Propriétés des intervalles temporels circulaires et application à l'analyse harmonique

François Pachet, Jean Carrive

LAFORIA-IBP, Université Paris 6, 4 Place Jussieu, 75252 Paris Cedex,
pachet@laforia.ibp.fr

Résumé : Le contexte de ce travail est la construction de systèmes d'analyse harmonique automatique de séquences d'accords de Jazz. Une des particularités de ces séquences d'accords est leur circularité temporelle. Cette circularité joue un rôle important pour l'analyse, mais les modèles temporels classiques, en particulier le formalisme de Allen, ne la prennent pas en compte. Nous étudions ici les propriétés des modèles temporels circulaires, vus comme des extensions du formalisme de Allen, avec une application à la construction d'un système d'analyse.

I. Introduction

Nous développons depuis plusieurs années une vision de la représentation de connaissances musicales fondée sur le développement de divers systèmes à base de connaissances résolvant un certain nombre de "tâches musicales" clairement définies. Ces systèmes ont comme but d'identifier les types de connaissances sous-jacent aux tâches musicales. La "mise en machine" de ces tâches nous sert à valider un certain nombre de techniques de représentation, la musique jouant un rôle de générateur de problèmes idéaux, par la nature des connaissances mise en jeu, leur caractère consensuel, et la complexité des problèmes qu'elle suscite. Comme toujours, dans ce type de modélisation, nous observons alors un effet de retour : en explicitant ces connaissances, nous apprenons aussi beaucoup sur la nature de ces processus, et les relations complexes qui existent entre créativité musicale, et résolution de problèmes. Toutefois, nous nous intéressons dans le cadre de cet article à la musique tonale comme *prétexte* à la mise en oeuvre de techniques de modélisation sophistiquée, et non pas comme un but en soi.

I.1. Le système MusES

Cette entreprise de modélisation multiple s'organise autour de la constitution d'un noyau de connaissances de base en musique tonale. Ce noyau, appelé MusES, contient une représentation considérée comme raisonnablement déclarative, des concepts de base de la musique tonale, et de leur propriétés principales. Le système MusES utilise les mécanismes de la programmation par objets (héritage, instanciation, polymorphisme, etc.). Il contient des descriptions des concepts tels que les notes, les pitch-classes, les gammes, les intervalles, les accords, les mélodies, etc. Le système contient environ 80 classes et 1500 méthodes. Chaque méthode de MusES est considérée comme une *primitive* du langage musical tonal. Une description complète du système peut-être trouvée dans [Pachet 94; 94b].

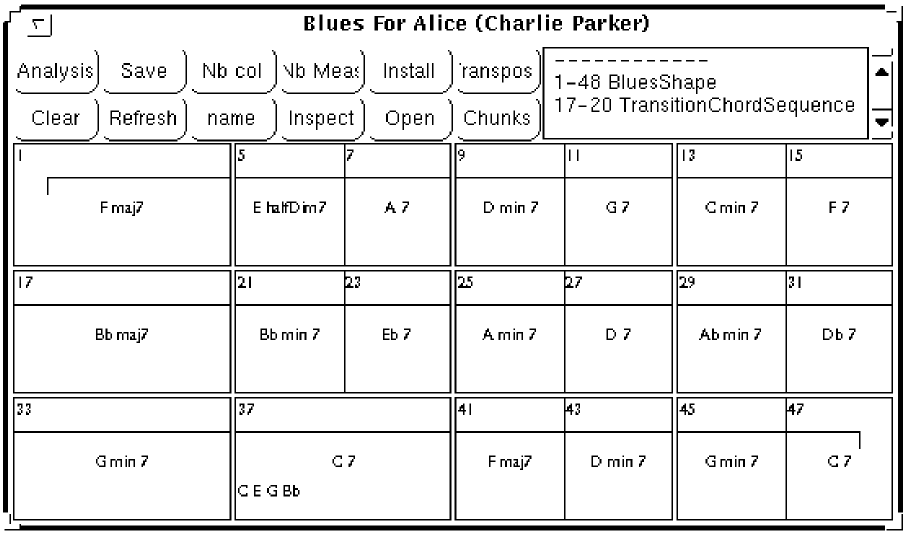
Chaque système modélisant une tâche particulière est alors construit comme une extension du système MusES de base. Les systèmes réalisés jusqu'à présent de cette manière résolvent les problèmes suivants :

- Harmonisation automatique, en utilisant des techniques de satisfaction de contraintes [Pachet & Roy 95], [Pachet & Roy 95b],
- Analyse harmonique [Pachet 94c], [Mouton & Pachet 95],
- Simulation d'improvisation [Ramalho & Pachet 94], [Ramalho & Ganascia 94],
- Interface, basé sur le concept d'*espace harmonique* de Simon Holland [Holland 89].

Ce papier décrit un développement théorique et appliqué concernant le système d'analyse de grilles de jazz.

I.2. Analyse automatique de grilles de jazz

Le problème d'analyse que nous cherchons à résoudre ici est un problème particulier, mettant en jeu des connaissances en musique tonale spécifique au domaine du Jazz. Les grilles de Jazz sont des séquences d'accords, considérées ici comme indépendantes d'une mélodie donnée. Chaque accord est décrit par son nom, sans indication de chiffrage (au contraire des basses chiffrées), De nombreuses grilles de Jazz dites standards sont ainsi disponibles sous la forme d'ouvrages comme le Real Book [Real 81] ou le Fake Book [Fake 83]. Un exemple typique de grille modélisée en MusES est représenté Figure 1.



Blues For Alice (Charlie Parker)

1-48 BluesShape
17-20 TransitionChordSequence

1		5	7	9	11	13	15
F maj7		E halfDim7	A 7	D min 7	G 7	C min 7	F 7
17		21	23	25	27	29	31
Bb maj7		Bb min 7	Eb 7	A min 7	D 7	Ab min 7	Db 7
33		37		41	43	45	47
G min 7		C 7 C E G Bb		F maj7	D min 7	G min 7	C 7

Figure 1. La grille *Blues for Alice* (Charlie Parker).

L'analyse d'une grille d'accords consiste à identifier, pour une grille donnée, la tonalité sous-jacente à chacun de ses accords. Cette tonalité est déterminée à partir de l'accord lui-même, ses accords voisins, et les diverses formes musicales que l'on peut reconnaître, dont l'accord fait partie.

Il est important de noter ici que plusieurs analyses sont en général possibles pour un accord donné, en fonction du point de vue choisi. Par exemple, un accord peut être analysé dans une tonalité donnée (C majeur), mais faire partie d'une partie qui elle est globalement en F; cette partie elle-même étant une partie modulante d'un morceau qui est globalement en Bb.

Dans la suite d'accords illustrée par la figure 1, l'accord [Ab min 7] peut être analysé dans trois tonalités différentes selon la forme considérée. C'est seulement une fois qu'un contexte plus large est connu que l'on peut affirmer que le premier deux-cinq qui était analysé en sol bémol est en réalité en deux-cinq substitué en fa, et ainsi de suite. On s'aperçoit donc qu'un accord ne peut s'analyser que dans un contexte, et que la notion même de tonalité n'a de sens que si l'on tient compte de ce contexte. La figure 2 montre un exemple partiel d'analyse à plusieurs niveaux hiérarchiques.

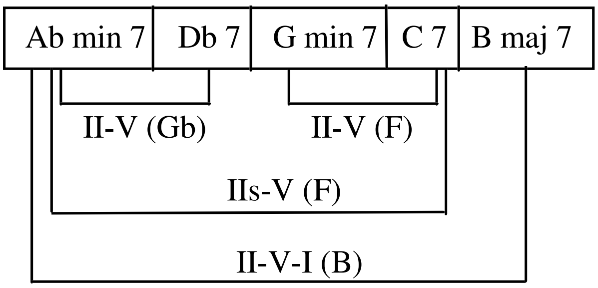


Figure 2. Structure hiérarchique d'une analyse.

Ainsi, l'analyse que l'on souhaite obtenir est un arbre dont les nœuds représentent des formes d'autant plus abstraites que leur profondeur est faible. La racine de cet arbre représente la grille dans sa totalité, et décrit, dans les cas les plus favorables, une forme typique telle qu'un Blues, une forme AABA, etc. Les formes les plus « profondes », celles qui sont le plus près des accords de la grille, décrivent des types de séquences d'accords standards du jazz, comme les *deux-cinq* (deuxième degré suivi du cinquième degré), les *résolutions* (ou *cadences*), les *substitutions*, etc. Les formes intermédiaires décrivent les structures intermédiaires harmonique de la grille (passages modulants, marches, etc.). A chacune des formes reconnues par les procédures d'analyse est associée une tonalité. Ainsi, un même accord se trouvera-t-il analysable dans plusieurs tonalités différentes, selon la distance avec laquelle on examine cet accord dans cet arbre d'analyse. On trouvera en annexe l'analyse complète de la grille de la Figure 1.

I.2.1. Le système d'analyse

Notre travail consiste à modéliser la tâche d'analyse telle qu'elle est résolue par les musiciens professionnels. Si de nombreux ouvrages ont été écrits sur les propriétés harmoniques diverses des accords de Jazz et des grilles (voir par exemple l'ouvrage de référence de [Siron 90]), aucun, à notre connaissance, ne propose de modèle opérationnel rendant compte de l'*activité même* de l'analyse. La plupart des travaux en analyse automatique reposent sur des théories du processus d'analyse construites de manière ad hoc. En particulier, de nombreux systèmes se sont inspirés des travaux sur les grammaires génératives, dans la lignée de [Lerdahl & Jackendoff 83]. Dans le domaine du Jazz, le modèle proposé par [Steedman 84] est le plus précis. Il décrit sous forme de règles de grammaire un modèle des grilles de Blues. Ce modèle, s'il rend bien compte d'un large corpus des grilles de Blues, est cependant non opérationnel, car il utilise un formalisme non implémentable (les grammaires dépendantes du contexte). D'autres travaux sur les grammaires ont été proposés, comme le système de Winograd pour l'analyse de Chorals de Bach [Winograd 93], mais les mécanismes mis en jeu ne sont pas directement applicables à notre cas, en particulier à cause de la nature différente des résolutions harmoniques en jazz et en musique baroque. De manière plus pragmatique, des travaux ont tenté d'appliquer des techniques diverses pour résoudre le problème : purement algorithmiques [Ulrich 77], par systèmes experts [Maxwell 92], [Winold & Bein 83], par systèmes de propagation de contraintes [Steels 79].

Notre approche consiste à premièrement élaborer une théorie du processus d'analyse, dégageant ses caractéristiques principales. Dans un deuxième temps, nous appliquons un certain nombre de techniques pour modéliser ce processus. Nous allons brièvement décrire ce processus.

I.3. Caractéristiques du processus d'analyse

Le processus d'analyse tel que nous l'envisageons est basé sur principes : 1) les principes de base, héritiers directs de la théorie harmonique classique, 2) les perturbations, qui expliquent un certain nombre d'idiosyncrasies du domaine, et 3) le caractère récursif de l'analyse. Examinons chacun de ces points plus en détail :

A) Principes de base

Les principes de base sont les deux suivants :

1) Principe de légalité

Ce principe dit que chaque accord, *pris isolément*, indépendamment de tout contexte, peut être analysé dans un nombre fixe de tonalités. Une tonalité est représentable par la donnée d'un couple {degré, gamme}. Ainsi, un accord de Do majeur hors contexte peut il être analysé comme : {I degré de C majeur}, {IV degré de G majeur}, {V degré de F majeur}, etc. Notons que le calcul de l'ensemble des tonalités "légales" est entièrement déterministe. Il est représenté comme une des primitives du systèmes MusES.

2) Principe de minimisation

Pris cette fois en contexte, un accord s'analysera bien sûr en fonction de sa relation avec les accords voisins. L'idée de base ici est que le choix de la "bonne" tonalité sera celui qui minimisera les modulations, i.e. qui sera commun au plus grand nombre possibles d'accords adjacents. Par exemple, la séquence (C / F / E min / A min) ne possède qu'une tonalité commune à tous ses accords : C Majeur. C'est (très probablement) la bonne.

B) Perturbations

La simple théorie précédente se complique par l'introduction de phénomènes qui échappent à une formalisation rigoureuse, mais qui sont néanmoins essentiels pour rendre compte de la nature du processus : les perturbations. Ces perturbations, constitutives des corpus de Jazz considérés se présentent sous la forme de substitutions et de constructions idiomatiques.

Premièrement, certains accords peuvent être substitués par d'autres. Les substituts, hélas, violent en général le principe de légalité précédent. Par exemple, un accord de septième qui résout (C 7 / F) peut être substitué par un accord de septième dont la fondamentale est le triton de la fondamentale (F# 7 / F). Le substitut devra être analysé comme étant en Fa majeur, bien que Fa majeur ne fasse pas partie de la liste des tonalités légales de Fa# 7.

Deuxièmement, un certain nombre d'idiotismes viennent étendre ces perturbations. Ces idiotismes sont caractérisés par le fait qu'ils portent en eux leur propre sémantique, en échappant aussi au principe de légalité. Ainsi, on trouve fréquemment dans les grilles de Jazz des "deux-cinq" non orthodoxes (D 7 - Db 7), ou des anatoles particulièrement illégales comme (C majeur / Eb maj 7 / A b maj 7 / Db maj 7). Dans ce dernier cas, l'analyse de la forme sera C majeur, bien que C majeur ne fait pas partie de l'ensemble légal de Db majeur. En d'autres termes, Db maj 7 in abstracto ne peut être analysé en C majeur, mais peut l'être dans le contexte bien précis d'une forme idiomatique.

Ces formes musicales ont en général une analyse directe (C majeur pour ces exemples), qui découlent de leur nature idiomatique : même si un raisonnement pourrait être fait a posteriori pour justifier ces distorsions, nous les considérons comme première ici, dans le cadre d'un système d'analyse (pourquoi réinventer systématiquement ces formes, si elle sont si fréquentes).

C) Récursivité

Enfin, un troisième point essentiel pour comprendre l'analyse de ce type de corpus est la nature récursive de la notion de forme musicale : chaque forme musicale reconnue peut jouer un rôle d'atome pour une forme plus générale, et ce, jusqu'à arriver à la forme la plus générale qui soit ; la grille elle-même.

Ce processus récursif explique la nature hiérarchique de l'analyse. Par exemple, des accords de septième résolvant peuvent ainsi être considérés, une fois reconnus comme tel, comme formant un tout avec leur résolution. Si la résolution est elle-même un accord de septième, on peut alors expliquer les enchaînements (marches harmoniques) typiques en Jazz comme : "A7 / D7 / G7 / C", qui peut ainsi être entièrement analysée en C major, par l'application récursive de la même règle (voir Figure 3).

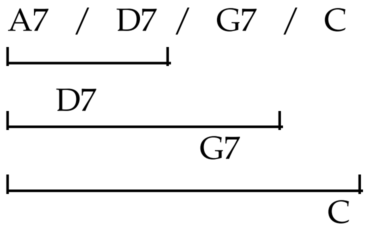


Figure 3. La nature récursive de l'analyse d'un groupe d'accords.

Cette récursivité joue à tous les niveaux. Un Blues est ainsi identifiable par la succession de trois formes couvrant la grille entière, et telles que la tonalité de la forme du milieu soit celle de la première et de la dernière, transposée d'une quarte. Ce principe peut être représenté par une règle de grammaire du type :

Blues ::== <Forme en X> < Forme en X + 4> < Forme en X>

Les autres structures typiques des grilles de Jazz peuvent être décrites de manière similaires (AABA, ABAB, etc.).

I.4. Représentation du processus d'analyse

Une fois le processus décrit, notre problème est de construire un système à base de connaissance qui réalise/modélise le processus de manière raisonnable. La construction d'un tel système et la validation de ses résultats constituent pour nous un élément tangible de discussion sur la nature du processus d'analyse.

Le processus d'analyse se déroule en deux phases principales : une phase dite de *reconnaissance*, et une phase dite de *remplissage*. La phase de reconnaissance consiste à identifier des formes relativement simples et bien connues, telles que les deux-cinq et les résolvantes. La nature récursive du processus est représentée par le caractère récursif de l'application des règles : chaque forme reconnue peut elle-même participer à la reconnaissance d'autres formes plus générales.

La seconde phase est une phase de *remplissage*, qui agglomère les formes reconnues suivant des règles typiques d'agrégation. Il s'agira par exemple, dans le cas d'un accord isolé précédant une forme reconnue, d'intégrer cet accord à cette forme, pour peu que certaines conditions de cohérence harmonique soient respectées, en particulier concernant la tonalité de l'accord et de la forme agrégeante.

De manière pratique, nous nous inspirons du modèle de raisonnement par "agrégation et oubli", développé dans [Dojat 94]. Dans ce modèle, le raisonnement est représenté par une alternance de règles d'agrégation (construction des formes à partir de formes de plus bas niveau) et d'oubli (destruction de formes considérées comme obsolètes à un moment du raisonnement). On pourra se reporter à [Dojat & Sayettat 94] pour les justifications conceptuelles et théoriques de ces deux mécanismes de base.

I.5. De la nécessité d'un modèle temporel circulaire

Dans une version naïve de notre système, une grille de Jazz est modélisée par une séquence linéaire d'accords. La majorité des règles du système sont alors exprimables par des comparaisons d'intervalles sur une échelle linéaire.

Cependant, une des caractéristiques d'une grille de jazz qui est souvent considérée comme secondaire est son aspect cyclique : non seulement la grille est répétée plusieurs fois, mais encore la dernière partie de la grille est le plus souvent une *préparation* à la reprise de la grille à son début. Ainsi il est fréquent qu'une grille se termine par un accord de septième qui ne résout pas, à interpréter comme résolvant sur le début de la grille.

Certaines formes à reconnaître peuvent ainsi se situer à « l'intérieur » de la grille, ou bien « à cheval » entre la fin et le début. Il faut noter que ces circularités ne sont pas anecdotiques, mais fondamentales, pour les musiciens, et surtout pour un système informatique. Dans l'exemple de la grille "Blues for Alice", on remarquera que la première partie de la grille doit être analysée en Fa majeur (le premier accord est un Fa majeur). Ensuite il y a une modulation en Bb majeur. La fin de la grille résout de nouveau en Fa majeur (G m7 / C 7). Le risque ici est que, si l'on ne prend pas garde à considérer la fin de la grille, le F majeur initial soit "mangé" par la sous-séquence en Bb : en effet, Fa majeur peut très bien être analysé en Bb (Bb fait partie de l'ensemble légal des tonalités de Fa), et l'analyse de Fa comme V de Bb minimise les modulations, puisqu'ainsi les mesures 1 à 20 peuvent être analysées en Bb.

Intuitivement il y a bien sûr ici une erreur : il paraît difficile de considérer l'accord de Fa initial comme un cinquième degré. Plus précisément, c'est bien la fin de la grille qui nous indique de manière sûre que ce Fa initial est bien un premier degré : c'est la suite logique des dernières mesures!

Pour rendre compte de ce phénomène, un modèle temporel particulier doit être introduit. En effet, nous allons montrer que les propriétés des intervalles linéaires ne sont pas immédiatement transposables au cas d'un référentiel circulaire.

Le reste de ce papier décrit les modèles temporels circulaires que nous avons développés, comme des extensions du modèles de Allen pour les intervalles linéaires. Leur application au système d'analyse est alors évidente : c'est ce modèle qui nous permettra d'en dire plus à la machine", en particulier de résoudre le problème des grilles bouclantes comme "Blues for Alice".

II. Modèles linéaire et circulaire

Dans une représentation linéaire, James F. Allen [Allen 84] dénombre et nomme treize relations distinctes possibles entre deux intervalles de temps (Figure 4).

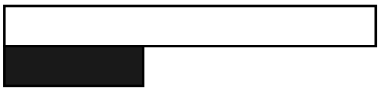
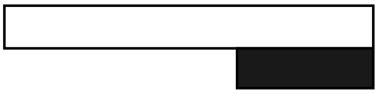
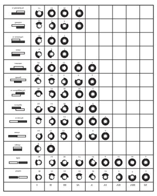
Relation	Relation réciproque	Exemple
X before Y	*Y after X*	
X equal Y		
X meets Y	*Y is met by X*	
X overlaps Y	*Y is overlapped by*	
X during Y	*Y contains X*	
X starts Y	*Y is started by X*	
X finishes Y	*Y is finished by X*	

Figure 4. Les treize relations de Allen.

Passer de cette représentation linéaire à une représentation circulaire revient à faire boucler la fin d'un des intervalles sur le début de l'autre. Nous allons détailler ici ce processus et le décliner sous la forme de trois modèles différents : circularité pure, avec origine, et sans phase.

II.1. Circularité Pure

Le premier type de circularité consiste simplement à faire boucler les intervalles des relations autour de l'axe temporel. Pour chaque relation linéaire de Allen, on peut construire plusieurs relations circulaires, en étirant la fin d'un des intervalles, et en le faisant parcourir les différentes positions topologiques possibles. Le même processus est alors répété avec le premier intervalle, pour couvrir tous les cas possibles. Ce bouclage donne lieu à un nombre variable de relations circulaires, en fonction de la nature de l'intervalle de Allen de départ. Ainsi, la relation linéaire *before*, donne-t-elle lieu à 9 relations circulaires.



La Figure 5 montre l'ensemble des relations circulaires obtenues par ce processus, appliqué à chacune des relations de Allen. Le numéro de colonne indique le nombre d'étirements réalisés. Intuitivement, plus ce nombre est élevé, plus la relation est "repliée".

Figure 5. Relations entre le modèle linéaire et circulaire pure. Chaque colonne indique une manière de boucler la relation linéaire de départ. les nombres 1 à 26 désignent les relations de la Figure 6.

Notons que cette procédure nous permet d'énumérer tous les intervalles circulaires possibles, mais introduit des redondances. Ainsi, la relation circulaire 1 peut être obtenue à partir de trois relations linéaires : *equal* avec le bouclage 2, *starts* avec le bouclage 3, et *is started by* avec le bouclage 3. Le nombre total de relations circulaires différentes est alors de 26 (2*13!). Ceux-ci sont représentés en Figure 6.

Figure 6. Les 26 relations circulaires pures.

II.1.1. Nommage des relations circulaires pures

Le processus de création des intervalles circulaires conduit à un nommage simple de ces relations (on verra que ce n'est plus le cas pour les autres types de circularité). La première méthode de bouclage donne lieu à 13 relations circulaires, qui sont intuitivement similaires à leurs correspondant de Allen. Par exemple, les relations 8 et 9 de la Figure 6 sont les équivalents respectifs des relations *equals* et *start* de Allen). Ces relations gardent le nom de Allen correspondant. Nous n'avons que 11 relations de ce type et non pas 13 car les relations de Allen *after* et *before* donnent lieu à la même relation circulaire et seraient donc indifférentiables.

Cinq autres relations circulaires peuvent être vues comme satisfaisant simultanément 2 relations de Allen. Nous les nommons par la concaténation des noms de Allen. Par exemple, la relation n. 11 de la figure 6 est obtenue par la relation de Allen *meet* ET *is met by*; son nom est donc "meet-is met by".

Les 10 relations restantes ont toutes la propriété suivante : un des deux intervalles couvre l'étendue du référentiel circulaire. Nous les nommons en ajoutant le préfixe "period" à la relation de Allen obtenue en raccourcissant l'intervalle couvrant (ou les deux le cas échéant) d'une quantité infinitésimale, sans changer la structure topologique. Ainsi, la relation 1 de la figure 6 se nomme *equals-period*.

Cette convention de nommage est bien sûr arbitraire, mais elle a l'avantage d'être bijective : à chaque relation circulaire correspond un nom et un seul. De plus, le nom et la convention de nommage permettent de reconstituer les intervalles à partir de leurs noms. La table 1 résume les noms donnés aux relations par ce procédé.

1	*equals-period*	10	*is started by*	19	*during*
2	*starts-period*	11	*meets-is met by*	20	*is overlapped by*
3	*finishes-period*	12	*is finished by-period*	21	*meets*
4	*overlaps-period-is overlapped by-period*	13	*is met by-overlaps*	22	*is finished by*
5	*overlaps-is overlapped by-period*	14	*is met by*	23	*contains-period*
6	*during-period*	15	*meets-is overlapped*	24	*overlaps*
7	*is started by-period*	16	*is overlapped by-overlaps-period*	25	*contains*
8	*equals*	17	*finishes*	26	*before-after*
9	*starts*	18	*overlaps-is overlapped*		

Table 1 : Les noms des 26 relations circulaires pures.

II.2. Le problème de l'origine

Dans la représentation linéaire comme dans la circulaire pure, on peut ajouter une *origine* au repère temporel. D'un point de vue intuitif, cette origine joue un rôle important. Par exemple, les grilles de Jazz, tout en étant intrinsèquement circulaires, ont une origine bien précise : on ne les commence pas n'importe où.

Le nombre de relations binaires entre intervalles est alors considérablement augmenté, l'origine pouvant se situer à n'importe quel configuration topologique particulière de la relation, les moments particuliers étant les débuts et fins d'intervalles, ainsi que les sous-intervalles compris entre une fin (ou un début) et la fin (ou le début) suivant. Sur l'exemple de la figure 7 une origine peut être introduite dans 7 configurations différentes.

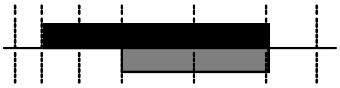


Figure 7 : différentes manières d'introduire une origine dans une relation linéaire.

Un calcul exhaustif montre que l'on obtient alors 101 relations linéaires avec origine, et 150 relations circulaires pures avec origine (Cf. Figure 8). Un modèle linéaire avec origine est a priori de peu d'intérêt : l'origine n'est qu'un moment particulier, et l'introduction d'un moment singulier dans un modèle général est maladroite. Dans le cas circulaire il en est autrement, puisque l'origine se répète un nombre infini de fois dans le référentiel.

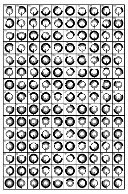


Figure 8. Les 150 relations circulaires avec origine (représentée par une ligne verticale).

II.3. Le problème de la phase

On peut être amené à distinguer dans certains cas les intervalles de temps couvrant une période entière, car dans ce cas la notion de début et de fin de l'intervalle n'a plus nécessairement de sens, tout au moins eu égard aux relations temporelles existant avec un autre intervalle de temps. En effet, un intervalle de temps inférieur à une période est nécessairement inclus dans un intervalle de temps couvrant toute une période. Certaines des relations binaires obtenues précédemment sont alors identiques. Cela revient à ne pas prendre en compte l'information de *phase*. Dans ce cas, nous obtenons un nombre total de relations de 19 pour le modèle circulaire pure, et de 113 pour le modèle circulaire avec origine. La notion de phase n'a bien sûr pas de sens pour le modèle linéaire.

La figure 9 montre les 19 relations circulaires ainsi obtenues pour le modèle circulaire pur.

Figure 9. Les 19 relations circulaires sans origine et sans phase.

La table suivante donne un résumé du nombre total de relations pour chacun des modèles étudiés.

Modèle circulaire	Sans origine	Avec origine
avec phase	26	150
sans phase	19	113
Modèle linéaire	13	101

Notons que le nombre élevé de relations pour les modèles avec origine pose un problème linguistique délicat : comment nommer ces relations ? Dans le modèle de Allen standard, les noms ont un sens intuitif, que nous avons tenté de conserver pour les relations circulaires pures. Pour les modèles avec origine, le nommage des relations reste un problème ouvert.

Application

L'application de ce modèle à notre problème d'analyse est immédiate. Les règles d'agrégation que nous employons pour modéliser le raisonnement peuvent être formulées à l'aide de ces intervalles circulaires , évitant ainsi des redondances et des tests qui compliquaient le système initial. Par exemple, la règle d'agrégation qui permet de reconnaître une structure de 2-5-1 s'exprimait dans le modèle non circulaire par une règle du type :

```
regle DeuxCinqUn
SI ∃ des formes f1 f2 f3 telles que
          meets(f2, f1) , meets (f3, f2)
          tonique (f2) = transpose (tonique (f1), 5),
          tonique (f3 = transpose (tonique (f2, 5),
          estMineur (f1), estSeptieme(f2)
Alors f := Construire Forme(start(f1) , (end(f3)) de type DeuxCinqUn
```

Mais cette règle ne s'applique pas au cas d'un 2-5-1 dont la résolution est en début de grille. Ce cas particulier doit alors être traité par une règle spéciale. Le même phénomène s'applique aux autres structures à reconnaître.

Si l'on utilise le modèle circulaire simple, sans origine, les choses se simplifient : il suffit alors de changer la règle ci-dessus en remplaçant les prémisses de tests d'intervalles de Allen par des relations circulaires plus générales. En l'occurrence, la relation circulaire appropriée est celle dénommée aussi « meets ». La règle ne change donc pas, et il suffit de changer l'interprétation des relations circulaires.

Une forme plus rigoureuse de représentation de cette règle pourrait faire intervenir le modèle avec origine. Dans ce cas, on pourrait distinguer la relation entre le 2 et le 5 (un « meet » ne passant pas par l'origine) ; et celle entre le 5 et le 1 (un « meet » passant éventuellement par l'origine). Dans la pratique cependant, nous gardons le modèle sans origine qui est suffisamment précis pour représenter nos règles d'agrégation harmonique sans ambiguïtés.

III. Algèbre

Les propriétés que nous avons décrites ci-dessus concernent les relations binaires d'intervalles. Dans le contexte musical, il est fréquent d'avoir à manipuler des relations d'arité plus grande. Par exemple, la règle dite d'emprunt modal stipule qu'une forme située au milieu de deux autres formes de même tonalités, et qui peut s'analyser dans la tonalité mineure de cette tonalité, peut être considérée comme un "emprunt modal", et donc s'agréger avec ses formes voisines (voir exemple Figure 2).

On peut aisément étendre ces modèles pour prendre en compte des relations n-aires. On note I_p l'ensemble des intervalles circulaires définis pour une période *p*. *p* correspond alors dans une grille d'accords au nombre de temps dans la grille, et l'on cherche à définir les lois de compositions internes *union*, *intersection* et *complément* sur I_p. Cela nous amène à redéfinir la notion d'intervalle circulaire.

III.1. Différentes classes d'intervalles

Afin de pouvoir définir les lois de composition interne sur I_p, on est amené à généraliser la notion d'intervalle circulaire, et à définir les notions d'*intervalle vide*, et d'*intervalle non connexe*. L'intervalle vide se définit comme l'intersection de deux intervalles circulaires liés par la relation *meets*, *is met,* ou *before-after*. Un intervalle non connexe est un ensemble d'intervalles dont l'intersection est l'intervalle vide. L'union de deux intervalles circulaires liés par la relation *before-after* ou l'intersection de deux intervalles circulaires liés par la relation *overlaps-is overlapped* donne un intervalle non connexe.

III.2. Lois de composition interne

On définit les lois de composition internes d'union, d'intersection et de complément pour les intervalles connexes et les intervalles non connexes.

III.2.1. Intervalles connexes

Le complément d'un intervalle connexe *i* est définit comme l'intervalle de même période que *i*, dont l'instant de début est égal à l'instant de fin de *i*, et l'instant de fin égal à l'instant de début de *i*. Une fois connue la relation unissant deux intervalles, il est facile d'en déterminer l'union et l'intersection. On définit donc pour chaque relation binaire circulaire les fonctions représentant les lois d'union et d'intersection. L'exemple de la figure 11 illustre l'union de deux intervalles circulaires liés par la relation *overlaps*, et de deux intervalles liés par la relation *overlaps-is overlapped*. Connaissant les dates de début et de fin de chacun des intervalles, la déduction de leur union ou de leur intersection est immédiate.

Ceci ne concerne que les intervalles circulaires connexes. Il reste donc à définir l'union et l'intersection avec des intervalles non connexes, avec l'intervalle vide, et l'union et l'intersection entre des intervalles non connexes.

L'intervalle vide ne pose pas de problème. On a toujours, quel que soit l'intervalle i, et si on note Ø l'intervalle vide, $Ø \cup i = i$ et $Ø \cap i = Ø$.

Il reste maintenant à définir union et intersection des intervalles non connexes. On définit tout d'abord l'union et l'intersection d'un intervalle connexe et d'un intervalle non connexe.

III.2.1. Intervalles non connexes

Le complément d'un intervalle non connexe i se définit de manière aussi intuitive que le complément d'un intervalle connexe comme l'ensemble des intervalles connexes dont l'instant de début est égal à l'instant de fin d'un intervalle connexe c de i, et l'instant de fin est égal à l'instant de début de l'intervalle connexe de i suivant c (Cf. Figure 10).

Figure 10. Complément d'un intervalle non connexe.

L'intersection d'un intervalle non connexe i et d'un intervalle connexe c est définie comme l'ensemble des intersections des intervalles connexes de i avec c. Si cet ensemble est vide, l'intersection est l'intervalle vide ; si cet ensemble est réduit à un élément, l'intersection est un intervalle connexe. Si cet ensemble a plusieurs éléments, l'intersection est un intervalle non connexe.

L'union d'un intervalle non connexe i et d'un intervalle connexe c est définie comme l'ensemble des intervalles connexes de i ayant une intersection nulle avec c (ces intervalles sont laissés tels quels), ensemble auquel est ajouté l'intervalle connexe constitué par l'union globale de tous les intervalles connexes de i ayant une intersection non nulle avec c et de c. Cette union globale se définit de manière récursive. Soit a un intervalle connexe et B un intervalle non connexe dont tous les éléments $b_{i\,\{i=1..N\}}$ ont une intersection non nulle avec a. Soit la formule récursive suivante :

$$u_0 = a$$

$$u_{n+1} = u_n \cup \mathbf{B}_n$$

L'union globale de a et B est u_N, avec N le nombre d'intervalles connexes de B.

L'union et l'intersection de deux intervalles non connexes se définissent de manière récurrente, d'une façon analogue à la définition de l'union globale vue plus haut. Soient A et B deux intervalles non connexes, et N le nombre d'intervalles connexes contenus dans B. Soient les formules de récurrences suivantes :

$$v_0 = A \qquad\qquad w_0 = A$$

$$v_{n+1} = v_n \cup B_n \qquad\qquad w_{n+1} = w_n \cap B_n$$

L'union de A et B est v_N et leur intersection est w_N.

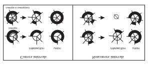


Figure 11. Union et intersection d'intervalles connexes et non connexes.

Ainsi sont définies sur I_P, ensemble des intervalles circulaires de période p, ces intervalles pouvant être l'intervalle vide, des intervalles connexes ou non connexes, les lois de composition internes de complémentation, d'union et d'intersection.

III.3. Un treillis booléen

On remarque ici, sans le démontrer, que l'ensemble I_p muni de la relation d'inclusion $\subseteq$ constitue un treillis booléen. La relation d'inclusion est définie par rapport aux relations circulaires déterminées plus haut. On dit que $x \subseteq y$ si x et y sont mis en relation par l'une des relations : *equals-period* (1*)*, *starts-period* (2), *finishes-period* (3), *overlaps-is overlapped-period* (4), *is overlapped-period* (5), *during-period* (6), *equal*s (8), *starts* (9), *finishes* (17)ou *during* (19)[1]. Cette relation correspond à la notion intuitive de «pendant».

L'union et l'intersection dans I_p sont idempotentes, commutatives et associatives, et les lois d'absorption $x \cap (x \cup y)=x=x \cup (x \cap y)$ sont vérifiées ; de plus, elles sont distributives l'une par rapport à l'autre. Par théorème, on en conclut qu'on peut définir sur I_p une unique relation d'ordre $\subseteq$ telle que $(I_p, \subseteq)$ soit un treillis distributif. En outre, I_p possède un plus petit élément, l'intervalle vide, et un plus grand élément, l'intervalle de durée égale à p. $(I_p, \subseteq)$ est donc un treillis booléen.

IV. Application à l'analyse de grilles

En pratique, le modèle temporel retenu pour le système d'analyse harmonique est le modèle d'exécution sans origine et sans regroupement (26 relations circulaires binaires distinctes, illustré par la Figure 6) [Pachet & al 95]. Ce choix a été dicté par un soucis d'équilibre entre simplicité et expressivité. Le modèle le plus simple - comportant le moins de relations binaires - est le modèle sans origine et avec regroupement (19 relations). le modèle le plus expressif, soit le modèle comportant le plus d'information, est le modèle avec origine et sans regroupement. Ce dernier comporte 150 relations binaires distinctes, ce qui le rend difficile à utiliser en pratique. Le modèle le plus simple ne permet pas de distinguer les relations faisant intervenir une ou plusieurs formes couvrant une période entière. Ce dernier point ne semble pas, à l'heure actuelle, présenter un handicap significatif, mais le gain obtenu (7 relations) ne paraît pas justifier une perte d'information pouvant se révéler préjudiciable pour des développements ultérieurs.

La perte de l'information d'origine peut être facilement palliée dans l'écriture même des règles. Cette information est en effet toujours disponible : les intervalles sont caractérisés par une date de début, une

---

[1] . Les numéros entre parenthèses renvoient à la figure XX.

durée, ainsi que la durée totale d'une période ; l'origine étant représentée ici par la date 1 (1er temps de la 1ère mesure), il est simple de tester si un intervalle contient ou ne contient pas l'origine.

Le regroupement des intervalles couvrant une période entière peut être obtenu pour les relations binaires en « hiérarchisant » de manière adéquate les relations binaires.

On constate donc que, même si l'un des modèles a été privilégié dans la réalisation informatique, on peut d'une certaine façon « naviguer » entre les différentes représentations, avec ou sans origine, avec ou sans regroupement. Il paraît donc justifié de dire que le modèle sous-jacent à la représentation temporelle des différentes suites d'accords est le modèle le plus complet, soit le modèle avec origine et sans regroupements, même si la réalisation est fondée sur un modèle plus simple (sans origine et sans regroupements).

V. Conclusion

Le système d'analyse de grilles de jazz enrichi de ce modèle du temps circulaire permet d'analyser correctement la plupart des grilles standards : dans un grand nombre de cas, les analyses fournies sont satisfaisantes, et se rapprochent suffisamment de l'analyse que pourrait faire un musicien [Carrive 95], même si des améliorations du système, suggérées dans [Carrive 95], pourraient encore améliorer ses performances.

Notons enfin que ce modèle développé pour des motivations essentiellement musicales - la circularité des grilles de Jazz - s'est trouvé applicable à des domaines non musicaux. Un travail en cours [Pachet & Dojat 95], [Pachet & al 96] suggère que le modèle circulaire ainsi défini se révèle applicable directement à des domaines tels que le diagnostic médical temps réel. Cette recherche, en cours, nous conforte dans l'idée que la construction de systèmes réalisant des tâches musicales bien définies nous permet d'aborder, modestement mais de manière précise, des question fondamentales de représentation de connaissances.

VI. Bibliographie

[Allen 84] Allen J.F., *Towards a General Theory of Action and Time*, in Artificial Intelligence, vol. 23, pp. 123-154.

[Carrive 95] Carrive, J. Analyse de grilles de Jazz dans le système MusES. Rapport de DEA Iarfa, Université Paris 6, 1995.

[Dojat 94] Dojat, M. Contribution à la représentation d'expertises médicales dynamiques : application en réanimation médicale. Thèse de l'Université de Compiègne, 1994.

[Dojat and Sayettat 1994] Dojat M and Sayettat C. *Aggregation and forgetting: two key mechanisms for across-time reasoning in patient monitoring*. AAAI spring symposium on Artificial Intelligence in Medicine, Stanford University (Ca), 1994.

[Fake 1983] The World's Greatest Fake book. San Franscisco, Sher Music Co 1983.

[Holland 89] Holland S., *Artificial intelligence, Education and Music*, Ph.D. thesis, Open University, Milton Keynes (GB), 1989.

[Lerdahl & Jackendoff 83] Lerdhal F., Jackendoff R., *A Generative Theory of Tonal Music*, MIT Press.

[Maxwell 92] Maxwell H.J., *An Expert System for Harmonizing Analysis of Tonal Music*, in Understanding Music with AI: Perspectives on Music Cognition, K.E. & O.L.M. Balaban, Ed., AAAI press, pp. 335-353.

[Mouton & Pachet 95] Mouton, Pachet F., *The Symbolic vs. Numeric Controversy in Automatic Analysis of Tonal Music*, in Workshop IJCAI'95 on Music an Artificial Intelligence, Montreal, pp. 32-40.

[Pachet 94] Pachet F., *An Object-Oriented Representation of Pitch-Classes, Intervals, Scales and Chords: The Basic MusES*, in Rapport Interne Laforia 93/38, Laforia.

[Pachet 94b] Pachet F., *The MusES System: an Environment for Experimenting with Knowledge Representation Techniques in Tonal Harmony*, in Proceedings First brazilian symposium on Computer Music - SBC&M'94, Caxambu, Minas Gerais, Brazil, pp. 195-201.

[Pachet 94c] Pachet F., *Vers un modèle du raisonnement dans les langages à objets*, in Langages et Modèles à Objets (L.M.O.), Grenoble, pp. 111-123.

[Pachet & al 95] Pachet, F. Ramalho, G. Carrive, J. Cornic, G. Representing temporal musical objects and reasoning in the MusES system. International Congress in Music and AI, Edinburgh University, pp. 33-48, Sept. 1995.

[Pachet & al 95] Pachet, F. Dojat, M. Carrive, J. Representation of circular time in medical knowledge-based systems. Submitted to the special issue of Computers in Biology and Medicine on Time-Oriented Systems, 1996.

[Pachet & Dojat 95] Pachet F., Dojat M., *Représentation de connaissances temporelles par objets et règles*, in Rapport interne Laforia 95/19.

[Pachet & Roy 95] Pachet, F. Roy, P. Mixing constraints and objects: a case study in automatic harmonization. *TOOLS Europe '95*. Prentice-Hall. pp. 119-126.

[Pachet & Roy 95b] Pachet, F. Roy, P. Integrating constraint satisfaction techniques with complex object structures. *15th Annual Conference of the British Computer Society Specialist Group on Expert Systems, ES'95*. Cambridge, pp. 11-22. December 1995.

[Ramalho and Ganascia 1994] Ramalho G and Ganascia J-G. *Simulating Creativity in Jazz Performance*. 12th AAAI Conference, Seattle, AAAI Press, p. 108-113,1994.

[Ramalho & Pachet 94] Ramalho G., Pachet F., *From Real Book to Real Jazz Performance*, in International Conference on Music Perception an Cognition, Lièges, Belgium, pp. 349-350.

[Real 1981] The Real Book, The Real Book Press 1981.

[Siron 90] Siron, J. La partition intérieure, Editions Outre mesures, Paris, 1990.

[Steedman 84] Steedman M.J., *A Generative Grammar for Jazz Chord Sequences*, in Music Perception, vol. 2, pp. 52-77.

[Steels 1979] Steels L. Reasoning modeled as a Society of Communicating Experts, Technical Report MIT AI Lab., AI-TR-542, 1979.

[Ulrich 77] Ulrich W., *The Analysis and Synthesis of Jazz by Computer*, in Proceedings 5th IJCAI, MIT, Cambridge, Ma, pp. 865-872.

[Winograd 1968] Winograd T. Linguistic and Computer Analysis of Tonal Harmony. Journal of Music Theory 12 : 2-49, 1968.

[Winold and Bein 1983] Winold, A and Bein, J. BANALYSE: An Artificial Intelligence System for Harmonic Analysis of Bach Chorales, Technical Report Indiana University, Unpublished manuscript, 1983.

VII. Annexe : exemple d’analyse

Voici l'analyse de la grille Blues for Alice de la Figure 1. Les numéros indiquent les temps. Les noms des formes sont les noms des classes des objets reconnus par le système d’analyse.

```
1-48 BluesShape in F MajorScale
    45-4 ResolvanteEndToBeginning in F MajorScale
      45-48 ChordSubstitution in F MajorScale
        45-48 TwoFive in F MajorScale
            [G #min 7 ]
            [C 7 ]
      [F #maj7 ]
    5-20 ChordSubstitution in Bb MajorScale
      5-20 Resolvante in Bb MajorScale
        5-16 ChordSubstitution in Bb MajorScale
            5-16 TwoFive in Bb MajorScale
               5-14 ChordSubstitution in C HarmonicMinor
                  5-14 Resolvante in C HarmonicMinor
                      5-12 ChordSubstitution in C MajorScale
                          5-12 TwoFive in C MajorScale
                            5-10 ChordSubstitution in D HarmonicMinor
                               5-10 Resolvante in D HarmonicMinor
                                   5-8 ChordSubstitution in D HarmonicMinor
                                       5-8 TwoFive in D HarmonicMinor
                                          [E #halfDim7 ]
                                          [A 7 ]
                                   [D #min 7 ]
                            [G 7 ]
                      [C #min 7 ]
               [F 7 ]
        [Bb #maj7 ]
    17-48 ExtendedShape in F MajorScale
      17-46 ExtendedShape in F MajorScale
        17-44 ExtendedShape in F MajorScale
            [Bb #maj7 ]
            21-44 ExtendedShape in F MajorScale
               21-42 ChordSubstitution in F MajorScale
                  21-42 Resolvante in F MajorScale
                      21-40 ChordSubstitution in C MajorScale
                          21-40 Resolvante in C MajorScale
                            21-32 ChordSubstitution in Db MajorScale
                               21-32 Resolvante in Db MajorScale
                                   21-28 ChordSubstitution in D MajorScale
                                       21-28 Resolvante in D MajorScale
                                          21-24 ChordSubstitution in Ab MajorScale
                                           21-24 TwoFive in Ab MajorScale
                                                [Bb #min 7 ]
                                                [Eb 7 ]
                                          25-28 ChordSubstitution in G MajorScale
                                           25-28 TwoFive in G MajorScale
                                                [A #min 7 ]
                                                [D 7 ]
                                   29-32 ChordSubstitution in Gb MajorScale
                                       29-32 TwoFive in Gb MajorScale
                                          [Ab #min 7 ]
                                          [Db 7 ]
                            33-40 ChordSubstitution in F MajorScale
                               33-40 TwoFive in F MajorScale
```

[G #min 7]
[C 7]
[F #maj7]
[D #min 7]
[G #min 7]
[C 7]