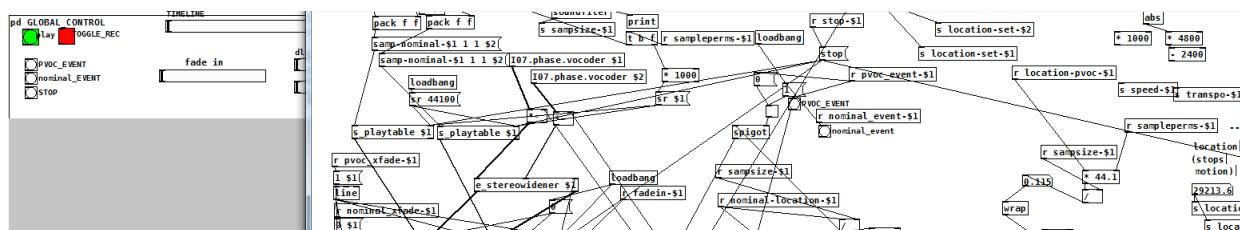


# PVOC\_SNAPSHOT\_TIMELINE : POST MORTEM

## (Prototype Pure Data proposant des contrôles interactifs du modèle de lecture Phase Vocoder conçu par Miller Puckette)



### Introduction :

En signal audio numérique il existe différentes méthodes pour la lecture d'un fichier audio. On définit ici la lecture du fichier par deux paramètres. La valeur de l'adresse en entrée d'un tableau de mémoire. La valeur de la combinaison correspondante à cette adresse. En lecture nominal d'un fichier audio d'une durée de une seconde (formaté par un taux d'échantillonnage de 48kHz) on admet que le lecteur parcourt de la première adresse jusqu'à la dernière adresse les 48000 adresses existant et ceux en une seconde. Simultanément en sortie le tableau, affiche la valeur correspondante. Si l'on veut lire le fichier plus lentement il suffit alors auquel cas de ne plus parcourir les 48000 adresses non plus en une seconde mais en deux secondes par exemple, auquel cas la lecture du fichier complet est deux fois plus long et la tonalité perçue descend d'une octave. En effet en une seconde on lit 24000 samples les périodes des signaux audios ressentis sont deux fois plus long donc deux fois plus grave que lors d'une lecture nominale. D'autres méthodes permettent de rallonger ou raccourcir la durée de la lecture sans pour autant changer la tonalité perçue du signal. Le phase vocoder conçu par Miller Puckette en est une. Elle propose, pour rallonger la lecture du fichier, une méthode de répétition de tronçon linéaire d'adresse plus ou moins long. Chaque tronçon du signal a son premier échantillon en cohérence de phase avec le dernier sample du tronçon précédant. Chaque tronçon a son niveau convolué par un fenêtrage (sorte de fade in fade out constant pour chaque tronçon) plus ou moins long. En résultat cela permet de rallonger la lecture d'un son voir de le stopper ou de accélérer. La tonalité reste constante car la lecture de chaque tronçon est nominal, Mais on peut très bien transposé la vitesse de lecture de chaque tronçon. cependant en fonction de la configuration et de la matière sonore le résultat ressenti subit des modifications et la perception figurée du son dans son espace acoustique et stéréo change. C'est ce qui m'a intéressé et c'est ce que j'ai voulu développer avec ce patch. Un outil qui permet d'ouvrir les paramètres du patch de Miller Puckette (disponible dans pure data 0.47.1 : 107.Phase.Vocoder.pd) pour réaliser différents effets.

### Transition d'état :

La première chose qui m'a semblé être intéressante de faire avec cette méthode de lecture était de réaliser une transition entre deux états de mix. Entre la version nominale de l'audio et la version Phase.Vocoder. J'aurais pu tout simplement commander par exemple la lecture à 100%(speed) du fichier puis le faire évoluer. Cependant la lecture audio à 100% comporte déjà des transformations de perceptions comparées à une réelle lecture nominale linéaire. J'ai donc conçu un moteur de lecture audio basé sur deux lecteurs audio. Un linéaire, un Phase.vocoder. Une fonction de crossfade permet de basculer l'un avec l'autre. Les boutons PVOC\_EVENT et nominal\_EVENT permettent d'opérer la bascule avec automatisation dans le temps. On peut donc réaliser une bascule plus ou moins longue. Dans ce système nos deux lecteurs audios sont autonomes et même à 100% le lecteur PVOC présente une déviance très légère par rapport au lecteur nominal. Lorsque l'on bascule de l'état PVOC à l'état nominal, la dernière adresse de lecture du lecteur PVOC est imposée au lecteur nominal. Ainsi lors de la bascule on perçoit pendant le crossfade que les deux signaux sont cohérents il n'existe pas de délai entre les deux signaux (en fonction de la configuration de l'automatisation).

### **Contrôle de la *timeline* :**

Lorsque l'état PVOC est actif, l'utilisateur peut avoir la main sur la position de la lecture du fichier (Slider : TIMELINE) Cette fonction rétro-active (le lecteur envoie l'adresse à la *timeline*, la *timeline* renvoie l'adresse au lecteur) permet à l'utilisateur de revenir en arrière ou plus loin dans le fichier. Le lecteur balaie les adresses avec une vitesse de lecture dérivé de l'évolution de l'adresse. (plus on opère vite sur la *timeline* plus la vitesse de lecture est grande) lorsque l'on arrête le contrôle de la *timeline*, la vitesse de lecture revient à sa position initiale.

### **Stéréo :**

Sur les fichiers stéréos le lecteur PVOC a un impact que je n'avais pas anticipé. Il semble qu'une rotation de phase se trouve être exagéré (en fonction de l'incidence de phase pré-existante entre le canaux gauche et droite du fichier), mais aussi la composante commune d'un fichier stéréo ne subit jamais le même traitement entre le lecteur PVOC gauche et droite ce qui crée incohérence entre les deux canaux. Il y donc une instabilité quand à la perception du signal qui se trouve généralement beaucoup plus ouvert que dans la perception stéréo nominal. J'ai mis en place une fonction qui permet de réaliser plus ou moins la sommation des deux canaux entre-eux, pour pouvoir « fermer » ou « ouvrir » la stéréo.

### **Exemples :**

à ce stade du développement, j'ai fini de faire évoluer le patch et je continue de tester différentes matières sonores dans différentes situations pour connaître la versatilité des textures créées. Ces exemples sont extraits directement en sortie du patch sans aucune modification montage ou traitement sinon un encodage .mp3.

**1ère exemple :** Crossfade et contrôle de la *timeline* sur un fichier 92kHz.

Ambiance place stalinegrad Paris réalisé par Dorian disponible sur [sounddesigners.org](http://sounddesigners.org)

#### Exemple 1

Il est possible de modifier la vitesse de lecture du lecteur nominal en fonction du taux d'échantillonnage, ici le lecteur nominal est configuré à 92kHz. Cependant le lecteur PVOC est toujours par défaut configuré à 44,1 kHz (je souhaite changer cela dans une prochaine MAJ). Ainsi lorsque l'on passe à l'état PVOC le son perçu est à l'octave en dessous qu'en lecture nominal (avec la transposition à -149). Le lecteur PVOC est à 100% de sa vitesse. Dans cet exemple on peut entendre un effet de transition où le son perçu devient plus bas sans dissonance notable (car l'intervalle est une octave). De plus l'enregistrement initial en 92 kHz a pour résultat de préserver la qualité du signal lorsque celui-ci est transposé une octave en dessous. À noter que le circuit est bouclé. Lorsque le lecteur arrive à la fin du fichier il revient au premier sample. Cette boucle s'applique aussi bien sur le lecteur nominal que sur le lecteur PVOC.

**2ème exemple :** Scène de dialogue. Transition d'état et contrôle de la *timeline*.

Film « The Sky Queen », réalisé par Simon Maignan. [Youtube](https://www.youtube.com/watch?v=...)

#### Exemple 2

Dans cet exemple je souhaitais mettre en valeur la possibilité d'interactivité des paramètres de ce patch avec une scène sonore. Deux personnages se parlent, il est possible de stopper le temps, revenir en arrière dans la scène ou aller plus loin pour capter toutes les informations. En résultat, l'effet est notable car elle ouvre d'un coup la stéréo et à énormément d'emphase sur l'acoustique présent dans l'audio initial. Pour autant ce qui se passe dans la scène est toujours intelligible mais pas compréhensible. On identifie les deux voix des personnages mais pas ce qu'ils se disent. En préservant la vitesse de lecture du PVOC à 100% on récupère cette intelligibilité, mais on perd l'impression que le temps s'arrête. On peut noter la capacité des deux moteurs de lecture à toujours se synchroniser lorsque il est nécessaire de basculer de l'état à un autre, c'est ce qu'il y a de plus important pour que l'effet de transition soit crédible.

### 3ème Exemple : Musique.

Arvo Part – Tintinnabuli The Tallis Scholar

#### Exemple 3

Lorsque j'ai découvert ce patch et ce qui m'a décidé à développer ce patch. C'était la ressemblance des textures créées par le PVOC est celle que l'on peut entendre dans l'œuvre musicale de Johann Johanson sur le film « Arrival » de Denis Villeneuve. L'impact notable est la possibilité en terme de montage sonore de modifier la perception de l'enveloppe du son et naturellement sa temporalité. Dans « Arrival » nous pouvons entendre des scènes sonores composées de notes de piano qui durent très longtemps avec des variations de niveaux, et de superpositions de plans dans l'acoustique. Le PVOC n'est donc pas une finalité dans le processus de création, mais un moyen de créer de la matière qu'il faut par la suite travailler. C'est le même constat que je fais avec mon patch. Cependant, l'essai que j'ai réalisé sur l'œuvre d'Arvo Part pour chœur, démontre qu'il est possible d'opérer une transition subtile entre les deux états, mais aussi d'exagérer la tenue des notes la perception de l'acoustique mais aussi la perception des voyelles et des sibilances. Cette méthode de lecture s'intègre bien pour un contenu interactif, cela peut être même à la base d'une mécanique d'un jeu vidéo. Ainsi liant la réalisation artistique de la musique à l'approche ludique du jeu.

#### **Fast Setup :**

1. Installer PURE DATA 0.47.1 <https://puredata.info/downloads/pure-data>
2. Télécharger la dernière release du patch PVOC\_SNAPSHOT\_TIMELINE [https://github.com/Baptisteq/Prototype\\_pvoc\\_snapshot\\_timeline/releases](https://github.com/Baptisteq/Prototype_pvoc_snapshot_timeline/releases)
3. ouvrir pure data 0.47.1, charger le patch PVOC\_SNAPSHOT\_TIMELINE.pd
4. activer le DSP
5. charger un fichier audio PCM16.wav ou PCM24.wav
6. configurer le sample rate
7. configurer la durée du crossfade et son délai, configurer la vitesse cible et le temps d'évolution vers la vitesse cible. Idem pour la transposition.
8. Play.
9. Apuyer sur PVOC\_event ou nominal\_event pour basculer entre les deux états. Le contrôle de la *timeline* ne s'opère que dans le mode PVOC.

Baptiste Quéméner Septembre 2017

[quemeneb@gmail.com](mailto:quemeneb@gmail.com)

<https://github.com/Baptisteq>