

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

COSM
IanniX
Blender
Trajectoires
Dans jeux vidéos, etc.

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.

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 paramétriser.

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).

$y = \sin(x)$

et $x < t$

ou

$x = \cos(t)$ $y = \sin(t)$

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

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 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 on utilise 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 ?

5. CONCLUSION

6. REFERENCES

- [1] Author, E. "Titre du papier", *Proceedings of the International Symposium on Music Information Retrieval*, Plymouth, USA, 2000.
- [2] Untel, A. *Titre du livre*. L'Armada, Paris, 2005.

4.1. Édition

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

Les opérations de base sont déplacement, translation, rotation.