 2. fft remplit automatiquement les données avec des zéros pour augmenter la taille de l'échantillon. Cela peut accélérer considérablement le calcul de la transformation.



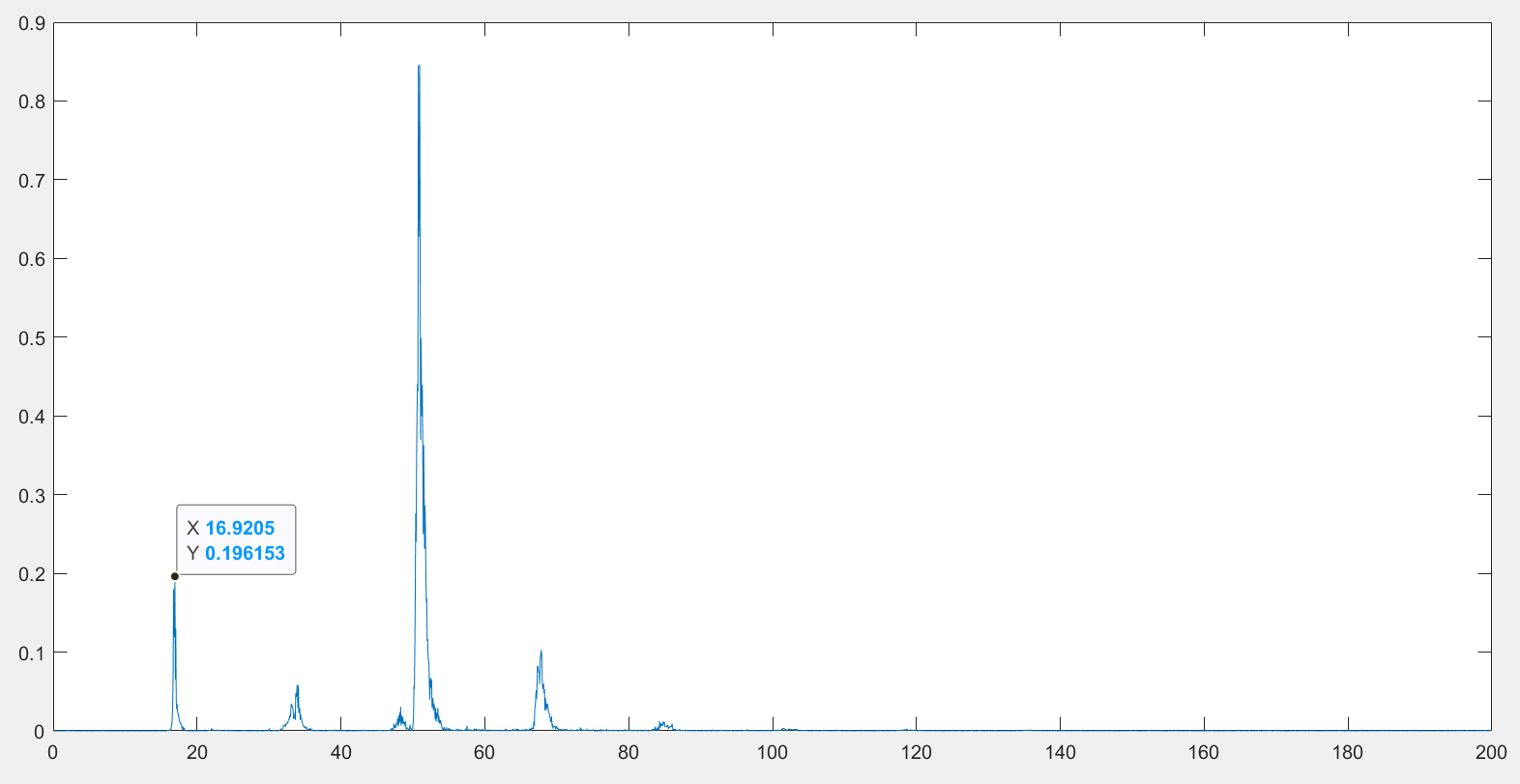


1. Spécifiez une nouvelle longueur de signal qui sera une puissance de 2, puis tracer la densité spectrale de puissance du signal.





1. Déterminer à partir du tracé, la fréquence fondamentale du gémissement de rorqual bleu : 16-17Hz



* TP2 : Jeux de mots Synthèse et analyse spectrale d’une gamme de musique
* Objectifs
* Comprendre comment manipuler un signal audio avec Matlab, en effectuant certaines opérations classiques sur un fichier audio d’une phrase enregistrée via un smartphone.
* Comprendre la notion des sons purs à travers la synthèse et l’analyse spectrale d’une gamme de musique.
* Jeux de mots

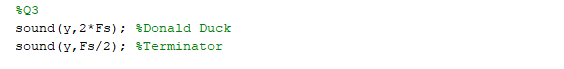
1. Sauvegardez ce fichier sur votre répertoire de travail, puis charger-le dans MATLAB à l’aide de la commande « audioread ».



1. Tracez le signal enregistré en fonction du temps, puis écoutez-le en utilisant la commande « sound »

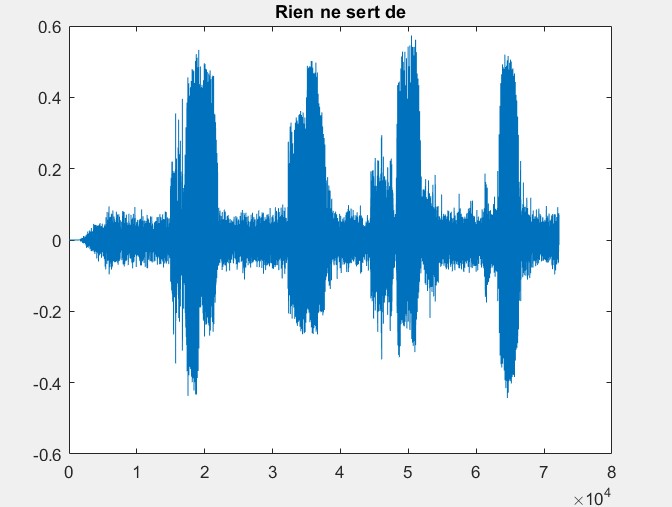


1. Cette commande permet d’écouter la phrase à sa fréquence d’échantillonnage d’enregistrement. Ecoutez la phrase en modifiant la fréquence d’échantillonnage à double ou deux fois plus petite pour vous faire parler comme « Terminator » ou « Donald Duck ». En effet, modifier la fréquence d’échantillonnage revient à appliquer un changement d’échelle y(t) = x(at) en fonction de la valeur du facteur d’échelle, cela revient à opérer une compression ou une dilatation du spectre initial d’où la version plus grave ou plus aigüe du signal écouté



1. Tracez le signal en fonction des indices du vecteur x, puis essayez de repérer les indices de début et de fin de la phrase « Rien ne sert de ».

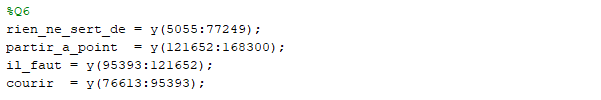




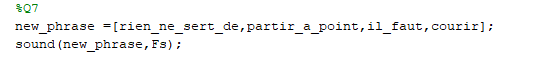
1. Pour segmenter le premier mot, il faut par exemple créer un vecteur « riennesertde » contenant les n premières valeurs du signal enregistré x qui correspondent à ce morceau. Créez ce vecteur, puis écoutez le mot segmenté



1. Segmentez cette fois-ci toute la phrase en créant les variables suivantes : riennesertde, courir, ilfaut, partirapoint.



1. Notez que le signal initial de parole est un vecteur colonne contenant un certain nombre de valeurs (length(x)). Réarrangez ce vecteur pour écouter la phrase synthétisée « Rien ne sert de partir à point, il faut courir ».

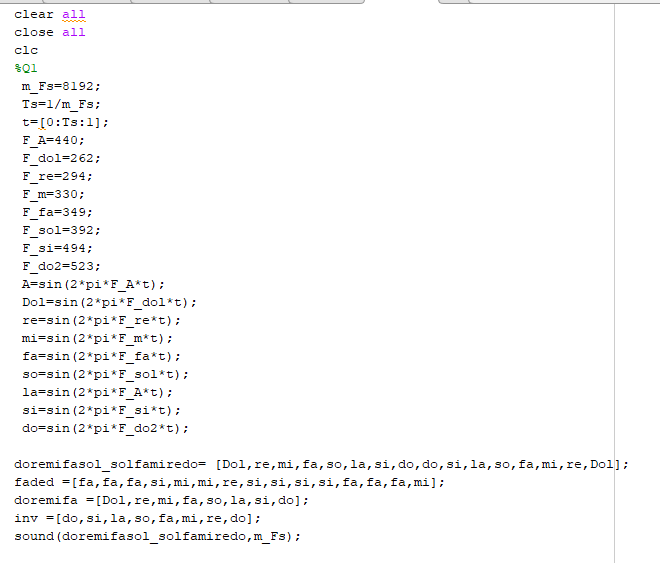


* Synthèse et analyse spectrale d’une gamme de musique
* Synthèse d’une gamme de musique

Les notes de musique produites par un piano peuvent être synthétisées approximativement numériquement. En effet, chaque note peut être considérée comme étant un son pur produit par un signal sinusoïdal. La fréquence de la note « La » est par exemple de 440 Hz

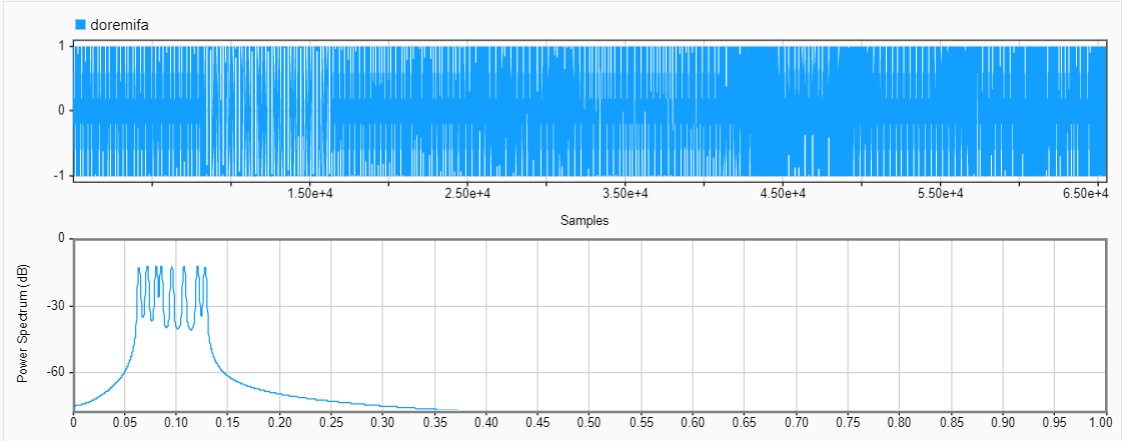
1. Créez un programme qui permet de jouer une gamme de musique. La fréquence de chaque note est précisée dans le tableau ci-dessous. Chaque note aura une durée de 1s. La durée de la gamme sera donc de 8s. La fréquence d’échantillonnage fe sera fixée à 8192 Hz.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Do1 | Ré | Mi | Fa | Sol | La | Si | Do2 |
| 262 Hz | 294 Hz | 330 Hz | 349 Hz | 392 Hz | 440 Hz | 494 Hz | 440 Hz |

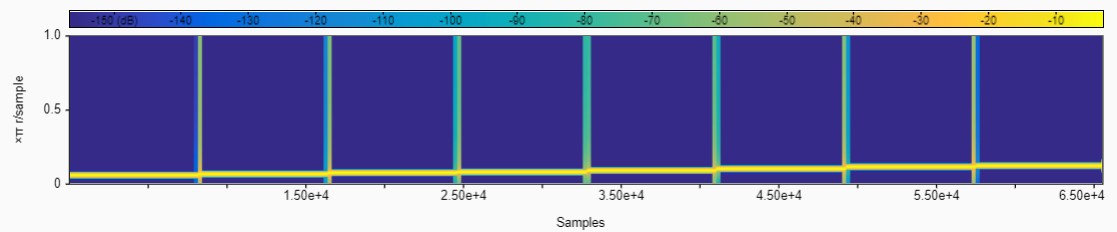


* Spectre de la gamme de musique

1. Utilisez l’outil graphique d’analyse de signaux signalAnalyzer pour visualiser le spectre de votre gamme. Observez les 8 fréquences contenues dans la gamme et vérifiez leur valeur numérique à l’aide des curseurs.

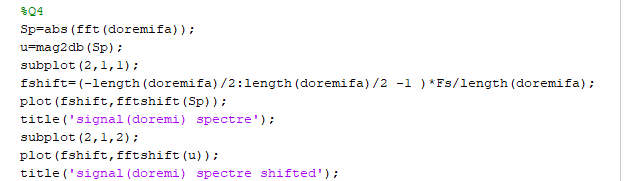


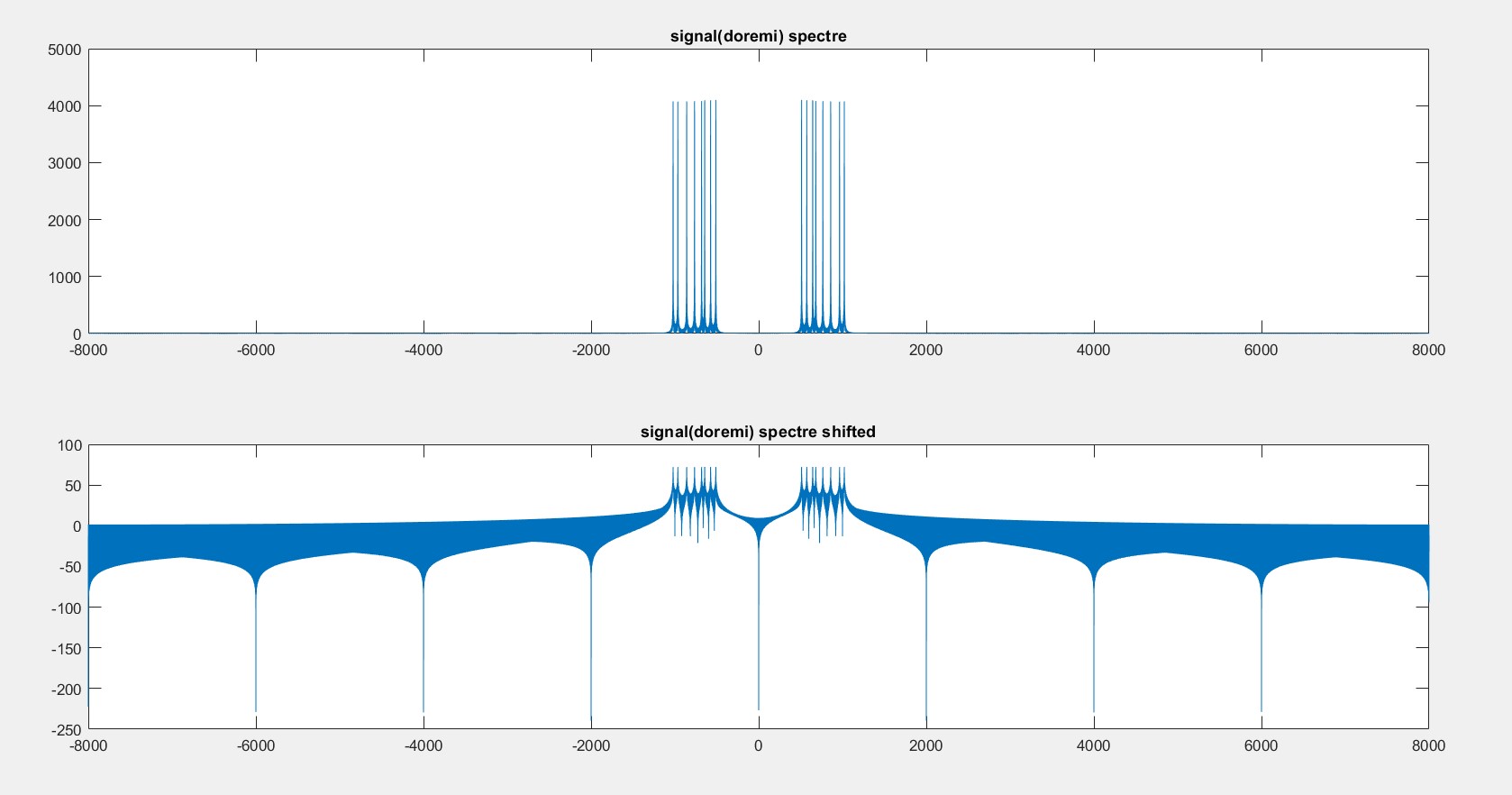
1. Tracez le spectrogramme qui permet de visualiser le contenu fréquentiel du signal au cours du temps (comme le fait une partition de musique) mais la précision sur l’axe des fréquences n’est pas suffisante pour relever précisément les 8 fréquences.



* Approximation du spectre d’un signal sinusoïdal à temps continu par FFT

1. Le spectre d’un signal à temps continu peut être approché par transformée de Fourier discrète (TFD) ou sa version rapide (Fast Fourier Transform (FFT). Afficher le spectre de fréquence de la gamme musicale crée en échelle linéaire, puis avec une échelle en décibels.





|  |
| --- |
| * TP3 : Traitement d’un signal ECG * Objectifs * Suppression du bruit autour du signal produit par un électrocardiographe. * Recherche de la fréquence cardiaque. * Suppression du bruit provoqué par les mouvements du corps  1. Sauvegarder le signal ECG sur votre répertoire de travail, puis charger-le dans Matlab à l’aide la commande load.      1. Ce signal a été échantillonné avec une fréquence de 500Hz. Tracer-le en fonction du temps, puis faire un zoom sur une période du signal.        1. Pour supprimer les bruits à très basse fréquence dues aux mouvements du corps, on utilisera un filtre idéal passe-bas. Pour ce faire, calculer tout d’abord la TFD du signal ECG, régler les fréquences inférieures à 50Hz à zéro, puis effectuer une TFDI pour restituer le signal filtré.        1. Tracer le nouveau signal ecg1, et noter les différences par rapport au signal d’origine.        * Suppression des interférences des lignes électriques 50Hz  1. Appliquer un filtre Notch idéal pour supprimer cette composante. Les filtres Notch sont utilisés pour rejeter une seule fréquence d'une bande de fréquence donnée.      1. Visualiser le signal ecg2 après filtrage.      * Amélioration du rapport signal sur bruit   Le signal ECG est également atteint par des parasites en provenance de l’activité musculaire extracardiaque du patient. La quantité de bruit est proportionnelle à la largeur de bande du signal ECG. Une bande passante élevée donnera plus de bruit dans les signaux, et limiter la bande passante peut enlever des détails importants du signal.   1. Chercher un compromis sur la fréquence de coupure, qui permettra de préserver la forme du signal ECG et réduire au maximum le bruit. Tester différents choix, puis tracer et commenter les résultats.      1. Visualiser une période du nouveau signal filtré ecg3 et identifier autant d'ondes que possible dans ce signal      * Identification de la fréquence cardiaque avec la fonction d’autocorrélation   La fréquence cardiaque peut être identifiée à partir de la fonction d'autocorrélation du signal ECG. Cela se fait en cherchant le premier maximum local après le maximum global (à tau = 0) de cette fonction   1. Ecrire un programme permettant de calculer l’autocorrélation du signal ECG, puis de chercher cette fréquence cardiaque de façon automatique. Utiliser ce programme sur le signal traité ecg3 ou ecg2 et sur le signal ECG non traité. NB : il faut limiter l’intervalle de recherche à la plage possible de la fréquence cardiaque |
| * TP4 : Filtrage Analogique * Objectifs * Appliquer un filtre réel pour supprimer les composantes indésirables d’un signal. * Améliorer la qualité de filtrage en augmentant l’ordre du filtre. * Filtrage et diagramme de Bode  1. Définir le signal x(t) sur t = [0 5] avec Te = 0,0001 s.      1. Tracer le signal x(t) et sa transformé de Fourrier. Qu'observez-vous ? (Essayez de tracer avec Te = 0,0005 s. Remarques ?)       La fonction H(f) (transmittance complexe) du filtre passe haut de premier ordre est donnée par : H(f) = (K.j.w/wc) / (1 + j. w/wc)  Avec K le gain du signal, w la pulsation et wc la pulsation de coupure. On se propose de tracer le diagramme de Bode de ce filtre et de l'appliquer au signal.   1. Tracer le module de la fonction H(f) avec K=1 et wc = 50 rad/s.        1. Tracer 20.log(|H(f)|) pour différentes pulsations de coupure wc, qu'observez-vous ? (Afficher avec semilogx)        1. Choisissez différentes fréquences de coupure et appliquez ce filtrage dans l'espace des fréquences. Qu'observez-vous ?        1. Choisissez wc qui vous semble optimal. Le filtre est-il bien choisi ? Pourquoi ? 2. Observez le signal y(t) obtenu, puis Comparer-le avec le signal que vous auriez souhaité obtenir.  * Dé-bruitage d'un signal sonore   Dans son petit studio du CROUS, un mauvais futur ingénieur a enregistré une musique en « .wav » avec un très vieux micro. Le résultat est peu concluant, un bruit strident s'est ajouté à sa musique. Heureusement son voisin, expert en traitement du signal est là pour le secourir :  « C'est un bruit très haute fréquence, il suffit de le supprimer. » dit-il sûr de lui   1. Proposer une méthode pour supprimer ce bruit sur le signal.   Utilisons un Filtrage analogique avec le diagramme de Bode   1. Mettez-la en oeuvre. Quelle influence à le paramètre K du filtre que vous avez utilisé ? |
|  |
| Quelles remarques pouvez-vous faire notamment sur la sonorité du signal final. Nous remarquons que nous entendons mieux le son et il y a moins de bruit et en augmentant l’ordre du filtre le bruit disparait. |
|  |