Dan Timis

THEORIE DES CODES ET ANALYSE MUSICALE

These pour le

D. E. A. d'Esthetique et Sciences de l'Art

option Musique et Scenographie

THEORIE DES CODES ET ANALYSE MUSICALE

Mots et Monoides

Soit un ensemble A nomme alphabet (on va nommer ses elements des lettres, notees "a", "b" etc.). Une suite finie de lettres s'appelle un mot, et l'ensemble des suites de lettres c'est l'ensemble des mots qu'on peut former avec l'alphabet(les mots seront notes "w", "u" etc. et l'ensemble des mots formes sur A sera note A*). Exemple:

$A = \{ a, b \}$

$A^* = \{$'a', 'b', 'aa', 'ab'....'aababb' etc.$\}$

On definit avec l'ensemble des mots une operation nommee concatenation(notee ^). Cette operation est associative mais non comutative:

$a \frown b \neq b \frown a$

$(a \frown b) \frown c = a \frown (b \frown c)$ (1)

(on utilisera la notation "ab" au lieu de "a ^ b" la ou il n'y aura pas risque de confusion)

Quand deux mots sont concatenes on dit qu'ils sont contigus. Si une lettre ou une suite de lettres se trouve a l'interieur d'un mot on dit que le mot contient une occurence de la lettre ou de la suite de lettres. Si une occurence se trouve au debut d'un mot c'est un facteur gauche, si elle se trouve a la fin d'un mot, c'est un facteur droit.

Dans l'ensemble des mots on trouve le mot vide (note "e") qui est element neutre pour la concatenation:

$$ae = ea = a$$

Le degre d'un mot est le nombre de lettres dans le mots. Il est note avec des barres verticales. Exemples:

$$|ababb| = 5$$
$$|e| = 0$$

Une suites obtenue par concatenation a comme degre la somme des degres de ses composantes. Par la concatenation des mots on obtient d'autres mots, donc la concatenation est une loi de composition partout definie. Un ensemble muni d'une loi de composition partout definie, associative et possedant un element neutre s'apelle un monoide. L'ensemble A* muni de l'operation de concatenation devient un monoide libre sur A.

Le codage

On peut considerer un code comme l'application d'un monoide dans un autre monoide. Les deux monoides seront un ensemble des suites V* formee sur l'ensemble V (vocabulaire) et l'ensemble des suites A* formee sur A ,les deux munis de l'operation de concatenation.

L'application "f" est telle que pour des suites vi et vj de V on a:

$$f(vi\text{^}vj) = f(vi)\text{^}f(vj)$$

Si on prend par exemple un code binaire, soit A = {0, 1} et V = {v1, v2, v3, v4} avec:

f(v1) = 00
f(v2) = 01
f(v3) = 10
f(v4) = 11

on aura par exemple:

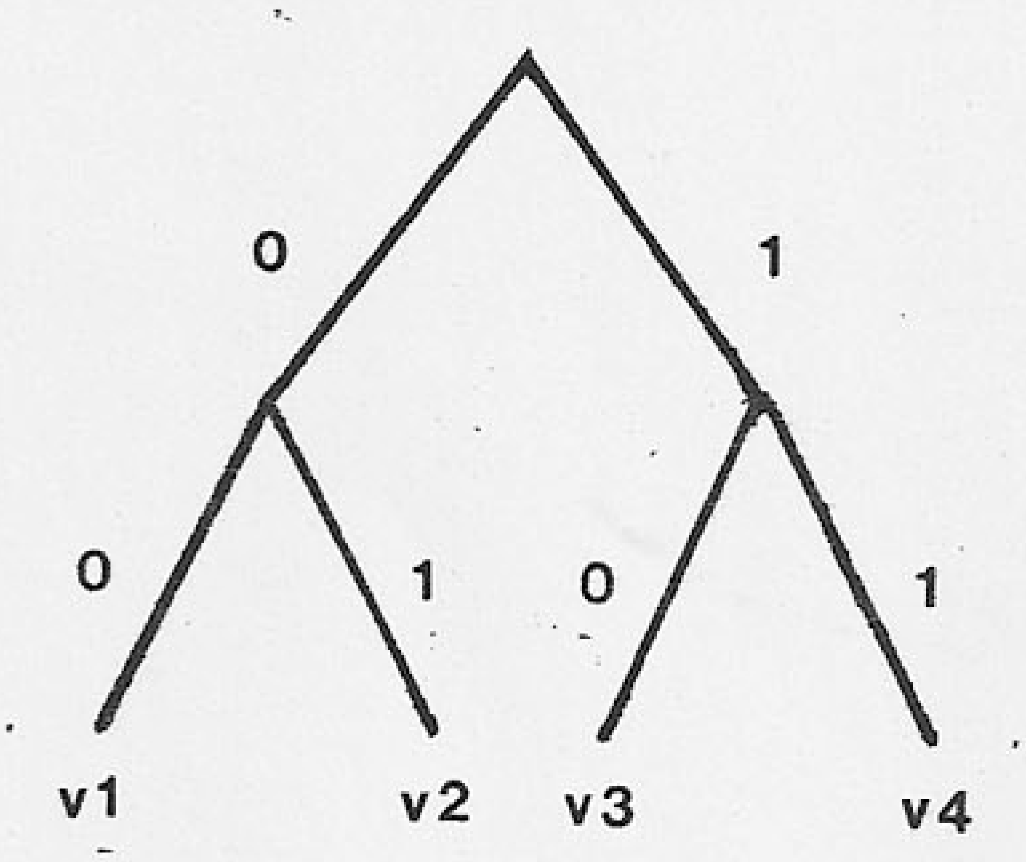
$$f(v2\text{^}v3) = f(v2)\text{^}f(v3) = 0110$$

ce qui revient a dire qu'on va coder la suite v2v3 par la suite 0110.

L'aplication "f" est un isomorphisme entre les suites du vocabulaire V et un sous-ensemble des suites de l'alphabet A. Dans l'exemple precedent, chacune des suites de V a une suite correspondante de A mais on n'a pas de correspondance inverse pour toutes les suites de A (par exemple pour 101). Plus precisement, dans notre exemple, le sous-ensemble des suites de A est l'ensemble des suites de longueur paire.

Le terme codage designe plusieurs operations comme le codage, la transmission, la reception, le decodage etc. En general on remplace une suite de V par une suite de A correspondante, les suites de A pouvant etre utilisees plus facilement dans un systeme de transmission (l'alphabet etant presque toujours beaucoup plus restreint que le vocabulaire). Pour reconstituer le contenu d'un message on doit proceder a l'operation inverse, le decodage, en remplacant la suite de A par la suite de V correspondante.

Ce n'est pas notre objectif de rentrer dans les details de la theorie des codes, mais il est necessaire de preciser qu'il y a differents types de codes (codes generaux, codes en arbre, codes anagrammatiques etc.). Dans notre exemple les suites de A qui correspondent a un seul element de V ont la meme longueur (2). C'est un code uniforme. Si on concoit un systeme de decodage comme un automate en forme d'arbre:

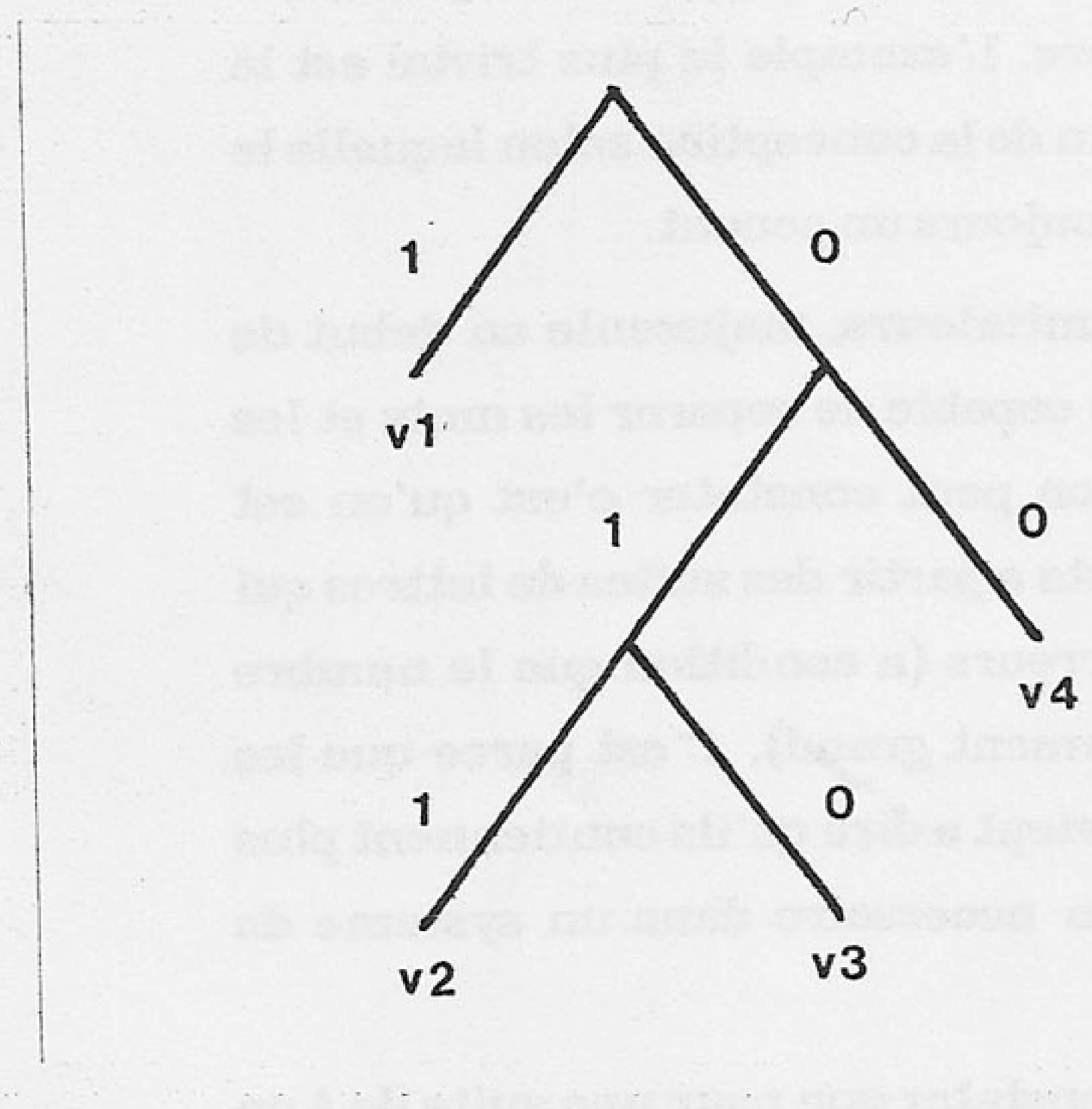


on trouve l'inconvenient suivant de notre code: si au cours de la transmission du message un bruit ou une erreur provoque la disparition d'un element de A la suite du decodage est compromise. Il existe des codes ou cet inconvenient est elimine, les codes a auto-synchronisation. Dans ces codes la frontiere des mots(on poura considerer le mot comme une suite de A correspondant a un seul element de V), est toujours indiquee par une suite particuliere de lettres. Voila un exemple de code a auto-synchronisation:

$V = \{v1, v2, v3, v4\}$ et $A = \{0, 1\}$

$f(v1) = 1$
$f(v2) = 011$
$f(v3) = 010$
$f(v4) = 00$

L'arbre de decodage sera:



Si par exemple on code la suite v2v3v4 par 01101000 et suite a une mauvaise transmission on recoit le message sans le premier 0 on decodera v1v1v3v4, et apres un certain temps on parviendra a se synchroniser.

Les codes naturels

Plus complexes et beaucoup plus interessantes pour notre demarche sont les codes naturels. Prenons l'exemple des langues parlees*. On va eliminer la composante phonologique qui pose certains problemes qui depassent le cadre de notre etude et on va se placer devant un texte ecrit**.

Ce qu'on peut constater tout de suite c'est que l'autocorrelation est assuree par la presence d'un ensemble de delimitateurs (espace, point, virgule etc.) qui n'ont pas toujours un correspondant phonique et qui sont tres commodes mais pas absolument necessaires. Notons que dans le cas de la musique ecrite on peut avoir aussi des delimitateurs sans correspondance evidente dans le domaine sonore. L'exemple le plus trivial est la barre de mesure. On est deja loin de la conception selon laquelle le premier temps de la mesure a toujours un accent.

Si on elimine tous les delimitateurs, majuscule en debut de phrase incluse, on est toujours capable de separer les mots et les phrases. Une autre chose qu'on peut constater c'est qu'on est capable de reconstituer des mots a partir des suites de lettres qui ont ete transmises avec des erreurs (a condition que le nombre d'erreurs ne soit pas ecxessivement grand). C'est parce que les mots sont redondants ce qui revient a dire qu'ils contiennent plus d'information que le minimum necessaire dans un systeme de transmission sans faute.

En meme temps on peut constater que pour une suite de A on peut avoir plusieurs element de de V correspondants (le mot est l'image de plusieurs elements du vocabulaire). On parle alors d'ambiguite.

---

*) on va considerer la musique comme un langage naturel et par consequence on va essayer d'etablir quelques similitudes et quelques differences par rapport aux langues parlees et aux codes naturels.

**) on va eliminer de la meme facon la composante sonore de la musique en supposant soit qu'on est devant une partition, soit qu'en ecoutant une musique on a deja repere les hauteurs et les durees.

On va essayer de decrire un modele du decodage qu'un etre humain pourra faire a partir d'un texte ecrit. Notre texte n'a pas de delimitateur et il contient un nombre acceptable* de fautes (des lettres qui manquent, des lettres en trop ou erronees). En plus il commence et il finisse au milieu d'un mot. Meme si on ne le trouve pas tel quel, dans la realite, un cas pareil peut etre considere comme une simplification de la situation suivante : un sujet ecoute un bout de discours; le discours est prononce sans aucune intonation (donc pas de separateurs); il y a des bruits qui favorise l'aparition des erreurs dans la perception de la composante phonologique; on a deja repere les phonemes donc on poura eventuellement ecrire une suite de lettres ou de symboles phonetiques.

Voila le modele (dans un langage proche des langages de programmation):

pour chaque lettre on fait:

> pour chaque suite de lettres commencant avec cette lettre et qui a une longueur "acceptable" on doit faire:
>
> > chercher s'il y a occurence entre cette suite et une suite du dictionaire (le dictionnaire etant l'ensemble des mots de A* qui correspondent a un seul symbole de V) ou s'il y a une resemblance "acceptable" entre notre suite et une ou plusieurs suites du dictionnaire
>
> fin de la recherche pour chaque suite.

---

*) la mesure de l'acceptabilite est un chose tres delicate. On va beaucoup utiliser le mot "acceptable" pour separer des phenomenes qui peuvent se mesurer sur une echelle de valeurs, le point de separation entre "acceptable" et "non-acceptable" etant assez subjective.

fin de la recherche pour chaque lettre.

(jusqu'ici c'est assez clair)

parmi tous les segmentations possibles chercher celle qui semble la plus vraissemblable (en tenant comte eventuellement de la signification).

(ca c'est deja tres vague)

Une chose qui s'impose par rapport a la premiere definition de la notion de code et l'introduction de la notion de "ressemblance". Au lieu d'avoir une correspondance entre chaque element de V et une suite de A, on va avoir une echelle de "ressemblance" entre tous les suites de A et tous les suite du sous-ensemble des suites de A qui ont un correspondant dans V. Cette echelle peut aller de la totale ressemblance a la dissemblance absolu en passant par un certain nombre de degres intermediaires.

Bien sur notre modele n'est pas optimal. Un sujet humain va utiliser d'autres methodes plus rapides. Il ne va pas chercher, par exemple, a rallonger des chaines qui n'ont pas de chance d'etre des debuts de mots; il va se servir d'un possible repertoire de syllabes. Probablement un bon modele sera toujours sous forme d'arbre, mais un tel arbre est beaucoup plus complexe pour etre decrit ici.

En plus de l'ambiguite due au fait que pour un mot de A on peut avoir plusieurs elements de V correspondants, la segmentation peut etre difficile parsqu'on a des mots qui peuvent etre des facteurs d'autres mots ou bien parce qu'on peut rencontrer le cas ou le facteur droit d'un mot et le facteur gauche d'un autre mot contigu au premier forment un autre mot. Les choses sont encore plus complexes quand on est devant la composante phonologique parce que des mots qui se prononcent de la meme facon s'ecrivent differament ce qui facilite le decodage d'un texte ecrit. L'exemple suivant d'ambiguite de segmentation est bien connu:

Gal, amant de la reine, alla (tour magnanime),

Galamment de l'arene a la Tour Magne, a Nimes.

Dans une situation semblable a celle decrite plus haut, c'est a dire devant un texte musical sans separateurs, qui commence et finisse au milieu d'une phrase, avec un nombre "acceptable" d'erreurs, on est toujours capable de separer des unites musicalement significatives. On est oblige alors de chercher un autre type de code et de dictionnaire (pas forcemment tres eloigne de notre modele).

Dans le cas des langues parlees, la capacite de decodage est acquise par la formation dans la memoire des liaisons entre les suites de A (suites de sons ou de lettres) et les elements de V (objets, categories d'objets, idees etc.), la nature des elements de V entrant dans la competance de la semantique. En admettant que l'acquisition par l'experience d'un certain nombre de categories de structures musicales est une condition indispensable pour la reception du phenomene musical on peut essayer la formalisation d'un code musical.

L'algebre des phenomenes temporelles

On poura considerer le temps comme une droite sur laquelle on a des points qu'on va nommer moments. Ces moments auront une valeur numerique et seront ordonnes (suivant par exemple la droite des reels). On va definir un phenomene simple comme etant un triplet (N, x, y) avec N - nature du phenomene, x - moment du debut du phenomene et y - moment de fin du phenomene. Alors on poura associer a chaque phemonene une valeur appelee duree et qui sera numeriquement egale a y - x. Pour definir la nature du phenomene on peut considerer un ensemble C (caracteristiques), la nature etant un sous-ensemble de cet ensemble.

Lorsqu'on considere l'ensemble de tous les phenomenes (tous les triplets avec les sous-ensembles de C et l'ensemble des moments) on peut definir une relation d'equivalence comme suit:

$(N, x, y) \sim (M, z, w)$ si et seulement si $N = M$ et $y - x = w - z$
(c'est a dire les deux phenomenes ont la meme nature et la meme duree)

On obtient ainsi des classes d'equivalence sur l'ensemble des phenomenes. Pour chaque classe d'equivalence on aura un representant "etalon" (N, x, y) avec $x = 0$.

Bien sur on aura des phenomene avec $x = y$ (duree 0) et d'autres avec $x > y$ (duree negative). En ce qui concerne la nature des phenomene on aura parmi les sous ensemble de C l'ensemble vide (notee θ) et pour chaque sous-ensemble non-vide N on aura un inverse (notee !N) de tel que

$N \cup !N = \theta$

On peut definir maintenant une premiere egalite fondamentale:

$(N, x, y) = (!N, y, x)$ (2)

Ainsi un phenomene avec une duree negative sera l'equivalent du phenomene de nature contraire et de duree positive et numeriquement egale a l'inverse de la duree negative. En combinant les deux relation d'equivalence on obtient des classes d'equivalence, chacune ayant un representant etalon de duree positive.

Les structures

Une structure simple sera un couple (N, d) avec N nature du phenomene et d duree. L'egalite (2) va devenir alors:

$$(N, d) = (!N, -d)$$

Ainsi on aura un isomorphisme entre l'ensemble des structures de duree positive et les classes d'equivalence definie auparavant sur l'ensemble des phenomenes.

Les ensembles de phenomenes

On definit maintenant un phenomene complexe comme etant un ensemble de phenomenes simples. Pour creer des phenomenes complexes on va utiliser la reunion des phenomenes simples. On va avoir les egalitees suivantes:

$$\{ (N, x, y) \} \cup \{ (M, z, z) \} = \{ (N, x, y) \}$$
$$\{ (N, x, y) \} \cup \{ (\theta, z, w) \} = \{ (N, x, y) \} \quad (3)$$

(le phenomene de duree nulle et le phenomene de nature vide sont neutres pour l'operation de reunion)

Deux phenomenes simples sont contigus si et seulement si le moment de fin d'un phenomene est egal au moment de debut de l'autre. Exemple:

(N, x, y) et (M, y, z) sont contigues

Voici maintenant une deuxieme egalite fondamentale des phenomenes temporels:

$$(N, x, y) \cup (M, x, y) = (N \cup M, x, y) \quad (4)$$

Premiere consequence importante:

$$(N, x, y) \cup (M, z, w) = (N, x, z) \cup (N \cup M, z, y) \cup (M, y, w) \quad (5)$$

pour toute paire de phenomenes simples

(toutes les phenomenes sont des ensembles de phenomenes simples contigus)

Demonstration:

$$(N, x, z) \cup (N \cup M, z, y) \cup (M, y, w)$$
$$= (N, x, z) \cup (N, z, y) \cup (M, z, y) \cup (M, y, w)$$

(conformement a (4))

$$= (N, x, z) \cup (!N, y, z) \cup (!M, y, z) \cup (M, y, w)$$

(conformement a (2))

en appliquant la formule (5) une fois pour les deux premiers termes et une deuxime fois pour les deux autres on obtient:

$$(N, x, y) \cup (N \cup !N, y, z) \cup (!N, z, z)$$
$$\cup (!M, y, y) \cup (!M \cup M, y, z) \cup (M, z, w)$$
$$= (N, x, y) \cup (M, y, z)$$

(conformement a (3))

Deuxieme consequence importante:

$$(N, x, y) \cup (N, y, z) = (N, x, z) \quad (6)$$

(deux phenomenes de meme nature et contigus forment un seul phenomene simple)

En effet:

$$(N, x, y) \cup (N, y, z) = (N, x, y) \cup (!N, z, y)$$

(conformement a (2))

en apliquant la formule (5) on obtient:

$$(N, x, z) \cup (N \cup !N, z, y) \cup (!N, y, y) = (N, x, z)$$

Tout phenomene complexe peut s'ecrire sous la forme d'une reunion de phenomenes simples de duree positive contigus. On va appeler cette ecriture forme normale. Les caracteristiques de la forme normale sont:

1) Tous les duree sont positives et non-nulles.
2) On a un seul phenomene de debut (dont le moment de debut differe de tous les autres moments) et un seul phenomene de fin (dont le moment de fin differe de tous les moments des autres phenomenes.

3) Tous les autres phenomenes (a l'exception des deux decrits plus haut) ont leur moment de debut identique au moment de fin d'un et d'un seul autre phenomene et leur moment de fin identique au moment du debut d'un et d'un seul autre phenomene

Pour exprimer un ensemble de phenomenes dans la forme normale on utilise les regles suivantes:

1) Tous les phenomenes de duree negative sont transformes en phenomene de duree positive en utilisant (2). Les phenomenes de duree nulle sont elimines conformement a (3). Les phenomenes de meme nature et contigus sont simplifies conformement a (6) et on reunit les natures des phenomene qui ont le meme temp de debut et le meme temps de fin en apliquant (4).
2) Ensuite on va ordonner les phenomenes dans l'ordre croissant des moments de debut et pour les phenomenes successifs non-contigus on applique (5).
3) Si par l'application de (5) on a de nouveau des durees negatives on recommence les operations 1) et 2)

En faisant correspondre a chaque ensemble de phenomenes normalises une suite de structures simple, chaque structure etant en correspondance avec un phenomene simple (meme nature, duree egale a la difference des moments de debut et de fin) on obtient des classes d'equivalences sur l'ensemble des phenomenes complexes.

Le code musical

Dans ce cas notre ensemble V sera forme d'un certain nombre d'elements chacun representant une categorie de structure musicale (rithmique, melodique, harmonique etc.). On poura trouver dans cet ensemble des elements tels que: accord majeur, mouvement conjoint, arpege ascendent etc. L'ensemble A est le produit des ensembles H (hauteurs) et D (durees). C'est l'ensemble des structures simples.

On dispose de deux operation: concatenation (notee ^) et superposition (notee /). Conformement a (1) la concatenation est associative et non commutative. La superposition est commutative et associative:

$$(a\hat{}b)\hat{}c = a\hat{}(b\hat{}c)$$
$$a\hat{}b \neq b\hat{}a$$

$$(a/b)/c = a/(b/c)$$
$$a/b = b/a$$

La concatenation a comme elements neutres tous les sons de duree nulle et la superposition a comme elements neutres tous les sons de hauteur nulle. Le son de hauteur et de duree nulle "e" est element neutre pour les deux operations.

$$a\hat{}e = e\hat{}a = a$$
$$a/e = e/a = a$$

A l'aide des deux operations on va former des structures complexes. On va avoir l'ensemble des structures A* et les structures simples vont faire aussi partie de cet ensemble a cote des structures complexes. Cet ensemble est ferme par rapport au deux operations, la concatenation et la superposition. En plus entre chaque structure et une classe d'equivalence dans l'ensemble des phenomenes il y a correspondance.

Dans ce qui suit on va s'intereser aux structures formees par la concatenation et la superposition des structures simples de duree positive et non-nulle. On peut etablir des classes d'equivalences a l'interieur de l'ensemble des structures en les transformant dans la forme normale. Cette forme normale va etre definie comme une concatenation de strctures simples. Toutes les structures peuvent etre ecrites sous la forme normale. Pour cela il faut attribuer a la structure qu'on veut transformer un phenomene complexe equivalent (eventuellement commencant avec le moment 0). Ensuite le phenomene sera ramene a sa forme normale, apres quoi on retransforme le phenomene dans sa structure.

Le degre d'une structure sera alors le nombre de sructures simples de sa forme normale. Sa duree sera la somme des durees des composantes simples.

La fonction qui va faire correspondre un phenomene a une structure sera definie de la facon suivante par rapport au deux operations:

Si on a deux structures a = (N, d1) et b = (M, d2) alors

$f(a\frown b) = (N, x, y) \cup (M, y, z)$

avec

$d1 = y - x$ et $d2 = z - y$

$f(a/b) = (N, x, y) \cup (M, x, z)$

avec

d1 = y - x et d2 = z - x

Il estevident que la superposition est commutative et que la concatenation ne l'est pas.

La duree d'une concatenation sera la somme des durees des composants. La duree d'une superposition de structures simples sera la duree maximale des durees des composants. Son degree sera le nombre de durees differantes de toutes les concatenation. De cette maniere on peut calculer la duree et le degree d'une structure sans la transformee dans un phenomene equivalent. Exemple:

on va noter par # la fonction "nombre des valeurs differantes" et par "max" le maximum

Soit la structure:

(a / b) ^ (c / (b ^ a))
avec a = (N, 5)
b = (M, 2) et c = (K, 7)

la duree sera:

max (5, 2) + max(7, (2 + 5) = 12

le degree sera: # (5, 2) + # (7, (2, 2+5)) = 4

La concatenation est distributive par rapport a la superposition:

a^(b/c) = (a^b)/(a^c)

En effet:

$$(N, x, y) \cup ((M, y, z) \cup (K, y, w))$$
$$= ((N, x, y) \cup (M, y, z)) \cup ((N, x, y) \cup (K, y, w))$$

Si on admet la presence d'un element inverse pour touts les elements de A (notee !a) de telle facon que:

$$a\hat{}(!a) = (!a)\hat{}a = e$$
$$a/(!a) = (!a)/a = e$$

l'ensemble des structures qu'on peut former sur A avec les deux operations ^ et / forment un corps.

On va admettre qu'on peut appliquer les deux operations aux elements de V (l'ensemble des categories de structures). On va pouvoir avoir des superpositions ou des concatenations d'accords, de gammes, de series dodecaphoniques etc.

Soit l'ensemble des structures de A (note S), on peut avoir des sous-ensembles de structures (S1, S2, etc.). On definie alors une fonction "f" qui va faire correspondre a chaque element de V un sous-ensemble de S.

$$f(vi) = Si$$

Le code musical sera alors une application des structures de V dans les structures de A telle que:

$$f(vi\hat{}vj) = f(vi)\hat{}f(vj)$$

$$f(vi/vj) = f(vi)/f(vj)$$

Par la concatenanation des ensemble de structures on comprend l'ensemble de toutes les structures formees par la concatenation d'une structure appartenant au premier ensemble avec une autre structure appartenant au deuxieme ensemble. La superposition de deux ensembles de structures sera l'ensemble de toutes les superpositions d'une structure contenue dans le premier ensemble avec une autre contenue dans le deuxieme.

Mais ce qui nous interesse plus c'est l'application inverse (on va l'appeler g). On peut avoir des sous-ensembles dans V (notes V1, V2, etc.). Alors "g" sera telle que si:

$$f(v1) = S1,\ f(v2) = S2 \ldots f(vi) = Si$$
$$\text{et } s \in S1,\ s \in S2 \ldots s \in Si$$

alors on aura:

$$g(si) = v1 \cup v2 \cup \ldots \cup vi$$

Par exemple si a la notion d'"accord" correspond un ensemble de structures et si une structure particuliere fait partie de cet ensemble, alors, a cette structure correspond un ensemble de notions, et la notion "accord" fait partie de cette ensemble. A la structure:

on va faire correspondre l'ensemble {accord, majeur, premier degre, agpege, mouvement melodique ascendant, valeurs egales, monophonie, mouvement rapide etc.}

Toutes les structures de A n'ont pas forcement un correspondant dans V (chaque langage, chaque style de musique a son dictionnaire, et la correspondance entre les structures des deux ensembles est fonction de l'experience de l'auditeur).

Le decodage de la musique

Il est necessaire maintenant d'introduire une echelle de "ressemblance" entre les structures de A qui ont un correspondant dans V, avec toutes les structures de A. Sur cette echelle on va fixer un point d'acceptabilite. Maintenant on peut essayer d'esquisser un model de decodage musical:

La structure est ramenee a sa forme normale.

pour chaque son (paire hauteur-duree) faire:

pour chaque superposition existante de ce son avec d'autre sons faire:

pour chaque concatenation de cette superposition avec d'autres superpositions existantes jusqu'a une limite "acceptable" faire:

chercher si il y a une

ressemblance "acceptable" entre la structure ainsi obtenu et une ou plusieurs structures qui ont un correspondant dans V.

fin de la recherche pour chaque structure.

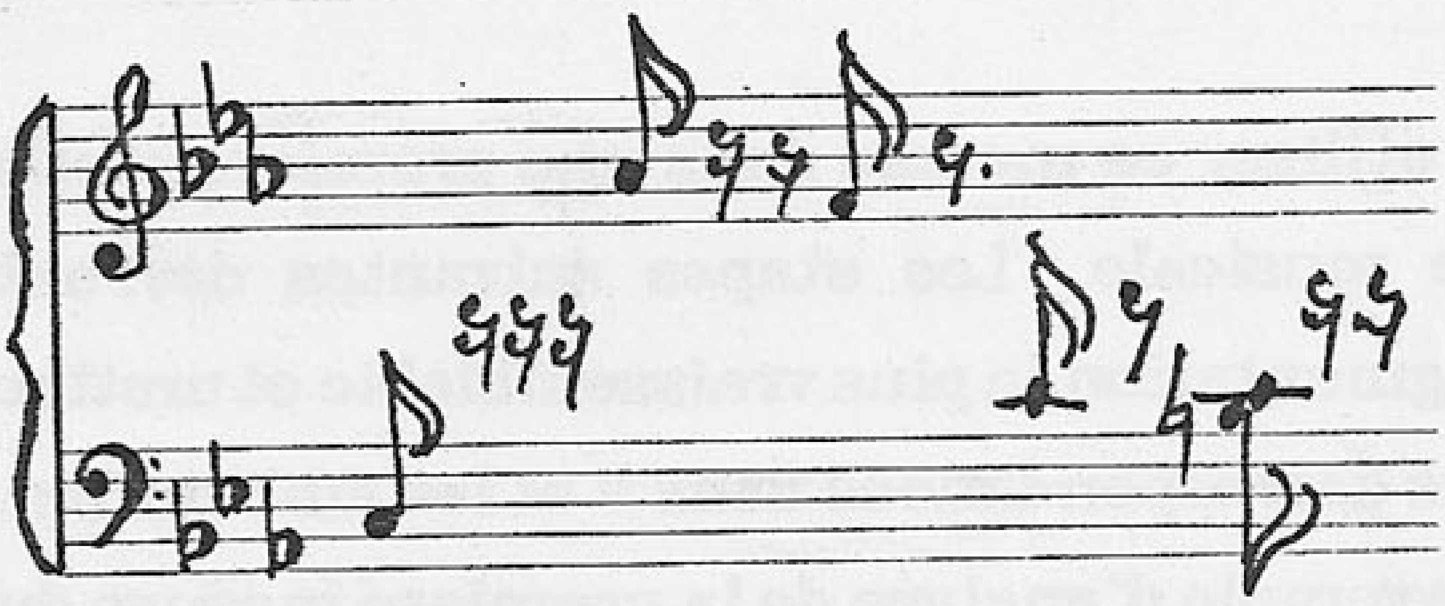
fin de la recherche pour chaque superposition.

fin de la recherche pour chaque son.

On poura utiliser ce modele dans une premiere etape d'une analyse musicale. Les etapes suivantes devront chercher la segmentation la plus vraissemblable et mettre en evidence les correspondances dans V et les ambiguites. Voici un petit exemple d'analyse de la premiere mesure du deuxieme prelude du "Clavecin bien tempere" deuxime volume de J. S. Bach:

La forme normale est:

On pourra isoler cette structure qui n'a pas de correspondant en V:

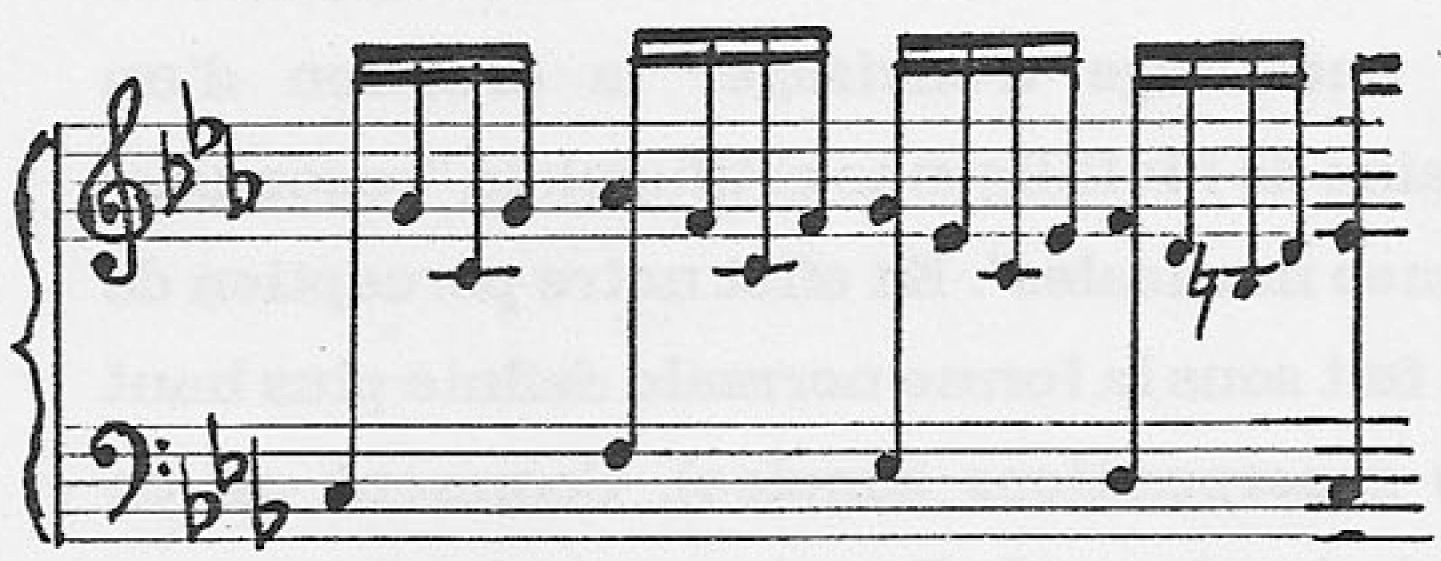


Mais on peut en isoler d'autres:

superposition de deux mouvements melodiques conjoints descendants ou bien:

la meme chose mais avec des rithmes syncopes

En meme temps on trouvera:



concatenation d'accords figures (T, S, T, D, T)

En superposant cette derniere structure avec une concatenation de broderies inferieures:

on obtient de nouveau la forme normale qui est une concatenation de structures simples (un son ou un ensemble de sons avec une duree positive), qui peut s'ecrire sous la forme d'une superposition de deux concatenations comme elle apparait dans l'exemple original.

Ce n'est pas du tout un inconvenient que de trouver

Conclusions

Bien sur une telle methode d'analyse peut s'appliquer a d'autres niveaux de la forme musicale. Il reste a definir les elements du vocabulaire de categories "V" et les regles generatives de construction des structures correspondantes a chaque element de "V".

Une bonne formalisation mathematique est necessaire si on veut implementer notre model sur ordinateur. L'utilisation des ordinateurs est importante non pour remplacer l'homme mais pour elucider les mecanismes de la perception et de la comprehension de la musique. Les progres faits dans le domaine de la reconnaissance des formes nous encourage d'envisager la creation d'un nouveau domaine de l'intilligence artificiel: la "reconnaissance des formes musicales". En effet notre perception de la musique se fait sous la forme normale definie plus haut (une suite de superpositions simples). Comment est ce qu'on arrive a isoler des melodies, des accords et d'autres structures plus complexes ? les musiciens, les mathematiciens, les informaticiens et les psychologues peuvent essayer ensemble de donner une reponse a cette question.

On considere que l'analyse musicale qui essaye de donner une "solution" unique n'est pas la bonne analyse. Ce n'est pas du tout un inconvenient que de trouver plusieurs segmentation possible ou d'obtenir des structures isolees ambigues. C'est peut etre dans l'equilibre de la clarte et de l'ambiguite que reside le secret de la creation musicale.

Bibliographie

Chomsky, N et Miller, G. A. - *L'analyse formelle des langues naturelles*, Gauthier-Villars, 1968.

Gross, M. et Lentin A. - *Notions sur les grammaires formelles*, Gauthier-Villars, 1967.