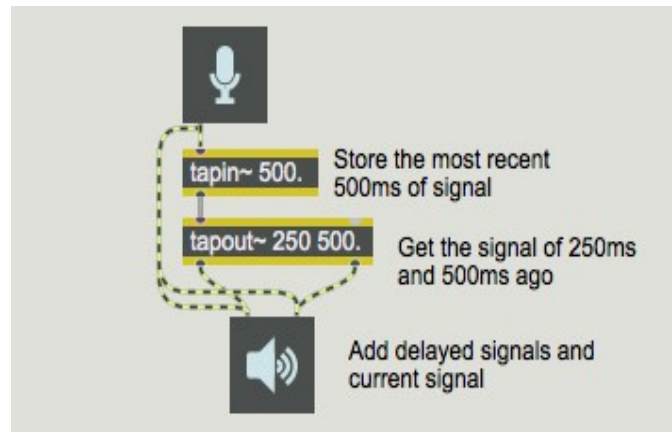


45-Lignes de retard avec feedback

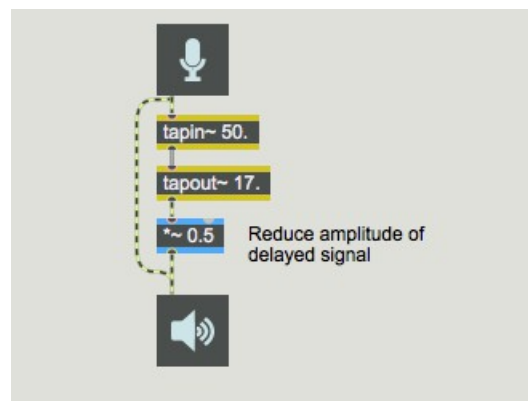
Le delay émule la réflexion

Vous pouvez retarder un signal pendant une durée spécifique en utilisant les objets *tapin~* et *tapout~*. L'objet *tapin~* est une mémoire continuellement mise à jour qui stocke le signal le plus récent qu'il a reçu et *tapout~* accède à cette mémoire à un ou plusieurs points spécifiques passés.



Retarder un signal avec *tapin~* et *tapout~*

Combiner un son avec une version retardée de lui-même est un moyen simple d'émuler une onde sonore qui se réfléchit sur un mur avant d'atteindre nos oreilles; nous entendons le son direct suivi de près par le son réfléchi. Dans le monde réel, une partie de l'énergie sonore est en fait absorbée par le mur réfléchissant, et nous pouvons émuler ce fait en réduisant l'amplitude du son retardé, comme le montre l'exemple suivant.



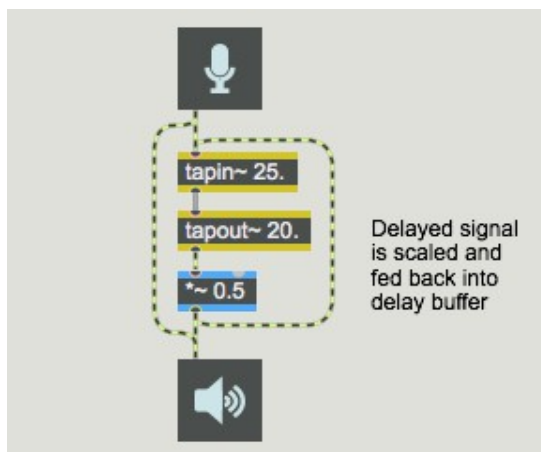
Mise à l'échelle de l'amplitude d'un signal retardé, pour émuler l'absorption

Détail technique: Les différents matériaux absorbent le son à des degrés divers, et la plupart des matériaux absorbent le son de manière qui dépend de la fréquence. En général, les hautes fréquences sont plus absorbées que les basses fréquences. Ce fait est ignoré ici.

Retarder le signal retardé

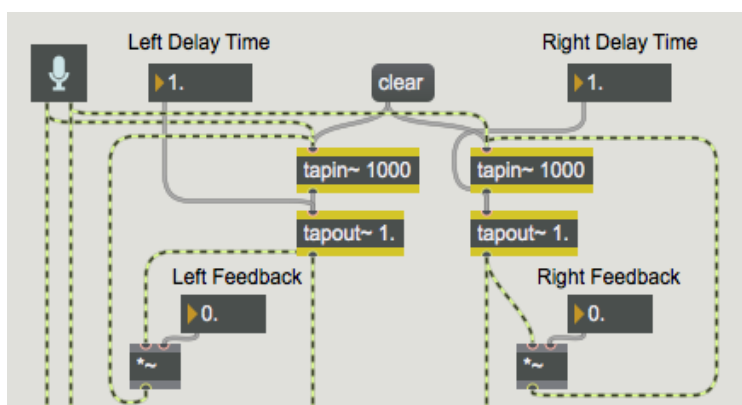
De plus, dans le monde réel, il y a généralement plus d'une surface qui reflète le son. Dans une pièce, par exemple, le son est réfléchi par les murs, le plafond, le sol et les objets de la pièce de multiples façons, et ces réflexions sont à leur tour réfléchies sur d'autres surfaces. Une façon simple de modéliser cette "réflexion des réflexions" consiste à réintroduire le signal retardé dans la ligne à

retard (après en avoir "absorbé" une partie).



Retard avec feedback

Une simple ligne à retard avec feedback comme celle ci-dessus est trop simpliste pour ressembler à une situation acoustique réelle, mais elle peut générer un certain nombre d'effets intéressants. Le délai stéréo avec feedback est implémenté dans le patch d'exemple de ce tutoriel. Chaque canal d'entrée audio est retardé, mis à l'échelle et réinjecté dans la ligne à retard.



Délai stéréo avec des délais individuels et des valeurs de feedback

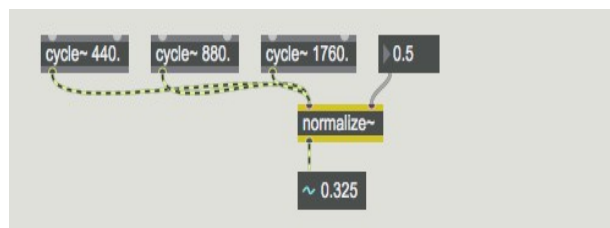
- Définissez la boîte de *nombre* intitulée «Output Level» sur **0,25** et déplacez le *curseur* vers sa position centrale de sorte que les boîtes de *nombres* «Direct Level» et «Delay Level» indiquent **0,5**. Activer l'audio et envoyer du son dans l'entrée audio de l'ordinateur. Expérimentez avec différents temps de retard et quantités de feedback. Par exemple, vous pouvez utiliser les paramètres indiqués ci-dessus pour obtenir un effet de flou. Augmentez les quantités de feedback pour obtenir une plus grande résonance au taux de rétroaction (1 000 divisé par le temps de retard). Augmentez les temps de retard pour obtenir des échos discrets. Vous pouvez modifier le mélange Dry/Wet avec l'aide du *curseur*.

Notez que chaque fois que vous réinjectez un signal audio dans un système, vous risquez de le surcharger. C'est pourquoi il est important de mettre le signal à l'échelle par un facteur inférieur à 1,0 (avec les objets *~ et les objets de la boîte de *nombre* 'Feedback') avant de le réinjecter dans la ligne de retard. Sinon, le son retardé continuera indéfiniment et augmentera même s'il est ajouté au nouvel audio entrant.

Contrôle de l'amplitude: *normalize ~*

Puisque ce patch contient des paramètres de niveau variables (notamment les niveaux de réinjection) et que nous ne savons pas quel son sera introduit dans le patch, nous ne pouvons pas vraiment prédire comment nous devons mettre à l'échelle le niveau de sortie final. Si nous avons utilisé un objet ** ~* juste avant *ezdac ~* pour mettre à l'échelle l'amplitude de sortie, nous pourrions définir le niveau de sortie, mais si nous augmentons ensuite les niveaux de réinjection, l'amplitude de sortie pourrait devenir excessive. L'objet *normalize~* permet de gérer de telles situations imprévisibles.

L'objet *normalize ~* vous permet de spécifier l'amplitude de crête (maximale) que vous voulez envoyer à sa sortie. Il examine l'amplitude de crête de son entrée et calcule le facteur selon lequel il doit mettre à l'échelle le signal afin de maintenir l'amplitude de crête au maximum spécifié. Ainsi, avec *normalize~*, l'amplitude de crête de la sortie ne dépassera jamais le maximum spécifié.



normalize ~ envoie l'entrée actuelle * sortie crête / entrée crête

Un inconvénient potentiel de *normalize ~* est qu'une seule crête dans le signal d'entrée peut amener *normalize ~* à réduire l'intégralité du signal, même si le reste du signal d'entrée est très faible. Vous pouvez donner à *normalize ~* une nouvelle valeur de crête d'entrée à utiliser en envoyant un nombre ou un message **reset** dans l'entrée gauche.

- Désactivez le son et fermez la fenêtre du Patcher avant de passer au chapitre suivant.

Résumé

Une façon de créer plusieurs versions retardées d'un signal consiste à réinjecter la sortie de *tapout ~* dans l'entrée de *tapin ~*, en plus de l'envoyer au DAC. Comme le signal retardé réinjecté sera ajouté au signal entrant entrant actuel à l'entrée de *tapin ~*, il est judicieux de réduire la sortie de *tapout ~* avant de le réinjecter à *tapin ~*.

Dans un patch impliquant l'addition de signaux d'amplitudes variables, il est souvent difficile de prédire l'amplitude du signal additionné qui sera envoyé au DAC. Une façon de contrôler l'amplitude d'un signal est de se servir de *normalize ~*, qui utilise l'amplitude de crête d'un signal entrant pour calculer de combien il doit réduire l'amplitude avant d'envoyer le signal.