

PRESENTATION DES TEXTES POUR LES JIM

Auteur 1
Organisme
Adresse électronique

Auteur 2
Organisme
Adresse électronique

Auteur 3
Organisme
Adresse électronique

RÉSUMÉ

Le résumé doit être placé en haut de la colonne gauche et doit contenir entre 150 et 200 mots.

1. INTRODUCTION

Nécessité d'un outil de conception spatiale généralisé répondant aux besoins de la muséographie de la musique spatiale de la robotique des outils de spectacle

Présentation d'un modèle adapté pour l'exécution, intégré avec les développements précédents -> quelle interaction de l'espace et du temps ? Problème de l'écriture et des outils graphiques

Problème de la visualisation de l'écoulement du temps en 2D (animation).

1.1. Existant

Une présentation de l'état actuel de l'écriture spatiale en musique est donnée dans [4]. Notamment, la question de la notation dans le cadre de partitions impliquant des éléments spatiaux est abordé. Ces partitions peuvent être spatiales uniquement dans leur représentation, mais peuvent aussi indiquer des manières d'interpréter dans l'espace, notamment à l'aide de symboles spécialisés [3]. Une taxinomie des possibilités de création dans l'espace en musiques électro-acoustiques est présentée par Bertrand Merlier dans [9]. Elle est étendue dans l'ouvrage *Vocabulaire de l'espace en musique électro-acoustique*[8].

Des outils logiciels existent pour ces partitions – ils sont souvent spécialisés. Par exemple, la bibliothèque **ENP!** (**ENP!**)[7] permet de concevoir des partitions graphiques telles qu'en fig. ?? à l'aide d'un éditeur lui aussi graphique et d'un langage basé sur LISP.

Une des problématiques actuelles pour la représentation de l'écriture musicale est celle du geste, et de son lien avec la partition : comment notamment annoter le geste du musicien avec précision ? Et, inversement, comment à partir d'un geste créer un son correspondant ? Ces questions sont abordées dans la description de Soundstudio 4D[12], dans le cadre d'un système de conception de trajectoires pour spatialisation à l'aide d'interactions en trois dimensions.

Une autre question est l'association entre l'aspect graphique et le résultat. Ainsi, des outils tels que HoloEdit et HoloSpat permettent de travailler avec des trajectoires, mais sont extrêmement spécialisés pour des objets audio.

C'est notamment du à la nécessité de composer en ayant conscience à chaque instant des fortes contraintes techniques du moyen de restitution de l'œuvre. Il serait intéressant d'utiliser ces trajectoires pour contrôler non pas des sources sonores mais des éléments dans des espaces de paramètres quelconques.

Le logiciel IanniX[5] (fig. ??) dispose aussi de nombreuses possibilités d'écriture spatiale : les partitions sont des ensembles d'éléments graphiques définis paramétriquement ou bien à l'aide d'un langage de programmation dédié, que des curseurs vont parcourir. L'information de position de chaque curseur est envoyée en OSC, ce qui permet l'intégration à d'autres logiciels.

Une méthode d'écriture de la spatialisation par contraintes est proposée par Olivier Delerue avec le système MusicSpace[2]. Cela permet une approche déclarative à l'écriture de partition, en spécifiant des contraintes telles que « deux objets ne doivent jamais être à plus de deux mètres l'un de l'autre » ou bien « l'angle entre deux objets et l'auditeur doit être supérieur à 90 degrés ». Les objets peuvent être notamment des sources sonores. Une édition graphique de ces contraintes est proposée, et elles sont représentées en termes de cercles et de segments reliant les objets qu'elles contraignent.

Des données spatiales peuvent aussi être utilisées directement pour créer des mappings sonores. C'est le cas notamment de la bibliothèque Topos[10], qui permet de capter le mouvement de danseurs et d'en extraire des informations pouvant être utiles pour la conception de pièces de musique interactives. Une fois que le mouvement du danseur est capturé via un périphérique externe, il devient possible d'extraire des informations telles que le volume occupé par le danseur, sa vitesse, ou bien diverses mesures relatives à l'évolution de deux ou quatre points dans le temps, comme l'instabilité ou les collisions entre différentes parties du corps. Ces données peuvent ensuite être réutilisées dans Pure Data pour de la génération de musique.

Enfin, il convient de noter la richesse pour ce qui est des modes d'entrée et d'interaction. Par exemple, il existe plusieurs possibilités de composition musicale à l'aide de tables interactives comme la Reactable[6] et différentes approches dérivées qui peuvent être spécifiquement axées sur la spatialisation du son[11].

Un modèle plus complet d'espace en trois dimensions est fourni par COSM[13]. Implémenté dans Max/MSP, il offre une grande richesse d'écriture. En plus de lieux et de trajectoires, il est possible d'écrire l'interaction dans une

certaine mesure, ainsi que la communication entre différents agents. Une nouveauté est la possibilité de travailler avec des champs définis mathématiquement. Ces champs peuvent varier dans le temps et être sonifiés par la suite.

Une approche de contrôle spatial est possible via le logiciel Blender, qui sert à l'origine à réaliser des images et films de synthèse. Blender peut être contrôlé via une API Python et il est notamment possible de déplacer des éléments et tester pour des collisions ou d'autres propriétés géométriques. Néanmoins, cela se fait à la vitesse de son moteur d'exécution qui est fixée à 60 Hz. Les messages reçus entre deux trames sont accumulés.

Enfin, le monde des jeux vidéos dispose aussi d'outils adaptés à l'écriture spatiale : la bibliothèque OpenAL a été développée à l'origine pour offrir aux jeux une couche d'abstraction permettant de bénéficier de spatialisation simplement en donnant une position et une orientation à des sources sonores ponctuelles. Cette position et orientation peuvent évoluer dans le temps. Par la suite les implémentations sont libres d'utiliser les méthodes qu'elles souhaitent pour réaliser la spatialisation : cela peut aller d'un simple panning gauche-droite à l'utilisation de HRTFs, comme le fait la bibliothèque OpenAL-soft.

2. EXEMPLES

2.1. Exemple : sonopluie

Utilisation de OpenAL

Description du scénario :

On a plusieurs sources sonores dans un espace. Plusieurs personnes se déplacent et sont géolocalisées. Lorsqu'elles se rapprochent d'une source, elles l'entendent plus fort.

On désire utiliser i-score pour l'écriture de tels scénarios. Par la suite, il faut scénariser : par exemple, une fois qu'une zone a été passée on veut activer d'autres zones pour faire un parcours et non une simple balade.

2.2. Exemple : robots

Avec logiciel de simulation à côté (simule pannes, etc.).

On conçoit une chorégraphie de robots et drones.

Ils ont chacun une trajectoire qui peut évoluer dans le temps, en fonction de ce qu'il se passe.

Les drones portent des hauts-parleurs tandis que les robots font du rythme en tapant des pieds.

3. MODÈLE

3.1. Conception

On veut :

- une intégration avec l'écosystème existant.
- Décrire finement des évolutions dans le temps (ou en fonction d'autres paramètres).
- pouvoir écrire des scènes 2D, 3D basiques potentiellement en utilisant des guides (GMaps, image).

- animer soit via le temps soit via d'autres paramètres (mapping).

3.1.1. Choix du modèle

RCC8 envisagé mais n'est adapté principalement que pour de la reconnaissance. Ici on veut faire de l'écriture.

Fournir la possibilité d'écrire directement des zones en code : via extension, permet moins de possibilités d'analyse par la suite.

Le plus simple pour mappings (du temps ou autre) : les représenter à part.

3.2. Description

Approche paramétrique, non focalisée sur le son.

On définit des zones par un ensemble d'équations que l'on peut paramétrer.

Chaque équation possède des variables.

Pour chaque variable, on choisit si elle correspond à une dimension de l'espace, à un paramètre fixe, ou à une adresse externe.

Notion de zones : modèle générique, puis spécialisations.

Pas de curseur de temps : si on désire manipuler du temps, on peut le faire directement par OSC car c'est un paramètre explicite.

Fonctionne pour > 3 dimension avec notion de viewport. -> rendu par voxelisation ? ou méthode d'approximation de polygones / surfaces ?

Donner l'écriture de trajectoires simples (sin, etc).

$$y = \sin(x) \text{ et } x < -t$$

ou

$$x = \cos(t) \text{ } y = \sin(t)$$

Écriture de la masse spatiale avec des aires / volumes ?

Utilisation d'un arbre OSC.

Zones, puis calculs sur ces zones.

Possibilité d'écrire de tels calculs (mappings) en JavaScript.

Exemple : calcul de collision ; calcul de distances.

Rendre apparents les calculs ?

Les trajectoires sont définies à côté et font un mapping du temps vers l'espace.

Problème avec GiNaC : pas de multi-thread.

4. IMPLÉMENTATION

4.1. Sémantique d'exécution

Sémantique d'exécution : résultats obtenus au tick d'après.

Discuter les multiples sémantiques possibles : mise à jour en temps réel, ou bien "State".

Dans le premier cas : Le logiciel i-score expose son arbre et les processus spatiaux sont des parties intégrantes de l'arbre ; elles se mettent à jour de manière réactive lorsqu'un message est reçu et déclenchent une chaîne d'évaluation qui va de calculs en calculs.

Dans le second cas, on suit la sémantique des processus i-score qui est par "State" agissant à l'exécution. Cela

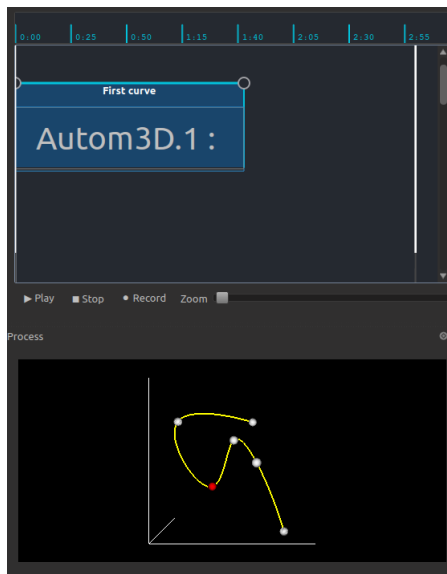


Figure 1. Une spline 3D dans i-score. On affiche la courbe dans le panneau inférieur

pose le problème de la visualisation : on ne peut pas voir lorsqu'on édite un scénario l'effet des changements de paramètres.

Cependant, si on a envie de voir le rendu en temps réel à l'écriture, il faut que l'outil lui-même s'expose de manière permanente via OSC et soit à l'écoute des données de l'arbre.

Dans un tick, on calcule d'abord les valeurs actuelles des zones.

Puis on exécute les computations qui écrivent dans l'arbre.

Si plusieurs processus utilisent la sémantique d'i-score (les processus sont empilés).

Nécessité d'un graphe d'exécution pour l'ordre ?

Affichage : pour l'instant simple.

Par la suite, utiliser VTK ?

Ne spécifier qu'une équation (ex : $x=0$ pour une ligne) ?

Afficher le résultat d'un solving.

(ex. : $y < 1+x^2$ and $y > -1-x^2$: si on a un curseur x , y , pour vérifier la collision on remplace juste les variables et on vérifie que ce soit vrai.)

TODO faire un identifiant "any" pour la collision de pointeur vers tout

nécessité de trouver une bibliothèque pour l'affichage rapide

Animation : la durée d'une boîte correspond à la durée d'une spline d'animation ?

4.2. Édition

L'édition va modifier pour les formes connues des paramètres définis.

Les opérations de base sont déplacement, translation, rotation.

On applique les transformations à l'objet :

— La translation est triviale.

— Rotation par un angle θ dans le plan :

$$\begin{cases} x' = x \cos \theta - y \sin \theta \\ y' = x \sin \theta + y \cos \theta \end{cases}$$

Une formule générale de la rotation en n -dimension est donnée dans [1].

— Mise à l'échelle : pas de formule générale ; on fait un rendu et on met à l'échelle par la suite ?

5. CONCLUSION

6. REFERENCES

- [1] Antonio Aguilera and Ricardo Pérez-Aguila. General n -dimensional rotations. 2004.
- [2] M Olivier DELERUE. Spatialisation du son et programmation par contraintes : le système MusicSpace.
- [3] Emile Ellberger, Germán Toro Perez, Johannes Schuett, Giorgio Zoia, and Linda Cavaliero. Spatialization symbolic music notation at ICST.
- [4] Dominique Fober, Jean Bresson, Pierre Couprie, and Yann Geslin. Les nouveaux espaces de la notation musicale. In *Journées d'Informatique Musicale*.
- [5] Guillaume Jacquemin, Thierry Coduys, and Matthieu Ranc. Iannix 0.8. pages 107–15.
- [6] M Kaltenbranner, Sergi Jorda, Gunter Geiger, and Marcos Alonso. The reactable : A collaborative musical instrument. In *Enabling Technologies : Infrastructure for Collaborative Enterprises, 2006. WE-TICE'06. 15th IEEE International Workshops on*, pages 406–411. IEEE.
- [7] Mika Kuuskankare and Mikael Laurson. Expressive notation package. 30(4) :67–79.
- [8] Bertrand Merlier. *Vocabulaire de l'espace en musiques électroacoustiques*.
- [9] Bertrand Merlier. VOCABULAIRE DE L'ESPACE ET DE LA SPATIALISATION DES MUSIQUES ÉLECTROACOUSTIQUES : PRÉSENTATION, PROBLÉMATIQUE ET TAXINOMIE DE L'ESPACE. In *EMS : Electroacoustic Music Studies Network ?Beijing 2006*, page 247.
- [10] Luiz Naveda and Ivani Santana. ?topos ? toolkit for pure data : exploring the spatial features of dance gestures for interactive musical applications.
- [11] Yuya Sasamoto, Michael Cohen, and Julian Villegas. Controlling spatial sound with table-top interface. In *Awareness Science and Technology and Ubimedia Computing (iCAST-UMEDIA), 2013 International Joint Conference on*, pages 713–718. IEEE.
- [12] James Sheridan, Gaurav Sood, Thomas Jacob, Henry Gardner, and Stephen Barrass. Soundstudio 4d : A VR interface for gestural composition of spatial soundscapes. In *ICAD*.

- [13] Graham Wakefield and Wesley Smith. *Cosm : A toolkit for composing immersive audio-visual worlds of agency and autonomy*. Ann Arbor, MI : MPublishing, University of Michigan Library. 00007.