HAL open science

# DIFFÉRENTES UTILISATIONS DE L'ESPACE EN MUSIQUE À L'AIDE D'UN LANGAGE DE PROGRAMMATION DÉDIÉ AU CALCUL SPATIAL

Louis Bigo, Antoine Spicher, Olivier Michel

**To cite this version:**

Louis Bigo, Antoine Spicher, Olivier Michel. DIFFÉRENTES UTILISATIONS DE L'ESPACE EN MUSIQUE À L'AIDE D'UN LANGAGE DE PROGRAMMATION DÉDIÉ AU CALCUL SPATIAL. Journées d'Informatique Musicale, May 2011, Saint-Etienne, France. hal-03104801

**HAL Id: hal-03104801**

**https://hal.science/hal-03104801v1**

Submitted on 9 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.

# DIFFÉRENTES UTILISATIONS DE L'ESPACE EN MUSIQUE À L'AIDE D'UN LANGAGE DE PROGRAMMATION DÉDIÉ AU CALCUL SPATIAL

*Louis Bigo*
LACL - IRCAM
Université Paris Est Créteil
Louis.Bigo@lacl.fr

*Antoine Spicher*
LACL
Université Paris Est Créteil
Antoine.Spicher@u-pec.fr

*Olivier Michel*
LACL
Université Paris Est Créteil
Olivier.Michel@u-pec.fr

**RÉSUMÉ**

Dans cet article, nous proposons de reformuler à l'aide de la notion d'espace des problèmes portant sur des objets musicaux. La construction et la manipulation d'espaces pour ce type d'étude se fait à l'aide du langage de programmation MGS dédié au calcul spatial que nous présenterons, avant de nous pencher sur trois problèmes musicaux.

## 1. INTRODUCTION

Le *calcul spatial* est un domaine de recherche émergeant qui vise à développer des formalismes, des langages et des architectures matérielles permettant de prendre en compte la notion d'espace en informatique, que ce soit comme ressource ou comme résultat. Plus particulièrement, ce domaine cherche à manipuler de manière *intentionnelle* des objets *spatialement structurés*.

Un certain nombre de travaux dans le domaine de la musique, notamment concernant différents aspects de sa nature mathématique, ont eu recours à des outils géométriques [1] ou topologiques [2]. Nous proposons ici de montrer que l'utilisation de l'espace en musique peut s'exprimer à différents niveaux de manière élégante dans un langage de programmation dédié au calcul spatial.

Dans la section 2 nous proposons une introduction rapide au langage de programmation MGS. Puis à travers différents exemples, nous montrerons section 3 comment ce langage, et plus largement le calcul spatial, constituent des outils pertinent pour la formalisation de problèmes musicaux.

## 2. PRÉSENTATION DU LANGAGE MGS

MGS est un langage déclaratif de programmation expérimental [3, 4] proposant l'utilisation de règles de réécriture pour la manipulation de structures de données spatialement organisées. Les concepts de MGS sont fondés sur des notions empruntées topologie algébrique [5] et ont été validés dans le cadre de la modélisation de systèmes complexes.

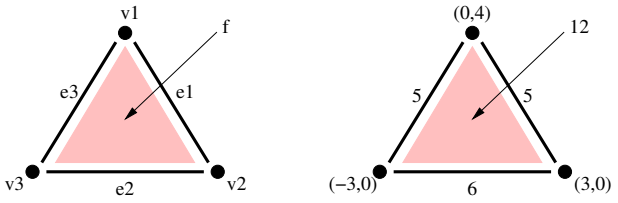



**Figure 1**. Sur la gauche, un exemple de complexe cellulaire : il est composé de trois 0-cellules ($v_1$, $v_2$, $v_3$), de trois 1-cellules ($e_1$, $e_2$, $e_3$), et d'une 2-cellule $f$. Le bord de $f$ est constitué de ses cellules incidentes $v_1$, $v_2$, $v_3$, $e_1$, $e_2$ et $e_3$. Plus particulièrement, les trois arcs sont les faces de $f$, et par conséquent, $f$ est une coface de $e_1$, $e_2$ et $e_3$. Sur la droite, des données sont associées aux cellules topologiques : des positions pour les sommets, des longueurs pour les arcs et une aire pour $f$.

### 2.1. Collections topologiques

En MGS, toutes les structures de données sont unifiées sous la notion de *collection topologique* [6]. Une collection topologique est un *complexe cellulaire* décoré par des valeurs arbitraires. Le complexe cellulaire a le rôle de support et les valeurs correspondent aux éléments de la structure de données.

Un *complexe cellulaire* est un espace construit en assemblant des espaces élémentaires appelés *cellules*. Une cellule correspond à l'abstraction d'un espace pour une dimension donnée. En particulier, les 0-cellules sont des sommets, les 1-cellules des arcs, 2-cellules des surfaces, 3-cellules des volumes, etc. Ainsi, les graphes sont un exemple de complexe cellulaire de dimension 1. Un autre exemple est illustré sur la partie gauche de la Figure 1.

Dans un complexe cellulaire, les cellules sont organisées suivant leur *relation d'incidence*. Cette relation (définissant un ordre partiel sur les cellules) est proche de la notion intuitive de *bord* : le bord d'une cellule correspond à l'ensemble des cellules de plus petite dimension qui l'entourent dans le complexe. Par exemple, le bord d'un arc (*i.e* une 1-cellule) est composé de deux sommets (*i.e* deux 0-cellules). Le bord d'une surface triangulaire correspond à trois 1-cellules et trois 0-cellules. Deux cellules sont dites *incidentes* si l'une appartient au bord de l'autre. En particulier, une $p$-cellule $c$ est une *face* d'une

$q$-cellule $c'$, ce qui est noté $c < c'$, si elles sont incidentes et si $p = q - 1$. La cellule $c'$ est appelée *coface* de $c$ (voir exemple figure 1). Des voisinages plus élaborés peuvent être définis à partir de la relation d'incidence. Deux cellules $c$ et $c'$ sont *$p$-voisines* s'il existe une $p$-cellule incidente à la fois à $c$ et $c'$. Si les deux cellules sont de dimension $n$, on dit qu'elles sont *$(n, p)$-voisines* . Un *$(n, p)$-chemin* est une séquence de $n$-cellules tel que deux cellules consécutives sont $(n, p)$-voisines. Par exemple, la notation usuelle d'un chemin dans un graphe correspond à la notion de $(0, 1)$-chemin : une séquence de sommets reliés par des arcs. Finalement, une *collection topologique* est un complexe cellulaire décoré. Un exemple de collection topologique est donné sur la partie droite de la Figure 1.

Les types de collection diffèrent au niveau de la spécification de leur complexe cellulaire sous-jacent. Dans l'implémentation actuelle du langage MGS, les structures de donnée usuelles (ensembles, séquences, arbres, listes, etc.) sont représentées par des graphes dont les sommets sont décorés par les éléments de la structure de données et dont les arcs représentent les accessibilités relatives depuis un élément vers un autre.

Les GBF [7] sont un exemple de type de collection topologique qui généralise la notion usuelle de réseau, en décorant un complexe cellulaire généré par un groupe mathématique. Un tel groupe $G$ peut être spécifié par sa présentation (*i.e* en termes de générateurs et de relations) et peut être représenté par un complexe cellulaire à 1 dimension appelé *graphe de Cayley* : chaque sommet du graphe correspond à un élément du groupe ; il existe un arc entre deux sommets $h$ et $h'$ si $h' = h + g$ où $g$ est un générateur appartenant à la représentation de $G$. Par exemple, la représentation suivante

```
<Quinte,TierceM;4*Quinte=TierceM,12*Quinte=0>
```

correspond au graphe cyclique de Cayley de la figure 2. Deux générateurs sont définis : `Quinte` et `TierceM`, correspondant respectivement aux directions diagonales et verticales sur la figure. La structure du groupe fournit les directions opposées aux générateurs. Ainsi, chaque sommet possède quatre 1-voisins. Nous ne considérerons ici que les groupes abéliens ; ainsi chaque paire de générateurs commute. Dans notre exemple nous avons

```
Quinte+TierceM=TierceM+Quinte
```

signifiant que partant d'un sommet, un même sommet est atteint que l'on suive successivement les directions `Quinte` puis `TierceM` ou bien `TierceM` puis `Quinte`. Sans relations entre générateurs, cette présentation génèrerait une grille carré infinie. Les deux relations rendent le graphe périodique. Suivre 12 fois la direction `Quinte` est équivalent à ne pas se déplacer ; suivre quatre fois la direction `Quinte` est équivalent à suivre la direction `TierceM` une fois.

### 2.2. Transformations

Les collections topologiques sont transformées à l'aide d'ensembles de règles appelés *transformations*. Une règle est une paire `motif => expression`. Lorsqu'une règle est appliquée sur une collection topologique, les sous-collections respectant ce motif sont remplacées par la collection topologique calculée par l'évaluation de `expression`. Il existe différents moyens, appelés stratégies, pour contrôler l'application d'un ensemble de règles sur une collection. Plusieurs stratégies différentes existent en MGS (mais les détails de la stratégie utilisée dans les exemples présentés plus bas ne sont pas nécessaire à leur compréhension). Il faut cependant noter qu'une stratégie ne détermine pas nécessairement complètement les occurrences d'application des règles : on dit alors que le programme est non-déterministe. Dans ce cas, MGS calcul l'un des résultats possibles et une autre exécution du même calcul peut donner un résultat différent. Une présentation formelle du mécanisme de réécriture est donnée dans [8]. On se restreindra ici à la spécification des motifs.

Une *clause* spécifie une cellule devant respecter certaines contraintes dans la collection topologique. Une variable de motif `x` filtre une $n$-cellule $\sigma$ de la collection décorée par une valeur $v$. La variable permet alors de référer à l'élément filtré en partie droite de la règle :

- la variable `x` prend la valeur $v$
- la variable `^x` prend la valeur $\sigma$

Cette convention est arbitraire mais elle permet de simplifier l'expression du filtrage (moins de variables sont introduites) et de simplifier l'expression des parties droites (qui manipulent souvent uniquement les décorations). Par exemple, l'expression `c / c = 3` permet de filtrer toutes les cellules décorées par la valeur 3. La variable `c` peut être utilisée dans la contrainte (et partout ailleurs dans la règle) afin de spécifier la valeur de la cellule satisfaisant la règle. On utilisera la variable `^c` pour désigner la cellule elle-même.

Un motif est une liste de clauses séparées par des opérateurs. Il y a trois opérateurs de composition :

1. La composition spécifiée par une simple juxtaposition (*e.g.* "`x y`") n'impose pas de contrainte sur les membres de la composition.

2. Le dernier opérateur de composition correspond à l'opérateur de face : un motif "`x < y`" (resp. "`x > y`" filtre deux cellules $x$ et $y$ tel que $x < y$ (resp. $x > y$).

3. Lorsque deux clauses sont séparées d'une virgule (*e.g.* "`x,y`"), cela signifie que les cellules respectant les clauses `x` et `y` doivent être *$p$-voisines*.

Les motifs sont *linéaires* : deux variables distinctes du motif référencient toujours deux cellules distinctes.

## 3. QUELQUES CAS D'UTILISATION DE L'ESPACE EN MUSIQUE

Dans cette section, nous présentons différents exemples utilisant l'espace à différents niveaux pour l'analyse musicale.

### 3.1. Réseaux de notes

De nombreux travaux ont été réalisés autour de la théorie néoriemannienne, montrant la pertinence d'utiliser des réseaux de notes (souvent désignés par le terme *Tonnetz*) pour l'analyse musicale [9, 10, 11]. Ce type de représentation propose notamment une visualisation des classes de hauteur organisées spatialement suivant des critères de consonance. Un réseau de notes est organisé selon certaines directions correspondant à des intervalles. Le réseau est ainsi généré en faisant agir le groupe des intervales sur l'ensemble des notes [12]. Ainsi le tonnetz d'Euler (Figure 2) répartit les notes suivant les intervalles de tierce majeure et de quinte. Dans la suite, nous nous intéressons au réseau de note hexagonal généré par les intervales de tierce mineure, tierce majeure et quinte (Figure 3).

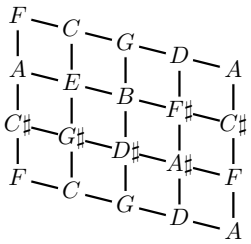



**Figure 2**. Le *Tonnetz* de L. Euler (1739). La direction diagonale correspond à l'intervale de quinte, celle vers le bas à l'intervale de tierce majeure.

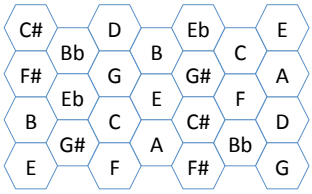



**Figure 3**. Réseau de notes hexagonal généré suivant les intervalles de tierce mineure, tierce majeure et quinte.

En MGS, les collections GBF décrites plus haut permettent facilement l'implémentation de tels réseaux de notes dans le contexte du calcul spatial. Il s'agit de considérer un complexe cellulaire décoré par les notes et généré par un groupe composé de certains intervalles. La spécification MGS du GBF, comme exposé dans la section 2, associé au réseau de note hexagonal de la Figure 3 est alors la suivante :

```
< Tiercem, TierceM, Quinte;
    Tiercem + TierceM = Quinte, 4*Tiercem = 0,
    3*TierceM = 0, 12*Quinte = 0 >
```

Le graphe de Cayley correspondant (Figure 3), est un maillage hexagonal révélant un grand nombre de propriétés dans le cadre de la musique tonale. On peut noter par exemple que les trois notes $do, mi, sol$, constituant un accord parfait majeur, constituent la forme la plus compact possible de l'espace.

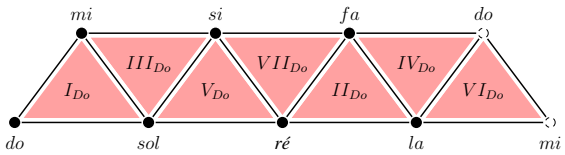



**Figure 4**. Représentation de la tonalité de Do majeur.

### 3.2. Séquences d'accords

Une analyse musicale mathématique repose généralement sur la définition d'un modèle en tant qu'objet mathématique, dont les propriétés spécifiques décrivent certaines caractéristiques musicales. Dans la section précédente, nous avons suivi cette approche pour définir des espaces topologiques utilisés afin de mettre en évidence certaines propriétés musicales. Dans cette section, nous nous concentrons sur le processus utilisé pour construire un tel modèle mathématique. Ainsi, on considère ici l'espace comme *résultat* d'un calcul spatial. On décrira en particulier un mécanisme d'auto-assemblage utilisé pour définir une représentation spatiale de la tonalité basée sur les accords triadiques. Puis on propose d'utiliser ce mécanisme pour l'étude de la tonalité basée sur des accords à quatre notes.

**Représentation de la tonalité.** Une tonalité peut être associée à une gamme de notes. Par exemple, la tonalité de Do Majeur est caractérisée par la gamme diatonique $(do, re, mi, fa, sol, la, si)$. A partir d'une telle gamme, on peut définir la tonalité par le recouvrement de ses notes par sept accords triadiques appelés les degrés. Dans la gamme de $Do$ Majeure, ces degrés sont les suivants :

$$I_{Do} = \{do, mi, sol\} \quad II_{Do} = \{re, fa, la\}$$

$$III_{Do} = \{mi, sol, si\} \quad IV_{Do} = \{fa, la, do\}$$

$$V_{Do} = \{sol, si, re\} \quad VI_{Do} = \{la, do, mi\}$$

$$VII_{Do} = \{si, re, fa\}$$

Cette définition de la tonalité peut être interprétée de manière géométrique. Chaque note de la gamme est représentée par un sommet. Puis, deux notes appartenant au moins à un des degrés sont reliées par un arc. Enfin, les accords triadiques sont identifiés dans ce réseau par des surfaces triangulaires. La Figure 4 représente cette interprétation géométrique de la tonalité de Do Majeur.

La surface résultante correspond à un ruban de Mœbius [13]. Cette particularité remarquable fût mis en évidence en premier lieu par Schoenberg et fût considérablement étudiée depuis pour l'analyse musicale. Parmi les propriétés de cette surface, on remarque que son bord unique (un $(0, 1)$-chemin fermé), révèle le cycle des quintes. Plus récemment, ce ruban a été utilisé comme outil pour

l'élaboration d'un modèle mathématique musicale basé sur un point de vue fonctoriel [2].

Le ruban de Mœbius s'avérant être une représentation géométrique pertinente des sept degrés triadiques et de leurs relations harmonique, nous proposons ici un algorithme d'auto-assemblage pour la construction systématique de telles structures topologiques représentant des ensembles arbitraires d'accords. L'idée générale est de laisser *réagir* les accords entre eux à la façon de molécules chimiques en solution, la règle de réaction consistant en l'identification d'éléments communs partagés par les accords. L'étude de l'objet final (obtenu au point de stabilité quand aucun accord ne peut plus réagir) peut révéler d'intéressantes propriétés sur l'ensemble d'accords.

**Représentation d'accords.** On propose de représenter un accord de $n$ notes par un $(n-1)$-simplexe. Un *$p$-simplexe* est une $p$-cellule qui possède exactement $p+1$ faces. Par exemple, un arc est un simplexe mais un cube n'en est pas un (un cube est une 3-cellule mais a 6 faces). Les simplexes sont souvent représentés géométriquement par l'enveloppe convexe de leurs sommets comme présenté en Figure 5 pour les $p$-simplexe avec $p \in \{0,1,2,3\}$. Dans le cadre de la représentation simpliciale des accords, une 0-cellule représente une note seule, une 1-cellule un accord de deux notes, etc.

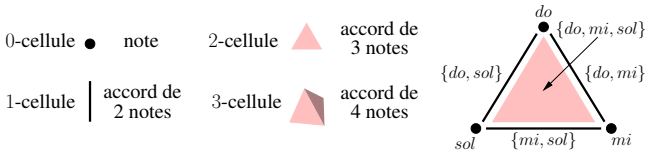



**Figure 5**. Représentation d'un accord par un simplexe. Le complexe de droite correspond au premier degré $I_{Do}$ de la tonalité de Do major ainsi que l'ensemble des notes et accords à deux sons qui y sont inclus.

**Processus d'auto-assemblage.** Nous proposons de construire la représentation simpliciale d'un ensemble d'accords à l'aide d'un processus incrémental d'auto-assemblage [14]. Ce processus est basé sur l'identification des bords des simplexes. Cette opération topologique est valable pour toute dimension. Par exemple, l'assemblage des accords $I_{Do}$ et $III_{Do}$ dans la construction du ruban de Moebius nécessite l'identification de trois éléments : les sommets $mi$ et $sol$ ainsi que l'arc $\{mi, sol\}$.

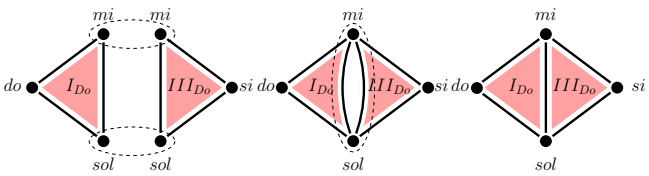



**Figure 6**. Identification des bords.

Une manière simple de programmer cette identification consiste à appliquer récursivement jusqu'à un point fixe la fusion de cellules topologiques ayant exactement les mêmes faces. Dans notre exemple, cela consiste à faire fusionner les sommets $mi$ et $sol$ dans $I_{Do}$ et $III_{Do}$, puis les arcs $\{mi, sol\}$ comme illustré Figure 6. La modification topologique correspondante peut être exprimée en syntaxe MGS de cette manière :

```
transformation identification = {
  s1 s2 / (s1==s2 & faces(s1)==faces(s2) )
  =>
  let c = new_cell ( dim(s1) )
                   ( faces(s1) )
                   (union ( cofaces(s1) )
                          ( cofaces(s2) ))
  in  s1*c
}
```

La règle spécifie que deux éléments `s1` et `s2`, décorés par la même valeur et possédant les même faces dans leurs bords, fusionnent en un nouvel élément `c` possédant comme cofaces l'union des cofaces de `s1` et `s2` décoré par la même valeur que `s1` (qui est aussi la décoration de `s2`). Cette décoration se fait grâce à l'expression `s1*c`.

Dans la Figure 6, la transformation `identification` est appelée deux fois. A la première application, (du complexe de gauche à celui du milieu), les sommets sont identifiés. Les deux opérations topologiques sont effectuées en parallèle. A la seconde application (du complexe du milieu à celui de droite), les deux arcs reliant $mi$ et $sol$ partageant le même bord fusionnent. L'arc résultant possède les 2-simplexes $I_{Do}$ et $III_{Do}$ comme cofaces, correspondant à la réunion des cofaces des arcs ayant fusionné. Enfin (à droite), plus aucune opération de fusion ne peut avoir lieu et le point fixe est atteint.

Comme attendu, l'itération jusqu'au point fixe de l'application `identification` sur les accords triadiques $I_{Do}$, ...,$VII_{Do}$ aboutit à la construction du ruban de Moebius illustré Figure 4.

**Application aux accords de quatre notes.** Il est courant dans certains styles de musique comme le jazz, de privilégier les accords à 4 sons pour caractériser les degrés d'une tonalité. Pour la gamme de Do Majeur utilisée ci dessus, le premier degré $I_{Do}$ est alors associé aux notes $\{do, mi, sol, si\}$. La représentation géométrique d'un accord à 4 sons est un tétraèdre (voir Figure 5). La conception du complexe simplicial associé à l'assemblage des sept accords est quasiment impossible à la main. Grâce au caractère générique de la transformation `identification`, le complexe simplicial résultant de l'auto-assemblage est automatiquement construit par MGS. Après avoir calculé les caractéristiques d'Euler et les coefficients d'orientabilité du complexe, sa topologie s'avère être celle d'un *toroïd*. Contrairement au ruban de Moebius non orientable dont le bord révèle le cercle des quintes (1 dimension), la topologie du tore est orientable et révèle deux dimensions.

### 3.3. Construction d'un espace pour les classes de hauteur et d'intervalles

On propose ici une représentation spatiale arbitraire pour la manipulation des classes de hauteurs et d'inter-

valles. Des travaux sur la recherche de séries ayant été à l'origine de la conception de cet espace, on s'attache ici à la manipulation des classes de hauteurs plutôt qu'à celle des notes elles-mêmes. Cela signifie qu'on ne distingue pas les notes à une octave près, d'où l'utilisation du terme *classe de hauteur*. Après avoir décrit cet espace, nous proposerons l'utilisation qu'il est possible d'en faire pour une énumération optimisée des séries tous-intervale (STI).

**Construction de l'espace.** On propose dans un premier temps de représenter une classe de hauteur par un sommet (0-cellule) et un intervalle reliant deux classes de hauteur par un arc (1-cellule) reliant les deux sommets concernés. A ce stade, la complexe cellulaire résultant peut être représenté par un graphe complet (Figure 7) car il existe bien un intervalle associé à toute paire de classes de hauteur.

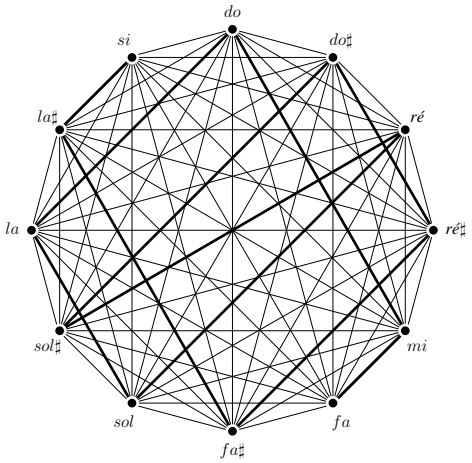



**Figure 7**. Représentation spatiale de la série tous-intervalle de Alban Berg (en gras dans le graph complet).

Attribuer à chacun de ces deux types d'objet une dimension voisine permet grâce à la notion de face définie plus haut de faire apparaitre des relations de voisinage intuitives entre deux éléments (en l'occurrence un sommet et un arc). Prolongeant cette démarche, on propose de représenter une classe d'intervalle par une surface (2-cellule) dont les faces sont constituées par les arcs associés aux intervalles précédemment évoqués, que l'on peut aussi désigner comme instances de ces classes d'intervalle.

Les classes d'intervalle étant au nombre de onze, il s'agit d'introduire onze surfaces au sein du complexe de la Figure 7. Le complexe résultant n'est pas visualisable dans sa globalité. Il est toutefois possible d'en avoir des représentations locales en se restreignant par exemple à une seule classe d'intervalle et à ses bords (Figure 8).

Cet espace peut facilement être construit suivant la méthode d'auto-assemblage précédemment décrite.

**Recherche des STI.** L'énumération et la classification des STI est un problème connu en informatique musicale. Une STI est une série dodécaphonique comprenant les onze intervalles réduits à $\mathbb{Z}_{12}$. Ces séries comportent de nombreuse propriétés [15]. L'une des utilisations les plus con-

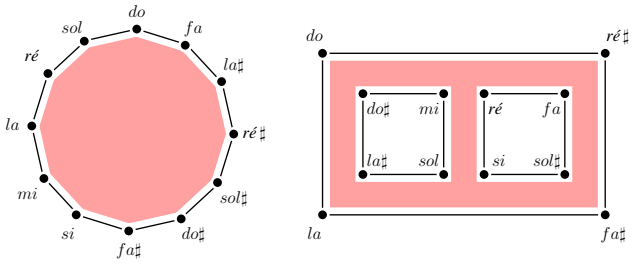
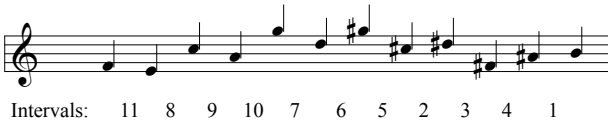



**Figure 8**. Représentation spatiale des classes d'intervalle associées à la Quarte (à gauche) et à la Tierce mineure (à droite)

nues de ce type particulier de série est probablement celle de Alban Berg dans se *Suite Lyrique* (1928) :




L'une des premières énumérations des 3856 séries fût effectuée par André Riotte [16]. Depuis, de nombreuses méthodes ont été proposées pour leur énumération et leur classification. Nous proposons ici une nouvelle approche pour cette énumération, fondée sur l'utilisation de l'espace topologique décrit au paragraphe précédent. Dans cet espace, la spécification d'une STI est entièrement structurelle. Elle consiste à rechercher un chemin, composé de 0, 1 et 2-cellules, à la fois 0-Hamiltonien et 2-Hamiltonien [1]. Nous proposons le motif filtrant suivant pour spécifier les STI.

- les sommets associés aux classes de hauteur sont désignés par les variables `n0, n1, ..., n11`
- les arcs associés aux intervalles entre classes de hauteur sont désignés par les variables `i1, i2, ..., i11`
- les surfaces associées aux classes d'intervalle sont désignées par les variables `I1, ..., I11`

Le motif est le suivant :

```
n0 < i1  < I1  > i1  > n1
   < i2  < I2  > i2  > n2
   ...
   < i11 < I11 > i11 > n11
```

La figure 9 illustre l'instanciation de ce chemin sur les cinq premiers éléments de la STI de Alban Berg. Il est facile de voir sur cette figure que la contrainte spatiale consiste à déterminer une 2-cellule supplémentaire respectant le motif filtrant. Plus précisément, deux notes consécutive `n`$p'$ et `n`$p$ avec $p' = (p-1)$ doivent être $(0,1)$-voisins via `i`$p$ et $(0,2)$-voisins via `I`$p$ tel que les cellules `I`$p$ et `i`$p$ soient incidentes.

Cette démarche pour spécifier les STI a été approfondi dans le cadre d'une classification topologique [17]. D'autre

1 . Dans un graphe, un chemin visitant chaque sommet exactement une fois est appelé *Eulerien*. Cependant il n'existe pas de terme général pour un chemin dans un complexe visitant chaque $p$-cellules exactement une fois. Nous appelons un tel chemin un chemin $p$-Hamiltonien.

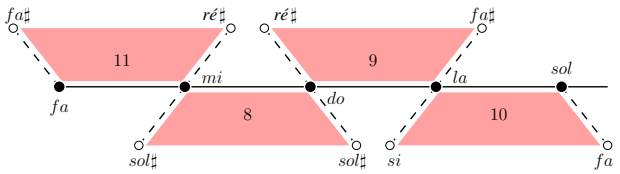



**Figure 9**. Représentation spatiale des cinq premières notes de la STI d'Alban Berg.

part, nous travaillons actuellement sur un espace d'accords construit suivant une démarche similaire et révélant certaines transformations entre accords [18].

## 4. CONCLUSION

Nous avons montré dans cet article liminaire, comment trois types d'objets musicaux, les réseaux de notes, les séquences d'accords et les STI, pouvaient être manipulés à la lumière du calcul spatial. Ces travaux préliminaires semblent montrer que la construction d'espaces abstraits constitue un outil performant pour l'étude de ce type d'objets et nous pensons que cette démarche peut s'avérer utile pour le développement de nouveaux outils pour assister musicologues et compositeurs dans leur travail. Les problèmes exposés ici ont déjà été abordé, mais notre démarche, fondée sur des notions topologiques supportées par le langage de programmation MGS, est générique. Il est par exemple possible, prolongeant la méthode présentée en section 3 d'étudier des accords de $n$ notes avec $n > 4$ malgré le fait qu'ils soient associés à des objets non visualisables de par leur dimension.

Nous travaillons actuellement sur l'énumération de chemins hamiltoniens dans différents espaces d'accords d'un point de vue spatial. Dans la prolongation de notre démarche pour l'énumération des STI, nous menons une réflexion sur la répartition spatiale de ces STI dans le cadre d'une classification topologique. Enfin, nous réfléchissons au développement d'outils qui pourraient offrir aux utilisateurs de ces espaces divers possibilités d'interaction.

**Remerciements.** Les auteurs aimeraient remercier l'équipe Représentations Musicales de l'IRCAM, en particulier Moreno Andreatta et Jean-Louis Giavitto, ainsi que Mikhail Malt et Jean-Marc Chouvel.

## 5. REFERENCES

[1] D. Tymoczko, "The geometry of musical chords," *Science*, vol. 313, no. 5783, p. 72, 2006.

[2] G. Mazzola *et al.*, *The topos of music : geometric logic of concepts, theory, and performance*. Birkhäuser, 2002.

[3] J.-L. Giavitto and O. Michel, "MGS : a rule-based programming language for complex objects and collections," in *Electronic Notes in Theoretical Computer Science*, M. van den Brand and R. Verma, Eds., vol. 59. Elsevier Science Publishers, 2001.

[4] J.-L. Giavitto, "Topological collections, transformations and their application to the modeling and the simulation of dynamical systems," in *Rewriting Technics and Applications (RTA'03)*, ser. LNCS, vol. LNCS 2706. Valencia : Springer, Jun. 2003, pp. 208 – 233.

[5] J. Munkres, *Elements of Algebraic Topology*. Addison-Wesley, 1984.

[6] J.-L. Giavitto and O. Michel, "Data structure as topological spaces," in *Proceedings of the 3nd International Conference on Unconventional Models of Computation UMC02*, vol. 2509, Himeji, Japan, Oct. 2002, pp. 137–150, lecture Notes in Computer Science.

[7] ——, "Declarative definition of group indexed data structures and approximation of their domains." in *Proceedings of the 3nd International ACM SIGPLAN Conference on Principles and Practice of Declarative Programming (PPDP-01)*. ACM Press, Sep. 2001.

[8] A. Spicher, O. Michel, and J.-L. Giavitto, "Declarative mesh subdivision using topological rewriting in mgs," in *Int. Conf. on Graph Transformations (ICGT) 2010*, ser. LNCS, vol. 6372, Sep. 2010, pp. 298–313.

[9] R. Cohn, "Neo-riemannian operations, parsimonious trichords, and their "tonnetz" representations," *Journal of Music Theory*, vol. 41, no. 1, pp. 1–66, 1997.

[10] J.-M. Chouvel, "Analyser l'harmonie - aux frontières de la tonalité."

[11] ——, "Au-delà du système tonal."

[12] D. Lewin, *Generalized musical Intervals and Transformations*. Yale University Press, 1987.

[13] A. Schoenberg, *Harmonielehre*, U. Edition, Ed., 1911.

[14] J.-L. Giavitto and A. Spicher, *Systems Self-Assembly : multidisciplinary snapshots*. Elsevier, 2008, ch. Simulation of self-assembly processes using abstract reduction systems, pp. 199–223, doi :10.1016/S1571-0831(07)00009-3.

[15] R. Morris and D. Starr, "The structure of all-interval series," *Journal of Music Theory*, vol. 18, no. 2, pp. pp. 364–389, 1974. [Online]. Available : http ://www.jstor.org/stable/843642

[16] A. Riotte and M. Mesnage, *Formalismes et modèles musicaux*. Delatour, 2006.

[17] L. Bigo, J.-L. Giavitto, and A. Spicher, "Building spaces for musical objects," in *Mathematics and Computation in Music*, ser. Communications in Computer and Information Science. Springer, 2011.

[18] G. Albini and S. Antonini, "Hamiltonian cycles in the topological dual of the tonnetz," in *Mathematics and Computation in Music*, ser. Communications in Computer and Information Science, E. Chew, A. Childs, and C.-H. Chuan, Eds. Springer Berlin Heidelberg, 2009, vol. 38, pp. 1–10.