HAL
open science

Le modèle 'COSMOS'

Sinan Bökesoy

To cite this version:

Sinan Bökesoy. Le modèle 'COSMOS'. Journées d'Informatique Musicale, Association Française d'Informatique Musicale; Centre de recherche en Informatique et Création Musicale, Jun 2005, Saint-Denis, France. hal-03121635

HAL Id: hal-03121635

https://univ-paris8.hal.science/hal-03121635v1

Submitted on 26 Jan 2021

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.

LE MODELE 'COSMOS'

Sinan Bokesoy
CICM, Université de ParisVIII
MSH Paris Nord
bokesoy.sinan@wanadoo.fr

RÉSUMÉ

Le modèle "Cosmos" est un système de distribution d'événements dynamiques qui génère et organise des entités sonores évoluant sur de multiples échelles de temps. Inspiré par les modèles de génération d'événements stochastiques de Iannis Xenakis, ce modèle apporte de plus des aspects importants comme l'autosimilarité dans la structure et le comportement émergents de l'évolution sonore résultante, tout en préservant la simplicité structurelle. L'originalité de ce modèle par rapport aux modèles basiques de synthèse stochastique sera présentée dans cet article, ainsi que la réalisation Max/MSP du modèle et ses aspects techniques.

1. INTRODUCTION

Iannis Xenakis a développé ses méthodes de composition en fonction du timbre en introduisant les modèles mathématiques stochastiques (1). En commençant par les compositions telles que *Metastasis* (1953-54), *Achorripsis* (1956-57), *Analogique A/B* (1958-59) et les séries de compositions *ST* (1956-62), Xenakis a utilisé de façon intensive les modèles statistiques et a introduit les concepts de désordre-ordonné statistiquement dans le domaine de la composition.[1] Le mécanisme organise les entités sonores qui sont définies dans des espaces de timbre en tant que quanta acoustiques individuels :

- Les événements séparés sont distribués avec une certaine densité dans une cellule qui comprend un espace temporel défini par une certaine durée. Parmi les variables du système se trouvent la densité de l'événement, le temps de départ de l'événement et la durée qui sont calculés avec des fonctions stochastiques. (Voir Figure 1)

- Les événements sont assignés à des entités sonores avec des paramètres sonores comme la hauteur et l'intensité qui sont modulés dans le temps en utilisant les fonctions de courbes stochastiques continues.

- La définition d'un mécanisme de plus haut niveau, comme la stochastique markovienne dans *Analogique A/B* a été introduite afin de formuler le deuxième ordre d'évolution du système d'événements.

[1] Je vais faire référence à de nombreuses reprises dans le texte à l'oeuvre de *Xenakis* précédemment citée à propos des systèmes de génération d'événements en comparaison avec mon modèle.

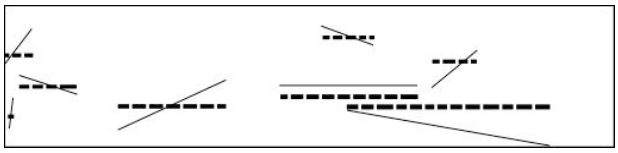

Figure 1. L'espace d'événement dans le modèle *Achorripsis* (Voir FM, p.28-38) calculé et individualisé à l'intérieur de Max/MSP. Les événements représentés par des lignes en pointillés sont distribués à l'intérieur d'une cellule d'une certaine durée, avec la distribution exponentielle définissant leur temps de départ et leurs durées. Xenakis a utilisé la distribution triangulaire pour déterminer les valeurs offset entre les glissandi comme ils sont représentés en traits fins sur la figure. La vitesse des glissandi est déterminée par la distribution de Gauss.

2. LE MODELE 'COSMOS'

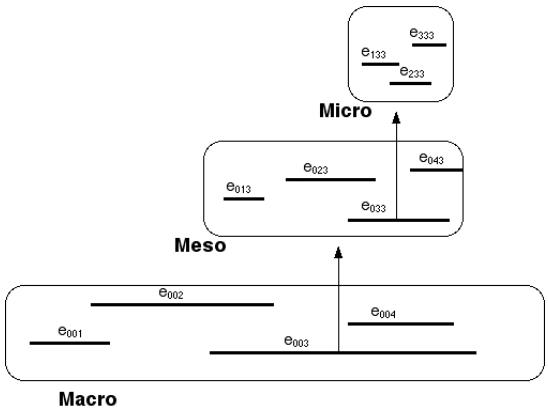

Figure 2 L'autosimilarité dans la structure du modèle Cosmos. Chaque événement dans l'espace existant ouvre un autre espace dans l'échelle de temps de plus bas niveau avec le même mécanisme de distribution. Nous voyons aussi les méthodes d'indexation des événements distribués.

Pendant que les modèles de Xenakis opèrent sur une échelle temporelle, le modèle Cosmos (Voir figure 2) permet d'opérer sur de multiples échelles de temps simultanément, où nous pouvons observer la structure sonore évoluer à différents niveaux. Les événements de

l'espace macro contiennent des sous-événements qui sont distribués sur l'échelle méso de temps. Chaque événement dans l'espace macro définit une région temporelle de l'espace méso égale à sa durée, et les sous-événements sont distribués à l'intérieur de cette durée de temps. La même organisation s'applique aussi aux événements de l'espace méso. Chaque événement dans un espace méso définit une région temporelle d'un espace micro égal à sa durée, et les sous-événements sont distribués à l'intérieur de ces cellules.

La règle de base dans ce système de distribution d'événements est :

$dur\ e_{00x} = dur\ C_{ex}$ et $dur\ e_{0yx} = dur\ C_{iy}$ (1)

La durée d'un événement dans l'espace macro est égale à la durée de la cellule qu'elle crée dans son espace méso. Comme nous pouvons voir dans la figure 2, des événements se chevauchent dans leurs cellules et aussi entre les cellules. A l'arrivée du temps de départ de chaque événement dans l'espace temporel macro, méso ou micro, des sous-espaces pertinents sont créés.

$e_{00x} \rightarrow C_{ex}$ et $e_{0yx} \rightarrow C_{iy}$ (2)

Le cas particulier réside dans la distribution de l'espace macro, où la taille absolue de l'espace macro peut changer dynamiquement selon la distribution des événements de l'espace macro. Cela signifie que selon les information sur le temps de départ du dernier événement de l'espace macro, la véritable longueur de l'espace pourrait rétrécir, s'élargir, ou pourrait rester constante en tant que valeur déterministe comme spécifiée par l'utilisateur.

Les fonctions stochastiques ne délivrent aucune dépendance entre les distributions de valeurs précédentes et suivantes de l'espace de l'événement en ce qui concerne les cellules, puisque ce sont des mécanismes sans mémoire et qui réagissent selon l'événement. La dépendance dans la distribution est entre les niveaux de la structure, où chaque événement dans l'espace de l'événement définit la taille de l'espace de l'événement de plus bas niveau. Ceci introduit de nouveaux aspects comme les multiples niveaux d'organisation structurelle, les propriétés émergentes et la complexité. Tout en décrivant le modèle Cosmos comme un système, j'aimerais citer la phrase suivante :

« Les quatre idées de système les plus fondamentales sont ainsi l'émergence, la hiérarchie, la communication, et le contrôle : ce sont les idées nécessaires à la description de la métaphore du noyau du système. » (4)

2.1. A propos des faits structurels

Un objet autosimilaire est exactement ou approximativement similaire à une de ses parties. Le mécanisme de distribution d'événement du modèle Cosmos est statistiquement autosimilaire: des parties montrent les mêmes propriétés statistiques à des échelles temporelles macro, méso et micro (Voir Figure 2). Il n'est pas possible d'utiliser une seule fonction stochastique pour la distribution d'événements comme dans le modèle d'Achorripsis pour générer le même type de résultats. Par conséquent, le résultat pour le modèle Cosmos a une complexité qui va au-delà des mécanismes de distribution d'événements que Xenakis a proposé dans Achorripsis et ses séries de compositions ST.

Il est préférable d'utiliser, et c'est une décision méthodologique, l'organisation d'unités minimales pour introduire des sonorités secondaires (3), où le compositeur prend les décisions d'appliquer à la plus petite échelle de temps, et où les choses se produisent qui résultent à grande échelle (propriétés émergentes), des attributs perceptibles que nous pourrions appeler timbre et texture (2). Par conséquent nous pouvons assigner les événements créés à des degrés plus bas en tant qu'entités sonores tout en s'attendant à ce que l'organisation de plus haut niveau créé les sonorités secondaires avec les propriétés émergentes.

L'émergence est le processus de dérivation de nouvelles structures cohérentes, de modèles et de propriétés dans un système complexe. Un comportement émergent ou une propriété émergente est montrée lorsque un nombre de simples entités opère dans un environnement, formant des comportements plus complexes en tant qu'ensemble. Les propriétés émergentes surviennent lorsqu'un système complexe atteint un seuil combiné de diversité, d'organisation et de connectivité. Le modèle de Cosmos établit cette relation en créant un pont entre les articulations au niveau micro et l'organisation au niveau macro. Chaque événement sonique sur la couche de bas niveau peut identifier du quanta acoustique, et les couches de plus haut niveau organisent ces grains, et lorsqu'ils se chevauchent, ils créent des phénomènes tels que la densité locale perçue dans la texture résultante. Les phénomènes créés au niveau méso sont susceptibles d'amener l'épiphénomène qui pourrait être perçu en tant qu'effets de timbre non linéaires au niveau macro. Les couches au-dessous du niveau macro organisées par le mécanisme d'autosimilarité introduisent des propriétés émergentes qui deviennent visibles au niveau macro en créant des épiphénomènes.

Nous devrions mettre l'accent sur le fait que pendant qu'un phénomène émergent à l'échelle macroscopique n'existe pas directement à l'échelle microscopique, son existence à l'échelle macroscopique peut toujours être expliquée en prenant en compte les interactions entre tous les composants microscopique d'un objet macroscopique.

L'idée de mettre en application un comportement émergent dans des structures sonores entraîne aussi la question de l'unification du timbre et de la forme musicale à travers la composition micro structurelle, où les entités sonores deviennent le timbre et le développement structurel établit la forme musicale (2).

Les inconvénients inhérents des fonctions stochastiques - le fait que même si un processus stochastique peut avoir une direction, il avance toujours vers l'équilibre - posent une contradiction au mécanisme émergent (5). Puisque la stochastique markovienne est un système ergodique, cet énoncé pourrait aussi être correct en ce qui concerne le mécanisme stochastique que Xenakis a établi, qui se termine par un équilibre sans l'introduction de perturbations manuelles dans Analogique A/B. (1) Lorsque la densité d'informations produite par un système stochastique est au-delà des limites du système d'écoute, il ne sera pas possible de suivre le mouvement individuel, seulement le mouvement de masse. Par conséquent, les contours du système deviennent visibles où les unités individuelles deviennent invisibles, ce qui provoque, sur le système auditif, la prédiction que l'évolution resterait toujours à l'intérieur de cette marge. Et lorsqu'il n'y a pas de mécanisme émergent comme l'établissement d'une dépendance entre les entités, le résultat deviendrait banal.

A l'intérieur du modèle Cosmos et avec les fonctions stochastiques disponibles dans le mécanisme de distribution de l'espace des événements, les marges du système sont bien plus difficiles à prévoir à cause des effets non linéaires introduits dans la complexité et par conséquent, le résultat sonore devient moins prévisible. La principale raison vient des multiples couches qui opèrent sur différentes échelles de temps. Les espaces d'événements se chevauchent aussi et l'autosimilarité dans la structure crée des connections invisibles entre les couches.

Cosmos n'est pas sensible à l'événement en terme de changement automatique de comportement sur l'occurrence d'états particuliers. Mais ce n'est pas 'Stochos' (6), ce qui signifie que c'est plus qu'un désordre organisé statistiquement comme dans les modèles de Xenakis, où la distribution d'événement se produit sur une seule échelle de temps.
Il est important d'observer que les niveaux méso et micro dans le modèle Cosmos dépendent de paramètres initiaux, qui manipulent l'espace macro de l'événement.

3. LA REALISATION DU MODELE 'COSMOS'

Le modèle Cosmos a été réalisé avec l'environnement de programmation en temps réel Max/Msp, grâce au support JavaScript de la version V4.5. Pour réaliser le mécanisme de distribution autosimilaire, on doit utiliser la programmation récursive, qui est nécessaire à la mise en application sur un environnement de programmation basé sur le script tel que JavaScript.

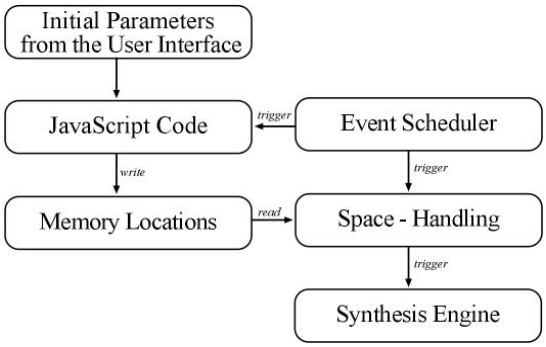

Figure 3. Dans la structure de programmation, le code JavaScript calcule toutes les informations nécessaires à la distribution de l'événement et les écrits dans la mémoire.

L'application effectue les tâches suivantes :

- Elle établit un lien contrôlé avec l'utilisateur en présentant une interface d'utilisateur pour transférer le résultat de l'utilisateur vers le code du programme.

- Elle calcule la distribution de données et l'écrit dans la mémoire qui sera interprétée par le patch MSP qui l'accompagne.

- Elle fait fonctionner le programmateur d'événements dans le patch MSP qui compte le temps DSP et organise la liste de temps de départ qui devra être exécutée par le moteur de synthèse.

- Elle calcule les générateurs de courbes pour chaque événement au niveau macro, méso et micro en tant que sources de modulation et les applique au moteur de synthèse.

3.1. L'interface de *Cosmos*

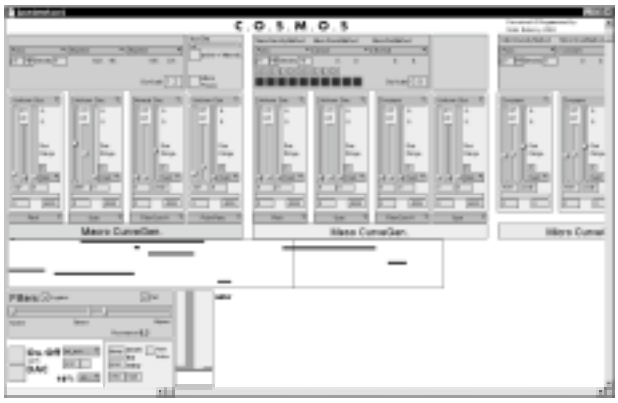

Figure 4. L'interface utilisateur de Cosmos. La partie supérieure est l'interface de contrôle pour la distribution dans l'espace macro, méso et micro. Dans la partie du milieu nous voyons la distribution macro de l'espace avec le temps de départ de l'événement et l'affichage de la durée. En dessous, les paramètres du moteur de synthèse sont affichés.

L'interface utilisateur est divisée en trois sections, qui sont aussi autosimilaires. Elles organisent les paramètres de distribution dans l'espace conduits par l'utilisateur

qui applique des données déterministes au système (Voir Figure 4).

Le changement de paramètres de l'espace macro introduit par l'utilisateur va se refléter sur le système au prochain cycle de distribution de l'espace macro. La même chose s'applique aux paramètres d'espace de l'événement méso et micro.

Lorsqu'on introduit un changement dans les paramètres de l'espace macro, tous les sous-espaces sont aussi redistribués à l'intérieur du prochain cycle de distribution de l'espace macro, comme nous pouvions le prévoir. Mais si ce sont les paramètres d'espace méso qui sont modifiés, la distribution de l'espace macro reste la même et le changement sera reflété dans le prochain cycle de distribution de l'espace méso mais seulement au niveau méso de l'espace. Cette solution nous assure que la résolution de l'interruption de l'utilisateur s'accommode à l'échelle de temps sur laquelle l'utilisateur opère, ce qui signifie moins de calculs pour le programmateur d'événement.

En tant que paramètre critique, la longueur de cellule de l'espace macro peut être modifié par un réglage continu en temps réel. Il est connecté à un *phasor~*, objet dans MSP qui conduit le programmateur d'événements temporels DSP. Les changements dans le réglage correspondent à un changement de fréquence du *phasor~*, et par conséquent de la fréquence de l'horloge du système, le ralentissant ou l'accélérant par une transition continue.

Si la nouvelle valeur de la longueur de la cellule macro était directement introduite dans le code du script, le système devrait alors attendre le prochain cycle de l'espace macro pour réaliser ce changement, parce que c'est un paramètre de l'espace macro. Le temps de départ et la distribution de la durée de l'événement pour les espaces macro, méso et micro peuvent être contrôlés par les fonctions stochastiques *uniforme*, *exponentielle*, *triangulaire*, *normal* et *cauchy*, mais ces méthodes pourraient aisément être étendues à d'autres processus, soit à des modèles déterministes ou non déterministes.

3.2. Le programmateur d'Evénements

La structure récursive dans le code du programme appelle la définition de la fonction *celldist*, ce qui définit les principaux objets de la distribution de l'espace.

```
function
celldist(celL,distmethod,distrange,onsetmethod,durmethod,
courbe1meth,courbe2meth,courbe3meth,courbe4meth,
param1,param2,param3,param4)
{ var dens = densdist(arguments[1],arguments[2]);
  var cospace = new space(dens,arguments[0]);
  for (j=0;j<dens;j++) {
cospace.absonsets[0] = 0;
cospace.durs[j]=durdist(arguments[4],arguments[0],dens
, arguments[11],arguments[12]);
cospace.onsets[j]=onsetdist(arguments[3],arguments[0],
dens,arguments[9],arguments[10]);
cospace.courbe1[j] = courbedist(courbe1meth,celL,dens);
cospace.courbe2[j] = courbedist(courbe2meth,celL,dens);
cospace.courbe3[j] = courbedist(courbe3meth,celL,dens);
cospace.courbe4[j] = courbedist(courbe4meth,celL,dens);
}
return (cospace);
};
```

A chaque fois que l'espace macro, méso ou micro est créé, la fonction *celldist* est appelée avec les paramètres d'entrée appropriés selon la longueur de la cellule, la fonction de densité de l'événement, la valeur moyenne de la densité, la fonction de distribution du temps de départ de l'événement, la fonction de distribution de la durée d'événement, les méthodes de quatre générateurs de courbes et leurs paramètres de fonction.

La boucle *for* à l'intérieur de la fonction *celldist* assigne le temps de départ, la durée et les valeurs de la courbe, l'objet *cospace* pour les événements indexés par « j ». L'objet *cospace* est crée avec les lieux de stockages de valeurs dans la mémoire (arrays) selon la valeur de densité calculée et la longueur de la cellule.

Le mécanisme de distribution calcule l'intervalle relatif du temps de départ entre les événements dans chaque espace dans lequel ils sont définis (Voir Figure 5).
Le code de programme convertit cette information vers l'information absolue du temps de départ en passant par le programmateur d'événements.

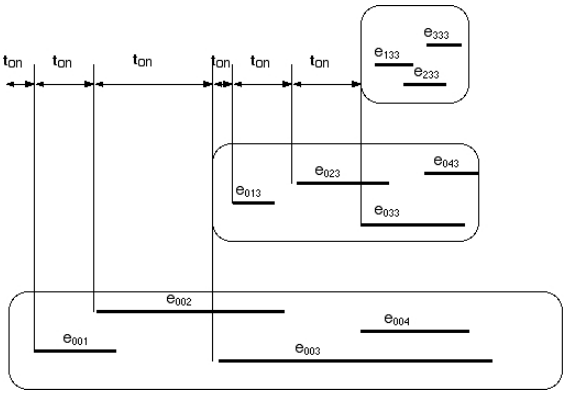

Figure 5. Additionner les vecteurs relatifs du temps de départ.

Expression mathématique pour calculer le temps de départ absolu $T_{on}e_{xyz}$ dans laquelle x, y et z spécifient les index macro, méso et micro de l'événement.

$$\mathbf{T}_{on}\mathbf{e}_{xyz} = \sum_{m=0}^{z}\sum_{l=0}^{y}\sum_{k=0}^{x} \mathbf{t}_{on}\mathbf{e}_{klm} \qquad (3)$$

Dans Cosmos, la densité de l'événement de l'espace macro a la limite supérieure de dix événements / cellule.

Les temps de départ calculés des événements macro sont comparés avec la valeur de *phasor* dans le patch MSP, qui évolue entre les valeurs 0-96000, représentant la longueur de cellule macro normalisée et envoie ensuite la valeur d'index pour présenter le temps DSP. Puisque chaque événement dans l'espace macro représente un espace de l'événement méso, un résultat logiquement 'vrai' de la comparaison entre le phasor index et la liste de temps de départ de l'événement enverra un message 'bang' aux patches correspondants qui s'occupent des processus dans l'espace méso, numérotés entre 1-10. (Voir Figure 6)

Pendant que chaque patch qui s'occupe des processus dans l'espace méso reçoit la valeur index en provenance de l'objet *phasor~* dans la précision audio, les données de distribution de l'espace méso de l'événement écrit par le code du programme sur la mémoire sera comparé afin de déclencher les événements de l'espace méso sur les temps de départ adéquats.

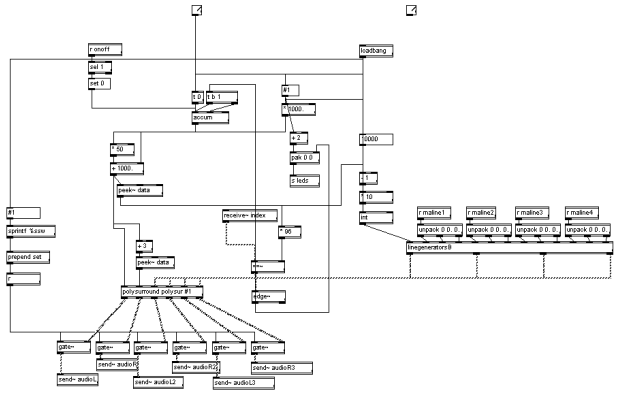

Figure 6. Ce patch qui s'occupe des processus dans l'espace méso lit les temps de départ et les durées des événements méso avec *peek~*, objet venant de la mémoire qui compare la valeur de l'index pour donner le temps absolu. L'objet *edge~* envoie un '*bang*' et les données sont passées à l'objet *poly~* qui existe dans chaque patch de l'espace méso.

Dans cet exemple d'application, les événements de l'espace méso sont interprétés comme des entités sonores. Les données des événements de l'espace micro sont utilisées pour la génération de la micro modulation pour chaque événement de l'espace méso.

Le patch *poly~* dans MSP reçoit les informations du temps de départ sur chaque événement méso, et le chevauchement de différents événements dans les espaces méso seront assignés dynamiquement à un exemple de voix *poly~* libre.

3.3. La construction des sources de modulation

Les générateurs de courbes dans Cosmos sont des sources de modulation à être placées selon les paramètres de synthèse qui sont assignés à des entités sonores construites avec l'espace de l'événement. Il y a quatre différents générateurs de courbes pour les trois différentes échelles de temps d'espaces macro, méso et micro.

Il en découle que nous pouvons combiner différentes sources de modulation appartenant à trois différentes échelles de temps, allant de macroscopique à microscopique, pour obtenir une complexité plus grande. Chaque générateur de courbe peut être assigné à une destination unique, et s'il y avait plus qu'une source de modulation pour une destination, elle serait superposée à une source de modulation avec la complexité de représenter l'évolution de multiples échelles de temps.

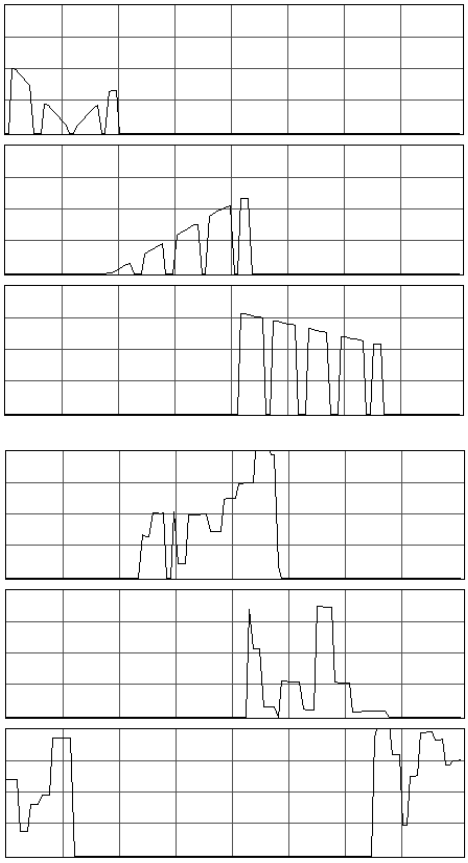

Figure 7. Les trois affichages supérieurs de MSP montrent les générateurs de courbes au niveau macro pour les trois premiers événements de l'espace macro. Chaque espace a quatre événements méso. Les pentes sont douces parce qu'il n'y a que la génération de courbes au niveau macro. Dans la figure ci-dessous, les générateurs de courbes du niveau méso sont superposés aux courbes du niveau macro, pour que la courbe montre les composants fractals comme résultat de génération de courbes de multiples échelles de temps.

Les valeurs d'ordonnées des courbes sont générées avec les fonctions stochastiques dans Cosmos, elles sont similaires aux segments de lignes des générateurs 'glissandi' dans le modèle Achorripsis de Xenakis. Les fonctions dans l'équation (4) sont conduites par des fonctions stochastiques.

$$f_{offsetdist} + f_{Vdist} * dur_{event} = y_{new} \quad \textit{(Achorripsis)}$$

$$(f_{Ynew} - y_{old}) / dur_{event} = v_{glissandi} \quad \textit{(Cosmos)} \tag{4}$$

$f_{offsetdist}$: Fonction qui distribue la valeur "offset"

f_{Vdist} : Fonction qui distribue la vitesse de glissandi

y_{new} : La valeur y finale du générateur du courbe

y_{old} : La valeur y précédente du courbe

Avec le temps de départ de chaque événement, les quatre nouvelles valeurs d'ordonnée sont assignées à des fonctions de courbes, pour qu'elles dessinent un segment de ligne commençant de la valeur d'ordonnée précédente vers la nouvelle donnée d'ordonnée à l'intérieur d'un intervalle de temps définit comme une durée d'événement. Pour résumer, l'événement macro a 4 générateurs de courbes indépendants, où les événements méso à l'intérieur de cet événement macro ont chacun 4 générateurs de courbes indépendants, ce qui est identique pour les événements micro. Dans une situation où 4 événements macro se chevauchent, avec 5 densités d'événements méso dans chacun, et 6 densités d'événements micro dans chaque événement méso, 120 générateurs de courbes sont distribués sur de multiples échelles de temps respectivement au total, ce qui donne une idée de la complexité du système.

4. CONCLUSIONS

Il y a un grand potentiel dans l'investigation des résultats sonores de l'application Cosmos. Le concept d'ordonner le désordre à travers la manipulation de ce système complexe de génération d'événements donne la possibilité de composer le son en introduisant des aspects importants. La mise en application d'autres mécanismes, puis les systèmes stochastiques pour la génération d'événements et la recherche de manipulation au niveau Meta de l'application seront les prochaines étapes.

5. REFERENCES

[1] Xenakis, I. *Formulized Music - Revised Edition* Pendragon Press , New York, 1992.

[2] Di Scipio, A. "The Problem of 2nd order Sonorities in Xenakis' electroacoustic music" in Organised Sound, 2(3) : 1997, pp 165-178

[3] Bregman, A.S.; *Auditory Scene Analysis: the perceptual organization of sound*. MIT Press, Cambridge, MA, 1990.

[4] Holland, J.H. *Emergence from Chaos to Order*, Oxford University Press, 1998.

[5] Morin, E. *Teorie dell' evento*, Milano, Bompiani, 1972, p. 31.

[6] Bokesoy, S. ; Pape, G. "Stochos: Software for Real-Time Synthesis of Stochastic Music" in *Computer Music Journal*, 27(3) : 2003, pp.33-43