

9-Révision des principes de base

Exercices sur les fondamentaux de MSP

Dans ce chapitre, nous vous proposons quelques tâches à programmer qui vous permettront de tester votre compréhension des principes fondamentaux de MSP présentés jusqu'à présent dans les tutoriels. Quelques conseils sont inclus pour vous aider à démarrer. Essayez d'abord ces trois exercices progressifs par vous-même, dans un nouveau fichier de votre choix. Ensuite, consultez le patch d'exemple pour voir une solution possible et lisez la suite de ce chapitre pour obtenir une explication du patch de solution.

Exercice 1

Ecrivez un patch qui joue la note E au-dessus du C moyen pendant une seconde, dix fois de suite, avec un timbre semblable à celui d'une guitare électrique. Assurez-vous que tout ce que vous avez à faire est de cliquer une fois pour activer le son et une fois pour jouer les dix notes.

Voici quelques astuces:

- 1-La fréquence de E au-dessus du C moyen est de 329,627557 Hz.
- 2-Pour un 'timbre de type guitare électrique', vous pouvez utiliser le fichier AIFF *gtr512.aiff* utilisé dans le didacticiel de base 3. Vous devrez lire ce fichier dans un objet *buffer ~* et accéder au *buffer ~* avec un objet *cycle ~*.
- 3-Votre son aura également besoin d'une enveloppe d'amplitude caractéristique d'une guitare: une attaque très rapide, une décroissance rapide et un sustain assez constant (qui ne diminue que légèrement). Essayez d'utiliser une liste de segments de ligne (valeurs cibles et temps de transition) dans un objet *line ~* et d'utiliser la sortie de *line ~* pour mettre à l'échelle l'amplitude de *cycle ~*.
- 4-Pour jouer la note dix fois de suite, vous devez déclencher l'enveloppe d'amplitude à plusieurs reprises à vitesse constante. L'objet *metro* de Max convient parfaitement à cette tâche. Pour vous arrêter après dix notes, votre patch doit soit compter les notes soit attendre un laps de temps, puis désactiver le *metro*.

Exercice 2

Modifiez votre premier patch de façon à ce que, au cours des dix notes répétées, le son de la guitare électrique soit fondu avec un son sinusoïdal à la 12th supérieure. Utilisez un fondu enchaîné linéaire, l'amplitude d'un son passant de 1 à 0, tandis que l'autre son passe de 0 à 1. (Nous aborderons d'autres modes de fondu enchaîné dans un prochain chapitre.) Envoyez le son de la guitare sur le canal de sortie audio gauche. et le son sinusoïdal sur le canal droit.

Astuces:

- 1-Vous aurez besoin d'un second objet *cycle ~* pour produire le son à la 12th au dessus.
- 2-Pour obtenir la fréquence parfaite à la 12th au dessus de E, il suffit de multiplier simplement 329,627557 par 3. La fréquence qui est égale à la 12th parfaite au dessus de E est 987,7666 Hz. Utilisez le réglage que vous préférez.
- 3-En plus de l'enveloppe d'amplitude de chaque note, vous devrez modifier l'amplitude globale de chaque son au cours des dix secondes. Pour ce faire, vous pouvez utiliser un objet **~* supplémentaire pour redimensionner l'amplitude de chaque note, en modifiant lentement le facteur d'échelle de 1 à 0 pour un son et de 0 à 1 pour l'autre.

Exercice 3

Modifiez votre deuxième patch de façon à ce que, au cours des dix notes répétées, les deux notes en fondu enchaîné effectuent également un diminuendo général, en diminuant à **1/32** de leur amplitude initiale (c'est-à-dire de 30 dB).

Astuces:

- 1-Cela nécessitera encore un autre facteur de mise à l'échelle de l'amplitude (probablement un autre objet * ~) pour réduire progressivement l'amplitude d'un facteur de 0,03125.
- 2-Notez que si vous faites passer l'amplitude linéairement de 1 à 0,03125 en dix secondes, le diminuendo semblera commencer lentement et s'accélérer vers la fin. Cela est dû au fait que la distance linéaire entre 1 et 0,5 (une réduction de moitié) est beaucoup plus grande que la distance linéaire entre 0,0625 et 0,03125 (également une réduction de moitié). Les 6 premiers dB du diminuendo se produiront donc dans les premières 5,16 secondes, mais la dernière réduction de 6 dB se produira dans les dernières 0,32 secondes. Donc, si vous voulez que le diminuendo soit perçu comme linéaire, vous devrez vous ajuster en conséquence.

Solution à l'exercice 1

- Double-cliquez sur l'objet *patcher p exercise_1* pour voir une solution possible à cet exercice.

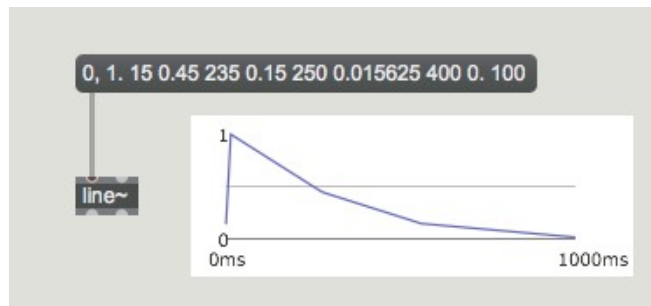
Pour créer un oscillateur avec une forme d'onde semblable à une guitare, vous devez lire le fichier audio **gtr512.aiff** (ou une forme d'onde similaire) dans un *buffer ~*, puis faire référence à ce *buffer ~* avec un *cycle ~*. (Voir le didacticiel de base 3.)

Notez qu'il y a une limite à la précision avec laquelle Max peut représenter des nombres décimaux. Lorsque vous enregistrez votre patch, Max peut modifier légèrement les valeurs de **float**. Dans ce cas, vous n'entendrez pas la différence.

Si vous voulez que le fichier audio soit lu dans le *buffer ~* immédiatement lorsque le patch est chargé, vous pouvez taper le nom du fichier comme deuxième argument dans l'objet *buffer ~* ou vous pouvez utiliser un objet *loadbang* pour déclencher un message de lecture au *buffer ~*. Dans notre solution, nous avons également choisi de fournir la fréquence à partir d'une boîte de *nombre* - ce qui vous permet de jouer d'autres hauteurs - plutôt que comme argument à *cycle ~*, donc nous envoyons également à *cycle ~* une valeur de fréquence initiale avec *loadbang*.

Maintenant que nous avons un oscillateur produisant le son désiré, nous devons fournir une enveloppe d'amplitude pour façonner une note.

Nous avons choisi l'enveloppe présentée ci-dessous, composée de segments de lignes droites. (Voir le *didacticiel de base 3*.)



Enveloppe d'amplitude dans le style "guitare"

Cette enveloppe d'amplitude est imposée à la sortie de *cycle~* avec une combinaison de *line~* et **~*. Un *metro* est utilisé pour déclencher l'enveloppe une fois par seconde, et le *metro* est désactivé après un délai de 10 secondes.

Solution à l'exercice 2

- Double-cliquez sur l'objet *patcher p exercise_2* pour voir une solution possible à cet exercice.

Pour le canal de sortie droit, nous voulons un son sinusoïdal à trois fois la fréquence (le troisième harmonique du son fondamental), avec la même enveloppe d'amplitude. Pour effectuer un fondu enchaîné entre les deux sons, l'amplitude du premier son doit passer de 1 à 0, tandis que l'amplitude du second son passe de 0 à 1. Ceci peut être obtenu en combinant *line~* et **~* pour chaque son. Nous avons utilisé une petite astuce pour être plus économes. Plutôt que d'utiliser un objet *line~* séparé pour atténuer le second son de 0 à 1, nous soustrayons simplement 1 à la sortie existante de *line~*, ce qui nous donne une rampe de 0 à -1. Perceptivement, cela aura le même effet.

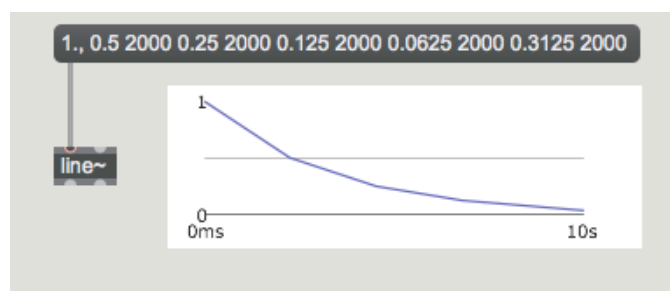
Ce fondu enchaîné est déclenché (via les objets *s* et *r*) par le même *button* qui déclenche le *metro*, de sorte que le fondu enchaîné commence en même temps que les dix notes individuelles.

Solution à l'exercice 3

- Double-cliquez sur l'objet *patcher p exercise_3* pour afficher une solution possible à cet exercice.

Enfin, nous devons utiliser une autre enveloppe d'amplitude pour créer un diminuendo global. Les deux sonorités vont vers un autre objet **~*, contrôlé par un autre *line~*. Comme indiqué précédemment, une diminution en ligne droite de l'amplitude ne donnera pas la perception d'un diminuendo constant de l'intensité.

Nous avons donc utilisé cinq segments de droite pour simuler une courbe qui diminue de moitié toutes les deux secondes. (L'objet *curve~* le fera automatiquement.)



Enveloppe d'amplitude globale diminuant de moitié toutes les deux secondes

Cette enveloppe d'amplitude globale est insérée dans le réseau de signaux pour réduire les deux sonorités d'un facteur de 0,03125 en 10 secondes.