

Numéros / n° 5 - Informatique et musique : Recherche et Création 1

« L'analyse orientée objet-opérateur de *Due di Uno* d'Agostino di Scipio pour la modélisation des fonctions de contrôle non linéaires dans *La Philosophie du temps* »

João Svidzinski et Alain Bonardi

Résumé

Dans cet article, nous allons montrer comment l'analyse de la pièce *Due di Uno* (2002 - 03) d'Agostino di Scipio nous a conduits à la composition d'une nouvelle pièce originale. Nous proposons, dans un premier temps, l'analyse par modélisation orientée objet-opérateur comme un modèle d'analyse créative faite par un compositeur, démarche s'appuyant sur les fondements théoriques posés par Horacio Vaggione.

Ensuite, nous interprétons les codes informatiques de la pièce *Due di Uno* en mettant en évidence un objet de contrôle algorithmique non linéaire, pour finalement l'émuler dans la composition d'une nouvelle pièce par l'un des auteurs en tant qu'analyste-compositeur, *La Philosophie du temps* pour piano et électronique en temps réel.

1. Introduction

L'analyse par modélisation orientée objet-opérateur est une démarche analytique créative en contexte numérique dont le but est d'achever un processus compositionnel en permettant à l'analyste-compositeur de réaliser une ou plusieurs compositions originales. Il s'agit donc d'une approche théorique réalisée par un compositeur. Il est essentiel qu'une analyse soit fertile en fournissant des données utiles à l'activité musicale. Cette exigence est donc au centre de notre démarche théorique déchiffrement d'une oeuvre musicale numérique en cherchant le système régissant l'organisation de l'oeuvre étudiée, partant du code numérique original comme source analytique principale, pour susciter la réflexion de

l'analyste-compositeur afin de le mener à un processus compositionnel créatif et original.

Le maniement de l'approche par objet-opérateur, héritée de la pensée d'Horacio Vaggione, considère l'analyse de la pièce *Due di Uno* (2002) pour flûte à bec, violon et électronique temps réel du compositeur italien Agostino Di Scipio né en 1962. La pièce analysée illustre la démarche assez particulière de Di Scipio. L'interaction acoustique/électronique selon un principe algorithmique lié à un système non

linéaire interactif est le concept principal de cette composition.

La pièce analysée a été composée sur la plateforme KYMA. Comme nous n'avons pas accès au patch original, le compositeur nous a aimablement fourni un *flow chart* de l'électronique temps-réel contenant les modules de traitement numérique ainsi que leurs paramètres. L'approche orientée objet-opérateur prévoit l'analyse du code (dans ce cas du *flow chart*) en l'interprétant comme un réseau opératoire où chaque objet ? matériau numérique codifié ? est opéré dynamiquement dans un réseau d'interactions.

Dans cet article nous allons d'abord aborder la méthodologie appliquée dans le cadre de la modélisation orientée objet-opérateur de *Due di Uno*, en particulier son système de contrôle, avec l'emploi d'une fonction non linéaire, qui aboutira *in fine* à la composition de *La Philosophie du temps*. Nous nous intéresserons notamment aux approches à travers lesquelles un compositeur peut se servir

d'une analyse

musicale pour composer sa propre musique, de façon à ce que la pièce résultante soit originale et non une simple version ou citation de l'oeuvre analysée.

2. Modélisation orientée objet-opérateur

Nous avons présenté les premiers résultats de cette démarche dans nos publications récentes (Svidzinski, 2013, 2014, 2015). Dans le cadre de notre recherche en master (2013), nous avons analysé la pièce *Circulos Ceifados* avec, comme matériel analytique, les codes Csound et les publications que le compositeur Rodolfo Caesar a livrées dans un livre (2008) pour interpréter les modules opératoires dans un réseau d'interactions. Le modèle généré par analyse a ensuite été utilisé par João Svidzinski en tant que compositeur-analyste pour la composition de trois pièces *Turdus* (2013), *M1910* (2014) et originales

Les Ames remerciées (2014). Dans un premier temps, nous avons qualifié cette méthodologie de « modélisation opérationnelle » ; dans cet article, nous préférons parler de modélisation orientée objet-opérateur pour nous rapprocher étymologiquement de Vaggione ainsi que de Gilles-Gaston Granger (1994). Vaggione précise sa notion d'objet-opérateur en citant Granger et conclut :

Lorsque nous parlons d'objet [...] il ne s'agit pas nécessairement d'objets sonores, mais d'opérations diverses de production musicale qui se trouvent assemblées dans une entité opératoire : quelque chose qu'on peut nommer, manipuler, faire circuler dans un réseau. (Vaggione et al., 2003).

C'est dans cette perspective opératoire que nous concevons la modélisation présentée dans cet article : l'objet logiciel comme catégorie opératoire.

L'approche compositionnelle objet fut proposée par Horacio Vaggione dans l'article *On object-Based Composition* de 1991. Suivant cette approche, ce compositeur-chercheur a composé les pièces *Fractal C* (1983-1984) pour ordinateur ? bande magnétique 16 pistes *Thema* (1985) pour saxophone basse et bande et *Ash* (1989-1990) électroacoustique. Le coeur de la théorie est le double concept d'objet et d'opération : le premier consiste en une « entité dotée de modalités de comportement spécifiques (méthodes), déterminés de manière numérique (codes) » (Budon et Vaggione, 2007) ; les opérations sont les tâches appliquées aux objets. Le rapport objet-opération est donc indissociable. Selon Vaggione l'objet est une « catégorie opératoire ». En bref, les objets sont tous les matériaux utilisés par le compositeur lors du processus créatif, et ceux-ci sont mis en relation systémique dans un réseau opératoire. Vaggione

attribue à ces objets les mêmes propriétés que dans les langages de programmation orientés objet ⁽¹⁾ : la clôture (encapsulations des propriétés d'un objet), l'héritage (l'héritage des propriétés d'un objet à des sous-objets) et le polymorphisme (objets différents recevant des messages identiques qui produisent des résultats différents).

Dans une publication plus récente (Vaggione, 2010), Vaggione se penche spécifiquement sur les représentations musicales numériques de l'approche orientée-objet, c'est-à-dire l'aspect technique de la composition musicale numérique du point de vue opératoire. Le compositeur considère les patches graphiques (comme dans le logiciel Max) « comme des ensembles de scripts alphanumériques encapsulés dans des objets graphiques qui constituent leurs abstractions » : plusieurs patches (ou sous-patches) concourant à une même fonction peuvent être assemblés dans une unité, appelée module, qui possède un rôle spécifique (par exemple, un module de traitement granulaire). L'ensemble des modules correspond, selon Vaggione, « à la totalité d'un dispositif donné ». Les modules sont donc des unités opératoires dont le fonctionnement s'établit à l'intérieur d'un réseau d'interactions. Un compositeur-analyste qui désire analyser une pièce faisant appel à l'informatique musicale cherche plutôt à comprendre comment la pièce fonctionne. Le réseau d'interactions (l'ensemble d'un patch) peut donc contenir des informations capitales pour une approche analytique créative. Comme le dit Jean-Claude Risset « l'analyse est un moyen de s'apprendre à composer : il est important de pouvoir étudier de telles "partitions" si l'on s'intéresse à composer le son lui-même. » (Risset, 2001).

Dans une approche analytique orientée objet, il est possible d'isoler un module (même s'il dépend de l'ensemble du réseau et des interactions) pour comprendre son fonctionnement. En vue de l'analyse et du recodage de la pièce *En Echo* (1993-94) de Philippe Manoury, dans le cadre du projet ANR ASTREE (Bonardi, 2011), Alain Bonardi a suivi les étapes suivantes pour étudier le module *harmoniseur* du patch : localiser et extraire le module, tester, documenter, modéliser (modélisation en langage Faust avec des améliorations), greffer la nouvelle version du module dans le patch. Ensuite, une nouvelle version du

patch a été diffusée, avec un module actualisé dont le rendu sonore est plus clair et qui diminue la consommation du processeur informatique. Dans un second temps, au niveau créatif, la modélisation de

ce module joue un rôle important dans le cycle *Pianotronics* du compositeur-analyste (Bonardi, 2015).

Pour l'analyse de *Due di Uno*, nous allons suivre une démarche similaire et établir le réseau opératoire en classant les objets selon les fonctions technico-musicales usuelles pour ensuite les analyser individuellement et établir leur potentiel créatif.

3. Analyse par modélisation orientée objet opératoire *Due di Uno* d'Agostino Di Scipio

3.1. Présentation de l'oeuvre

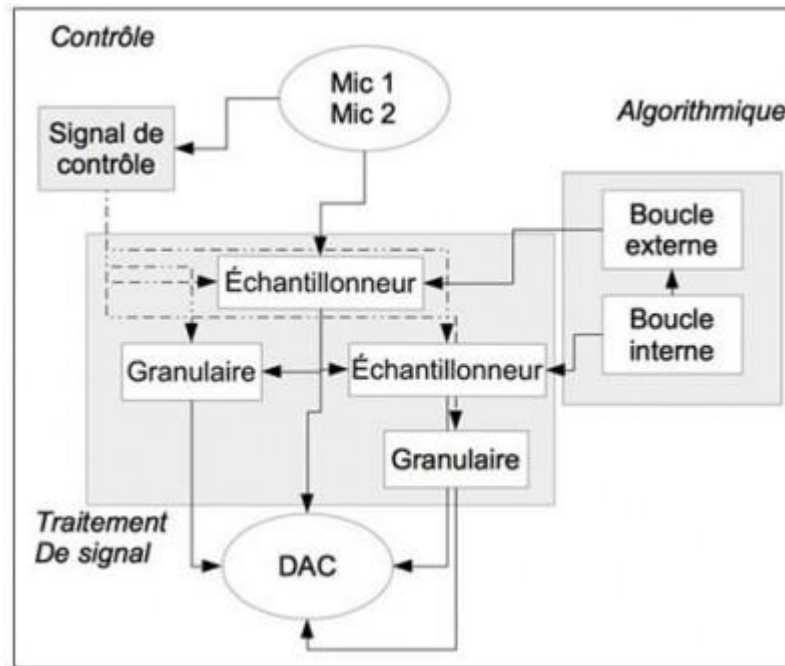
Due di Uno pour flûte à bec, violon et électronique fut composée par Agostino Di Scipio à l'occasion de l'anniversaire des 60 ans de Horacio Vaggione⁽²⁾. Le compositeur définit la pièce comme suit :

Due di Uno est une composition pour flûte à bec piccolo (sopranino en fa), violon et électronique en direct. Comme certaines autres de mes compositions, cette pièce représente une tentative d'élaborer et de mettre en scène une petite infrastructure technologique dont le résultat audible est la musique. C'est un essai de composition algorithmique ; simultanément, cette pièce implémente un échange dynamique entre les instruments et l'électronique en direct. Plus important encore, il s'agit essentiellement d'un travail dans le son ? c'est-à-dire d'une composition où le timbre, la polyphonie de textures et la densité ainsi qu'un ensemble d'artefacts bruiteux constituent des dimensions perceptuelles plus importantes que d'autres et sont traités créativement sous la contrainte du temps-réel (Di Scipio, 2007).

Notre modélisation du réseau d'objets de *Due di Uno* est schématisée ci-dessous (figure 1.). Il s'agit de trois catégories d'objets opératoires (boîtes en gris) : contrôle, algorithmique et traitement de signal. L'algorithme génère les données pour les paramètres de contrôle du traitement de signal (échantillonneur et granulaire), comme la durée, l'amplitude et la fréquence. Les signaux de contrôle sont générés par un suiveur d'amplitude des deux microphones placés sur les instruments acoustiques. Ces signaux (lignes pointillées) contrôlent les paramètres du traitement sonore, comme le niveau de sortie, et servent à *mapper* les paramètres générés par l'algorithmique. Le concept prédominant dans *Due di Uno* est l'objet

algorithmique : deux algorithmes identiques fonctionnent simultanément, l'un à une méso-échelle (boucle externe) et l'autre à une micro-échelle (boucle interne). Une fonction sinusoïdale récursive alimente le processus en boucle. Les paramètres de sortie de la boucle externe alimentent le traitement sonore aussi bien que la boucle interne. La boucle externe se répète dix fois (pour chaque section de la pièce) et à l'intérieur de cet objet, la boucle interne se répète également dix fois.

Figure 1. Schéma du réseau opératoire d'*Due di Uno*



Source : auteurs compositeur (Di Scipio,

; à partir du schéma original du cit2007, P. 304)

La pièce est divisée en dix sections de quarante secondes chacune, plus une coda d'environ quatre-vingts secondes. La partition instrumentale est le résultat d'un processus algorithmique similaire à celui décrit

ci-dessus, la structure contient deux boucles, une externe et une interne qui sont répétées chacune dix fois. À l'issue de ce processus, les durées, les temps de départ et les amplitudes de chaque note-événement sont générés. Concernant les hauteurs, le compositeur a utilisé un processus simple fait à la main alternant les notes *Mi aigu* et *si bémol* la plus grave disponible. Di Scipio procède également à des interventions manuelles, surtout pour la génération des ornements, glissandos et trilles et des changements selon les propriétés de chaque instrument. À l'exception de la coda, les partitions du violon et de la flûte à bec sont quasiment identiques (figure 2), la seule différence tient à des raisons idiomatiques qui joueront un rôle important pour l'interaction avec l'électronique (voir ci-dessous).

Figure 2. Premières mesures de la partition *Due di Uno*. En haut le violon, en bas la flûte à bec



Source : Di Scipio, partition *Due di Uno* (ouvrage non publié)

L'un des concepts de la pièce est l'obtention de résultats sonores particuliers qui émergent d'une structure à une micro-échelle. Pour cela, le compositeur fait appel à un algorithme (comme décrit ci-dessus) contenant une fonction de contrôle non linéaire récursive. Étant donné que l'algorithme de la composition instrumentale est similaire à celui du contrôle de l'électronique (3), la convergence entre l'acoustique et électronique est ainsi garantie par les traces génétiques d'un même processus algorithmique.

La partition instrumentale est générée grâce à un processus algorithmique déterministe. L'algorithme génère les paramètres des notes-événements à une échelle macro-temporelle, à savoir la durée et l'amplitude de la note. Cependant, le compositeur joue avec le fait que les deux instrumentistes ne vont jamais jouer rigoureusement simultanément, bien que ni la partition ni le compositeur ne demandent aux instrumentistes de profiter d'un probable décalage temporel. Di Scipio compte donc sur une

« circonstance », une « fatalité » du monde instrumental. Il transforme une contrainte musicale en un fait opératoire et par conséquent musical. La partition instrumentale est déterministe, même si le résultat

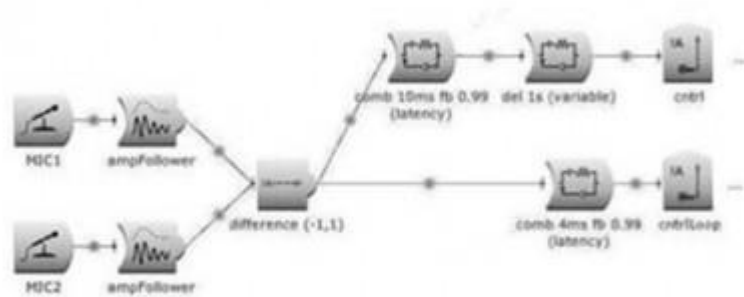
sonore n'est pas déterminé ; les modes de jeux utilisés par la flûte, ainsi que par le violon, sont pris en compte pour l'accomplissement du résultat sonore final attendu. Les trilles (à la fin de chaque section et

puis au début de 6 à 10) sont stratégiquement placés pour donner à chaque section la même durée de dix secondes.

Nous avons divisé en deux parties l'analyse de *Due di Uno* : d'abord la reconstruction de la partie électronique contenant les opérations numériques en Max à partir d'un *flow chart* fourni par le compositeur ; ensuite l'élaboration du réseau opératoire.

Le *flow chart* décrit les processus de traitement sonore numérique par des schémas contenant les modules et leurs paramètres (figure 3). Le document est divisé en onze *flow charts* qui fonctionnent simultanément.

Figure 3. Flow chart 9 de *Due di Uno*

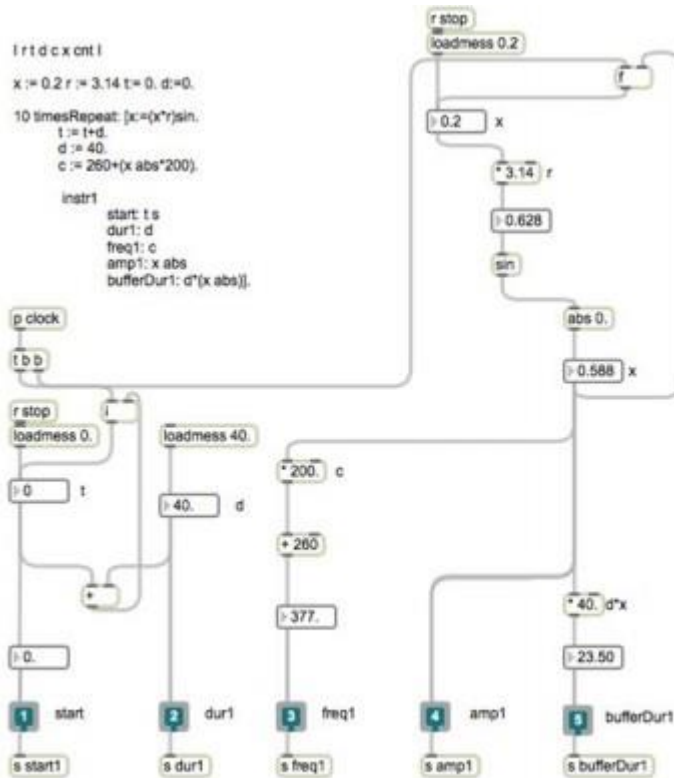


Source : Di Scipio *Due di Uno's DSP flow chart* (ouvrage non publié)

Le traitement granulaire ainsi que les autres modules de contrôle ont été modélisés et migrés vers le langage Max. Nous allons aborder les sections concernant l'objet algorithmique (*flow charts* 3 et 4) et de contrôle du signal (*flow charts* 9, 10 et 11).

L'algorithme en boucle utilisé sur deux échelles temporelles a été modélisé en Max sous la forme de deux sous-patches ci-dessous (figure 4).

Figure 4. Boucle externe de l'objet algorithmique en Max



Source : auteur

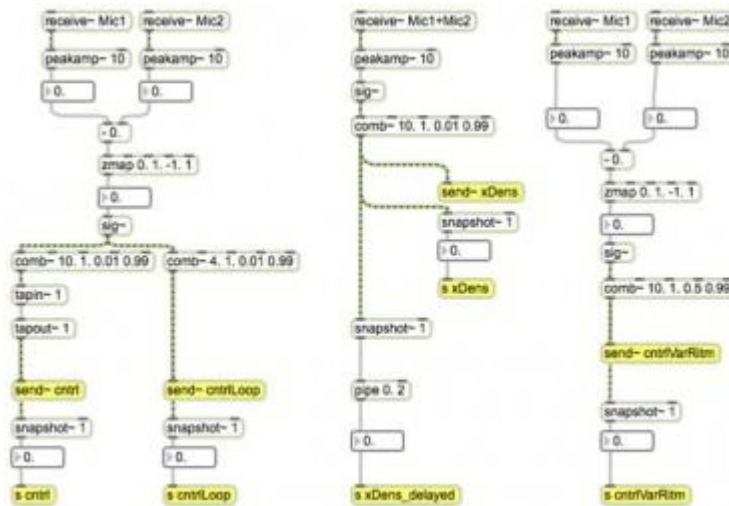
Il s'agit d'une structure (répétée dix fois pour la boucle externe et dix fois pour la boucle interne à chaque réitération de la boucle externe) qui implémente une fonction non linéaire sinusoïdale :

$$x = \sin (x * r)$$

Dans la boucle externe, x est une valeur initiale qui réinjecte le résultat de la fonction dans la boucle suivante. r est l'indice dont les valeurs déterminent le comportement des résultats (périodiques ou chaotiques par exemple). Cet algorithme génère les valeurs des paramètres pour les modules de traitement sonore : durée, fréquence, amplitude et panoramique. L'alimentation des données dans la boucle interne est fournie par les valeurs d'amplitude reçues de la boucle externe.

En sus de l'algorithme, les signaux de contrôle sont également responsables de la génération des données pour le contrôle des paramètres de la synthèse sonore et du panoramique. Il s'agit d'une série de données issues d'un *mapping* d'amplitude de deux microphones, *Mic1* pour le violon et *Mic2* pour la flûte à bec. Dans le patch Max (figure 5) sont générés cinq signaux de contrôle *cntrl*, *cntrlLoop*, *xDens_delayed*, *xDens* et *cntrlVarRitm* qui originellement ont été représentés dans les *flow chart* 9 et 10. Un même signal est utilisé pour contrôler plusieurs objets à la fois. Le signal *cntrl*, par exemple, est la somme du *Mic1*, suiveur d'amplitude du violon, et du *Mic2*, suiveur d'amplitude de la flûte à bec, il est responsable des paramètres des échantillonneurs de traitement granulaire et des valeurs des filtres passe-bande.

Figure 5. Objets de contrôle en Max



Source :auteur

L'objet algorithmique ainsi que l'objet de contrôle jouent un rôle important pour la mise en oeuvre des deux concepts principaux de la pièce : la convergence entre l'acoustique et l'électronique et l'émergence des sonorités à partir d'une structure à une micro-échelle. La fonction qui génère le résultat sur les deux supports (acoustique et électronique) est une fonction sinusoïdale récursive (comme mentionné

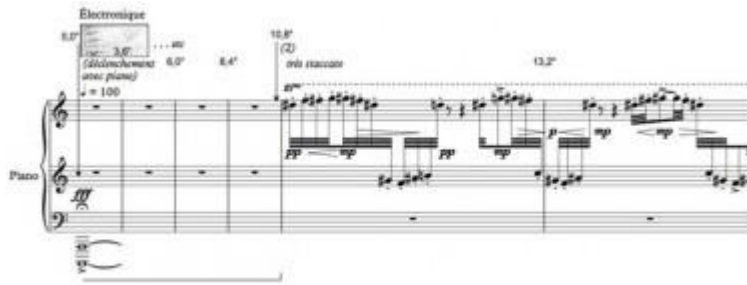
ci-dessus). L'utilisation de ce genre de système est courante dans la production compositionnelle de Di Scipio (Di Scipio, 1990, 1991, 2001). Dans la prochaine section, nous allons montrer comment l'analyse de l'application de ces fonctions a favorisé la composition d'une nouvelle musique par le compositeur-analyste, auteur de cet article.

4. *La Philosophie du temps* (2015) pour piano et électronique en temps réel

Le concept principal dans *La Philosophie du Temps* (2015) pour piano et électronique en temps réel est le contraste entre le temps linéaire et le temps non linéaire. Bien plus qu'une hypothèse ou une théorie, le titre est plutôt une invitation à une réflexion sur ce sujet. Nous ne voulons pas donner une réponse ni proposer une opposition au pied de la lettre, il ne s'agit donc que d'une réflexion sur le temps et les différentes méthodes de le générer et de l'opérer.

Le point de départ de la conception du modèle opératoire de *La Philosophie du temps* est une fonction non linéaire sinusoïdale, comme celle utilisée par Di Scipio dans *Due di Uno*. Dans un premier temps, nous avons procédé dans un environnement pour la composition algorithmique, *OpenMusic* avec la bibliothèque *OMTristan*, et ensuite la même fonction a été construite comme objet dans le logiciel Max, pour le traitement des données en temps réel. À partir de la cinquième mesure de la partition (figure 6.), un passage a été généré algorithmiquement en *OpenMusic* par l'emploi d'une fonction non linéaire pour la génération des notes et des rythmes.

Figure 6. Premières mesures de la partition de piano de *La Philosophie du temps*.



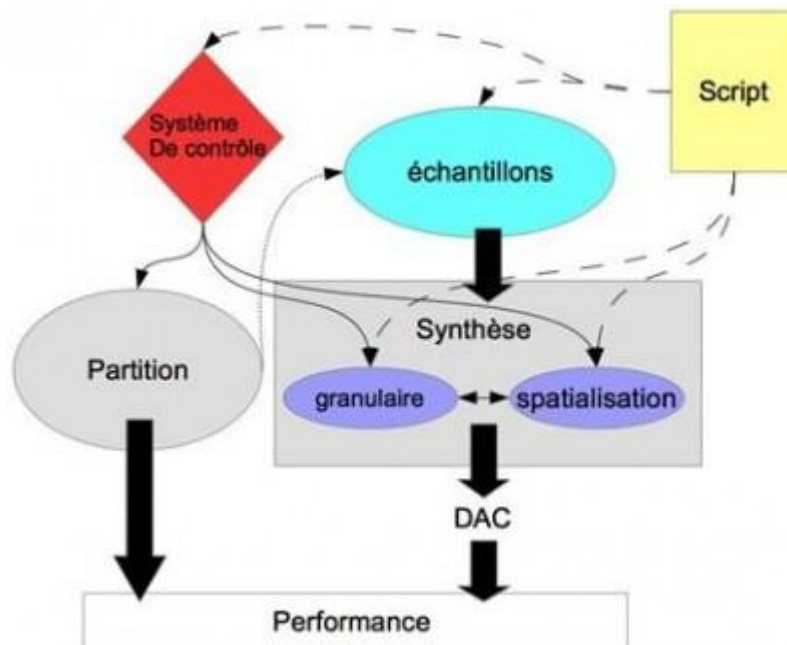
Source : auteur

La même idée a été utilisée pour le contrôle de la synthèse dans le patch Max. Nous avons choisi de travailler avec un système hybride, c'est-à-dire, avec des traitements en temps réel ainsi qu'avec des échantillons sonores préenregistrés. Deux objets musicaux de la partition acoustique (figure 6) ont été enregistrés et ont servi comme échantillons pour le traitement sonore numérique : la note grave dans la première mesure et le premier passage de la cinquième mesure.

Le réseau opératoire de *La Philosophie du Temps* est représenté sur le schéma ci-dessous (figure 7).

L'ordre des événements, ainsi que quelques paramètres sont définis par un script qui suit une temporalité linéaire chronométrique. Celui-ci ordonne les événements du système de contrôle, des échantillons et des modules de traitement numérique, notamment la synthèse granulaire et la spatialisation. Le système de contrôle, même s'il est déclenché par le script, est responsable de la génération des paramètres et attributs de la partition acoustique et des traitements numériques.

Figure 7. Réseau opératoire de *La Philosophie du temps*



Source : auteur

4.1 *La Philosophie du temps* ? émulation de l'objet de contrôle non linéaire

La méthode orientée objet (telle que théorisée par Vaggione dans un cadre compositionnel) prévoit

certaines propriétés des objets-opérateurs, ces propriétés ont été prises en compte lors de notre analyse de *Due di Uno*. Nous pouvons souligner que dans notre analyse les objets ont la propriété d'objet opératoire au sens de Vaggione (4) :

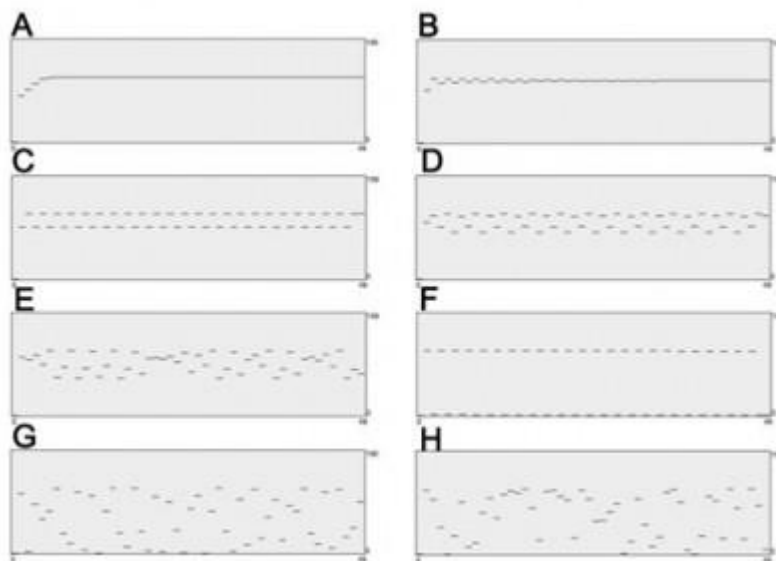
- la clôture : chaque objet est un sous-patch max, doté *di puts* ils peuvent ainsi être et *outputs* analysés individuellement, exportés et émulés dans d'autres réseaux opératoires ;
- le polymorphisme : un même signal de contrôle appliqué aux différents modules de traitement sonores numériques génère des résultats tout à fait différents ;
- l'héritage : certains attributs d'un même objet, comme par exemple l'algorithme interne, héritent des propriétés de l'algorithme externe en générant un sous-objet.

Les propriétés de clôture et d'héritage rendent possible l'émulation (5) d'un objet d'un réseau à l'autre, changeant par conséquent le contexte opératoire. Cela permet d'atteindre l'objectif d'une analyse créative dans le rapport analyse-crédation. L'émulation d'un objet-opératoire, grâce à ses propriétés inhérentes, permet d'exploiter ces dernières dans un réseau opératoire créatif. Selon notre méthodologie, nous avons donc repris la fonction non linéaire récursive sinusoïdale utilisée par Di Scipio dans l'algorithme

(figure 4). En l'émulant dans le réseau opératoire de *La Philosophie du temps*, on constate que ce dernier hérite des propriétés de l'objet. Les attributs étant les mêmes, certaines méthodes sont altérées produisant différentes formes de l'objet-père, rendant ainsi manifeste la propriété de polymorphisme.

Dans le cadre de cette recherche, la première étape d'une émulation opératoire a été l'application de la fonction utilisée par Di Scipio dans le langage Max pour ensuite examiner les comportements de cette fonction. Nous avons essayé plusieurs valeurs d'indice de comportement (figure 8) dans un sous-patch avec l'objet Max table. Il est possible de visualiser le comportement de la fonction sinusoïdale récursive.

Figure 8. Visualisation de la fonction $x_{n+1} = \sin(x_n * r)$ après 50 interactions, valeur initiale de $x = 0.3$; A : $r = 1.5$; B : $r = 2.2$; C : $r = 2.5$; D : $r = 2.7$; E : $r = 3$; F : $r = 5$; G : $r = 6$; H : $r = 9$; I : $r = 10$



Source : auteur

Lorsque $r < 3$, x se stabilise à une valeur nulle (figure 8.A et 8.B) ; pour $r = 2.5$, il se stabilise en oscillant entre 2 états (figure 8.C) ; pour $r = 2.7$, le cycle passe à 3 états (figure 8.D) ; puis un comportement chaotique avec $r = 3$ (figure 8.E) ; le comportement redevient périodique pour

ensuite il redevient chaotique en $r =$ et figure 8.G et figure 8.H).

Le comportement chaotique dynamique de cette fonction se montre fertile pour le réseau opératoire de La

Philosophie du temps. Nous avons choisi de l'utiliser pour le contrôle de la granulation temps réel. Dans le patch Max, il y a trois modules granulaires, dont deux sont destinés au traitement de la synthèse sur les échantillons préenregistrés et un pour le traitement temps réel. Les paramètres de ce dernier ? taille du grain, delay, feedback et raréfaction ? sont contrôlés par la fonction sinusoïdale récursive. La valeur de l'indice r est programmée systématiquement par le script, elle peut donner des valeurs fixées selon la texture granulaire désirée pour chaque section de la pièce, ou elle peut être dynamique, pilotée par un signal contrôlé par le suiveur d'amplitude du piano. Dans ce cas, l'interaction instrument acoustique et électronique est responsable de la création de textures sonores.

À la différence de Di Scipio, l'interaction n'aboutit pas directement à l'émergence de sonorités d'une structure en micro-échelle ; elle génère plutôt les paramètres globaux d'un traitement granulaire. Toutefois l'objet est précieux dans le réseau opératoire de *La Philosophie du temps*, permettant ainsi la réussite de la composition sur le plan de la mixité ; la convergence entre l'électronique et l'acoustique grâce à un

principe algorithmique commun et grâce au contrôle des modules numérique par un paramètre acoustique.

Conclusion

L'élaboration du modèle opératoire de *La Philosophie du temps* s'est faite grâce à l'analyse de *Due di Uno*

d'Agostino

Di

Toutefois, le comportement de l'objet contenant la fonction non linéaire récursive dans *Due di Uno*, puis émulé dans *La Philosophie du temps*, présente des fonctionnalités différentes dans le contexte de chaque pièce. Di Scipio entreprend depuis la fin des années 1980 des études sur l'emploi des fonctions non linéaires récursives pour des traitements granulaires en micro-échelle. Le réseau interactif dans *Due di Uno* permet l'émergence d'une sonorité à une micro-échelle sur un plan algorithmique pour laquelle l'objet étudié joue un rôle décisif, tandis que dans *La Philosophie du temps* l'objet émulé contrôle le traitement temps réel à une *échelle paramétrique* ; cela n'a de valeur que lié à l'ensemble du réseau opératoire. La poursuite de cette étude visera à analyser de manière exhaustive l'objet émulé, de façon à donner suite à la démarche initiée par Di Scipio en explorant diverses utilisations dans des contextes différents. Cependant, la mise en perspective de cet objet dans le cadre de notre création nous a d'ores-et-déjà permis de voir plus loin et nous à aider à définir notre propre démarche compositionnelle individuelle.

1. Vaggione est prudent en définissant les propriétés de l'objet-opératoire dans un cadre compositionnel. Il a d'ailleurs développé cette approche dans des environnements informatiques qui n'étaient pas orientés-objets. Il est donc inutile de procéder à une comparaison pragmatique avec les langages purement informatiques orientés-objets, notamment sur la notion de polymorphisme utilisée en langages de programmation pour donner la capacité à une même fonction de recevoir des messages de syntaxes différentes.

2. *Due di Uno* est décrit dans un article écrit par Di Scipio qui fait partie du livre « Espace composable : Essais sur la musique et la pensée musicale d'Horacio Vaggione » en hommage au 60e anniversaire de Horacio Vaggione (Solomos, Makis (dir.) *Espace composable essais sur la musique et la pensée musicale d'Horacio Vaggione*, Paris : l'Harmattan, 2007).

3. Vaggione se réfère davantage à la mixité dans une perspective opératoire, il nomme « vectorisation commune » le processus de convergence entre la partie électronique et la partie acoustique. Étant donné la complexité de ce sujet, nous ne l'aborderons pas exhaustivement dans cet article. L'approche de Vaggione est toutefois importante dans la modélisation objet-opératoire ; « une situation riche mais néanmoins délicate, où il s'agirait de définir des passerelles très fines afin de faire interagir les deux domaines [électronique et acoustique] d'une façon très rapprochée, au niveau du résultat sonore, mais aussi au niveau du processus de composition lui-même, en travaillant à partir de la même situation musicale » (Vaggione, 1998).

4. Voir note ci-dessus.

5. Nous considérons comme émulation, dans la modélisation objet-opérateur, le fait de « charger » un objet logiciel modélisé dans un différent réseau opératoire.

Pour citer ce document:

Alain Bonardi, « L'analyse orientée objet-opérateur de *Due di Uno* d'Agostino di Scipio pour la modélisation des fonctions de contrôle non linéaires dans *La Philosophie du temps* », *RFIM* [En ligne], Numéros, n° 5 - Informatique et musique : Recherche et Création 1, Mis à jour le 28/07/2017

URL: <http://revues.mshparisnord.org/rfim/index.php?id=387>

Cet article est mis à disposition sous [contrat Creative Commons](#)