Un composant de synthèse formantique adapté à l'interaction multimédia

Jen-Paul Smets-Solanes

Un composant de synthèse formantique adapté à l'interaction multimédia

Jean-Paul Smets-Solanes
IRCAM

Nous montrons dans cet article qu'une exploitation judicieuse de la puissance de calcul des processeurs RISC permet d'envisager l'implantation de tout modèle de synthèse au moyen de ressources peu onéreuses et compatibles avec la plupart des environnements composition assistée par ordinateur. Le modèle de synthèse choisi pour la démonstration est celui de la synthèse par formes d'onde formantiques. L'implantation est réalisée sous forme de composant capable d'interagir en temps réel au sein du cadre de travail multimédia Quicktime. Des formes d'interactions évoluées, fondées sur un modèle de composant actif, sont alors envisagées.
Mots-clés : forme d'onde formantique, Quicktime, MIDI, programmation par composants, observateur.

Introduction

Le terme « composant » est fréquemment utilisé par un grand nombre d'industriels du génie logiciel qui l'emploient chacun sous une signification vague et, a priori, différente. Son irruption dans la terminologie informatique courante correspond à une problématique majeure qui n'est encore que partiellement résolue : définir une architecture modulaire « orientée objets » indépendante d'un langage sous-jacent, permettant, entre-autre, de faire interagir des « composants logiciels » développés séparément.
Les solutions proposées actuellement par l'industrie reposent sur deux entités :

- Le **gestionnaire de composant**. Il s'agit d'une extension du système d'exploitation permettant de charger dynamiquement des objets ou des classes définis dans un modèle à objet uniforme sous-jacent.
- Le **cadre de travail** (framework). Il incite à construire une application ou un document composite en commençant par définir de nouvelles classes par héritage de classes existantes puis en utilisant des mécanismes prédéfinis de composition et d'interaction pour assembler des objets (voir figure 1).

Figure 1. Exemples de gestionnaires de composants et cadres de travail

Gestionnaire de composants	Cadre de travail	Exemples de composants
Component Manager (MacOS)	Quicktime	Compresseur vidéo Synthétiseur sonore Base de temps
System Object Model (IBM)	OpenDoc	Correcteur orthographique Butineur WWW Editeur de dessin

Nous allons montrer dans cet article qu'il est possible de fonder une démarche de recherche reposant sur l'utilisation de cadres de travail et de gestionnaires existant. Plus précisément, l'emploi d'un cadre de travail comme Quicktime autorise assez simplement une implantation temps réel d'un modèle de synthèse sonore par formes d'onde formantiques (FOF). Cette approche a le mérite de permettre au musicien ou chercheur de manipuler immédiatement le modèle FOF au sein d'outils logiciels existant sans pour autant restreindre ses potentialités de mise en oeuvre au moyen d'outils évolués de contrôle de la synthèse sonore.
Nous rappellerons dans une première partie ce qu'est la synthèse sonore par forme d'onde formantique. Nous monterons ensuite comment implanter un composant de synthèse temps réel au sein de Quicktime. Nous suggérerons finalement comment contrôler ce composant au moyen d'outils évolués fondés sur la programmation à objets concurrents.

1 Synthèse par forme d'onde formantique

La synthèse de la voix chantée a été étudiée depuis le milieu des années 70 par Xavier Rodet [1], pour aboutir dix ans plus tard à un système général de synthèse adapté à la voix chantée [2] : Chant. A la fin des années 80, le système Chant a été porté sur la station de travail dédiée de l'IRCAM pour proposer aux compositeurs un contrôle temps-réel de la synthèse sonore par forme d'onde formantique. Au milieu des années 90, Chant a été porté sur micro-ordinateur Macintosh afin de permettre aux compositeurs de l'utiliser dans un cadre de travail personnel, mais sans pouvoir bénéficier d'un contrôle temps-réel, ou même interactif, de la synthèse sonore.
L'implantation originale sous fome de composant, décrite ci-dessous, permet de combiner l'interactivité d'une implantation temps-réel et l'accessibilité d'une implantation pour micro-ordinateur.

1.1 La synthèse de la voix chantée

On considère que la voix humaine est le résultat d'une modification continuelle par le conduit vocal du signal émis par trois types de sources sonores :

- une **source voisée** : elle correspond à la vibration des cordes vocales et se présente sous la forme d'un signal quasi-périodique.
- Une **source fricative** : elle correspond aux turbulences engendrées par les rétrécissements en certains points du conduit buccal (lèvres, langue-palais, glotte).
- Une **source plosive** : elle correspond au bruit d'explosion engendré par la fermeture puis l'ouverture brusque du conduit buccal avec les lèvres ou la langue.

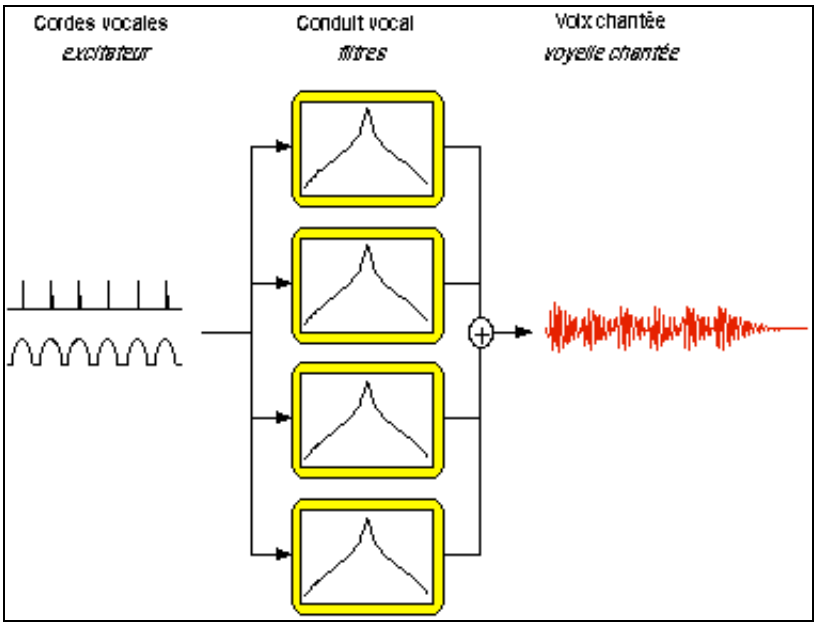
Dans le cas de voyelles chantées, les sources fricatives et plosives peuvent être omises.

1.2 Modèle excitateur-résonateur

Xavier Rodet modélise la production de voix chantée par un modèle de synthèse sonore du type excitateur-résonateur :

- l'**excitateur** correspond aux cordes vocales et définit le timbre de la voix; il est modélisé par une impulsion ou un arc.
- le **résonateur** correspond au conduit vocal et définit la voyelle chantée; il est modélisé par un jeu de filtres en parallèle. La fréquence centrale d'un de ces filtres est appelée un formant. Une suite de formants permet de créer une voyelle. En effet, la perception sonore de l'homme « détecte » les pics d'énergie dans le spectre d'un son quasi-permanent pour le caractériser.

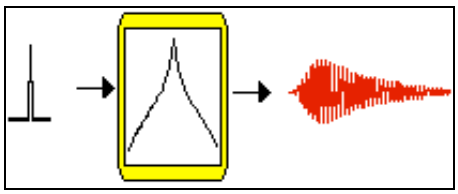
Figure 2. Modèle de synthèse de la voix chantée



1.3 Simplification du modèle

Le principe de la synthèse sonore par FOF est de considérer que les variations des paramètres des filtres sont suffisamment lentes pour être négligées lors du filtrage d'une excitation élémentaire. On peut calculer alors formellement l'expression correspondant à une impulsion élémentaire filtrée : c'est ce que l'on appelle une FOF ou forme d'onde formantique.

Figure 3. Une FOF est la réponse d'un filtre à une impulsion élémentaire



Ainsi, la réponse d'un filtre élémentaire à une impulsion est une sinusoïde amortie par une exponentielle (voir [2]).

$$y(t) = e^{-\alpha} \sin\left(\omega t + \phi\right)$$

Afin de mieux modéliser les excitations engendrées par les cordes vocales, on va « lisser » l'attaque de la sinusoïde amortie par une portion de cosinus :

$$t<0 \Rightarrow y(t)=0 \ (t<0)$$

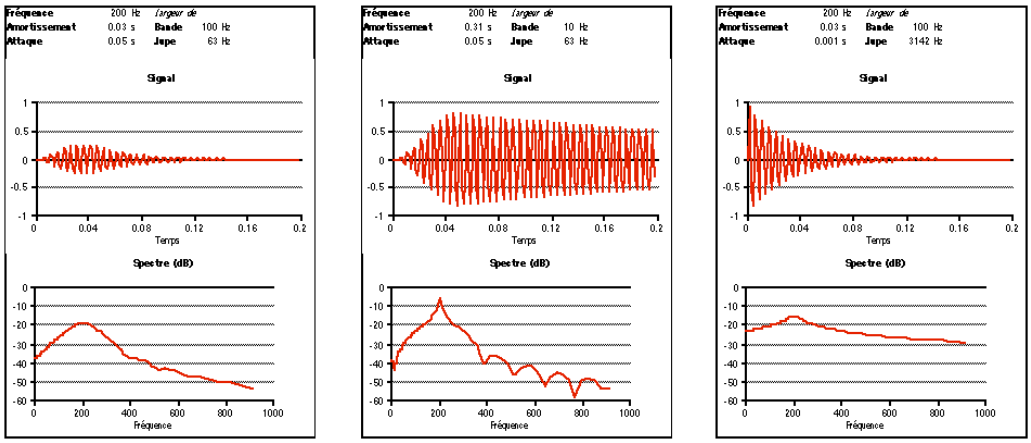
$$0<t<\frac{\pi}{\beta} \Rightarrow y(t)=\frac{1}{2}\left(1-\cos\left(\beta t\right)\right)e^{-\alpha}\sin\left(\omega t+\phi\right)$$

$$\frac{\pi}{\beta}<t \Rightarrow y(t)=e^{-\alpha}\sin\left(\omega t+\phi\right)$$

L'expression obtenue dépend de trois paramètres :

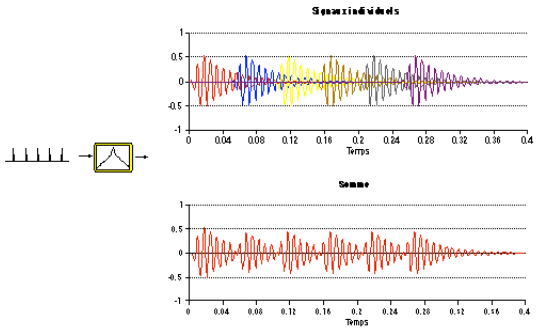
- ω : il correspond à la fréquence centrale du filtre;
- α : il correspond à la largeur de bande du filtre, c'est-à-dire à largeur du pic d'énergie à -3 dB du maximum;
- β : il correspond à la largeur de jupe du filtre c'est-à-dire à l'étalement du spectre dans les hautes fréquences.

Figure 4. Signal et spectre de trois FOF ayant la même fréquence d'oscillation mais différentes largeurs de bande ou de jupe



Enfin, pour calculer le signal résultant d'une excitation quasi-périodique, on va superposer un grand nombre de FOF décalées dans le temps de façon quasi-périodique.

Figure 5. Calcul par superposition de FOF décalées



1.4 Algorithme de calcul rapide

Afin de calculer rapidement une FOF, on utilise la méthode par itération développée initialement par Gerart Eckel et Francisco Iovino pour la station IRCAM [3] : en résolvant par différences finies une équation différentielle linéaire du premier ou du deuxième ordre, les valeurs successives correspondant au sinus, au cosinus ou à l'exponentielle de la FOF peuvent être obtenues en un minimum d'additions et de multiplications. Cette méthode est beaucoup plus rapide qu'une tabulation dans la mesure où l'accès aux champs d'un tableau est pénalisé par la taille réduite de la mémoire cache des machines à processeur RISC. De plus, elle ne dépend pas d'une évaluation de cosinus ou d'exponentielle par l'unité arithmétique du processeur.

2 Un composant Quicktime de synthèse en temps-réel

L'implantation de l'algorithme repose sur l'écriture d'un composant (au sens du Component Manager de MacOS) s'interfaçant avec le gestionnaire multimédia Quicktime. Cette implantation présente plusieurs avantages :

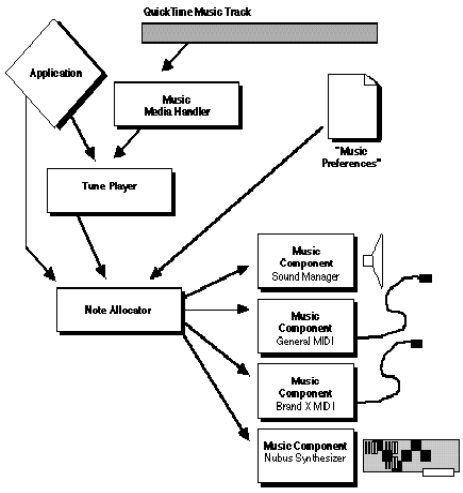
- Elle est **multi-plate-forme** : elle peut être portée sur tout système compatible avec Quicktime, comme Windows ou IRIX.
- Elle permet un **contrôle générique** : dans la mesure où les logiciels multimédias, MIDI ou compatibles OMS ont accès à Quicktime de façon transparente.
- Elle est **modulaire et réutilisable** : le code de synthèse est concentré sur une dizaine de lignes en C. En changeant ce code, on peut créer très rapidement de nouveaux modules de synthèse sonore.

2.1 Quicktime Music Architecture

L'architecture musicale de Quicktime (QMA)[3] définit quatre types de composants :

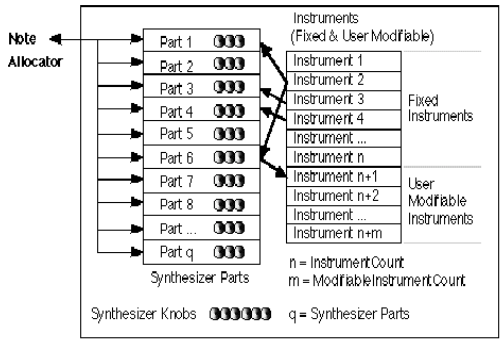
- **Music Data Handler** : ce composant gère un format de données musicales, comparable au format MIDI mais incluant un contrôle micro-tonal de la hauteur, un accès générique aux paramètres de synthèse et une description du timbre par nom et par mode de synthèse. Il permet de lire des données musicales dans une séquence multimédia.
- **Tune Player** : ce composant se charge de la mise en temps d'événement musicaux. Une application souhaitant jouer des notes de musique à des dates précises, pourra faire appel à une instance de ce composant.
- **Note Allocator** : ce composant gère la répartition de la polyphonie et des timbres à reproduire en fonction de l'environnement de synthèse disponible. Une application souhaitant jouer des notes de musique immédiatement, pourra faire appel à une instance de ce composant.
- **Music Synthesizer** : ce composant synthétise des notes en fonction de paramètres de hauteur, de sonie et de timbre. En créant de nouveaux composants synthétiseurs, on étend le nombre d'algorithmes de synthèse accessibles aux applications.

Figure 6. L'architecture musicale de Quicktime (crédit Apple)



L'implantation du synthétiseur de FOF passe donc par la conception d'un composant de synthèse. Dans QMA, ces composants comportent un certain nombre de voix (polyphoniques ou monophoniques) correspondant chacune à un timbre donné. L'ensemble de ces voix se partagent la polyphonie totale du composant. Elles peuvent chacune être contrôlées par un jeu de paramètres définis par le concepteur du composant.

Figure 7. Chaque composant de synthèse peut produire simultanément plusieurs timbres provenant d'une même banque de timbres (crédit Apple)



Generic Synthesizer Model

2.2 Paramètres de synthèse

Nous avons choisi de définir tout d'abord un jeu de 48 paramètres de contrôle de la synthèse. Ces 48 paramètres correspondent à un jeu de 6 formants par voix, chacun de ces formants étant défini par les 8 paramètres :

- Fund : définit fréquence fondamentale de l'excitation associée au formant.
- Freq : définit la période du formant.
- Bw : définit la largeur de bande des FOF associées au formant.
- Amp : définit l'amplitude du formant.
- Attack : définit la largeur de jupe du formant.
- Debatt : limite le temps de relaxation d'une FOF.
- Atten : définit le temps d'atténuation d'une FOF.
- Cross-over : définit le recouvrement de FOF maximum du formant.
- On définit aussi au niveau de chaque voix un jeu paramètres de contrôle :
- Relax : temps de relaxation du filtre du premier ordre appliqué aux données de contrôle pour éviter un effet « crémaillère ».
- Amp : amplitude de la voix.
- Pitch : hauteur de la voix.

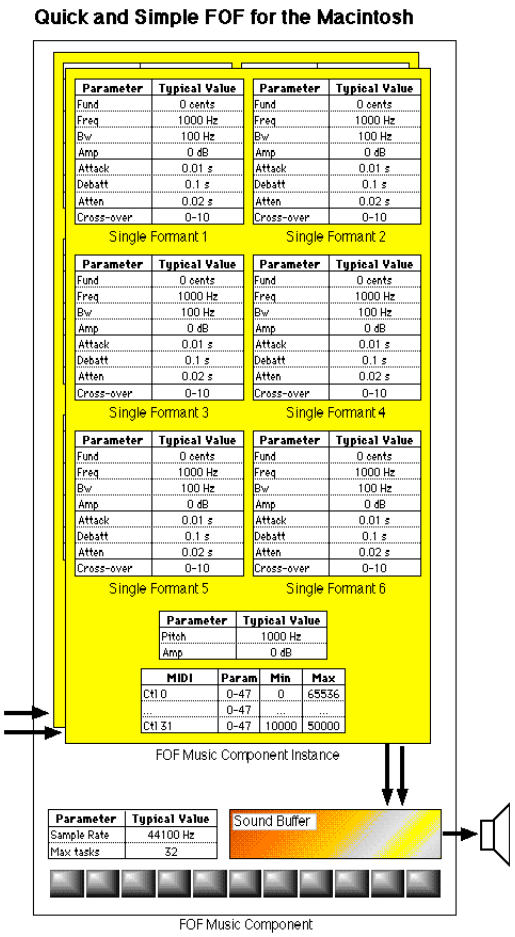
Enfin, au niveau du composant, on définit :

- SampRate : fréquence d'échantillonnage du signal
- MaxTask : polyphonie ou recouvrement maximal du composant.

2.3 Contrôle MIDI

128 autres paramètres permettent de piloter par contrôle MIDI les paramètres de synthèse en définissant pour chacun des 32 contrôles continus MIDI, un paramètre de synthèse cible, une valeur minimum et une valeur maximum.

Figure 8. Architecture du composant de synthèse de forme d'onde formantique pour Quicktime Music Architecture



2.4 Principe de fonctionnement

Lorsque le composant reçoit un message lui ordonnant de créer une note, il alloue pour chaque formant un certain nombre de tâches en fonction du recouvrement défini par les paramètres de synthèse. Dans le même temps, un système d'interruption est mis en place pour remplir régulièrement un buffer où est stocké le signal sonore. A chaque interruption, on parcourt la liste des tâches en calculant pour chaque tâche une portion de FOF correspondant à la taille du buffer. Lorsqu'une FOF a été complètement calculée, la tâche est ré-initialisée pour être exécutée ultérieurement en fonction de la fréquence fondamentale d'excitation.

2.5 Contraintes liées à l'architecture RISC

Lors de l'implantation, nous avons essayé de regrouper autant que possible les zones mémoires où sont définis les paramètres du modèle avec les zones mémoire où se trouve le buffer. Nous avons aussi éliminé toute utilisation de pointeurs dans la partie du calcul qui s'exécute sous interruption. En effet, un système informatique à processeur RISC souffre généralement d'un goulot d'étranglement situé au niveau de l'accès à la mémoire principale, qui ne s'effectue qu'à environ un dixième de la vitesse maximum théorique de traitement du processeur. C'est pourquoi, les concepteurs de systèmes à processeur RISC placent une mémoire cache à accès rapide entre le processeur et la mémoire principale. En tentant de regrouper les données, on augmente la probabilité de leur présence simultanée dans le cache et l'on exploite ainsi beaucoup mieux la vitesse théorique du processeur. Par exemple, le simple regroupement des paramètres de synthèse et du buffer s'est traduit par doublement des performances, qui atteignent à présent plus de 100 FOF élémentaires sur un PowerMacintosh.

3 Insertion dans un cadre de travail multimédia

Nous présentons dans cette partie une application, au sein d'un système multimédia complexe, du composant de synthèse par formes d'onde formantiques présenté précédemment.

3.1 Contrôle de la synthèse sonore

Le contrôle de la synthèse sonore a pour objectif de faciliter la manipulation de modèles de synthèse dépendant d'un grand nombre de paramètres, en offrant au musicien des outils capable de gérer, de représenter et de manipuler simplement le modèle. De nombreux outils de contrôle de la synthèse sonore fondés sur des interfaces graphiques ont déjà été développés [4,5]. A l'instar de générateurs d'interface utilisateur, ces outils permettent de placer sur l'écran de l'ordinateur des composants de contrôle et de spécifier leur relation avec le modèle de

synthèse.
C'est ce principe que nous avons choisi d'appliquer à la synthèse par forme d'onde formantique en proposant deux innovations :

- Une **interface animée en trois dimensions**. Le modèle graphique que nous avons choisi d'utiliser est fondé sur un système d'animation à modèle physique et résolution de contraintes mécaniques [6].
- Un **contrôle actif**. Le dispositif de contrôle utilise des composants dotés d'un comportement autonome propre capable de se manifester sans intervention directe de l'utilisateur.

La mise en interaction du modèle graphique avec le modèle de synthèse par FOF est obtenu par application d'un formalisme réflexif de l'interaction : le modèle acteur-observateur [7].

3.2 Acteurs-observateurs

Presque tous formalismes de l'interaction qui ont été développés sont caractérisés par une séparation nette :

- d'une part des composants d'un système, sur lesquels porte l'interaction;
- d'autre part d'un mode d'interaction entre composants, décrit comme un objet formel portant sur les composants du système et néanmoins étranger à celui-ci.

Cette formalisation de l'interaction permet certes, à l'instar de Quicktime, d'étendre les fonctionnalités d'un système par ajout de composants mais n'autorise pas l'extension des modes d'interaction entre composants. Au contraire, le modèle acteurs-observateurs autorise la co-existence d'interactions en nombre et type quelconque au sein d'un même cadre de travail grâce à un mécanisme de réflexion : toute interaction y est représentée formellement au sein d'un composant du système appelé observateur; l'extension des modes d'interaction passe alors par un ajout « traditionnel » de composants au système.

3.3 Un contrôle mécanique des voyelles

L'application du modèle acteur-observateur à la synthèse par FOF conduit à séparer :

- un **acteur de formant** pour définir le comportement autonome d'un formant isolé;
- un **observateur de voyelle** pour appliquer un jeu de contraintes et d'interactions au sein d'un ensemble de formants isolés et contrôler ainsi la synthèse de voyelle.

Le contrôle de la synthèse est obtenu soit par manipulation directe des acteurs de formants isolés, soit indirectement par des procédés d'animation ou d'interpolation des observateurs de voyelles.
L'application du modèle acteur-observateur à l'animation par modèles physiques et contraintes mécaniques conduit à séparer :

- un **acteur de point physique** pour définir les paramètres de position, vitesse, masse et inertie d'un point;
- un **observateur de dynamique** pour mettre en interaction, selon les lois de la dynamique newtonnienne, un point physique avec un base de temps;
- un **observateur de contrainte mécanique** pour satisfaire en permanence la contrainte mécanique faisant interagir un ensemble de points physiques.

Le contrôle de l'animation est assuré soit par manipulation directe des points physiques, soit indirectement par manipulation des observateurs de dynamique ou de contrainte.
La mise en interaction de ces deux modèles hétérogènes peut alors être obtenue en définissant un observateur mettant en interaction les observateurs de voyelle, dynamique et contrainte : c'est l'observateur de méca-voyelle.

3.4 Implantation

MetaMedia est une implantation du modèle acteur-observateur réalisée dans l'environnement SmalltalkAgents en utilisant, d'une part Actalk, l'extension concurrente de Smalltalk proposée par Jean-Pierre Briot [8], d'autre part le principe des messages sémantiques qui permet d'étendre et déformer simplement la sémantique standard de l'héritage et de l'instanciation dans Smalltalk [9]. Les modèles de voyelle ou d'animation décrits précédemment ont pu être implantés par dérivation des classes des bases de MetaMedia, écrites en Smalltalk. L'accès au composant de synthèse sonore par FOF au sein de cet environnement a été particulièrement facilitée par l'adéquation entre le modèle de composant de Quicktime et la sémantique du modèle objet de Smalltalk. Cette adéquation a été rendue possible grâce à la dynamicité et la neutralité de la programmation par composant par rapport à tout langage.

Conclusion

L'expérimentation décrite dans cet article prouve, à notre sens qu'implanter un modèle de synthèse sonore au moyen du cadre du travail Quicktime ne présente quasiment que des avantages pour le chercheur ou le musicien.

Il pourra par exemple commencer à tester en temps-réel son modèle au moyen d'outils de MAO existant avant d'envisager d'augmenter la polyphonie ou la résolution de son composant grâce aux nouvelles machines multiprocesseur Daystar RISC Genesis ou à la console de jeu Bandai @tmark. Le cadre de travail Quictime étant fondé sur la notion de composant, il est aussi aisé d'interfacer le composant développé avec un système à objets chargé du contrôle de la synthèse sonore. C'est pourquoi, nous considérons que la programmation par composants pourrait rendre beaucoup plus abordable, tant d'un point de vue technique que financier, la recherche en synthèse sonore. Elle devrait aussi favoriser l'émergence de nombreux modèles originaux qui trouveront une application musicale immédiate auprès du public nombreux et curieux des utilisateurs d''Internet.

Remerciements

Je suis très reconnaissant à Jean-Baptiste Barrière et Hugues Vinet de m'avoir permis d'entreprendre ce travail au sein des équipes de recherche à l'IRCAM. Je tiens à remercier tout particulièrement Gerart Eckel, Xavier Rodet, Gérard Assayag et Chris Rogers pour les discussions et explications fructueuses qui ont permis à ce projet d'aboutir.

Bibliographie

1 Rodet Xavier. Analyse du signal vocal dans sa représentation amplitude-temps. Synthèse de parole par règles. Thèse d'Etat. Université de Paris VI.1977.
2 Rodet Xavier Potard Yves Barrière Jean-Baptiste. Chant. De la synthèse de la voix chantée à la synthèse en général. Rapport de recherche N° 35. IRCAM. 1985.
3 Notes on the FOF-FTS implementation, G. Eckel, F. Iovino, IRCAM internal report, March 94.
4 VanBrink David. Music the Easy Way: The Quicktime Music Architecture. Develop 23, The Apple Technical Journal. 1995.
5 Interactors User Manual. TimeTech. 1991.
6 Luciani Annie Cadoz Claude. Informatique Musique Image Animée ACROE. Rapport de recherche INPG. Grenoble. 1980.
7 Marie-Paule Gascuel. An Implicit Formulation for Precise Contact Modelling Between Flexible Solids. Proceedings of SIGGRAPH'93, pages 313-320, August 1993.
8 Smets Jean-Paul. Synthèse Multimédia : une formalisation réflexive de l'interaction. A paraître en 1996.
9 Briot Jean-Pierre. Des Objets aux Acteurs, 1982-1989 : 7 Ans de Réflexion.
10 Quasar Knowledge Systems. SmalltalkAgents Reference Manual. Quasar Knowledge Systems. 1993