

25-Synthèse soustractive

Dans ce chapitre, nous allons voir comment utiliser les filtres de manière créative avec un groupe de générateurs audio MSP créant différents types de *noise* ~. La génération de bruit est un composant essentiel de la synthèse soustractive, une méthodologie de conception sonore qui fonctionne en prenant des signaux complexes et en les sculptant avec des filtres, en soustrayant de l'énergie du signal original (comparez cela avec la synthèse additive, qui fonctionne de manière opposée). En cours de route, nous discuterons des moyens de façonner ce bruit à l'aide d'un objet qui crée et contrôle une *banque* de filtres en parallèle.

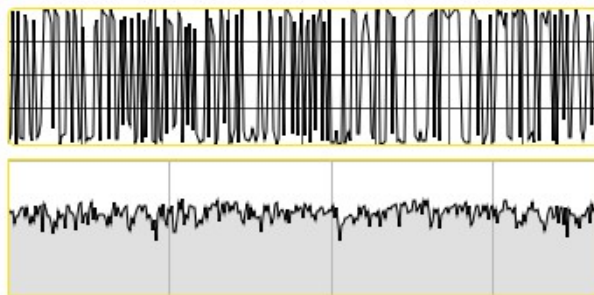
Noise

Jetez un coup d'oeil à notre patcheur de tutoriel. Il est composé de trois régions de patcheur. Si nous regardons la région étiquetée **1**, nous pouvons voir que nous avons trois nouveaux objets MSP connectés par des objets * ~ au *dac* ~.

- Démarrer l'audio dans le patcheur de tutoriel. Ajustez la boîte de *nombre* qui contrôle le volume de l'objet *noise* ~ et écoutez le résultat. Baissez-le et augmentez le volume de l'objet *pink* ~. Faites la même chose pour l'objet *rand* ~. Cliquez dans la boîte de *nombre* reliée à l'entrée de l'objet *rand* ~ (intitulée 'Fréquence'). Tapez **100** et appuyez sur Retour. Essayez **1000** et appuyez sur Retour. Essayez d'autres valeurs.

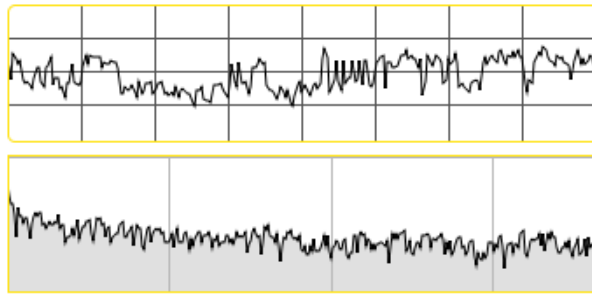
Les objets *noise* ~, *pink* ~ et *rand* ~ génèrent tous du bruit à un taux de signal. Le bruit, par essence, est un type de génération de nombres aléatoires; par conséquent, ces objets se comportent de manière similaire aux objets Max tels que *random* et *drunk*.

L'objet *noise* ~ génère du bruit blanc, ce qui signifie que toutes les fréquences possibles du spectre audio sont représentées de manière égale dans le temps. Le processus de génération de bruit blanc numériquement est assez simple: pour chaque échantillon, choisissez un nombre aléatoire compris entre **-1** et **1**:



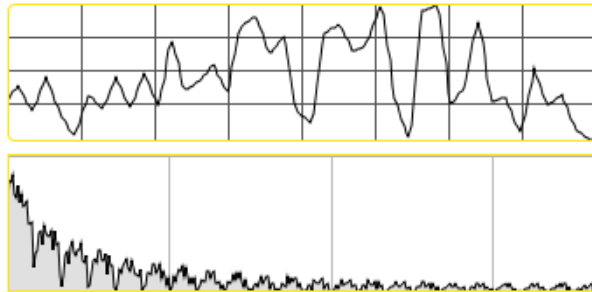
Un tracé de forme d'onde et spectrogramme du bruit blanc.

L'objet *pink* ~ génère un bruit rose, ce qui signifie que chaque octave du spectre audio a le même poids. On parle parfois de bruit $1 / f$, car la probabilité qu'une fréquence apparaisse est l'inverse de sa valeur, par ex. des fréquences de 100 Hz sont deux fois plus probables que celles de 200 Hz. La différence auditive entre les deux est assez évidente: le bruit blanc contient beaucoup plus de hautes fréquences et sonne plus «durs» que le bruit rose:



Bruit rose (1 / f): forme d'onde et spectrogramme.

L'objet *rand* ~ est un générateur de nombres aléatoires qui génère un signal et choisit une nouvelle valeur aléatoire pour ce signal à une fréquence variable. Il prend un argument (ou une valeur à son entrée) pour définir la fréquence de la sélection des nombres aléatoires. Une fréquence de **44100** rend l'objet impossible à distinguer du bruit blanc. Cela nous permet de créer un bruit à bande limitée dont la limite supérieure peut être spécifiée:



Un objet *rand* ~ sélectionnant des valeurs à 1000 Hz: forme d'onde et spectrogramme.

Filtrage du bruit

Le bruit ayant un tel contenu de fréquentiel à large bande, il peut être filtré et sculpté pour créer des timbres très précis. La technique de composition de la synthèse soustractive repose sur cet attribut de la génération de bruit; il est souvent plus facile (ou plus efficace) de commencer par le bruit et le filtrer, puis de tenter de créer le timbre désiré en ajoutant des oscillateurs.

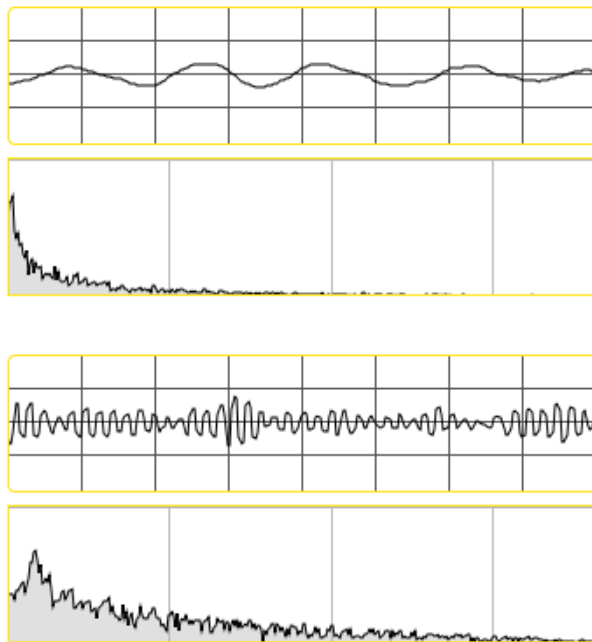
- Baissez les volumes dans la zone **1** de patcheur et examinez la zone de patcheur **2**. Augmentez le volume à l'aide la boîte de *nombre* située en bas de la chaîne de signal (contrôle de l'objet * ~ connecté au *dac* ~). Cliquez dans la boîte de *nombre* 'Frequency' connectée à l'objet *phasor* ~, tapez **0.1** et appuyez sur Retour. Tapez une fréquence plus élevée (par exemple 3,0) et appuyez sur Retour. Expérimentez avec différentes valeurs.

La zone **2** de patcheur contient un objet *noise* ~ qui envoie son signal dans un filtre *lores* ~. La fréquence du filtre passe-bas est modulée par un *phasor* ~, que nous avons mis à l'échelle pour une rampe entre **100** et **600** à la fréquence que nous spécifions. Par conséquent, la fréquence de coupure du filtre balaie à intervalles réguliers. Ceci est un exemple d'un LFO, ou **oscillateur basse fréquence**, utilisé pour moduler un paramètre d'un système de traitement audio. Comme vous pouvez l'entendre, l'objet *lores* ~ atténue les hautes fréquences émises par l'objet *noise* ~. De plus, la valeur de résonance de *lores* ~ fait que le filtre présente un pic juste en dessous de sa fréquence de coupure, ce qui donne un son particulièrement «aigu» au bruit filtré.

Banques de filtres

- Baissez le volume de la zone 2 dans le didacticiel et jetez un coup d'œil à la zone 3. Un par un, montez et descendez les curseurs de gain connectés à l'objet *dac* ~.

L'objet *fffb* ~ signifie **Fast, Fixed, Filter Bank**. Contrairement à l'objet *cascade* ~, qui implémente un certain nombre de filtres *biquad* ~ en série, l'objet *fffb* ~ dispose un certain nombre d'objets *réson* ~ en parallèle, ce qui signifie que les réglages d'un filtre n'affecteront aucun des autres. L'objet *fffb* ~ prend un certain nombre d'arguments qui définissent son comportement: le nombre de filtres, la fréquence de base du banc de filtres, le rapport entre les filtres et le Q des filtres. Tous les paramètres de l'objet, à l'exception du nombre de filtres, peuvent être modifiés avec les messages Max; le nombre est fixe car, comme nous pouvons le constater, chaque filtre est connecté à une sortie séparée. Cela nous permet de créer des banques de filtres, où nous pouvons «exploiter» chaque filtre passe-bande individuellement:



Sortie des filtres les plus bas et les plus élevés dans notre objet *fffb* ~: forme d'onde et spectrogramme.

- A l'aide de la souris, cliquez et faites glisser l'objet *dial* dans la zone de patcheur 3. Cela a pour effet de déplacer l'ensemble de la banque de filtres vers le haut ou vers le bas. Montez les différents curseurs de *gain* ~ pour entendre les résultats.

La valeur du dial est interprétée comme une hauteur MIDI, convertie en fréquence (via l'objet *mtof*) et utilisée pour formater le message **freqRatio** à l'objet *fffb* ~. Le message **freqRatio** prend deux arguments: la fréquence centrale du premier filtre (le plus bas) et le rapport entre celui-ci et les filtres suivants. La lettre **H**, lorsqu'elle est utilisée comme ratio, indique à l'objet *fffb* ~ de régler les filtres de la banque sur des multiples **harmoniques** de la fréquence de base. Ainsi, le message **freqRatio 100. H** configurerait nos dix filtres pour qu'ils soient centrés sur des incréments de 100 Hz.

- Cliquez dans les boîtes de *nombres* connectées à l'objet *pak*. Tapez **200.** dans la boîte de *nombres* de gauche et **1,5** dans la boîte de *nombres* de droite. Cliquez sur la boîte de *nombres* connectée à la boîte de message contenant le message **QAll \$ 1**. Entrez la valeur **100.** et appuyez sur Retour. Montez et descendez les différents curseurs de *gain* ~ pour entendre les résultats.

Nous pouvons facilement régler nos filtres dans un rapport de fréquence autre qu'une série harmonique. En réglant notre fréquence de base sur **200** et notre rapport sur **1,5** nous obtenons une banque de dix filtres réglés sur les fréquences **200, 300, 450, 675, 1012.5, 1518.75, 2278.125, 3417.1875, 5125.78125, 7688.671875 et 11533.0078125** Hz, respectivement. Les autres ratios les plus populaires incluent $1,4142 (2^{1/2})$ et $1,25992 (2^{1/3})$ pour un espacement d'une demi-octave et d'un tiers d'octave. Comme pour l'objet *reson* ~, nous avons le contrôle direct sur le Q de ces filtres. Un Q de **100** donne une largeur de bande de 1/100 de la fréquence, ce qui crée des filtres étroits et aigus.

- Cliquez sur la boîte de *message* à droite de la zone de patch **3** qui contient une série de listes. Écoutez les résultats en ajustant les curseurs de *gain* ~.

L'objet *fffb* ~ accepte de nombreux autres messages, ce qui nous permet de régler les filtres sans rapport du tout. L'envoi de listes au format **filter_# frequency Q** nous permet de régler chaque filtre de la banque individuellement. Dans notre exemple, nous avons réglé les dix filtres sur les fréquences d'un accord musical.

Vu-mètre

A côté de chaque curseur de *gain* ~, dans la zone du patcheur **3**, se trouve un objet d'interface utilisateur qui enregistre l'amplitude du signal qui lui est connecté. Ces objets *meter* ~ nous permettent de voir le niveau de chaque filtre dans le *pré-fader* de *fffb* ~, c'est-à-dire avant de l'écouter.

- Baissez tous les curseurs de *gain* ~ dans la zone du patcheur **3**. Cliquez dans la boîte de *nombre* qui déclenche le message **Qall**. Tapez **0.5** et appuyez sur Retour. Tapez **10**. et appuyez sur Retour. Écoutez les résultats et remarquez l'effet du Q sur le gain de chaque filtre, puis observez comment les objets *meter* ~ réagissent.

Comme l'objet *fffb* ~ fonctionne en parallèle, le gain de sortie de tous les filtres de la banque sera généralement *supérieur* au gain du signal d'entrée. Selon les valeurs de Q et des fréquences utilisées, le volume potentiel de sortie du *fffb* ~ peut être assez élevé. L'objet *meter* ~ vous permet d'observer visuellement vos volumes dans la fenêtre du patcheur avant d'écouter (et de vous faire mal aux oreilles).

Résumé

MSP dispose de trois objets générateurs de bruit simples à utiliser, qui génèrent du bruit blanc (*noise* ~), du bruit rose (*pink* ~) et des signaux aléatoires limités en bande (*rand* ~). Ces objets sont des candidats idéaux pour le filtrage. L'objet *fffb* ~ implémente un banc de filtres fixes, de filtres passe-bande parallèles qui peuvent être contrôlés via des ratios d'une fréquence de base ou individuellement. L'objet *meter* ~ vous permet de visualiser l'amplitude de n'importe quelle partie du signal MSP, et est incroyablement utile pour mesurer et déboguer vos patchers audio.