



HAL
open science

Logiciel de programmation par contrainte pour l'assistance à la composition

Antoine Bonnet, Camilo Rueda

► To cite this version:

Antoine Bonnet, Camilo Rueda. Logiciel de programmation par contrainte pour l'assistance à la composition. Journées d'informatique musicale 1994, Mar 1994, Bordeaux, France. hal-03347022

HAL Id: hal-03347022

<https://hal.science/hal-03347022>

Submitted on 17 Sep 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.

SITUATION

Logiciel de programmation par contrainte pour l'assistance à la composition

Antoine Bonnet
Camilo Rueda

Département représentation musicale
IRCAM
rueda@ircam.fr

APPROCHE GENERALE DU PROJET

a- Contexte

Alors que de nombreuses recherches ont été consacrées à la synthèse ou à l'interaction machine/instrument, presque rien n'a été entrepris dans le domaine de la composition assistée par ordinateur : les logiciels musicaux sont essentiellement axés sur l'édition, l'analyse, la performance ou le contrôle MIDI (*Finale, Note-writer, Performer, Max...*).

Seul *Patchwork* représente un effort dans le domaine de la CMAO. C'est cependant un outil de bas niveau qui ne permet pas de dépasser le stade d'élaboration du matériau : il interdit toute interaction dynamique des données. En outre, valorisant par défaut une conception *fonctionnelle* de la composition, il n'est pas neutre et ne peut apporter une aide substantielle qu'à certains types de problèmes très spécifiques.

b- Objectif

L'objectif était la conception d'un mode de représentation musicale assez général et de haut niveau pour permettre l'élaboration de processus compositionnels complexes et interactifs. Plus précisément, il s'agissait de construire un **outil d'aide à la composition** :

- neutre eu égard aux choix esthétiques,
- permettant la *programmation par contraintes*
- et comportant une *représentation explicite du temps* (actuellement, non encore réalisée)

IDÉES FORTES

a. Limites actuelles de la catégorie de *note*

On fait l'hypothèse que la catégorie de base de l'écriture musicale n'est plus la note.

Démonstration (schématique...) :

Toute pensée musicale suppose une *écriture*.

Toute écriture suppose une *notation*.

La catégorie fondamentale de la notation, en Occident, était la note.

La note était également la catégorie fondamentale de l'écriture puisqu'elle permettait de représenter, avec les paramètres de hauteur et de durée qui la formaient, les deux dimensions constitutives de la pensée musicale : horizontale et verticale.

Les compositeurs, depuis la seconde moitié du XVIII^e siècle, font preuve d'un intérêt croissant pour de nouvelles dimensions du son (intensité, profil dynamique, timbre...) et, depuis Wagner et Debussy, manifestent le désir de fonder, en un jaillissement qu'on a parfois qualifié de *diagonal*, les dimensions horizontales et verticales.

Aujourd'hui, bien qu'en charge d'un plus grand nombre d'indications paramétriques, la note ne parvient plus ni à représenter la complexité sonore imaginée, ni à initier de manière vraiment maîtrisée des constructions symboliques susceptibles de la représenter.

La musique de notre temps n'est par conséquent plus pensable à partir de la catégorie élémentaire de note; plus précisément, si la note demeure pour la musique instrumentale la catégorie de base de la notation, elle n'est plus la catégorie de base de l'écriture.

Pour avoir une représentation symbolique en accord avec la complexité du phénomène sonore de notre temps, il faut changer d'échelle, remplacer la note par une entité de plus haut niveau, la formaliser, la représenter et la tester en fonction de ses instanciations possibles.

b- Le concept de *situation*

Nous proposons le terme situation pour désigner la catégorie de base de l'écriture musicale d'aujourd'hui.

Définition : La situation est une entité dynamique et multi-dimensionnelle, pensable comme le mouvement interactif de ses éléments.

Cette conception s'accorde avec la physique/acoustique moderne qui ne conçoit plus le son comme un *objet statique* défini dans un *espace absolu*, mais comme un *phénomène dynamique* évoluant dans un *espace relatif*.

La situation nous semble recouvrir des entités proposées par des compositeurs d'esthétique très différente; elle paraît notamment englober ce que Tristan Murail appelle un objet complexe, ce que Marco Stroppa appelle un organisme, ce qu'un électro-acousticien appelle un timbre etc...

Nous préférons le terme situation au terme objet, à la fois trop neutre et trop inféodé à la terminologie informatique; au reste, informatiquement parlant, une situation n'est pas un objet mais une classe.

Tout objet élémentaire (une hauteur, une durée etc...) se définit donc ici comme une situation ne comportant qu'un seul élément.

En outre, le mot situation laisse entendre, à juste titre, une capacité à concevoir n'importe quel espace dans n'importe quelle dimension et à n'importe quel niveau.

D'un point de vue conceptuel, une situation se conçoit dans l'espace (au sens mathématique du terme) à partir de deux notions :

le *grain* (ou coupure du continuum), qui définit, pour tout paramètre ou toute dimension, une valeur unitaire de référence.

la *distance* (ou intervalle), qui définit, pour tout paramètre ou toute dimension, toute occurrence par rapport à toute autre.

c- Question :

Un des objectifs centraux de ce projet est de construire un outil assez *général* pour que puissent en bénéficier tous les compositeurs, quelque soit leur esthétique.

Nous proposons le concept de situation parce qu'il nous semble que ce qu'imagine un compositeur aujourd'hui, ce qu'il a en tête, ce n'est plus une idée réductible à un agencement de hauteurs et de durées mais un ensemble beaucoup plus complexe dont les éléments interagissent dans le temps.

Ces modes d'interaction nous paraissent obscurs, contrairement à ceux de la musique d'hier, au moins assez bien cernables à partir de la catégorie "mixte" (horizontale et verticale) de contrepoint. La non-maîtrise des modes d'interaction d'aujourd'hui est sans aucun doute une des causes majeures des "difficultés" de notre musique. Nous pensons que l'outil que nous envisageons permettrait de mieux contrôler ces modes d'interaction et, qui sait!..., de définir des *catégories* nouvelles de composition (il y avait autrefois *note* et *contrepoint*; il pourrait y avoir demain quelque chose comme *situation* et ...).

Il conviendrait donc, en premier lieu, que chaque compositeur s'interroge sur le bien-fondé de l'hypothèse, qu'il la mette à l'épreuve de son propre imaginaire : ce qu'il imagine lorsqu'il compose, ce qu'il appelle peut-être son idée musicale, son univers sonore, son projet formel... bref, toute image qu'il peut avoir quand il compose lui semble-t-elle être représentable et bénéfiquement corvéable dans le cadre que nous proposons?

Plus précisément, peut-il définir, à partir de relations, ce qu'il imagine ou souhaite?

Si non, de quoi d'autre?

La manière dont sont envisagées ici les relations lui paraît-elle satisfaisante?

Le concept de situation permettant *en principe* de représenter, hiérarchiquement ou non, n'importe quel niveau (du simple élément à l'oeuvre entière), subodore-t-il, à un niveau ou à un autre qu'il pourrait vouloir représenter, des impossibilités?

Lui paraît-il problématique de considérer tout objet simple (une hauteur par exemple) comme une situation réduite à un point, autrement dit comme un ensemble d'un seul élément?
etc...

DESCRIPTION DE L'ETAT ACTUEL DE SITUATION

1.Introduction

Situation est un moteur de satisfaction de contraintes écrit en Common-Lisp-CLOS et muni d'un interface graphique en PatchWork adapté à la construction de séquences harmoniques et rythmiques.

Situation est particulièrement pertinent pour la génération des objets musicaux lorsque :

- chaque objet dans une séquence est une entité décrite par un ensemble donné de propriétés reliant ses éléments;
- des objets dans des positions sélectionnées de la séquence doivent satisfaire un certain nombre de contraintes;
- les relations entre les objets établissent des contraintes souhaitables mais pas forcément obligatoires.

Situation permet ainsi la construction de séquences musicales à partir de la description de leurs propriétés - contrairement à l'approche fonctionnelle qui nécessite la spécification précise de l'algorithme de construction.

2.Principe : Contraintes et Instanciation d'Objets Musicaux

La construction et la manipulation d'objets musicaux se trouvent au centre de l'activité compositionnelle. Ces objets se conçoivent généralement en terme de relations entre leurs composantes. Dans les cas les plus simples, la construction d'objets musicaux se fait de façon

convenable au moyen de l'élaboration d'un algorithme spécifique, c'est-à-dire qui sache placer chaque composante selon la relation à satisfaire.

Quand la relation est complexe ou quand un nombre important de relations doit être satisfait, la définition de l'algorithme adéquat est lourde et difficile. La meilleure solution est alors de décrire les relations souhaitées afin de laisser la machine construire le bon objet. Les systèmes de résolution de contraintes, tel que Situation, ont cette approche.

Situation est conçu pour un utilisateur non expérimenté en informatique (si l'on fait exception des quatre modules cités plus haut, qui, eux, sont expérimentaux, et réclament une connaissance de la programmation en Common Lisp). L'inconvénient de cette exigence est le temps parfois très long nécessaire à l'obtention d'un bon résultat. Pour un environnement à vocation interactive, c'est un problème à prendre en considération.

Par souci d'efficacité, Situation a donc été optimisé en fonction des relations harmoniques et mélodiques les plus probables compte tenu du parti pris de départ de placer la notion d'accord au centre du projet. La possibilité de paramétrer les contraintes avec l'interface PatchWork en témoigne. Naturellement, les relations y figurant ne sont pas les seules possibles : leur présence est plus une question d'efficacité que de capacité.

Très généralement, on peut envisager le processus de résolution d'un problème de satisfaction de contraintes comme celui des raffinements successifs des solutions partielles. Pour trouver une séquence harmonique particulière, on pourrait partir d'une "solution" comportant toutes les séquences d'accords possibles et filtrer celles-ci par des contraintes permettant de sélectionner autant de sous-ensembles. Une solution valide serait ainsi l'intersection de tous les sous-ensembles.

L'utilisateur doit donc tenir compte du fait que si l'ensemble initial est plus restreint, le processus de résolution est plus efficace. Trouver par exemple des séquences particulières d'accords de sept hauteurs est beaucoup plus efficace si l'on impose un certain registre aux accords que si l'on considère au départ tous les accords possibles. Cet ensemble initial est appelé *Domaine* dans le jargon informatique.

Notion de contraintes "faibles"

L'interaction des contraintes données en entrée du module *harmonic-constraints* peut dans certains cas définir un problème inconsistant, sans aucune solution possible. Le module rendra nil en sortie mais après avoir exploré tout l'espace de recherche, ce qui peut entraîner un temps de calcul très important. Or, il n'est pas toujours évident de trouver la façon d'exprimer un problème en évitant cet écueil.

Situation propose donc de hiérarchiser les contraintes selon une échelle d'importance ou de nécessité. L'on peut associer à chaque contrainte un nombre entre zéro et 1 qui indique son importance relative. Une valeur de 1 (la valeur par défaut) signifie une contrainte obligatoire (à 100%); une valeur de zéro rend la contrainte inutile; une valeur intermédiaire donne son importance par rapport aux autres. Certaines contraintes s'inscrivent à l'intérieur d'une liste commençant par :

(n <degré> ...)

Ceci donne à la contrainte concernée un nombre <degré> correspondant à sa valeur de nécessité. Ce schéma induit aussi une hiérarchisation des solutions au problème. Une solution à valeur 1 est celle où toutes les contraintes ont été satisfaites. Une solution à valeur v inférieure à 1 est celle où n'a pas été satisfaite, à un degré de nécessité inférieur ou égal à $1 - v$, la contrainte non obligatoire la plus importante.

Les solutions sont calculées en ordre croissant de valeur.