



HAL
open science

Rejouer une œuvre avec l'électronique temps réel : enjeux informatiques et musicologiques

Alain Bonardi

► **To cite this version:**

Alain Bonardi. Rejouer une œuvre avec l'électronique temps réel : enjeux informatiques et musicologiques. Genèses musicales, Presses de l'Université Paris-Sorbonne, pp.239-254, 2015, 978-2-84050-991-2. hal-01242545

HAL Id: hal-01242545

<https://hal.science/hal-01242545>

Submitted on 22 Aug 2017

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.

Rejouer une œuvre avec l'électronique temps réel : enjeux informatiques et musicologiques

Alain Bonardi

Université Paris 8 et IRCAM-CNRS, UMR 9912, Sciences et techniques de la musique et du son

Mots-clé

Electronique temps réel, *patch*, génétique de la reprise des œuvres temps réel, dépendance au support technologique, élaboration/mise à jour homme-machine de relations et filiations.

S'il est commun de dire que toute œuvre musicale doit être jouée et surtout rejouée, la nécessité en est encore plus manifeste dans le répertoire contemporain, où un certain nombre d'œuvres nouvelles ne dépassent pas l'étape de la première audition. Au sein de ces corpus, nous nous pencherons plus particulièrement sur les approches temps réel de la musique numérique.

Précisons ce que recouvre ce domaine de la création musicale : les œuvres dites « temps réel » associent des musiciens humains et des ordinateurs. Les premiers exécutent leur partition ; les machines mises en œuvre captent soit le son émis par l'instrumentiste (grâce à des microphones) soit des paramètres relevant d'autres modalités comme le geste du musicien (par exemple, grâce à une caméra) ; ces informations sont ensuite traitées par les ordinateurs et permettent la génération de contenus sonores¹ en temps réel : les programmes informatiques sont capables de calculer et diffuser des sons à partir des entrées des musiciens humains dans un délai suffisamment court pour que le rendu de la machine semble instantané ; toutes les sources sonores semblent alors jouer « ensemble ». Une interaction entre musiciens et ordinateurs est ainsi proposée : il s'agit de simuler sur une machine une altérité musicale, un partenaire de jeu aussi souple et autonome qu'un musicien humain, notamment capable de suivre automatiquement le déroulement de la partition². Cette interaction des instrumentistes avec une modulation de leurs parties par des moyens électroniques est appelée *électronique temps réel*. Le calcul de l'ordinateur est déterministe mais donne un résultat sonore et un placement temporel qui ne peuvent pas être entièrement déterminés *a priori*, puisqu'ils sont liés à la source musicale en entrée.

Historiquement, ce type d'approche est apparu à la fin des années 1980. À cette époque, des centres de recherche musicale comme l'IRCAM tentaient d'enrichir les musiques mixtes, associant musiciens humains et sons électroniques. Jusqu'alors, une bande magnétique était le plus souvent utilisée, accompagnant les instrumentistes, mais imposant une temporalité figée. Le chercheur Miller Puckette développa alors une première version d'un logiciel intitulé *Patcher*, qui préfigurait les deux logiciels les plus utilisés aujourd'hui : *Max/MSP*³ et *PureData*⁴. L'informatique temps réel a progressivement investi plusieurs configurations musicales : le duo instrument-machine, avec par exemple les œuvres de Philippe Manoury (*Jupiter*, pour flûte et ordinateur, 1987-1992 ; *En Echo*, pour voix et ordinateur, 1993-1994) ; les œuvres pour ensemble et électronique, comme *Fragment de lune* (1985-1987) de Philippe Hurel ; les œuvres pour solistes,

¹ La génération de contenus visuels est également possible, mais sort du cadre de cet article qui n'envisage que les œuvres musicales et sonores.

² Philippe MANOURY, « Considérations [toujours actuelles] sur l'état de la musique en temps réel », *L'Étincelle*, Perspectives, 2007.

³ <http://www.cycling74.com>

⁴ <http://puredata.info>

ensemble et électronique, comme *Répons* (1981-1988) de Pierre Boulez ou *Diadèmes* (1986) de Marc-André Dalbavie. Plusieurs logiciels ont ensuite vu le jour, s'ouvrant au cours des dernières années à la vidéo et à l'image tri-dimensionnelle, comme *Isadora*, très utilisé dans le monde de la danse.

Le patch, pierre fondatrice de la musique interactive temps réel

Sans entrer dans des détails techniques qui seraient hors de propos, nous devons dire ici un mot de ces dispositifs, notamment au sujet de la notion de *patch*, omni-présente chez de nombreux compositeurs recourant à ces technologies. Cette courte digression nous semble nécessaire à la compréhension des problématiques exposées et des perspectives génétiques dégagées.

Les *patches* sont des modules logiciels de commande (c'est-à-dire de déclenchement et de synchronisation de dispositifs numériques) et de traitement numérique du signal utilisés dans les arts de la performance au sens large, comme la musique, la danse, le théâtre, la vidéo, ou encore les œuvres associant plusieurs arts. La plupart reprennent sous forme logicielle les paradigmes des dispositifs électroniques matériels utilisés sur scène depuis une cinquantaine d'années (par exemple les processeurs d'effets sonores, tels que les réverbérations sous formes de boîtiers installés en régie des concerts). Ces *patches* tirent leur efficacité et leur importante diffusion de leur fonctionnement en temps réel (à l'instar des appareils électroniques qu'ils remplacent) et de leur facilité de mise en œuvre grâce à un langage graphique sur ordinateur. Le *patch* marque le début des approches interactives dans lesquelles la machine est « en attente » d'informations venant du musicien (ou du *performer* au sens large⁵), informations acquises par des capteurs.

Pour permettre au lecteur de mieux comprendre ce qu'est un *patch*, nous présentons ci-dessous un module créé avec le logiciel *Max/MSP*, produisant deux sinusoïdes (ce sont des signaux de base, pauvres en termes d'expressivité, dont la courbe de l'amplitude dans le temps suit une fonction trigonométrique : sinus ou cosinus), à 100 et 200 Hz, et les additionnant, l'amplitude du résultat étant atténuée (multipliée par 0.25). L'aspect du *patch* fait penser à un montage électrique, comme ceux élaborés sur la paillasse d'un laboratoire de physique, avec des appareils qui sont les modules rectangulaires (tel l'objet *cycle~* qui produit une sinusoïde, à l'instar d'un générateur) reliés entre eux par des câbles, qui sont représentés par des traits hachurés. Il est extrêmement facile de réaliser ces *patches*, sans écrire de programme au sens classique, mais en utilisant le langage graphique du logiciel : une fenêtre vide est proposée à l'utilisateur, qui choisit dans une bibliothèque prédéfinie⁶ les objets qu'il souhaite utiliser, puis les positionne dans son espace de travail, en enfin les relie par des connexions⁷. La figure 1 schématise quelques étapes de confection du *patch*.

⁵ La notion anglo-saxonne de performer englobe tous les artistes présents sur scène : musiciens, comédiens, danseurs, etc. Elle n'est toutefois pas seulement générique mais apporte l'idée de la dimension « performative » de l'interprète : le terme « anglo-saxon » implique l'idée de dépassement d'un texte à interpréter, et peut englober l'improvisation. cf. Jerrold LEVINSON, « Performative versus Critical Interpretation in Music », *The Pleasures of Aesthetics, Philosophical Essays*, Cornell, 1996.

⁶ Il est également possible de créer ses propres objets, pour imaginer des fonctions qui ne seraient pas présentes dans le langage de départ.

⁷ Notons également que les *patches* autorisent la construction par encapsulation : un *patch* peut contenir des *sous-patches*, qui peuvent eux-mêmes en contenir d'autres...

Bibliothèque prédéfinie d'objets

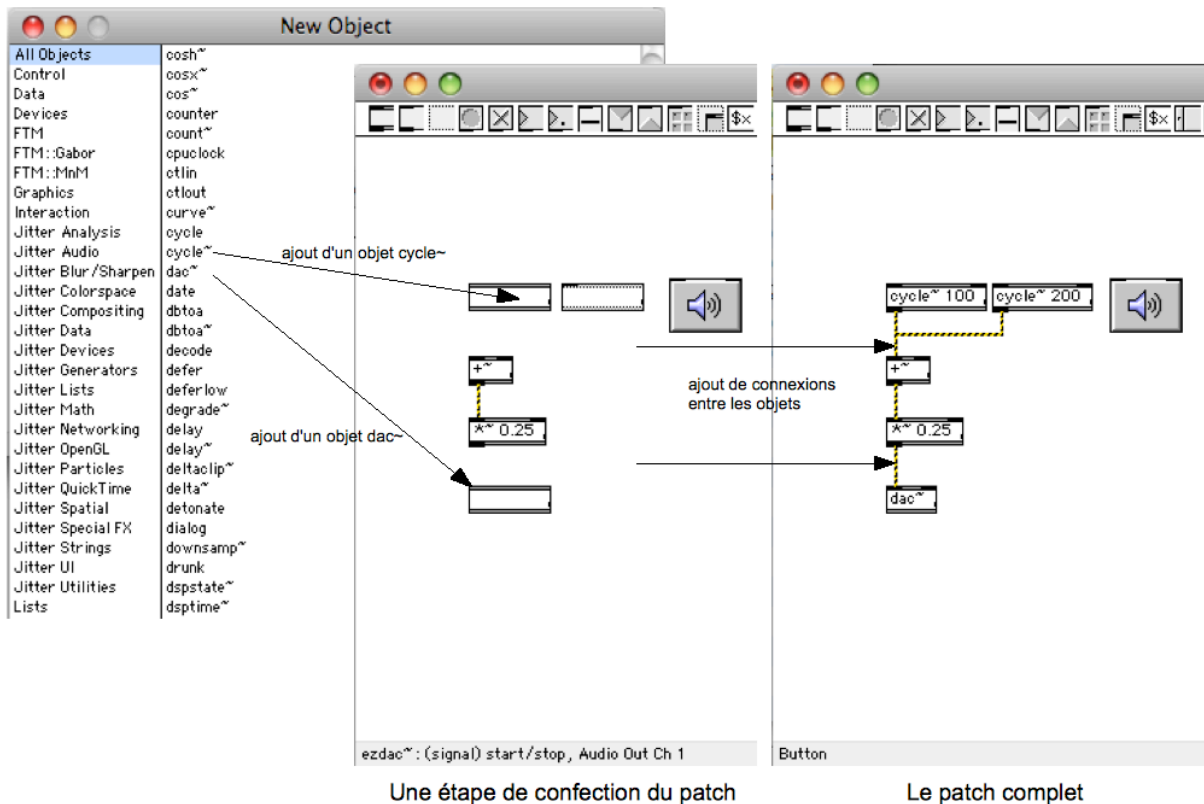


Figure 1. *Etapes de la confection d'un patch simple de synthèse additive à partir des objets prédéfinis dans la bibliothèque du logiciel Max/MSP.*

Cette facilité de déploiement n'a d'égal que la simplicité de mise en fonction du dispositif : il suffit de cliquer sur l'icône du haut-parleur, pour entendre le résultat sonore. La confection du *patch* et son exécution ne sont pas des phases séparées, mais simultanées, ce qui autorise la coordination des modifications et du suivi du rendu sonore. Cet environnement ouvre la porte à des pratiques de façonnage permanent associant perception, manipulation, et connaissances de domaines spécialisés comme la musique et le traitement du signal.

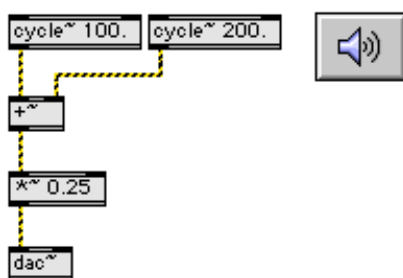


Figure 2. *Exemple élémentaire de patch Max/MSP produisant la somme de deux sinusoïdes (source : Alain Bonardi).*

Le *patch* simple de la figure 2 ne permet qu'une seule action de l'utilisateur : allumer ou éteindre le générateur sonore, en cliquant sur l'icône du haut-parleur. A l'opposé, il existe des *patches* extrêmement ouverts, complètement pilotables par l'utilisateur, par exemple ceux transformant le jeu d'un musicien. Par exemple, un *patch* d'*harmonizer*⁸ est un module logiciel construit selon ce

⁸ Curtis ROADS, *L'audio numérique – Musique et informatique*, traduction et adaptation française : Jean de Reydellet, Paris : Dunod, 2007, 658 pages.

principe d'objets reliés, et permettant de transposer vers l'aigu ou vers le grave un instrument ou une voix en entrée, ou encore de le répéter en le décalant dans le temps. Ainsi, dans une œuvre comme *Jupiter* de Philippe Manoury (1987-1992) pour flûte et électronique, se trouvent associés écriture musicale traditionnelle et transformation du son de la flûte par un *patch* temps réel. La partition papier contient la partie de flûte et les indications les plus précises possibles des traitements effectués par le *patch*, notamment des « échos » électroniques de l'instrument soliste avec des dispositifs transposeurs comme l'*harmonizer*.

Quand les réalisateurs en informatique musicale deviennent généticiens par nécessité

Les œuvres temps réel sont d'abord des compositions musicales tout court, et à ce titre suscitent des explorations génétiques des processus de leur constitution. Mais elles font également partie de ce que l'on appelle la musique avec ordinateur, ou *computer music*. À ce titre, nous renvoyons le lecteur à la contribution au présent ouvrage de Laura Zattra, qui met en évidence les concepts spécifiques de la critique génétique appliquée à la *computer music* au sens large. Le présent chapitre met quant à lui l'accent sur les processus visant à remonter des œuvres interactives, sur cette situation singulière de nouvelle investigation de l'œuvre, par exemple lorsqu'elle est programmée pour un concert ou un enregistrement quelques mois ou quelques années après sa première exécution. En effet, lorsqu'une œuvre interactive est reprise, plusieurs difficultés se présentent : la première, que nous ne ferons qu'examiner, est la simple préservation, au sens de la conservation, des *patches*. Ce qui est loin d'être évident : l'article d'Hannah Bosma⁹ et le rapport de Michael Longton¹⁰ montrent que dans un grand nombre de cas, seule la partition papier des œuvres interactives temps réel a fait l'objet d'une préservation systématique par une institution : dans l'enquête de Longton, 47% des compositeurs déclarent avoir perdu des fichiers relatifs à leurs œuvres sur leurs machines. La seconde difficulté, soulignée par Jean-Baptiste Barrière¹¹ dès 1993, est que les *patches*, même correctement conservés, fonctionnent en général mal ou plus du tout quelques mois après leur dernière mise à jour : les ordinateurs, les systèmes d'exploitation, les logiciels évoluent si rapidement, sans toujours garantir de compatibilité avec les versions précédentes, qu'il est quasiment certain que l'un au moins des composants de l'ensemble technologique nécessaire à la bonne marche de l'œuvre ne sera plus opérant.

Le projet de rejouer une œuvre interactive suppose donc toujours l'examen attentif de tous les éléments disponibles provenant des exécutions précédentes : partitions, *patches*, enregistrements, etc. Si nous définissons la critique génétique comme une approche qui vise non pas l'œuvre finie, mais le processus d'écriture, pouvons-nous considérer l'activité de remise à niveau technique que nous venons de décrire comme en participant ? La réponse est selon nous affirmative. Cette activité de « maintenance » créative constitue un versant original de la critique génétique, qui associe intimement lecture et écriture, dans un rapport inédit, imposé par l'artefact informatique, à un moment assez original dans la vie de l'œuvre, celui de sa « reviviscence ». Cette acception suppose bien évidemment que nous élargissions la définition de la critique génétique à des productions qui ne relèvent pas de l'écriture *stricto sensu*, mais de l'élaboration technique de situations d'interaction créative dans des œuvres musicales.

Précisons le statut du *patch*. D'une part, le *patch* ne relève pas à strictement parler de l'écriture musicale, même si les catégories habituelles de la musique y sont largement représentées. Le mode de production du *patch* s'apparente à un raffinement permanent, par retouches successives. Même les utilisateurs les plus aguerris aux possibilités de traitement du signal, ayant une idée

⁹ Hannah BOSMA, « Documentation and Publication of Electroacoustic Compositions at NEAR », *Proceedings of the 2nd Electroacoustic Music Studies Network International Conference* (EMS 05), Montréal, Canada, 2005.

¹⁰ Michael LONGTON, *Record Keeping Practices of Composers*, rapport d'enquête, site Internet du projet InterPares 2, disponible à <http://www.interpares.org>, lien vérifié le 28 septembre 2009.

¹¹ Jean-Baptiste BARRIERE, « Écriture musicale et ordinateur. Notes sur la genèse », *Genesis*, 4 (1993), p. 11-22.

claire du résultat à obtenir, procèdent ainsi : car tout ensemble de transformations sonores possède des paramètres qu'il convient de régler finement à l'oreille, et que l'on ne peut déterminer en amont de son implémentation, de manière abstraite. Il est impossible de se dispenser de cette phase empirique. D'autre part, le *patch* ne relève pas non plus d'une écriture informatique : il est par exemple assez difficile d'écrire une boucle contrôlée dans un *patch*, alors qu'il s'agit d'un mécanisme de base de la programmation structurée classique. Il n'y a plus de code à écrire, modifier et compiler, mais encore une fois un dispositif à façonner de manière interactive : l'utilisateur adapte les objets utilisés, les connexions et les paramètres des *patches* jusqu'à ce qu'il obtienne un résultat satisfaisant. Le *patch* peut être considéré comme un réceptacle empirique, support de l'œuvre, dans lequel sont encodées toutes sortes de connaissances non structurées, non explicitées, simplement posées-là par le truchement du langage graphique des programmes comme *Max/MSP* ou *PureData* et des mécanismes de l'informatique multimédia : copié/collé, encapsulation, etc.

Dans une institution comme l'IRCAM, les réalisateurs en informatique musicale confectionnent des *patches* en collaboration avec les compositeurs. Les réalisateurs en informatique musicale sont des « interprètes »¹² qui doivent non seulement jouer l'œuvre pendant le temps du concert mais aussi ré-interpréter la pièce en fonction des conditions techniques, lorsqu'elle est remontée. La migration¹³ ou portage du *patch* vers une nouvelle version est donc bien une ré-interprétation. Ainsi, le cycle de vie de l'œuvre interactive induit une démarche génétique du côté du producteur, de « l'auteur », ici le développeur de *patches* à côté de celle, plus habituelle, menée par le lecteur, par exemple un musicologue¹⁴. Nous pourrions ici faire un parallèle avec les écrivains qui ont abandonné papier et machine à écrire au profit de l'ordinateur, et ne cessent d'arpenter les différentes versions de leurs fichiers de textes. Mais les réalisateurs en informatique musicale sont encore plus dépendants du support technologique : les *patches* précédents ne fonctionnent plus nécessairement ; pour prolonger le parallèle, il faudrait imaginer des écrivains n'arrivant plus ou presque plus à accéder aux documents stockés sur leur ordinateur, qu'ils seraient sans cesse obligés de reconstituer. Les œuvres sur support informatique appellent elles aussi leurs *Sisyphes*.

Génétique de reconstitution et authenticité

Mais cette génétique du *patch* du côté de la production de l'œuvre est-elle suffisante pour permettre de la remonter correctement non pas seulement du point de vue technique, mais aussi et surtout du point de vue de l'authenticité ? Les œuvres interactives mixtes faisant appel à ce type de dispositif remettent cette problématique au goût du jour, alors qu'on la situait spontanément du côté de la musique baroque. La question de l'authenticité est un corrélat des problématiques de reconstitution, quel que soit le contexte. Le pianiste autrichien Paul Badura-Skoda, interprète du répertoire classique et romantique sur instruments d'époque s'est longuement penché sur la question¹⁵ : « je ne partage pas l'opinion de certains restaurateurs affirmant que les bonnes copies de piano-forte sonnent mieux que leurs modèles dans la mesure où le bois est plus jeune et n'est donc pas déformé.(...) Ces copies ne peuvent pas atteindre l'enchantement procuré par les Graf ou les Walter bien restaurés ». Le réalisateur en informatique musicale se trouve confronté aux mêmes nécessités de choix au moment de « transcrire » un *patch*. Si toutes les œuvres musicales trouvent une forme d'achèvement dans leur exécution, celles faisant appel aux *patches* en sont encore plus dépendantes, en l'absence de substrat culturel support

¹² Serge LEMOUTON, Raffaele CIAVARELLA, Alain BONARDI, « Peut-on envisager une organologie des traitements sonores temps réel, instruments virtuels de l'informatique musicale? », *Proceedings of the Fifth Conference on Interdisciplinary Musicology (CIM09)*, Paris, 2009.

¹³ Alain BONARDI, Jérôme BARTHELEMY, « The preservation, emulation, migration, and virtualization of live electronics for performing arts: An overview of musical and technical issues », *Journal on Computing and Cultural Heritage*, 1, 1 (2008), p. 6 :1-6 :16.

¹⁴ Ross FELLER, « E-sketches: Brian Ferneyhough's use of computer-assisted compositionnal tools », *A Handbook to Twentieth-Century Musical Sketches*, (Patricia Hall et Friedemann Sallis, éditeurs), Cambridge (UK) : Cambridge University Press, 2004, p. 176-188.

¹⁵ Paul BADURA-SKODA, *Etre musicien*, Paris : Hermann, 2007.

d'un système de notation et d'une organologie stables et reconnus. La genèse de ces pièces musicales ne s'arrête pas aux phases d'écriture, mais prend en compte les phases d'élaboration et de test technique des *patches*, ainsi que la performance. D'où une forme de génétique paradoxale, qui semble inachevable.

Prenons un exemple : faut-il conserver l'ancien format de fichier sonore des Stations d'Informatique Musicale IRCAM de la fin des années 1980 et début des années 1990, dont la fréquence d'échantillonnage est à 32 000 Hz, ou basculer vers la fréquence standard du CD, à 44 100 Hz ? Le premier format donne un son un peu métallique – mais n'est-il pas emblématique des musiques de ces années-là ? Tandis que le second donne un son qui nous semble plus « naturel » et plus « actuel ». Le réalisateur en informatique musicale Serge Lemouton raconte¹⁶ comment le même processus de migration de l'environnement Next à l'environnement Macintosh a provoqué des réactions fort variées chez les compositeurs concernés : si Philippe Manoury a considéré le résultat comme trop proche de l'ancienne version, et donc n'utilisant pas suffisamment les capacités techniques nouvelles, Michael Jarrell a trouvé pour sa part que la version révisée s'éloignait trop de l'original... Si le compositeur a le dernier mot, il faudrait aussi prendre en compte dans ces allers-retours entre intentions et réalisations le point de vue des interprètes au sens large, et plus particulièrement du réalisateur en informatique musicale qui se trouve au croisement des chemins selon plusieurs points de vue : interface entre les musiciens et le compositeur dans la mise en œuvre du dispositif technique interactif, interface entre les intentions le plus souvent formulées de manière poétique par le compositeur et l'implémentation technique dans le *patch*.

Le réalisateur en informatique musicale joue souvent le rôle de transcripteur, reprenant une ancienne version du *patch* pour la faire migrer vers un nouvel environnement. En tant que transcripteur, il est confronté à l'impossibilité d'être simultanément fidèle à l'esprit et à la lettre du texte qu'il a devant lui. Ce qui ne va pas sans poser de nombreux problèmes : au nom de l'authenticité et du respect du texte, le réalisateur en informatique musicale doit-il reproduire les erreurs manifestes du *patch* original, qui seraient l'équivalent de « fausses notes » dans une partition, mais qui peut-être contribueraient au son obtenu ? Nous n'avons évidemment pas de réponse absolue ni définitive à l'ensemble de ces questions ; nous préférons nous attacher à la description des approches méthodologiques mises en œuvre dans un cas particulier, la reprise d'une œuvre intitulée *Diadèmes*.

Une œuvre-témoin : *Diadèmes* de Dalbavie

Diadèmes de Marc-André Dalbavie est une pièce en trois mouvements pour alto, ensemble instrumental et électronique temps réel créée à l'IRCAM le 16 juin 1986 (Centre Georges Pompidou, Paris, alto : Nathalie Baudoin, Ensemble Itinéraire, direction : Arturo Tamayo). Cette œuvre a été jouée pour la dernière fois en 1992 à l'occasion de l'enregistrement d'un CD¹⁷. Elle a été distinguée par le jury du prix *Ars Electronica* en 1987 pour « la qualité de l'intégration des instruments traditionnels avec des sons numériques synthétiques ».

Diadèmes a été créée lors du colloque « Systèmes Personnels et Informatique Musicale »¹⁸ qui s'est tenu à l'IRCAM en 1986. A l'époque, on désignait par « systèmes personnels » toutes les nouvelles machines mettant à la portée du grand public les dispositifs de synthèse et de traitement sonores jusqu'alors réservés aux institutions. Yamaha avec ses synthétiseurs DX7, ses modules de réverbération et ses processeurs d'effets était l'un des principaux fabricants de ces « systèmes personnels », et l'IRCAM disposait d'un studio équipé spécialement par Yamaha.

¹⁶ Communication personnelle à l'auteur, octobre 2006.

¹⁷ Adès 205 202 - 1996, enregistrement 1992 / Accord 465 313-2 - 1996

¹⁸ Marc-André DALBAVIE, « De l'écriture au timbre », *Actes du Symposium Systèmes Personnels et Informatique Musicale*, Paris, Ircam, 1986.

Suite à la demande d'un organisateur de concert et de l'éditeur de la partition (Lemoine) concernant la création de cette pièce aux Etats-Unis en décembre 2008 ont été posés systématiquement les problèmes posés par l'exécution en concert de la partie technologique de cette œuvre, 23 ans après la première audition. Serge Lemouton, réalisateur en informatique musicale, a étudié avec un certain nombre de chercheurs¹⁹ de l'IRCAM participant au projet européen CASPAR²⁰ les mesures à envisager pour la mise à jour de cette pièce, puis réalisé une nouvelle version. Cette démarche constitue également une sauvegarde des éléments constitutifs de la pièce en vue d'en faciliter des réalisations ultérieures.

La partie électronique de *Diadèmes* de Dalbavie comporte deux types de dispositifs :

- des synthétiseurs générant des sons selon le principe de la synthèse FM²¹, procédé permet d'engendrer des sons par modulation de fréquence : deux oscillateurs, dont l'un pilote la fréquence de l'autre, suffisent à créer une grande variété de sons. Ce procédé, mis au point et breveté par le chercheur John Chowning en 1973 fut ensuite récupéré par Yamaha. Lors de la création, deux appareils Yamaha TX 816 étaient utilisés (chaque appareil comporte huit cartes électroniques sans clavier ; chaque carte embarque les composants nécessaires équivalent à un synthétiseur DX-7), pilotés par des claviers Yamaha KX 88.
- des modules de transformation temps réel du son de l'alto : un *harmonizer*²² qui transpose en hauteur un son en entrée, et permet également de le retarder dans le temps ; un module de *panning*²³, qui permet de distribuer spatialement un son en entrée ; un module de réverbération ; enfin un module d'écho ; ces effets étaient, dans la version de 1986, groupés dans deux boîtiers : un Yamaha SPX 1000 et un Yamaha REV5.

La figure 3 ci-dessous montre l'ensemble du dispositif : on aperçoit en bas à gauche et à droite les synthétiseurs TX 816, qui sont joués par deux musiciens utilisant les claviers électroniques KX 88.

¹⁹ L'ensemble de cette étude a été documenté en interne à l'IRCAM par Karin Weissenbrunner, étudiante de l'Université de Berlin en stage en 2008, au sein de l'équipe Services en Ligne.

²⁰ <http://www.casparpreserves.eu>

²¹ Roads, *op. cit.*

²² Roads, *op. cit.*

²³ Roads, *op. cit.*

ORIGINAL SETUP OF ELECTRONICS

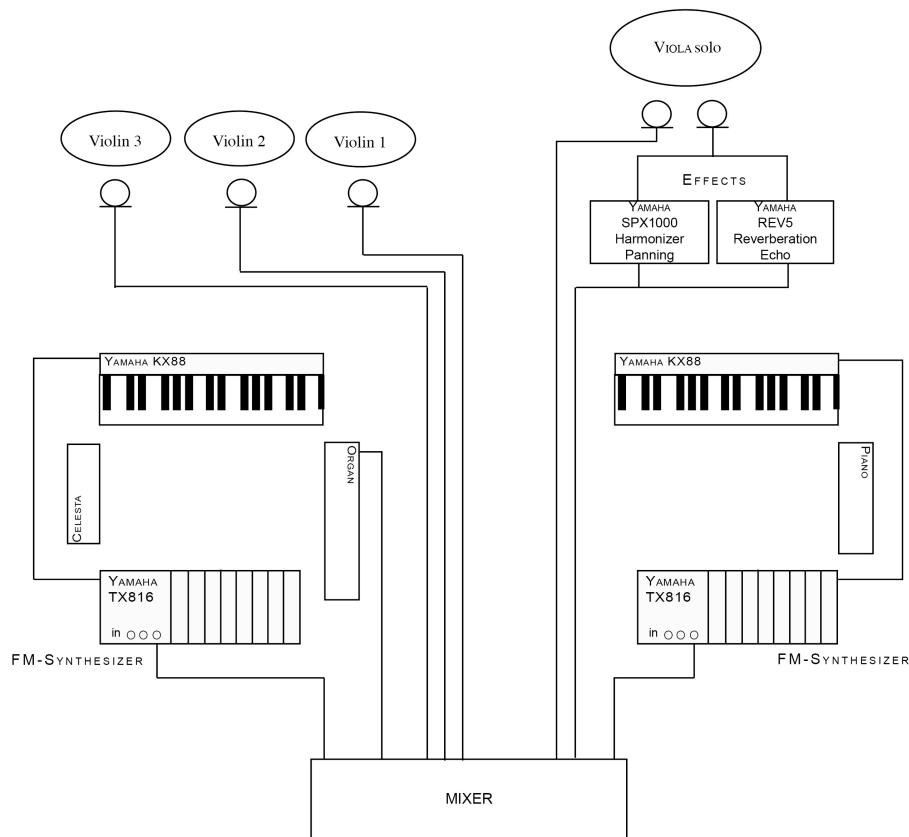


Figure 3. Schéma d'ensemble du dispositif électronique de Diadèmes. Source : Karin Weissenbrunner.

Pour aider le lecteur à comprendre le fonctionnement du dispositif, nous donnons deux exemples de réalisation. Le premier concerne la partie d'alto solo, mesures 79 à 84 de l'œuvre. La figure 4 montre la partie d'alto sur la troisième ligne à partir du haut. La deuxième ligne juste au-dessus de la partie instrumentale correspond à la notation de l'harmonizer (on distingue un H entouré en début de ligne), transposant l'alto de la mesure 79 à la mesure 81. La ligne du haut porte l'effet d'écho (indication « Echo » entourée mesure 80), de la mesure 81 à la mesure 84. Les deux flèches qui se croisent en bas du document indiquent un panning, c'est-à-dire un mouvement des sources sonores qui se croisent (ce qui était diffusé à droite vient à gauche, et vice-versa).



Figure 4. Un extrait de la partition de *Diadèmes*, mesures 79 à 84, montrant la partie d'alto solo et la notation des transformations temps réel. Source : Marc-André Dalbavie et Serge Lemouton.

Le second exemple concerne la synthèse FM par le TX 816. Mesure 56, lorsque le claviériste enfonce la touche correspondant au do1, le TX 816 produit les quatre accords indiqués figure 5, fournissant un son hybride entre vibraphone et marimba²⁴.



Figure 5. Un exemple d'accords générés par le synthétiseur DX7 mesure 56, lorsque le claviériste enfonce la touche du do1. Source : Marc-André Dalbavie.

Le problème principal dans le cas de *Diadèmes* est l'obsolescence des synthétiseurs FM TX 816 utilisés initialement. La situation est pour le moins difficile : les synthétiseurs Yamaha TX 816 ne sont plus fabriqués, ceux que possède l'IRCAM sont dans un état proche de la rupture. Il n'y a aucune possibilité de trouver dans un délai raisonnable un instrument en état de marche²⁵ ou de réparer ceux qui fonctionnent partiellement²⁶. Que faire ? Il existe des solutions d'émulation logicielle, à savoir des programmes informatiques qui prétendent imiter le son original des synthétiseurs Yamaha²⁷. Le compositeur a essayé plusieurs logiciels, sans qu'aucun ne lui donne satisfaction au niveau du résultat sonore.



²⁴ Dalbavie, *op. cit.*

²⁵ Il existe encore quelques appareils entretenus par des passionnés, et jalousement veillés par eux. Il n'y a quasiment jamais de vente.

²⁶ Les pièces détachées nécessaires (composants électroniques des années 1980) ne sont plus disponibles.

²⁷ Ce qui n'est pas chose aisée, car si le principe de la synthèse FM a été découvert et publié par John Chowning en 1973 à l'Université de Stanford aux États-Unis, sa mise en œuvre « industrielle » a été réalisée par Yamaha suite à une prise de brevet et une adaptation du procédé. Cf. John CHOWNING, « The synthesis of complex audio spectra by means of frequency modulations », *Journal of the Audio Engineering Society* (J.A.E.S.), 21, 7 (1973).

Figure 6. A l'IRCAM, un synthétiseur d'origine Yamaha TX 816 (largeur : 480 mm, hauteur : 176 mm), dont une grande partie des modules ne fonctionne plus. Source : Serge Lemouton.

En avril 2008, le réalisateur en informatique musicale Serge Lemouton a conçu une nouvelle version de la partie électronique de l'œuvre. Il a choisi de faire « revivre » ces synthétiseurs et les claviers qui les déclenchent en constituant ce que l'on appelle un échantillonneur ou *sampler*. Il s'agit en quelque sorte d'une base de données de sons d'un instrument, correspondant à son enregistrement à différentes hauteurs jouées avec différentes intensités sonores. En l'occurrence ont été enregistrées sur le « vieux » TX 816 de l'IRCAM (fonctionnant partiellement) les notes demandées par le compositeur dans la partition, avec différents programmes (c'est-à-dire instruments ou sons générés) et différentes vitesses (correspondant à la force d'appui sur la touche du clavier). Le principe du *sampler* permet de jouer un instrument unifié dans les registres et les intensités à partir de cette base de sons : lorsque l'on appuie sur une des touches du clavier commandant le *sampler*, deux cas sont possibles : soit la touche correspond à une note enregistrée, et cette dernière est jouée tout en étant « habillée » (attaque, décroissance, volume d'ensemble) ; soit la touche ne correspond à aucune note enregistrée, et le logiciel interpole une note à partir des notes enregistrées, supposée être proche de l'instrument d'origine.

Nous procédons en fait à une reconstitution d'un instrument ancien disparu, comme peuvent les mener parfois les luthiers en ré-inventant des instruments baroques. Avec des différences toutefois : nous disposons d'un spécimen, d'enregistrements de l'instrument en situation de concert, mais nous n'en faisons revivre qu'une partie, ne pouvant numériser complètement l'original. Il s'agit là d'une reconstitution des effets, et non des causes.

Concernant le deuxième ensemble de modules, ceux dédiés à la transformation temps réel de l'alto, la situation est bien plus simple : les effets existent encore ; ils sont désormais réalisés par logiciel en lieu et place de boîtiers électroniques ; de plus, leur qualité a progressé. Ici, le choix du compositeur se borne à une sélection parmi plusieurs éléments logiciels, de modules donnant à son avis le meilleur résultat en termes de réverbération, d'écho, d'*harmonizer* ou de *panning*. La notion de fidélité à l'ancien résultat sonore n'intervient pas, car la qualité des modules précédemment utilisés était très moyenne.

Si nous considérons l'ensemble du processus, étalé sur 23 ans, allant de la première audition à nos jours, quelle trajectoire « génétique » du patch pouvons-nous établir ? La technologie de la synthèse FM a été choisie dans les années 1980 comme la meilleure de l'époque pour réaliser des ponts sonores entre instruments acoustiques et électronique sous la forme d'accords, mais elle paraît aujourd'hui plus anecdotique, et certainement liée à l'époque²⁸. Les transformations sonores temps réel de l'alto ont peu évolué, peut-être parce qu'elles sont fondées sur des catégories musicales bien établies : on pourrait considérer sans difficulté l'*harmonizer* comme une version électronique du principe du canon musical...

Les œuvres interactives temps réel en danger

Les pièces interactives temps réel sont dans une complète dépendance à leurs supports matériels et logiciels ; elles sont conditionnées par les résultats des sciences du traitement du signal acoustique, ce qui provoque une situation inédite d'inadéquation aux paradigmes de la musique occidentale et une mise en danger de l'œuvre. La situation est tout à fait nouvelle : les œuvres interactives temps réel ne peuvent plus s'appuyer sur les paradigmes stables de ce que nous appelons au sens large la « musique classique occidentale », tels que la notation et l'organisation en familles instrumentales.

²⁸ Cette même synthèse FM était aussi utilisée par des groupes de musique électro-pop comme Depeche Mode...

Nous savions déjà que ces œuvres avec électronique ébranlaient les piliers de la composition musicale occidentale, en faisant vaciller les paradigmes de la partition, de l'instrument, voire de l'interprète et du compositeur. Nous découvrons désormais avec stupeur que nous ne pourrions peut-être plus entendre certaines d'entre elles. Le « recul » de l'universalité de l'écriture musicale, faisant face à l'explosion des notations voire à l'émergence de systèmes musicaux sans notation, s'étend désormais à la lutherie. Chaque œuvre devient un système en soi. Les accidents technologiques ne sont toutefois pas nouveaux : prenons le cas de l'arpeggione, instrument à six cordes, comportant des frettes²⁹, accordé comme une guitare, mais joué avec un archet comme un violoncelle. Inventé en 1823, il est rapidement tombé en désuétude. Sa disparition n'a pas entraîné celle des œuvres qui l'utilisaient, notamment la *Sonate en la mineur D. 821* de Schubert, pour arpeggione et piano. Cette dernière est désormais jouée soit au violoncelle, soit à l'alto. Cette adaptation est rendue possible par les paradigmes de l'organisation tempérée de l'espace des hauteurs, attribuant des équivalences à des sons aux timbres pourtant différents, et par celui des familles instrumentales de l'organologie classique. Une œuvre comme *Diadèmes* ne peut plus compter sur de tels paradigmes : d'une part, il n'existe pas de notation commune aux procédés électroniques de transformation sonore ; d'autre part, l'établissement de familles organologiques du domaine a peu de chance d'aboutir, tant la disparité des dispositifs en termes de paramètres et d'interface au sonore est grande. Ici, l'intension cède du terrain à l'extension...

Ajoutons que ces œuvres avec électronique temps réel sont pénalisées en termes de diffusion. En effet, les organisateurs de concerts hésitent à programmer des pièces telles que *Diadèmes*, pour des raisons techniques. Des exécutions récentes de cette œuvre ont été annulées. Tout un pan de répertoire de la musique électronique est menacé, les œuvres de cette époque sont en péril.

Premiers pas vers une génétique de l'œuvre interactive temps réel

Ce qui apparaît comme un danger important pour la transmission, voire même l'existence de notre culture, laisse aussi pressentir de nouveaux horizons de compréhension de cette dernière. Ainsi, la critique génétique des œuvres interactives faisant appel à l'électronique temps réel est à inventer.

Que pouvons-nous démêler de l'écheveau des différentes versions des *patches* d'une œuvre ? Que pouvons-nous induire de ces ensembles de traces en termes de parcours de création ? Quelles spécificités dans l'activité et quelles évolutions de la fonction de généticien ce corpus entraîne-t-il ?

A première vue, il est tentant de considérer le *patch* comme étant déjà une structure informatique, du seul fait qu'il est codé numériquement. En effet, à toute structure constituée, il est possible d'appliquer les techniques et méthodes automatiques de l'analyse et de l'extraction de connaissances³⁰. Or, ce codage du *patch* n'est qu'une sauvegarde du choix et du placement des objets et des liaisons entre eux dans l'espace de travail à l'écran, en quelque sorte une liste des opérations nécessaires à sa reconstitution telles que l'effectuerait le créateur à la souris. Le *patch* ne renvoie qu'au langage de base de son codage (noms des objets disponibles dans les programmes comme Max/MSP et PureData) et à certaines conventions non explicitées (ordre de placement à l'écran). Au nom de la facilité de mise en œuvre, il encode des connaissances qui font implicitement appel à un certain nombre de catégories, dans les domaines du traitement du signal, de la musique et de l'informatique. Donnons un exemple en retournant à notre *patch* élémentaire de la figure 2 : si tout le monde comprend que l'objet $+~$ permet d'ajouter deux sons, jamais la formule mathématique précise permettant de réaliser cette fonction n'est explicitée dans le fichier du *patch*.

²⁹ Une frette est une partie surélevée de la touche de certains instruments à cordes comme la guitare ou la viole, permettant de choisir la longueur de corde qui va entrer en vibration entre le chevalet et elle.

³⁰ Bruno BACHIMONT, *Ingénierie des connaissances et des contenus*, Paris : Hermès-Lavoisier, 2007.

Si le support des *patches* est bien informatique, nous savons d'emblée que toute tentative de génétique informatisée, se positionnant comme programme capable de travailler sur les logiciels que constituent les *patches*, selon des modèles *a priori*, est vouée à l'échec. Ce qui ne condamne ni à l'inaction ni au désespoir, mais oblige les chercheurs travaillant sur ces questions à des approches peut-être plus modestes : une piste raisonnable semble être l'association d'outils informatiques *ad hoc* et d'un ensemble de méthodes qui peuvent être systématisées sur une chaîne opératoire sous le contrôle d'opérateurs humains. Nous n'imaginons d'approche génétique des œuvres interactives temps réel que sous la forme d'une constitution interactive, homme-machine, de connaissances sur l'œuvre.

Nous avons distingué quatre domaines d'intervention de cette génétique, illustrés par des exemples de propositions émises par les chercheurs de l'équipe Services en Ligne³¹ de l'IRCAM :

- l'aide à la relecture des *patches* ;
- la modélisation de la provenance et du contexte de l'œuvre ;
- la virtualisation des *patches* ;
- la constitution d'organologies des modules de traitement temps réel du son.

Certaines de ces approches restent pour l'instant prospectives, mais il nous semble important de les évoquer, en sachant l'importance que prend la préservation active des œuvres avec électronique dans toutes les institutions de création qui constituent désormais des répertoires de pièces musicales que l'on souhaite pouvoir rejouer³².

L'aide à la relecture des *patches*

Les manuscrits des compositeurs ne sont pas seuls à être parfois illisibles, les *patches* peuvent le devenir rapidement. Le réalisateur en informatique musicale Olivier Pasquet nous a fait part du *patch* ci-dessous (figure 7), qui est tout à fait illisible. La plupart des modules n'atteignent pas ce degré de complexité, mais tous les « lecteurs » de *patches* soulignent qu'il est en général très difficile de parcourir un *patch*, même son propre *patch*, quelques mois après sa confection. La principale raison est la simplicité de mise en œuvre du logiciel : l'utilisateur opère principalement par copier-coller de *patches* existants, accumulant objets et connexions dans une même fenêtre. Si chaque groupe d'objet est, à l'instar du *patch* initial de la figure 2, très simple à saisir, leur entassement rend l'ensemble non seulement difficile à comprendre mais quasiment illisible.

³¹ Alain BONARDI, Jérôme BARTHELEMY, Guillaume BOUTARD, Raffaele CIAVARELLA, « Préservation de processus temps réel. Vers un document numérique », *Document Numérique*, 11, 3-4 (2008), p. 59-80.

³² Jamie BULLOCK, Lamberto COCCIOLI, « Modernising Live Electronics Technology in the Works of Jonathan Harvey », *Proceedings of the 2005 International Computer Music Conference*, Barcelona, Spain, 2005.

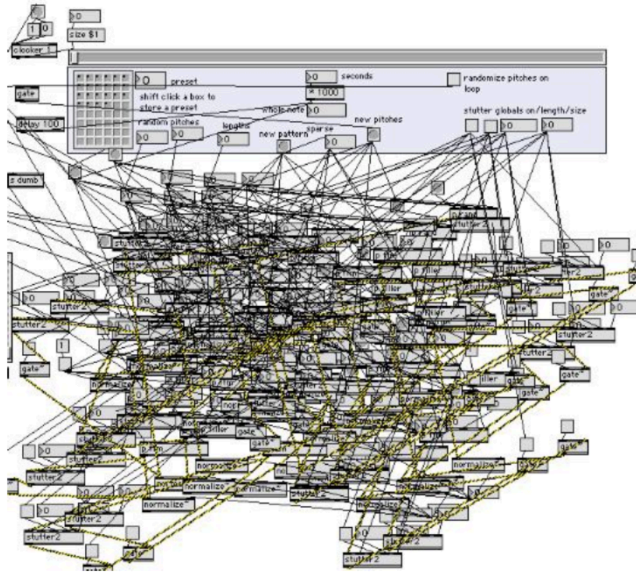


Figure 7. Exemple de patch complexe (source : Olivier Pasquet).

Les chercheurs de l'équipe Services en Ligne à l'IRCAM ont créé des outils permettant d'analyser les *patches* déjà développés, et d'en donner des vues synthétiques, par exemple l'application *PatcherTools*, dont les figures 8 et 9 montrent l'utilité. En effet, un *patch* est en réalité une arborescence de traitements, arborescence cachée par le logiciel même, qui n'en donne aucune représentation. Dans le cadre par exemple d'une démarche génétique de mise en perspective de versions d'un même *patch*, il est très important de pouvoir comparer les arborescences associées pour établir la croissance de ce *patch* dans le temps.

Philippe Manoury
En Echo
Msp Version 1

0 1 2 3 4 5 6 7

riviere jardin stop trevor broad photo betty visage table

p Scores

p Control
p Simul
p DSP
p Display

HDSP9652
SR 44.1
IOVS 512
sVS 64
overdrive / AudioInterrupt ON
optimize ON

next event (tab)

goto : 0

p Vas-y!

follower on check pitch

1 0

patcher pitch_in

11 C-2

pch

r exp-sectout 0

r EVT 0

r exp-note

dsp on/off (esc)

0. % cpu

0

s dto4

Figure 8. Copie d'écran du patch de En Echo, de Manoury, pour soprano et électronique temps réel (source : Serge Lemouton).

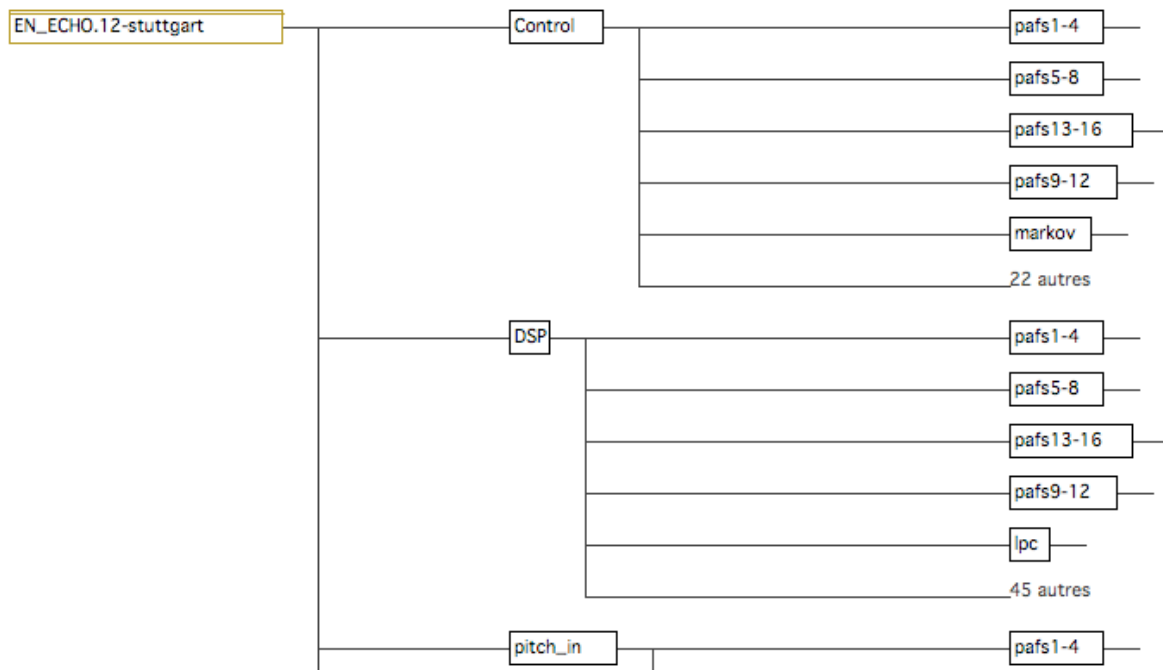


Figure 9. Copie d'écran du haut de la structure du patch de En Echo, de Manoury, obtenue dans le logiciel PatcherTools (source : Alain Bonardi).

Sur la figure 8 est présenté le *patch* de *En Echo*, de Manoury, tel qu'ouvert dans *Max/MSP*. Les différents modules de traitement de la voix de la soprano et de synthèse sonore sont disposés dans des *sous-patches* et donc non visibles au niveau du patch principal. Par exemple, pour trouver les modules de synthèse sonore par formant (*PAFs* ou *phase-aligned formants*), l'utilisateur doit double-cliquer sur l'objet *p DSP* (sur le côté gauche), qui est un *sous-patch*. Le parcours du patch dans son ensemble s'avère très fastidieux. Un outil de cartographie de patches, *PatcherTools*, a donc été proposé par l'équipe Services en Ligne de l'Ircam. La figure 9 montre le résultat obtenu dans ce logiciel : une visualisation du patch avec une profondeur de deux niveaux dans l'arborescence. Il est possible de raffiner le niveau de détail et de « zoomer » / « dé-zoomer » sur telle ou telle partie.

La modélisation de la provenance, du contexte et de l'authenticité de l'œuvre

Dans la production contemporaine d'œuvres « temps réel », la partie électronique est le résultat d'un processus complexe d'élaboration impliquant le compositeur, le réalisateur en informatique musicale, mais aussi les développeurs de modules logiciels spécifiques implémentés dans le *patch* de l'œuvre, ou encore l'ingénieur du son responsable du positionnement des haut-parleurs dans le lieu de concert. Les rôles des différents intervenants au cours des différents processus conduisant à l'exécution ou la reprise de l'œuvre sont difficiles à séparer et identifier. Toutefois, une formalisation minimale de la provenance, c'est-à-dire au moins des intervenants, de leurs rôles, de leur contribution, est à entreprendre pour contribuer à la pérennité de l'œuvre.

Dans le projet CASPAR, nous avons choisi d'exprimer les concepts de provenance et de contexte, en les modélisant sur la base des ontologies muséographiques CIDOC CRM³³ et de leur extension bibliographique FRBRoo³⁴. En informatique et en sciences de l'information, une ontologie est l'ensemble structuré des termes et concepts représentant le sens d'un champ

³³ Modèle CIDOC CRM, http://cidoc.ics.forth.gr/docs/cidoc_crm_version_4.2.2.pdf

³⁴ Modèle FRBRoo, http://cidoc.ics.forth.gr/docs/frbr_oo/frbr_docs/FRBR_oo_V0.8.3.pdf

d'informations, que ce soit par les métadonnées d'un espace de noms, ou les éléments d'un domaine de connaissances. L'ontologie constitue en soi un modèle de données représentatif d'un ensemble de concepts dans un domaine, ainsi que des relations entre ces concepts. Elle est employée pour raisonner à propos des objets du domaine concerné. Les concepts sont organisés dans un graphe dont les relations peuvent être des relations sémantiques ou des relations de subsomption (inclusion). La figure 10 montre une telle ontologie, modélisant la provenance et le contexte de l'opéra *Avis de Tempête* de Georges Aperghis, pour soprano, baryton, dispositif électronique et ensemble (2004). Il s'agit de modéliser l'œuvre à travers ses composantes (livret, œuvre vidéo, œuvre musicale) et ses manifestations (partition, livret, vidéo, patches, etc.), ce qui permet par exemple de gérer les versions, le contexte et la provenance. Sur la partie gauche de la figure, les œuvres comme concepts dans la colonne de gauche sont séparées de leurs « expressions ». En bas de la figure, l'expression de la musique est décomposée selon les éléments suivants : la partition, le patch Max/MSP et les échantillons sonores utilisés. On précise de plus le support de ces derniers (cartouche Syquest). L'approche selon des nœuds d'information œuvre/réalisation/intervenant permet de modéliser la provenance et le contexte.

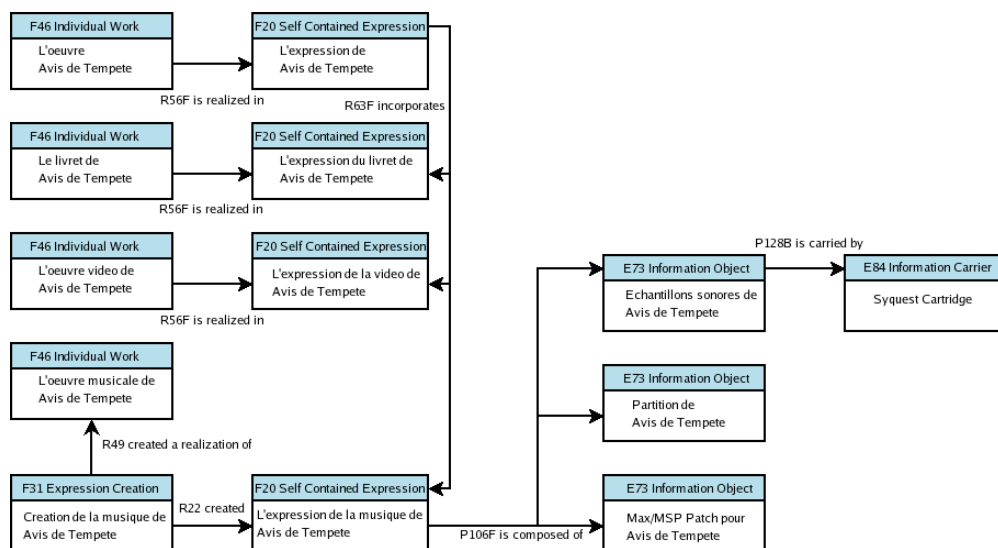


Figure 10. Une partie de la modélisation de la provenance et du contexte pour l'œuvre Avis de Tempête de Georges Aperghis (source : Guillaume Boutard).

Quant à l'authenticité³⁵, que nous évoquions dans un paragraphe précédent, elle doit être gérée comme un processus qui doit être défini le plus en amont possible ; ce processus devra s'appuyer sur des protocoles à exécuter à plusieurs étapes du cycle de vie de l'œuvre. Donnons un exemple de protocole d'authenticité correspondant à la migration d'un effet audio, passant d'une réalisation AFX1 à AFX2 ; il comporterait deux phases. La première serait la phase d'acquisition d'information avec les étapes suivantes : tout d'abord, la définition d'échantillons sonores de test unitaire en entrée ; puis leur application en entrée de l'effet AFX1 et le stockage du résultat audio ; enfin la conception des critères de comparaison utilisés lors de l'exécution avec AFX2. La seconde phase serait la phase de migration, au cours de laquelle on appliquerait les fichiers de test unitaire à AFX2, puis on stockerait les résultats que l'on comparerait à ceux d'AFX1 en employant les critères définis lors de la première phase.

La « virtualisation » de patches

Une des premières idées venant à l'esprit lorsque l'on se penche sur ces œuvres avec électronique temps réel est d'essayer d'exprimer les traitements effectués de manière indépendante du codage

³⁵ Mariella GUERCIO, Barthélemy, Jérôme, Bonardi, Alain, « Authenticity Issue in Performing Arts using Live Electronics », *Proceedings of the 4th Sound and Music Computing Conference (SMC 07)*, Lefkada, Grèce, 2007, p. 226-229.

proposé par les logiciels *Max/MSP* ou *PureData*. Elle ne passe pas nécessairement par l'informatique, si l'on considère la démarche du compositeur Pierre Boulez et de son réalisateur en informatique musicale Andrew Gerzso dans une œuvre comme *Anthèmes 2* pour violon et électronique (1997), tentant d'utiliser la notation musicale pour indiquer les traitements effectués : la figure 11 montre plusieurs portées placées sous la partie de violon, indiquant chacune l'action réalisée par un traitement, par exemple une ligne pour la réverbération infinie (notée *Inf. Rev.* pour « réverbération infinie »), une pour le *sampler IR* ou échantillonneur avec réverbération infinie, etc. Il s'agit de rendre compte de procédés du traitement du signal par la notation musicale du résultat escompté, lorsque ce dernier peut s'exprimer sous la forme de hauteurs identifiables dans un système de notation. Ce type de partition suscite deux commentaires : d'une part, il suppose que les effets électroniques utilisés sont connus universellement : de même que tout le monde sait ce qu'est un violon, tout le monde doit connaître ce qu'est une réverbération infinie ; d'autre part, cette partition ne note pas les actions à effectuer sur les appareils mais donne seulement une indication du résultat escompté.

anthèmes 2
pour violon et dispositif électronique (1997)

pierre boulez
(*1925)

Libre
Brasque
(♩ = 92)

♩ = 92 *rall.* (♩ = 66)

Violon

Spacialization F -11/-18/-18/2.0

Inf. Rev.
reverb. time 60"

Spacialization F -11/-18/-18/2.0

Sampler IR
reverb. time 60"

Spacialization F -11/-18/-18/2.0

Sampler
reverb. time 60"

Spacialization MR 4-12/-24/2.0

Freq. Shift

Violon

Spac. 10 11

IR

Spac.

S-IR

Spac.

FS
200 Hz 100 Hz 200 Hz 200 Hz

Spac. ML 4-12/-24/2.0

Figure 11. La première page de la partition d'Anthèmes 2, de Pierre Boulez, pour violon et électronique temps réel. Avec l'aimable autorisation des Editions Universal, Vienne, Autriche.

Mais si nous demeurons dans un environnement informatique, comment décrire automatiquement et de manière abstraite ce que réalise tel ou tel module, indépendamment du logiciel qui l'encode ? L'idée centrale de la solution développée notamment par Raffaele Ciavarella et Jérôme Barthélemy à l'IRCAM, est de convertir les patches en des programmes virtuels exécutables pour un « ordinateur virtuel universel » (UVC, *Universal Virtual Computer*), dont le concept a été développé à l'origine par Raymond Lorie³⁶. L'hypothèse sous-jacente est que la description de cet UVC est suffisamment simple pour pouvoir être appréhendée sur une vaste période historique. Dans la mesure où l'UVC peut être appliqué sur n'importe quel ordinateur, les programmes sur l'UVC deviennent indépendants de la technologie.

Un programme UVC peut être écrit pour décoder tout format de fichier spécifique. Des décodeurs ont déjà été développés pour les formats *Max/MSP* et *PureData*. Pour que cette solution soit utilisable pour l'interprétation des patches, un traducteur (décodeur) a été développé qui tourne sur l'UVC et traduit le patch en question en une « vue logique de données » (LDV, *Logical Data View*). La LDV est une description structurée de l'objet numérique, générée selon un schéma donné, le « schéma logique de données » (LDS, *Logical Data Schema*).

Pour analyser ou exécuter l'objet dans dix, quinze, ou même cent ans, un émulateur UVC pourra être écrit qui fera tourner un décodeur *Max/MSP* ou *PureData*, générant une LDV. Puisque le LDS est lui aussi préservé, les futurs programmeurs seront capables de comprendre la LDV et de développer des visualiseurs ou des interpréteurs.

Pour séduisantes et nécessaires qu'elles soient, ces approches ne résolvent pas complètement la question de la préservation, puisqu'il reste toujours un « traducteur » automatique dont il faut d'une part garantir la lisibilité de la représentation et d'autre part vérifier la cohérence et la complétude, pour assurer une authenticité minimale.

Prospective : vers la constitution d'organologies des modules de traitement temps réel du son

La constitution d'organologies des modules de traitement temps réel du son est un enjeu de recherche important, qui est loin d'être atteint actuellement. La question est au cœur du projet ASTREE³⁷ (Analyse et Synthèse des Traitements Temps Réel) qui a démarré début 2009. Nous employons volontairement le pluriel pour le mot « organologies », car nous pensons que les relations qui pourront être établies entre modules de traitement du son seront démultipliées et devront être appréhendées selon plusieurs plans.

L'organologie traditionnelle associe un système d'écriture musicale à un type d'interface homme-machine et un type de production sonore, en constituant des familles : la famille des cordes frottées de l'orchestre (violon, alto, violoncelle, contrebasse) fournit un exemple d'interface commune (un archet frottant des cordes sur un manche) avec un mode de production sonore commun. L'ensemble couvre tout le spectre, du grave à l'aigu, et s'organise en général selon le système tempéré.

A l'opposé, dans le monde des *patches*, il n'existe pas de système d'écriture musicale commun à plusieurs « modules ». Les interfaces d'intervention (potentiomètres, interrupteurs, etc. stylisés à l'écran) reprennent des éléments de base communs, mais chaque patch les articule différemment.

³⁶ Raymond LORIE, « Long term preservation of digital information », *Proceeding of the First ACM/IEEE-CS joint conference on Digital Preservation*, Roanoke, Virginia, United States, 2001, p. 346-352.

³⁷ Le projet ASTREE est financé par l'Agence Nationale de la Recherche, et associe l'IRCAM, le GRAME à Lyon, le Centre de Recherche en Informatique de l'Ecole des Mines de Paris, et le laboratoire CIEREC de l'Université de Saint-Etienne.

Chaque *patch* propose un mode instrumental propre. De plus, une même transformation du son peut être représentée par des *patches* fort différents : il existe ainsi de nombreuses variantes de réverbérations.

Il reste néanmoins fondamental de chercher, avec l'aide des ordinateurs, à établir des invariants, même s'ils n'ont qu'un périmètre de pertinence limité. Nous pouvons imaginer que de nombreuses techniques informatiques dans le domaine de l'ingénierie des connaissances, concernant notamment l'apprentissage et les représentations imprécises, pourront être mises au service de l'établissement de relations et filiations entre modules de transformation électronique, posant des bases pour la génétique de l'œuvre musicale avec électronique temps réel. Dans ce sens, le projet ASTREE propose de transcoder des *patches* existants en FAUST, qui est un langage de programmation dédié à la modélisation mathématique et la création de processeurs de flux (en quelque sorte la *lingua franca* des *patches*), puis d'analyser et classifier ces représentations.

Conclusion

En examinant les œuvres musicales avec électronique temps réel, nous pourrions penser que leur pérennité pose plus de difficultés que celle des manuscrits médiévaux. Chacune de ces œuvres interactives est en fait un système en soi, usant à la dérobée des catégories musicales et informatiques, sans toutefois s'établir sur un système *a priori*. Ce régime si particulier ouvre toutefois une large perspective à des approches génétiques intimement associées à la maintenance informatique de l'œuvre, en amont de celles menées par les musicologues. Le réalisateur en informatique musicale ne cesse d'interpréter – en concert –, et de ré-interpréter – sur son ordinateur – les *patches* mettant en œuvre le traitement temps réel des sons. Si le *patch* est stocké selon un format informatique, il n'en constitue pas pour autant un objet informatique structuré, car il encode des connaissances sans les représenter, et aucun traitement de « génétique automatique », spécifiant un programme qui traiterait d'autres programmes, ne pourrait donner de résultat satisfaisant. Dans ce domaine, les approches intensives doivent céder le pas face aux approches extensives. Il est donc nécessaire d'aborder les traitements selon des dynamiques homme-machine, dont nous avons présenté quelques exemples : l'aide à la relecture du *patch*, la modélisation de la provenance et du contexte de l'œuvre, la virtualisation de *patches* et la constitution d'organologies des modules de traitements temps réel du son. Si ces démarches peuvent paraître bien spécialisées, et réservées à un type d'œuvre bien particulier, il n'est pas exclu qu'elles ne débordent un jour de leur cours naturel, et intéressent la critique génétique, à l'heure où les manifestations numériques de l'œuvre ne cessent d'en interroger les fondements.

Remerciements

A Jérôme Barthélemy, Guillaume Boutard, Raffaele Ciavarella, Serge Lemouton, Olivier Pasquet.

Références bibliographiques

Larry AUSTIN, « John Cage's William Mix (1951-3): the restoration and new realisations of and variations on the fist octophonic, surround-sound tape composition », *A Handbook to Twentieth-Century Musical Sketches*, (Patricia Hall et Friedemann Sallis, éditeurs), Cambridge (UK) : Cambridge University Press, 2004.

Bruno BACHIMONT, *Ingénierie des connaissances et des contenus*, Paris : Hermès-Lavoisier, 2007.

Paul BADURA-SKODA, *Etre musicien*, Paris : Hermann, 2007.

Jean-Baptiste BARRIERE, « Ecriture musicale et ordinateur. Notes sur la genèse », *Genesis*, 4 (1993), p. 11-22.

Curtis ROADS, *L'audio numérique – Musique et informatique*, traduction et adaptation française : Jean de Reydellet, Paris : Dunod, 2007, 658 pages.

Alain BONARDI, Jérôme BARTHELEMY, « The preservation, emulation, migration, and virtualization of live electronics for performing arts: An overview of musical and technical issues », *Journal on Computing and Cultural Heritage*, 1, 1 (2008), p. 6 :1-6 :16.

Alain BONARDI, Jérôme BARTHELEMY, Guillaume BOUTARD, Raffaele CIAVARELLA, « Préservation de processus temps réel. Vers un document numérique », *Document Numérique*, 11, 3-4 (2008), p. 59-80.

Hannah BOSMA, « Documentation and Publication of Electroacoustic Compositions at NEAR », *Proceedings of the 2nd Electroacoustic Music Studies Network International Conference (EMS 05)*, Montréal, Canada, 2005.

John CHOWNING, « The synthesis of complex audio spectra by means of frequency modulations », *Journal of the Audio Engineering Society (J.A.E.S.)*, 21, 7 (1973)

Marc-André DALBAVIE, « De l'écriture au timbre », *Actes du Symposium Systèmes Personnels et Informatique Musicale*, Paris, Ircam, 1986.

Martin DOERR, Patrick LEOEUF, « Modelling Intellectual Processes: The FRBR - CRM Harmonization », *Proceedings of the Second DELOS Conference on Digital Libraries*, 2007.

Ross FELLER, « E-sketches: Brian Ferneyhough's use of computer-assisted compositionnal tools », *A Handbook to Twentieth-Century Musical Sketches*, (Patricia Hall et Friedemann Sallis, éditeurs), Cambridge (UK) : Cambridge University Press, 2004, p. 176-188.

Mariella GUERCIO, Barthélemy, Jérôme, Bonardi, Alain, « Authenticity Issue in Performing Arts using Live Electronics », *Proceedings of the 4th Sound and Music Computing Conference (SMC 07)*, Lefkada, Grèce, 2007, p. 226-229.

Serge LEMOUTON, Raffaele CIAVARELLA, Alain BONARDI, « Peut-on envisager une organologie des traitements sonores temps réel, instruments virtuels de l'informatique musicale? », *Proceedings of the Fifth Conference on Interdisciplinary Musicology (CIM09)*, Paris, 2009.

Jerrold LEVINSON, « Performative versus Critical Interpretation in Music », *The Pleasures of Aesthetics, Philosophical Essays*, Cornell, 1996.

Michael LONGTON, *Record Keeping Practices of Composers*, rapport d'enquête, site Internet du projet InterPares 2, disponible à <http://www.interpares.org>, lien vérifié le 28 septembre 2009.

Raymond LORIE, « Long term preservation of digital information », *Proceeding of the First ACM/IEEE-CS joint conference on Digital Preservation*, Roanoke, Virginia, United States, 2001, p. 346-352.

Philippe MANOURY, *La note et le son – Ecrits et entretiens, 1981-1998*, Paris : L'Harmattan, 1998.

Philippe MANOURY, « Considérations [toujours actuelles] sur l'état de la musique en temps réel », *Revue l'Étincelle*, Perspectives, Paris : Ircam – Centre Georges Pompidou, 2007.

Documentation en ligne

Modèle CIDOC CRM, http://cidoc.ics.forth.gr/docs/cidoc_crm_version_4.2.2.pdf

Modèle FRBRoo, http://cidoc.ics.forth.gr/docs/frbr_oo/frbr_docs/FRBR_oo_V0.8.3.pdf