



Musique électronique “ live ” et “ recasting ” : Trois cas d’étude

Antonio Sousa Dias

► To cite this version:

Antonio Sousa Dias. Musique électronique “ live ” et “ recasting ” : Trois cas d’étude. Revue Francophone d’Informatique et Musique, 2011, 1. <hal-02299050>

HAL Id: hal-02299050

<https://hal.science/hal-02299050v1>

Submitted on 27 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Numéros / n° 1 - automne 2011

« Musique électronique « live » et « recasting » : Trois cas d'étude »

António de Sousa Dias

Résumé

Ce texte porte une discussion autour d'un de nos axes de recherche : la récupération en vue de l'électronique « live ». Nous présentons trois cas d'étude : Jorge Peixinho, *Harmónicos* (1967), Jorge Peixinho, *Sax-Blue* (1982) et Cândido Lima, *Ŋcāncōa* (1995). Nous discuterons le besoin de la redistribution (Recasting) et la raison du choix de cette désignation. Puis, à propos des problèmes apportés dans chaque cas, nous présentons les solutions trouvées.

Introduction

Ce texte présente certains résultats portant sur un de nos axes de recherche : la récupération d'œuvres faisant appel à l'électronique « live ». Dans le cas des musiques mixtes, en particulier celles qui font appel au traitement du son en direct ? l'électronique « live » ?, l'obsolescence des équipements ou, du moins la difficulté dans l'accès à des appareils soit analogiques, soit des premières générations numériques, pose des problèmes pour la récréation des œuvres. De plus, même si le transfert ou l'actualisation sont possibles, restent des questions musicologiques à résoudre, notamment la pertinence des choix en face des différences dans les résultats pour infimes qu'ils soient.

Ainsi, partagés entre un désir d'accéder aux œuvres que nous aimerions bien récupérer pour promouvoir leur diffusion et de respecter les propositions de départ des compositeurs, nous avons été menés à prendre un certain nombre d'options. Pour présenter et discuter ici certains problèmes et options adoptés, nous avons choisi trois cas d'étude : du compositeur Jorge Peixinho (1940-1995), *Harmónicos* (1967) et *Sax-Blue* (1982) et de Cândido Lima (1939-), l'œuvre *Ŋcāncōa* (1995). Peixinho et Lima sont des compositeurs importants dans le cadre de la musique contemporaine portugaise, leurs activités se partageant entre la composition (ils ont été toujours des compositeurs très actifs), la diffusion (ils ont créé des groupes pour la diffusion d'œuvres) et l'enseignement.

Concernant les œuvres de Peixinho, ce texte reprend une partie substantielle d'un texte déjà publié en anglais (Sousa Dias, 2009), où nous exposons l'importance du travail de restauration (cf., par exemple Cannazza & Vidolin 2001 ou Landy 2006) et nous discutons les raisons qui nous ont menés à notre position dans ce domaine. En résumé, on peut dire que « nous devons être conscients que le transfert vers une autre technologie doit être soigneusement et clairement énoncé », toujours « en supposant un compromis : les solutions trouvées doivent respecter les instructions de la partition et elles ne doivent pas ajouter des possibilités non supposées par son auteur, les exceptions doivent être clairement énoncées et justifiées ».

Finalement, une de nos motivations pour ce travail reste dans le désir de contribuer à la promotion et la diffusion des œuvres appelant à l'usage des technologies et à la constitution d'un corpus d'œuvres accessible à l'étude par les nouvelles générations. Finalement, la motivation principale est que ce sont de très belles œuvres et que c'est un privilège de pouvoir les donner à entendre.

1. Le « recasting »

Si, d'une part, la réalisation des restaurations semble évidente, d'autre part, les problèmes y parvenant ne le sont pas. Et c'est dans le cadre des musiques mixtes que le travail de récupération devient plus sensible. De fait, il ne s'agit plus que de reconstituer des œuvres fixes, donc d'attendre un résultat pertinent par rapport à l'écoute. Il s'agit également d'assurer que les variables mises en jeu auront le même comportement au travers des différentes interprétations et que ce même comportement reste fidèle à l'œuvre et aux désirs du compositeur.

Le choix du mot « recasting » (redistribution) pour désigner ce travail de récupération nous rappelle que, souvent, il ne s'agit pas de refaire la composante technologique d'une œuvre avec la même technologie. En effet, il s'agit de redistribuer des rôles d'une œuvre à d'autres « acteurs ». C'est la raison pour laquelle nous préférons ce terme proposé par Chadabe (Chadabe 2001) à d'autres, par exemple, le terme « reforcer ». À notre avis, le mot *recast* souligne le caractère de compromis sous-entendu par le fait qu'il y aura toujours lieu à des adaptations. Nous avons affronté ce problème lors de nos transcriptions des fichiers de Jean-Claude Risset de *MusicV* vers *Csound* (Sousa Dias, 2007) où nous avons pris parti pour garder la ressemblance de style de programmation entre les deux langages, en ce qui concerne la structure de programmation et les noms de variables.

D'ailleurs, nous allons montrer que, parfois, un *recasting* avec la participation du compositeur peut aboutir à une solution qui n'a plus de correspondance avec ce qui est préconisé dans la partition, mais qui correspond quand même à la pratique musicienne.

Chowning le signale à propos de la reconstruction de *Stria* : pendant la reconstruction de l'œuvre, les artefacts dus à des problèmes technologiques (faibles taux d'échantillonnage, quantification du signal, etc.) ont été supprimés à la demande du compositeur (Chowning 2007).

2. Trois cas d'étude

Les œuvres choisies ouvrent à des problèmes distincts et ont requis des solutions différentes. Inscrites dans la catégorie « Musique électroacoustique instrumentale ? Transformations et interactions en direct » (AAVV 2000), d'une façon générale, tous les cas présentés ici exploitent le retard temporel. En cherchant dans un passé récent ce procédé, on peut l'inclure dans une des catégories proposées par Ernst (Ernst 1977 : 153) concernant l'application des procédés électromécaniques dans la performance : l'emploi des magnétophones dans la performance, inaugurée par Kagel avec *Transición II* (1958-1959). Cette catégorie porte, surtout sur la répétition à différents niveaux et échelles articulant trois procédures : les boucles, les lectures décalées et les réinjections.

Depuis les dispositifs où l'on fait passer la bande magnétique d'un magnétophone à l'autre pour obtenir des retards assez longs jusqu'aux chambres d'écho à bande magnétique où une boucle est enregistrée, puis lue successivement devant plusieurs têtes de lecture, la contrepartie numérique d'une ligne à retard consiste dans une structure de données appelée *queue circulaire* contenant des échantillons dont les valeurs sont récupérées au travers d'un pointeur de lecture, appelé de bascule. Ainsi, nous proposons différentes configurations et applications des lignes à retard numériques à différents résultats musicaux. Pour des précisions sur ce sujet, cf., par exemple (Gendre 1981), (Dalstein 1991) ou (Roads 1998).

Les versions actuelles des *recastings* présentés sont programmées dans le logiciel/environnement *Max*, version 5 (Zicarelli et al). *Max* semble assurer une certaine stabilité et durabilité au regard de son développement et depuis la version 5, le nouveau format de fichier des patchs permet une lecture du code plus claire et directe. Le centre des programmes, la ligne à retard, fait appel aux objets « *tapin~* » et « *tapout~* », dans trois configurations de base : lecture simple (bascule) pour *Harmónicos*, lecture multiple (multibascule) avec ou sans réinjection dans le cas de *Sax-Blue* et lectures simples en couche pour *Ñcãncôa*. Ces *recastings* incluent certains algorithmes appartenant à la bibliothèque *jimmies* (Settel, 1998). Nous avons choisi de les inclure comme sous-patchs et non comme des abstractions en vue d'obtenir un code autonome sans le risque de perte des liaisons externes. Toutefois, des références explicites à leur source d'origine et à son auteur sont indiquées dans le code.

Finalement, nous présentons les œuvres choisies dans un ordre différent de leur chronologie : d'abord *Ñcãncôa* (1995) de Candido Lima, puis *Harmónicos* (1967) de Jorge Peixinho et finalement *Sax-Blue*

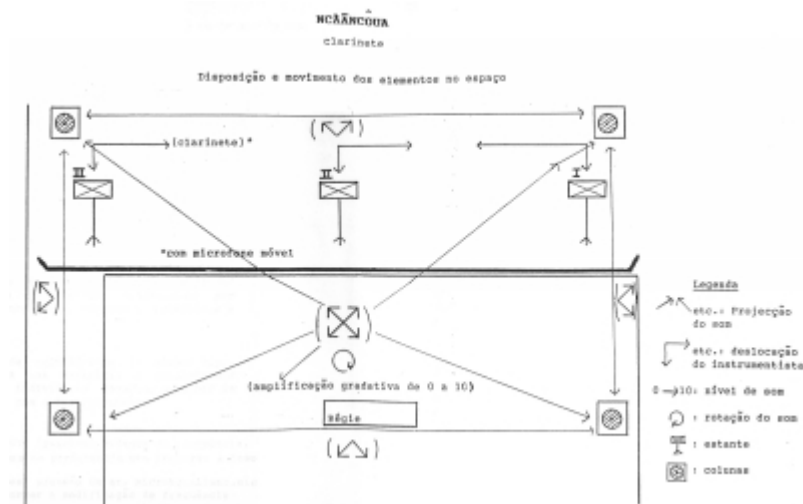
(1982) du même compositeur.

2. 1. Cândido Lima, *Ñcããncôa* (1995)

Cândido Lima (né en 1939 à Viana do Castelo, Portugal), est un compositeur pionnier dans l'informatique musicale au Portugal. Il a étudié la composition au conservatoire de Lisbonne et de Porto et, plus tard, avec Stockhausen, Ligeti, Boulez et Henri Pousseur, entre autres. Il a obtenu un doctorat en esthétique à Paris avec une thèse sous la direction de Xenakis. Il a également étudié des systèmes d'informatique musicale à Paris et à l'université de Vincennes et suivi des programmes de formation au CEMAMu et à l'Ircam. Il a été professeur titulaire de composition à l'École supérieure de Musique de Porto. Il dirige l'ensemble Musica Nova, qu'il a fondé en 1975. Lima est le compositeur portugais dont le catalogue contient un très grand nombre d'œuvres mixtes et sur support.

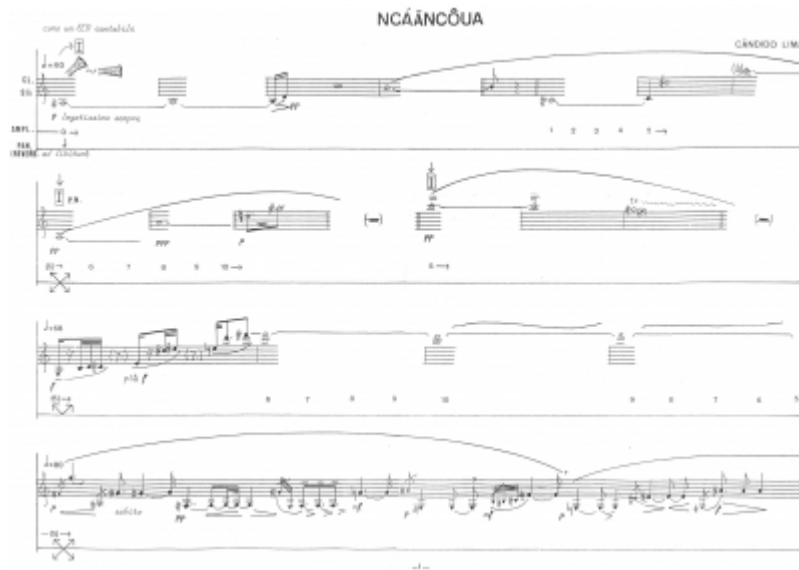
Apparemment, *Ñcããncôa* (1995) ne pose pas de problèmes. L'auteur étant vivant, nous avons toujours eu recours à ses indications. Et pourtant, la partition a toujours montré des problèmes vis-à-vis des usages dans la performance. En effet, la partition fait des références explicites à un dispositif comprenant une chambre de réverbération et la possibilité de faire circuler le son dans l'espace de la salle de concert. De plus, l'interprète doit changer de position dans la scène à certains moments indiqués dans la partition (voir figures 1 et 2).

Figure 1



Cândido Lima, *Ñcããncôa* : Partition ? Schème du dispositif et mise en scène

Figure 2



Cândido Lima, *Ncáãncôua* : Partition ? première page

Le problème est que, d'une part, la version la plus réussie, selon le compositeur, consiste à jouer la clarinette, sans traitement, accompagnée d'une bande contenant l'enregistrement de deux interprétations de l'œuvre superposées (une clarinette par canal). Les déphasages successifs des différentes interprétations apportent l'ambiance désirée pour l'œuvre, une sorte de prolongation ? parfois de rétrécissement ? de la clarinette jouée en direct. D'ailleurs, la superposition des trois clarinettes joue ainsi un rôle entre un écho non-automatique et une hétérophonie, en conservant un caractère toujours vivant.

Nous avons proposé à Lima de refaire la partition en essayant de trouver une solution garantissant le respect pour ses propos et, en même temps, permettant de reprendre le côté vivant de l'œuvre. C'est-à-dire qu'il s'agissait de trouver une solution dont le résultat découlerait totalement de la performance. Nuno Pinto, le plus récent interprète de cette œuvre, avait, lui aussi, mentionné le désir de se « débarrasser » de l'enregistrement car, à son avis il ne représentait plus un défi à cause de son immutabilité.

Ainsi, nous avons commencé à envisager des solutions possibles. D'une part, nous avons décidé de faire le travail en Max, car ceci permettait une large diffusion du patch accompagnant la partition. Il restait le choix de la technique de traitement du signal et son implantation.

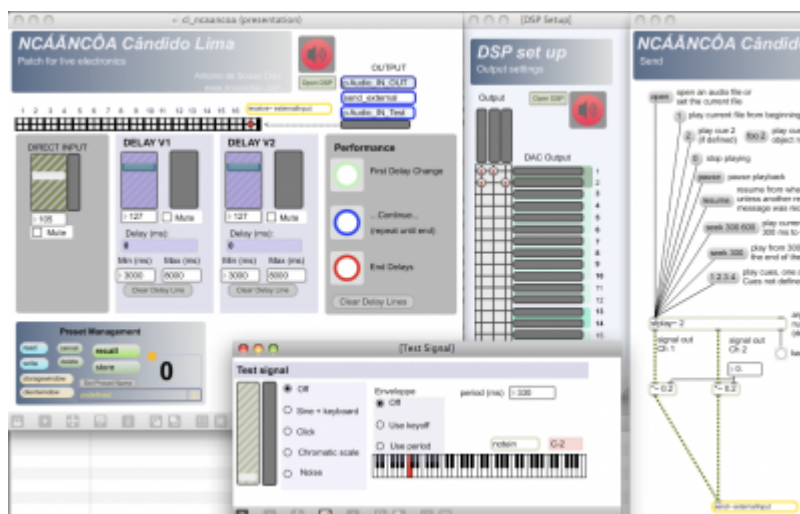
La première idée a été de réaliser une simulation d'interprétation de l'œuvre par trois clarinettes dont deux d'entre elles résultent de la transformation de la clarinette jouée en direct. Nous avons pensé à un dispositif pouvant faire des changements de temps (de la compression et de l'expansion temporelles) sans changer la hauteur. Ce dispositif était constitué par deux lignes de retard variables, chacune accouplée à un *harmoniseur*. Ceci pourrait rééquilibrer la justesse de hauteur en raison du changement de vitesse de lecture.

Ce système nous est apparu assez instable et les artefacts produits gênaient la qualité sonore. Ainsi, nous avons décidé de prendre une autre approche. D'une part, on voulait obtenir un son qui ne sonnait pas comme une clarinette enregistrée et transformée. D'autre part, il fallait une variabilité dans les changements de tempo pour créer l'idée de différentes interprétations en suggérant un certain degré d'autonomie.

La solution proposée consiste dans un système double de gestion de lignes à retard dont un contrôleur (une pédale, barre d'espaces, ou autre) déclenche des actions. Ces actions sont de trois types : (1) initialisation (dans la figure : « First delay change »), (2) continuation et (3) terminaison (« End Delays ») et elles ne peuvent être déclenchées que pendant des soupirs ou pauses dans la partition. Le premier et le dernier type ne sont déclenchés qu'une fois, le deuxième est déclenché un nombre indéterminé de fois pendant le déroulement de l'œuvre.

Un patch général permet les réglages du système selon les conditions de la performance (carte son, réglages de niveaux, etc.)

Figure 3

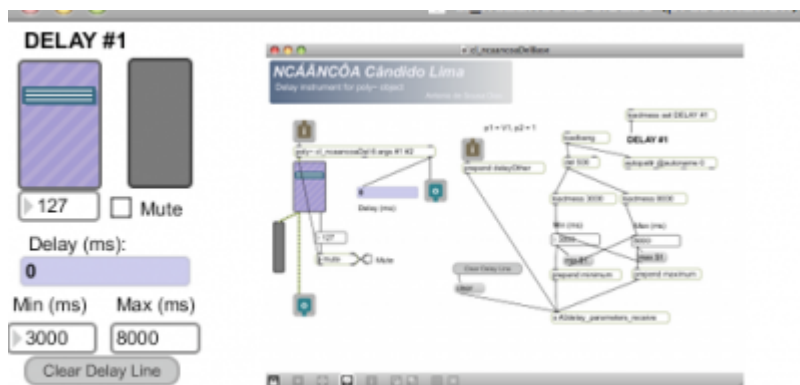


Cândido Lima, *Ncãncôa* : patch *cl_ncaancoa.maxpat* ? vue d'ensemble

Les deux patches opérant le retard du signal (DELAY V1 et DELAY V2, instances du patch *cl_ncaancoaDelBase.maxpat*) ont les mêmes construction et comportement.

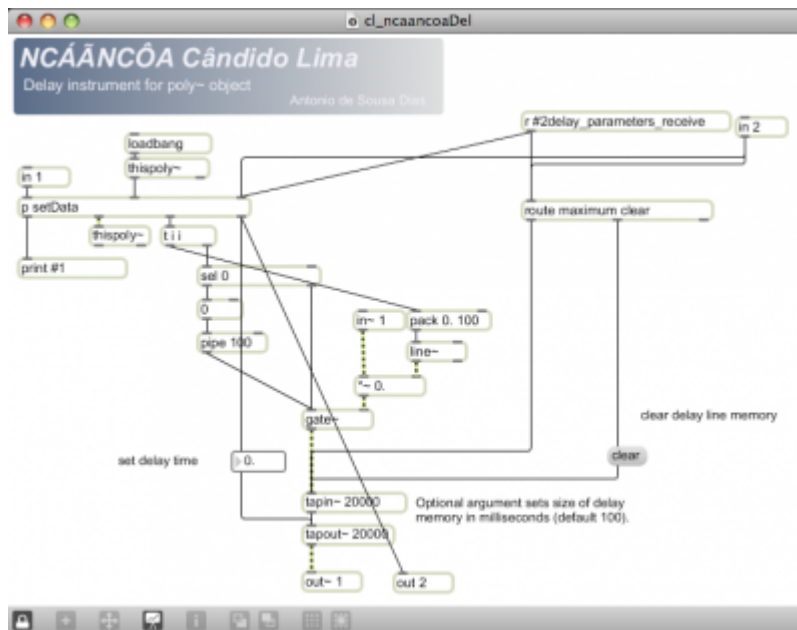
Ils sont constitués chacun par six lignes de retard (patch *cl_ncaancoaDel.maxpat*) préréglées pour un retard maximal de 20 secondes. Les entrées et sorties du signal des lignes sont fermées par défaut.

Figure 4



Cândido Lima, *Ncãncôa* : patch *cl_ncaancoaDel.maxpat*

Figure 5



Cândido Lima, *Ncãncôa* : patch *cl_ncaancoaDel.maxpat*

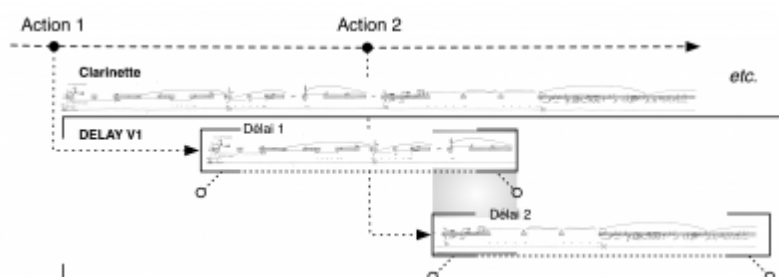
Dans la phase d'initialisation, chaque patch tire au sort une valeur de délai en millisecondes d'après une plage discrète de valeurs possibles et l'attribue à une variable. Ces valeurs sont choisies parmi 5 valeurs possibles entre les valeurs minimales et maximales attribuées. Ceci permet d'empêcher que des valeurs très proches comme, par exemple, 4000 et 4001 (millisecondes) soient obtenues.

Le premier patch, en plus, vérifie si les deux valeurs sont égales, au cas où il en résulterait un nouveau tirage, jusqu'à obtenir une valeur différente. Puis, dans chaque voix, le signal d'entrée est dirigé vers la première ligne à retard libre. Celle-ci est mise en activité et les entrées et sorties de signal sont ouvertes.

Pendant les déclenchements suivants, du type 2, le comportement est similaire : tirage au sort d'une valeur (et correction s'il en est le cas). Dès qu'une nouvelle valeur pour le délai est produite, les opérations suivantes sont effectuées : fermeture du signal d'entrée de la ligne à retard active, ensuite attribution de la nouvelle valeur à la première ligne à retard libre, puis activation et ouverture du signal d'entrée et sortie correspondante. Cependant, dès que le temps de délai précédent est accompli, la sortie de la ligne qui est restée active est fermée et la ligne devient inactive. Ceci permet de garantir que les phrases musicales sont entendues jusqu'à leur fin bien que, parfois, cela puisse entraîner une superposition des délais dans une voix (voir dans la figure ci-dessus, la région marquée en gris).

Pour finir la pièce, la dernière action ferme les entrées des lignes actives et, selon chaque valeur de délai ferme les sorties des lignes de retard et rend inactifs les délais.

Figure 6



Cândido Lima, *Ncãncôa* : patch *cl_ncaancoaDel.maxpat* ? schème de fonctionnement

De plus, ce système admet la mémorisation de différents ensembles de valeurs pour les délais permettant d'adapter l'échelle de valeurs possibles selon le caractère de la section de l'œuvre.

Le résultat a été approuvé par le compositeur (Lima 2010) et on peut l'écouter dans l'enregistrement (voir Lima 1995). Pour la suite, nous devons ajouter la réverbération (cf. les indications dans Lima 2010) et raffiner l'interfaçage avec l'interprète.

2. 2. Jorge Peixinho, *Harmónicos* (1967)

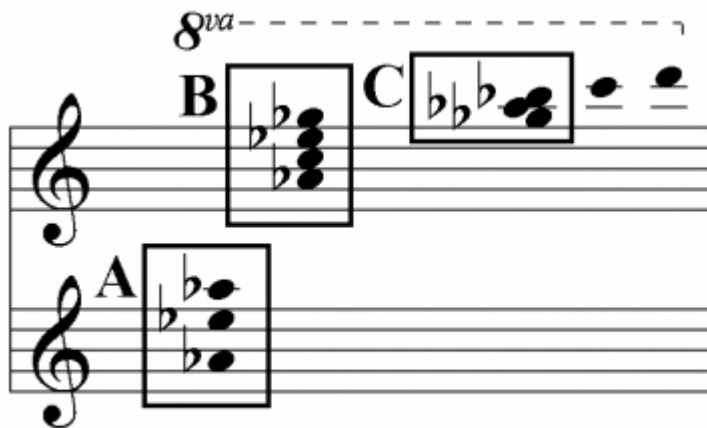
Jorge Peixinho (né en 1940 à Montijo ? décédé en 1995 à Lisbonne, Portugal) était compositeur, pianiste et chef d'orchestre. Peixinho a étudié la composition et le piano d'abord au Conservatoire de Lisbonne (où il enseignera plus tard et jusqu'à son décès), puis il a étudié la composition avec Porena et Petrassi à l'Accademia de Santa Cecilia à Rome. Après avoir travaillé avec Nono à Venise, il a étudié avec Boulez et Stockhausen à Darmstadt entre 1960 et 1970. En 1970, il crée le Groupe de musique contemporaine de Lisbonne. Pendant le reste de sa vie, il a continué à participer à différents festivals et événements autour de la musique contemporaine, jouant un rôle très important dans la promotion de la musique contemporaine au Portugal.

Harmónicos est une œuvre pour piano et électronique en direct datant de 1967, mais qui existe en plusieurs versions. Indiquant une durée minimale de 15 minutes, les différentes partitions se composent d'une série d'instructions pour l'exécution, en suggérant la façon dont l'interprète doit guider son improvisation construite à partir d'un ensemble de notes. Ces notes sont des approximations des harmoniques d'une fondamentale qui n'est jamais présente (Voir figure 7).

Figure 7a



Figure 7b



Jorge Peixinho, *Harmónicos* : réservoir de hauteurs

(7a) *Harmónicos* (1967), pour piano

(7b) *Harmónicos I ? 2b* (1986), pour quatre instruments, piano, harpe, clavecin et célesta

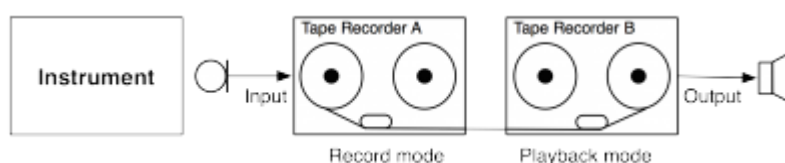
La performance est enregistrée et lue en temps réel avec un retard de 6 à 8 secondes. Nous pensons que *Solo* de Stockhausen (1965-1966) est, sans doute, une œuvre influente, car nous trouvons l'utilisation d'un vocabulaire similaire pour guider la performance (« varié », des matériaux « contrastés », et ainsi de suite) même si le système de retard utilisé dans *Harmónicos* reste beaucoup plus simple.

Le dispositif original est composé de deux magnétophones, un microphone et un amplificateur. Pendant la performance, le piano est enregistré, par un magnétophone et la bande est lue par le deuxième magnétophone. Cela introduit un retard en raison de la distance entre les magnétophones, générant un canon à l'unisson.

Depuis 1965, cette configuration a été une voie commune pour obtenir un long délai dans le signal et nous pouvons tracer différentes formes d'utilisation, avec ou sans réinjection simple ou double. Nous pouvons trouver quelques exemples, allant de compositeurs et œuvres de musique contemporaine comme *I of IV* (1966) de Pauline Oliveros ou l'œuvre citée ci-dessus, *Solo* de Stockhausen, jusqu'au rock progressif comme Brian Eno et Robert Fripp au début des années 1970, pour n'en nommer que quelques-uns.

Cependant, cette configuration analogique pose deux problèmes. D'une part, il est aujourd'hui difficile de trouver aisément deux magnétophones analogiques. D'autre part, il y a le risque des fluctuations, en raison de la distance entre les têtes de lecture et d'enregistrement (pour obtenir un temps de retard allant de 6 à 8 secondes à 38 cm/s, la distance doit varier entre 2,28 et 3,04 mètres) et du fait que, sur chaque magnétophone, la bande est bornée à un seul cabestan, donc la tension n'est pas respectée. De fait, et selon Keane, il faut garantir la tension correcte par rapport aux têtes de lecture et il faut également faire attention aux senseurs pour l'arrêt automatique qui doivent être débranchés (Keane 1980).

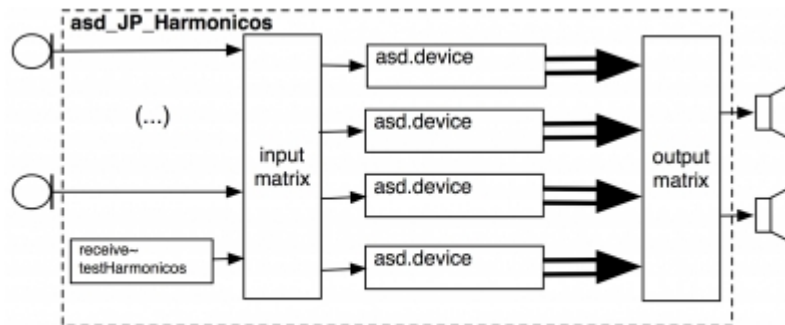
Figure 8



Jorge Peixinho, *Harmónicos* : diagramme du dispositif (adap. des instructions de la partition)

Afin de surmonter ces problèmes, nous avons décidé de programmer un patch Max/MSP simulant aussi près que possible ce dispositif, qui refond l'ensemble des différentes versions dans un même programme, *asd JP_Harmonicos.maxpat*. Conçu et programmé au départ dans Max/MSP, en 2002 (Macintosh iBook G3-Max/MSP 4.0.7), ce dispositif a été employé par Francisco Monteiro (Piano) et António de Sousa Dias (Mac, Max/MSP) lors d'un concert à Lisbonne, en mai 2002. Il a été utilisé régulièrement et, depuis 2006, cette œuvre est régulièrement jouée avec José Luis Ferreira assurant l'électronique. La version actuelle est programmée dans Max 5 et apporte un certain nombre d'améliorations. Sa structure est représentée ci-dessous (voir figure 9).

Figure 9



Jorge Peixinho, *Harmónicos* : diagramme de blocs simplifié du programme *asd JP_Harmonicos*

asd JP_Harmonicos.maxpat est une interface vers les souspatches simulant le dispositif et ses 16 entrées peuvent être redirigées vers un maximum de quatre appareils (pour la performance de *Harmónicos* I-2b). Il gère également les différents paramètres pour chaque périphérique et leur position dans un champ stéréophonique (voir figure 10). Il y a une entrée supplémentaire, à travers l'objet « receive~ », comme un moyen de fournir une entrée supplémentaire. La section « Contrôles généraux » permet de contrôler simultanément tous les appareils. Enfin, il y a aussi un « Gestionnaire de préréglages » pour mémoriser les informations concernant les réglages du patch principal et chaque périphérique.

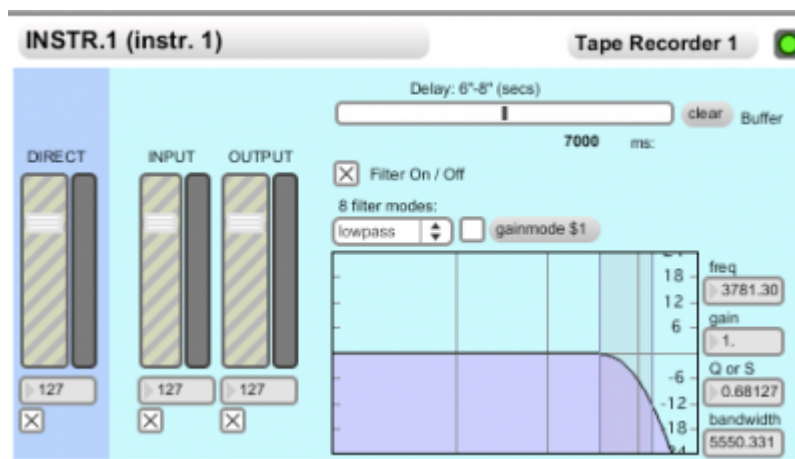
Figure 10



Jorge Peixinho, *Harmónicos* : fenêtre principale du patch *asd JP_Harmonicos*

L'abstraction Max *asd.device.maxpat* (voir figure 11) met en œuvre le système de délai. Pour chaque configuration, l'utilisateur peut contrôler la quantité de signal direct en entrée, le pourcentage envoyé dans la ligne à retard, ainsi que le temps de retard et les paramètres du filtre.

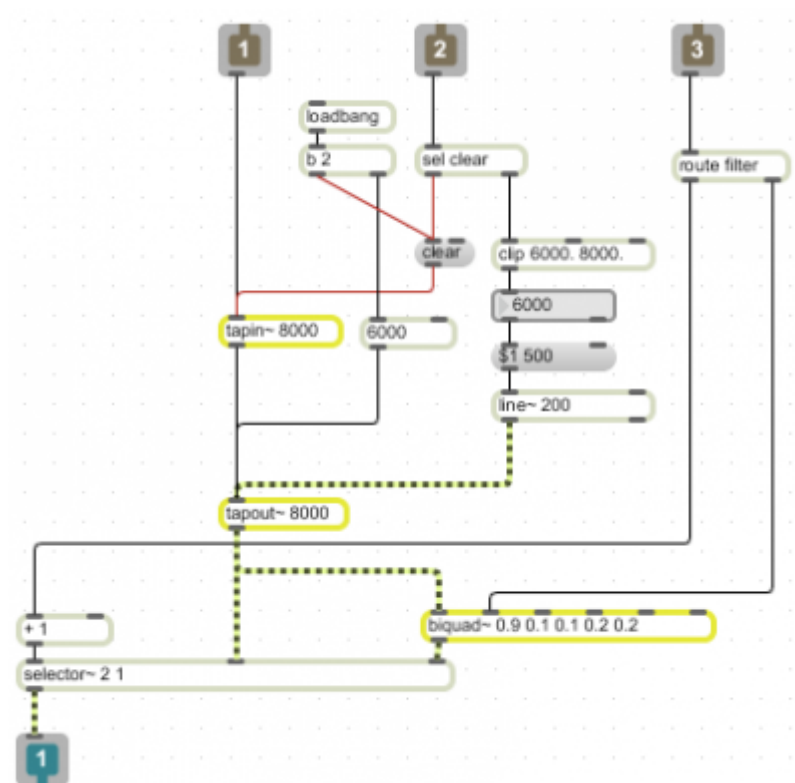
Figure 11



Jorge Peixinho, *Harmónicos* : patch *asd.device.maxpat*

L'abstraction *asd.device.maxpat* comprend le subpatcher *tapeDelay* contenant le code de la ligne à retard réalisant le retard par relecture de bande. Dans ce patch, l'entrée à gauche accepte le signal audio, celle du milieu, les messages entrants pour régler le temps de retard et l'entrée à droite, les messages pour l'activation du filtre et les réglages des paramètres (voir figure 12 ci-dessus).

Figure 12



Jorge Peixinho, *Harmónicos* : patch *tapeDelay* encapsulé dans le abstraction *asd.device.maxpat*. Ce souspatch contient le code de base pour la ligne à retard

Nous avons introduit des modifications à l'idée originale de Peixinho, paradoxalement, avec l'objectif de la respecter. Ces modifications comprennent :

- une option activant un filtre passe-bas, pour simuler une petite perte de hautes fréquences en raison de l'enregistrement : voir, par exemple, Revox A77 manuel, p12.1.4 : facteur de correction pour la courbe de réponse (*Revox A77*) ;
- une banque de mémoire pour stocker les paramètres de répétition et de concert ;
- la définition de l'emplacement de la source d'entrée et son image de retard dans le champ stéréophonique défini par les haut-parleurs.

La première modification nous a confronté avec le problème du respect trop strict concernant les conditions techniques de l'époque. La raison pour laquelle nous n'avons pas imposé une différence entre l'entrée et les sons retardés est basée sur l'hypothèse selon laquelle Peixinho aurait préféré une texture où les deux sources se mélangent subtilement ensemble. Dans la partition de *Harmónicos*, il déclare que les sons retardés constituent un « alter ego » du pianiste et, dans *Harmónicos I*, il affirme que « c'est de la polyphonie résultante du système constitué par le piano et le haut-parleur que l'œuvre sera formellement définie » (Peixinho 1967).

La deuxième modification nous a semblé justifiée comme une manière de garantir un minimum de commodité pour la performance. Les raisons invoquées pour la première modification s'appliquent également à la troisième modification : celle-ci a été mise en œuvre comme un moyen d'obtenir un équilibre entre les instruments et la partie électronique.

Enfin, il y a une option que nous n'avons pas encore mise en œuvre : la possibilité d'un arrêt automatique après un minimum de 15 minutes pour simuler la fin du défilement d'une bande. En effet, nous sommes presque certains que Peixinho indique la durée minimale de 15 minutes en tenant compte du diamètre des bobines de bande. Par exemple, dans une bobine d'un diamètre d'environ 18 cm (7 pouces), il y a 1200 pieds (environ 365 mètres) de ruban, ce qui permet d'enregistrer environ 16 minutes à une vitesse de 38 cm/s, mais le fait est que Peixinho n'a pas prévu explicitement cette option.

2. 3. Jorge Peixinho, *Sax-Blue* (1982)

La première version de *Sax-Blue*, pour saxophones soprano et alto et système électronique live, date de 1982 et est dédiée à Daniel Kientzy. Le système électronique nécessaire est composé d'une chambre d'écho à bande magnétique Korg Stage Echo 500 (SE500) et d'une pédale double Korg S-2 Dual Footswitch. De plus, l'interprète est supposé manipuler la SE500 pendant la performance, soit à la main sur le panneau frontal, soit grâce à des pédales.

Dans la partition, des passages écrits en notation proportionnelle alternent avec des passages écrits en notation traditionnelle avec et sans barres de mesure ou chiffrage de la mesure. La partition contient également des notes concernant la performance, comment mettre en place le système et une explication des symboles. Pour une description plus complète de l'utilisation de la SE500, Peixinho nous renvoie vers l'œuvre *Variantes-invariants* (1982) de Costin Mioreanu.

Figure 13



Jorge Peixinho, *Sax-Blue* : exemple de réglages pour la chambre Korg SE 500 dans la partition (adapt. Jorge Peixinho, *Sax-Blue*). Dans cet exemple, nous pouvons percevoir la tête de lecture sélectionnée (E II), la réinjection et des valeurs de vitesse (FB=5, SP=4) et la position de la pédale (pédale droite

déclenchée)

En 2001, le saxophoniste José Massarão a décidé de jouer *Sax-Blue* en concert. Des difficultés pour accéder à une SE500 nous ont conduits à envisager la mise en œuvre d'une simulation au travers de Max/MSP, comme une solution de compromis permettant la réalisation de *Sax-Blue* en toutes circonstances.

Le manuel d'opération (*Korg Stage Echo SE 500*) et les notes pour la performance, écrites par Peixinho, n'étaient pas suffisants pour comprendre intégralement le mode de fonctionnement de la SE500. Ni ceux qui sont inclus dans *Variantes-invariants*. La même chose pour l'enregistrement de l'œuvre ; les indications précieuses de Kientzy lui-même (au moment d'un concert à Lisbonne où il a joué *Sax-Blue*, en juin 2001), nous ont beaucoup aidé, surtout du point de vue de certains passages musicaux. Enfin, l'article de Minemier (Minemier 1979) ajoute de précieuses informations techniques. Par conséquent, la présentation de cette version de *Sax-Blue* a été le résultat de l'assemblage de tous ces indices y compris le diagramme technique de la SE500.

Comme mentionné précédemment, le système mis en place consiste principalement en un Korg Stage Echo 500 entraîné par un pédalier Korg S-2 Dual Footswitch.

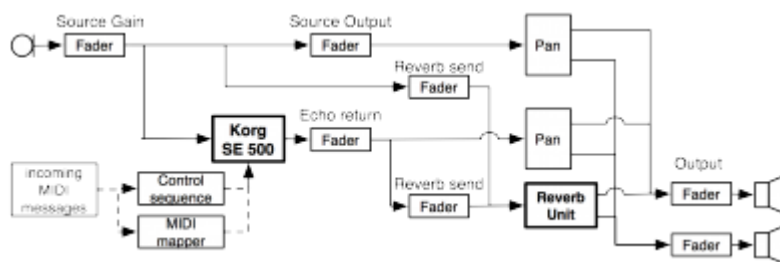
La SE500 est une chambre d'écho à bande magnétique permettant des retards entre 70 ms et 1500 ms. Ces retards sont obtenus grâce à un mécanisme par tête de lecture comprenant sept têtes, une tête d'enregistrement, une tête d'effacement et cinq têtes de lecture. Celles-ci sont assignées à des commutateurs étiquetés « 1 » (70 à 130 ms), « 2 » (200 à 370 ms), « 3 » (400 à 750 ms), « Long Delay » (800 à 1500 ms) et « Sound on Sound » (SOS). Celui-ci est une fonction spéciale qui permet à l'audition des sons préalablement enregistrés avec un retard allant de 20 à 30 secondes. L'utilisateur peut entendre des retards différents simultanément grâce à la combinaison des commutateurs. De plus, il y a deux autres commandes importantes : les potentiomètres rotatifs « Speed » et « Feedback ». « Speed » permet de changer le temps de retard à un taux approximatif de 1 :2 et « Feedback » contrôle la quantité du son enregistré réinjecté dans le système.

Le S-2 Dual Footswitch comprend deux interrupteurs chacun ayant leur propre connexion. Il était principalement destiné à une utilisation avec les machines de percussion Korg. Ici, il est couplé avec les contrôles externes de la SE500 : un commutateur entraîne le contrôle de l'« Effet » (pour couper les têtes de lecture) et l'autre le « Feedback » (annulant l'effet de rétroaction ? une seule répétition).

2. 3. 1. L'implémentation de la Korg SE500

Nous avons conçu un système intégrant tous les éléments nécessaires à l'exécution en direct selon le schéma simplifié de la figure ci-après.

Figure 14

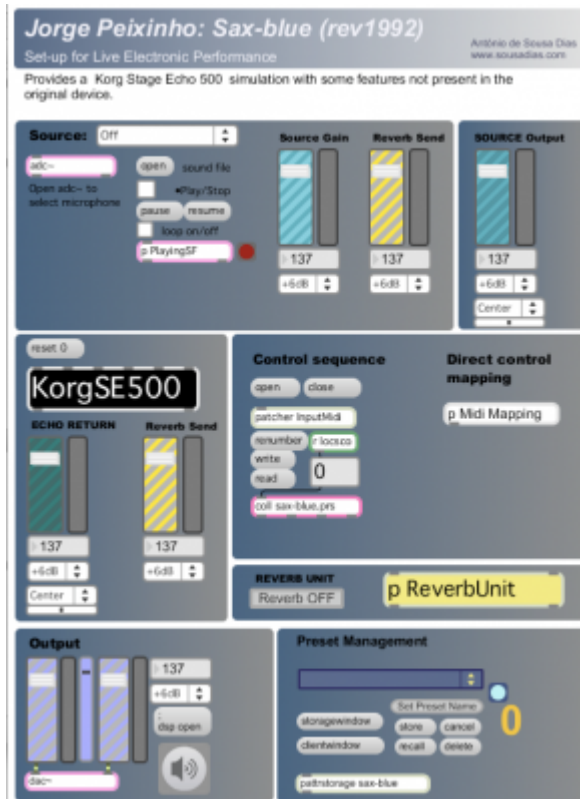


Jorge Peixinho, *Sax-Blue* : diagramme de blocs simplifié du dispositif

Le patch principal (voir la figure 15) intègre la Korg SE500 mais il intègre également une unité de réverbération (ici prise à partir du patch *rev1~* appartenant à la bibliothèque *jimmies*, Settel 1998) et un

système de mixage qui ne sont pas mentionnés dans la partition, mais entendus dans l'enregistrement. Il comprend également un mécanisme de mémorisation ainsi que des fonctions intégrées pour contrôler la SE500 soit au travers des contrôleurs MIDI soit par un séquenceur. Le séquenceur permet l'enregistrement au pas à pas des « gestes » contrôlant la SE500 sous forme de commandes. Ceux-ci, mémorisés dans un objet « coll », sont déclenchés pendant la performance. Les contrôleurs MIDI sont assignés aux paramètres de la SE500 au travers d'une matrice.

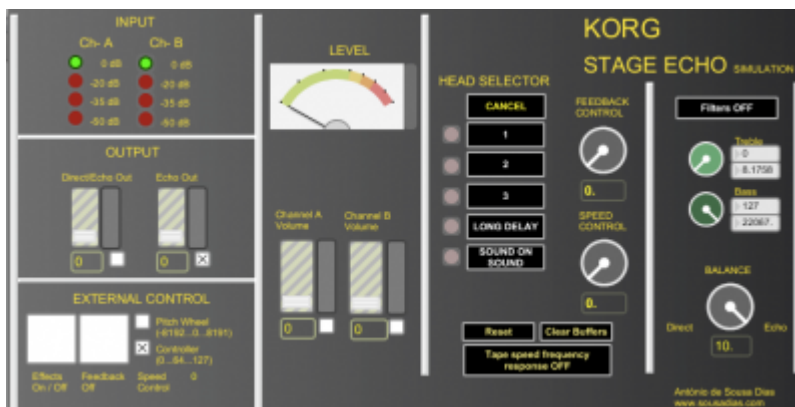
Figure 15



Jorge Peixinho, *Sax-Blue* : La fenêtre principale du patch *Sax-Blue-v5.maxpat* (d'après Jorge Peixinho, *Sax-Blue*)

Le patch SE500.maxpat simule le comportement de la SE500 et peut être utilisé en tant qu'abstraction Max ou intégré dans un objet « bypatcher ». Il ressemble au panneau frontal de la SE500 avec des modifications et adaptations.

Figure 16



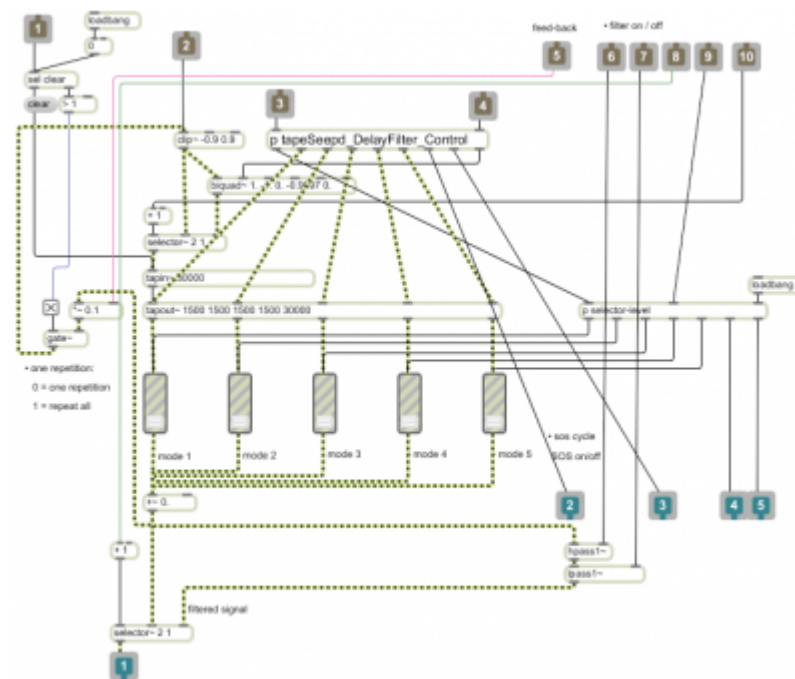
Jorge Peixinho, *Sax-Blue* : La fenêtre principale du patch *KorgSE500.maxpat* (Max, mode de « Présentation »)

Ce patch comprend cinq entrées correspondantes aux 5 entrées de la SE500 (deux signaux audio et trois interrupteurs pour les pédales de commande) et deux sorties correspondantes aux sorties audio. Il contient une entrée supplémentaire pour des commandes externes offrant un moyen de contrôle externe du panneau frontal.

Ainsi, tous les paramètres de la SE500 peuvent être modifiés en temps réel de deux façons : la manipulation directe du patch (souris ou pavé tactile) ou des messages de commande à distance via la sixième entrée. Dans le cas de *Sax-Blue*, ces messages de commande sont envoyés à travers le patch *Sax-Blue-v5.maxpat* où le patch SE500 est emboîté.

Le souspatch *EchoUnit* contient le code de base pour le traitement audio (figure). Il est implémenté comme une ligne à retard offrant de multiples points d'accès pour la lecture (multibascule). Ce résultat est obtenu grâce à une combinaison d'un objet « *tapin~* » avec cinq objets « *tapout~* ». Malgré l'apparence simple de cet appareil, nous avons eu à l'améliorer en introduisant un mécanisme de compensation de niveau des cinq sorties pour éviter la distorsion du signal et l'écrtage. Ce mécanisme prend en compte le nombre de têtes actives et ajuste leur gain en conséquence. Nous avons ajouté aussi un commutateur permettant le *bypass* du filtre passe-bas simulant la réponse en fréquence de 100-7500Hz à basse vitesse et 100-14000Hz à pleine vitesse signalées par Minemier (Minemier 1979) (on ne considère que les valeurs supérieures).

Figure 17



Jorge Peixinho, *Sax-Blue* : Modélisation du circuit de la Korg SE 500 (d'après Jorge Peixinho, *Sax-Blue*): le souspatch *EchoUnit* - simulation des têtes de lecture de la bande

Lorsque nous avons entrepris ce travail de conception de la simulation de la SE500, en 2001, nous nous sommes confrontés au problème de l'interface résultant de modes d'interaction entre l'interprète et le pédalier et les sélecteurs de mode.

Pour un musicien jouant un instrument sur scène, il n'est pas très confortable d'utiliser en plus une souris, un pavé tactile ou un clavier pour contrôler un dispositif logiciel et à l'époque nous ne disposions pas de contrôleur MIDI.

Une solution possible a été de mettre en ?uvre un système de suivi de partition, mais nous avons abandonné cette idée en faveur de l'utilisation d'une pédale. Nous pensons que le déclenchement de la pédale établit un lien physique minimal entre l'interprète et l'ordinateur, ce qui peut être considéré à peu près comme un équivalent de la relation que les musiciens entretiennent entre eux lorsqu'ils jouent ensemble (un regard, un geste, etc.).

Nous avons donc décidé de mettre en ?uvre un langage conçu pour décrire les actions sous la forme de messages de contrôle à distance. Ce langage a une syntaxe très simple et permet de traduire tous les gestes du saxophoniste par des messages déclenchés par une pédale MIDI (pour plus d'informations, cf. « *Sax-Blue* gesture transcription : Sequence control and command description » dans Sousa Dias 2009).

Ensuite, nous avons programmé un séquenceur déclenchant ces messages au pas à pas. Ces messages sont mémorisés comme des lignes de commande dans un objet « coll », « sax-blue.prs ». Ce séquenceur comprend un algorithme permettant la lecture de plusieurs commandes par ligne, un moyen plus facile de contrôler l'appareil : l'action de la pédale MIDI déclenche la lecture d'une ligne contenant plusieurs actions. En outre, chaque action peut être placée n'importe où dans cette ligne : un geste peut être constitué de plusieurs actions contenues dans une ligne qui à son tour contienne plusieurs commandes. Cela permet une meilleure efficacité concernant la création, l'édition et le contrôle et suppression d'erreurs des instructions.

Ces instructions sont déclenchées par le patch InputMIDI. Ce patch reçoit des valeurs envoyées par une pédale (affecté au contrôleur 64). Ces valeurs, 127 ou 0, correspondent aux modes ON ou OFF. La valeur 127 (ON) déclenche un accumulateur qui envoie le numéro de la ligne de commande à exécuter et la valeur 0 (OFF) n'est pas prise en compte. La décision d'utiliser seulement la valeur ON, reste une raison d'ergonomie pure : la pédale disponible à l'époque était une pédale Yamaha FC5 et l'interprète préférait jouer debout. Ce n'est pas une position confortable pour jouer une pédale si petite, d'ailleurs un dispositif qui n'est pas très commun pour les musiciens qui ne jouent pas de clavier ou d'instruments comme la guitare électrique.

Aujourd'hui, nous préférons une approche plus appropriée pour un interprète, à l'aide d'un contrôleur MIDI comme l'*Evolution UC-33* en raison de ses boutons programmables et les boutons rotatifs ainsi que des commutateurs au pied. C'est la raison qui nous a conduits à mettre en ?uvre un souspatch permettant le routage des messages MIDI, une façon plus commode pour faire l'assignation des contrôles.

Cette simulation de Korg SE500 a permis une performance adéquate dans le cadre strict de *Sax-Blue*. Néanmoins, elle présente quelques limites qui empêchent son utilisation générique. En fait, des travaux supplémentaires doivent être entrepris : par exemple, le comportement du mécanisme de contrôle de vitesse de la bande est seulement approximatif. La même chose s'applique aux valeurs de rétroaction et une étude plus détaillée de la SE500 devrait être effectuée pour vérifier ses particularités, car elles peuvent se révéler pertinentes d'un point de vue musical.

Nous avons ajouté quelques fonctionnalités et nous avons décidé de faire quelques changements. La possibilité de contourner la réponse en fréquence de la vitesse de la bande et des filtres passe-haut et passe-bas peut représenter une amélioration, mais elle soulève certaines questions sans réponse. Nous pensons que, dans ce cas, plutôt que de maintenir un strict respect pour le comportement de la machine originale, il est plus intéressant de fournir un moyen optionnel de se rapprocher de l'idée de composition pour lignes à retard multiples sans « effets secondaires ». Les autres caractéristiques ajoutées comprennent un bouton de réinitialisation générale et un bouton pour réinitialiser les mémoires tampon du délai.

Une des modifications effectuées est importante, car elle introduit un changement dans la partition : le mode de fonctionnement du sélecteur de têtes de lecture. Normalement, sélectionner une tête annule toutes les autres sauf si nous appuyons en même temps sur deux commutateurs ou plusieurs d'entre eux. Ce comportement n'était pas commode à simuler ayant seulement un interrupteur à pédale disponible. Accéder à l'ordinateur par le biais de ses interfaces habituelles (souris, clavier, etc.) n'a pas aidé beaucoup non plus. C'est la raison pour laquelle nous avons décidé de programmer un comportement global indépendant pour la sélection de chaque tête de lecture. Néanmoins, le commutateur « Cancel » fonctionne comme prévu.

Conclusions et perspectives

Nous pensons que les solutions choisies pour le *recasting* des œuvres présentées sont adéquates et ont fourni un moyen de répondre à certaines questions posées concernant les détails musicaux. Elles essaient, autant que possible, de respecter des consignes musicales des auteurs en présentant clairement les choix (et les déviations) par rapport aux consignes des compositeurs.

Le choix de Max comme langage permet d'améliorer la documentation et la diffusion de ces œuvres, car, de plus, les patches peuvent tourner sur deux systèmes d'exploitation (Mac OS et Windows).

Cependant, des travaux supplémentaires doivent être faits. Par exemple, le dispositif conçu pour Lima est en train d'être amélioré de façon à obtenir une version adéquate et commode pour accompagner la partition. De plus, il faut réaliser l'interfaçage avec un dispositif MIDI, par exemple, pour améliorer les déclenchements de délais ainsi que des changements de valeurs minimale et maximale selon les sections de l'œuvre. Bien que les patches pour les œuvres de Peixinho soient déjà disponibles, pour satisfaire pleinement aux exigences proposées par des auteurs comme Bernardini et Vidolin (Bernardini & Vidolin 2005), il faut encore tester les systèmes et réaliser des éditions critiques des partitions.

Ces solutions proposées ne sont pas censées être utilisées comme dispositifs généraux : elles ont été simplement mises au point pour la performance musicale des œuvres présentées et ont fait déjà preuve de leur efficacité (cf. par exemple Ferreira 2011). Comme exemple de la limitation, la mise en œuvre du système de ligne à retard pour *Harmónicos* est conçue pour imposer des limites (6 à 8 secondes) et n'autorise pas des valeurs en dehors de ces limites. Pour *Sax-Blue*, il en va de même : si, en général, le comportement de la Korg SE500 est plausible, il n'est adapté qu'aux besoins *Sax-Blue*. Nous n'avons pas vérifié si elle pouvait être utilisée pour l'œuvre de Miareanu mentionnée ci-dessus, bien que José Luis Ferreira nous a référé l'utilisation dans la réalisation de cette œuvre dans le même concert. De même que pour *Ñcãncôa*. Néanmoins, le code étant disponible, il est possible de l'adapter et réutiliser selon les besoins d'autres œuvres.

Cette contribution pour la préservation d'œuvres a un sens très pragmatique : elle vise à rendre disponible la partie technologique de ces œuvres aux interprètes en permettant ainsi que ces œuvres soient réalisées dans différentes circonstances. Comme l'a affirmé Burns, « L'interprétation crée la possibilité de partager l'œuvre avec de nouveaux publics, et encourage l'étude approfondie de la musique par les artistes » (Burns 2002).

Nous croyons que ces œuvres de Peixinho et Lima sont assez intéressantes pour mériter cette opportunité.

*

Ce travail a été développé dans le cadre d'une bourse de postdoctorat attribué par Fundação para a Ciência e a Tecnologia / Ministério da Ciência da Tecnologia e do Ensino Superior (Portugal).

Pour citer ce document:

António de Sousa Dias, « Musique électronique « live » et « recasting » : Trois cas d'étude », *RFIM* [En ligne], Numéros, n° 1 - automne 2011, Mis à jour le 07/10/2011
URL: <http://revues.mshparisnord.org/rfim/index.php?id=137>
Cet article est mis à disposition sous [contrat Creative Commons](#)