Dan Timis

UNE GRAMMAIRE GENERATIVE POUR LA MUSIQUE

These pour le

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

option Musique et Scenographie

UNE GRAMMAIRE GENERATIVE POUR LA MUSIQUE

Les grammaires generatives

La theorie des gramaires generatives et transformationelles a reusi a donner une bonne reponse a la question suivante: comment les etres umains sont capable de construire et comprendre un nombre infini de phrases tout en ayant a la disposition un ensemble fini de mots.

L'analyse des phrases d'un langage naturel peut nous montrer la structure profonde des productions verbales. Par exemple si on analyse la phrase suivante:

le garcon cueille un abricot

on arrive a isoler le syntagme nominal "le garcon" et le syntagme verbale "cueille un abricot". Le syntagme nominal est former d'un article est d'un nom. Alors "un abricot" est aussi un syntagme nominal. Le syntagme verbale est forme d'un verbe et d'un syntagme nominal. Les mots et les syntagmes sont les constituants de la phrase. Si on note:

P = phrase

Sn = syntagme nominal

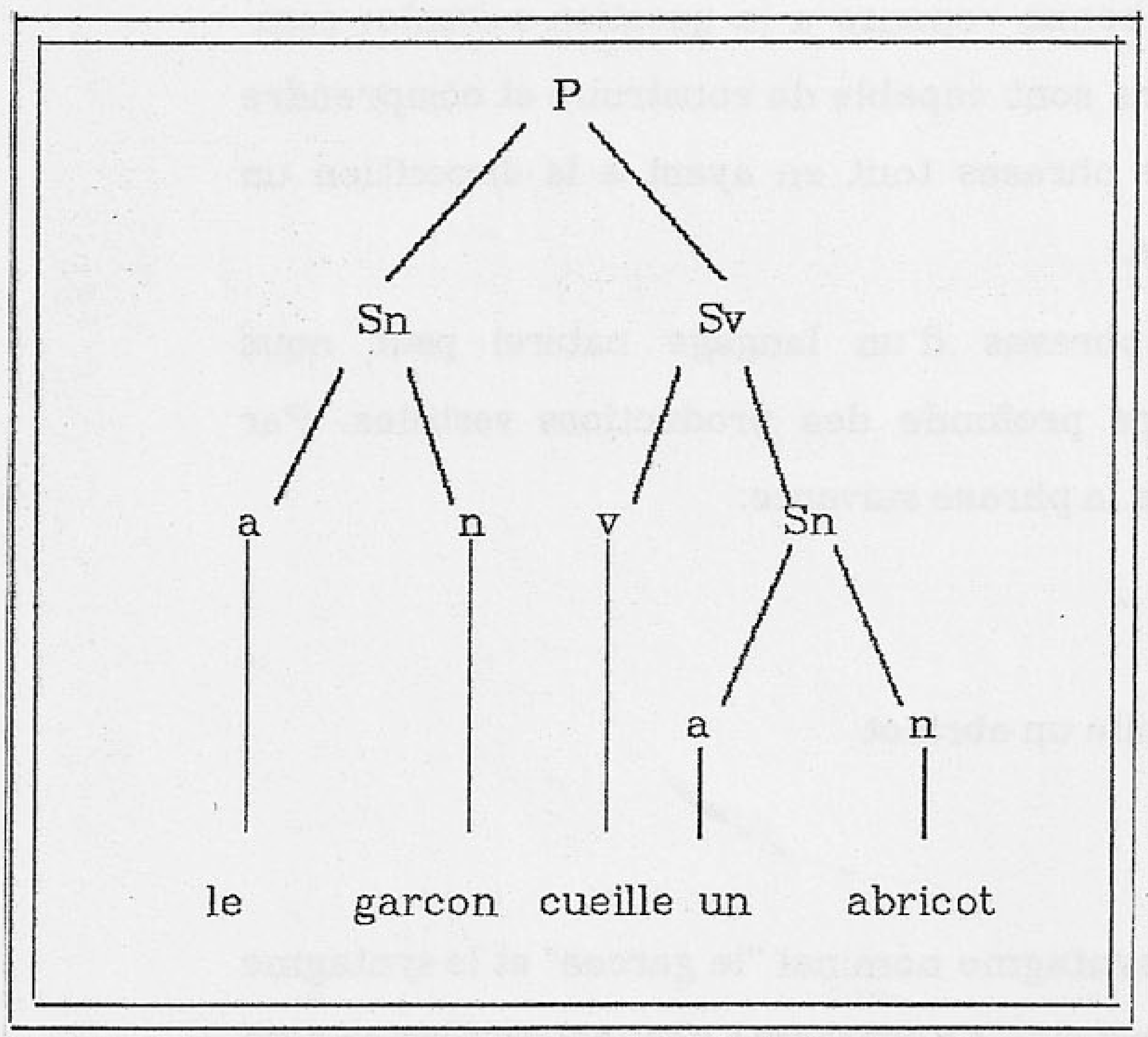
Sv = syntagme verbale

a = article

n = nom

v = verbe

on poura representer la phrase et ses constituants sous la forme d'un arbre:



Les constituant sont des suites de mots apartenant a un sous-arbre. Par exemple: "garcon", "cueille un abricot" sont des constituants mais pas "cueille un".

Maintenant on peut ajouter a l'ensemble fini des mots un ensemble fini des constituants et un ensemble fini de regles de production. Les regles de production capable de produire la phrase donner comme exemple pouront etre par exemple les suivantes:

P -> Sn Sv

Sn -> a n

Sv -> v Sn

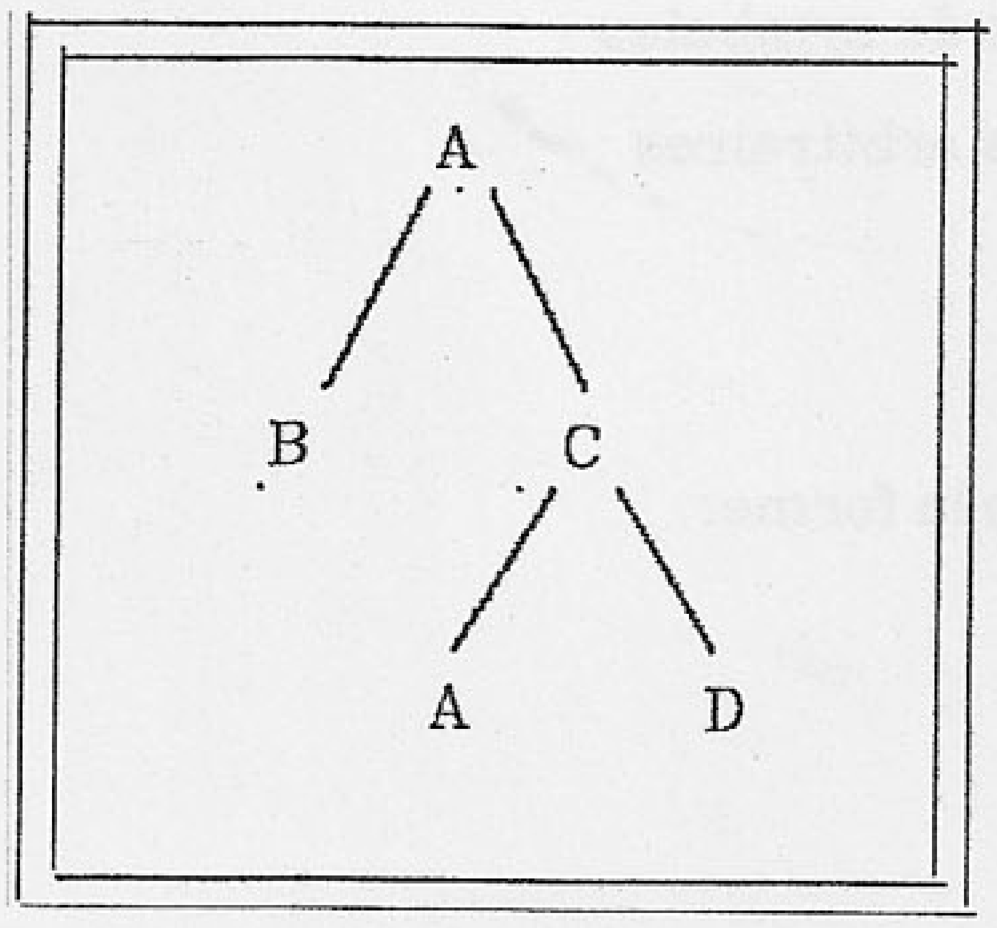
a -> un, le, la etc.

n -> garcon, abricot, maison etc.

v -> cueille, mange etc.

On va nommer l'ensemble de mots Vocabulaire terminal (notee Vt) et l'ensemble des syntagmes Vocabulaire non-terminal (notee Vn). Les elements du Vn sont nommer symboles.

L'arbre de la production d'une phrase sera son indicateur syntagmatique. Si dans un sous-arbre un trouve comme noeud le meme syntagme que la racine du sous-arbre on dit que le symbole respectif est recursif. Par emexple dans l'arbre:



le symbole A est recursif.

La recursivite nous permet de produire des phrases infiniment longues. Bien sur dans les langues naturels il y a une limite psychologique de la longueur.

Les grammaires formelles

Une grammaire formelle est une description structurelle d'un langage formel. On doit pouvoir etre capable avec un ensemble de regles de generer toutes les phrases du langage et de decider si une certaine phrase appartient ou non au langage. On va definir notre grammaire comme etant un quatruple {Vn, Vt, s0, R} avec Vn = vocabulaire non terminal ou ensemble des symboles, Vt = vocabulaire terminal ou ensemble des mots, s0 = symbole initial (c'est un element du vocabulaire non-terminal qui est la racine de toutes les arbre de derivation),et R = l'ensemble des regles.

On va utiliser la notation suivantes:

a, b, c... - les elements de Vt
A, B, C... - les symboles
e - le symbole vide (avec la propriete ae = ea = a)
w, v, u... - des suites d'elements termineaux
W, V, U... - des suites de symboles
m, n, p... - des suites arbitraires

Une regle sera alors de la forme:

m A n -> m p n

avec la signification suivante: un symbole dans un certain context (preceder et suivi des suites arbitraires) peut se reecrire sous la forme d'une suite arbitraire. Seulement les symboles peuvent se

Seulement les suites contenent des symboles peuvent se reecrire et si en partant du symbole initial apres une serie d'application de regles on arrive a une suite contenant seulement des elements non-termineaux, alors on a obtenu une phrase du langage engendre par notre grammaire.

Par exemple la grammaire:

Vt = {a, b}

Vn = {s0}

R = { s0 -> as0b, s0 -> e}

va engendrer le langage L suivant:

L = { e, ab, aabb, aaabbb etc.}

Noam Chomsky a etabli la hierarchie suivante des grammaires:

Les grammaires de type 0 sont les grammaires sans aucune restriction dans leur regles de production.

Les grammaires de type 1 contient seulement des regle du type:

m A n -> m p n

avec p differant de e et la condition que A n'aparrait pas dans la partie droite d'une regle de derivation. Ce sont les grammaires sensibles au contexte.

Les grammaires de type 2 ont des regles de la forme:

A -> p

on les apelle grammaires independentes de contexte

Les grammaires de type 3 ont des regles de la forme:

A -> wB ou A -> w

Grammaires de constituant

Dans *L'Analyse formelle des langues naturelles*, Chomsky et Miller parlent d'une classe de grammaires appeles grammaires de constituants ou grammaires syntagmatique. Ces grammaires sont censee de pouvoir fournir pour toutes les productions l'indicateur syntagmatique.

On doit alors introduire quelques restrictions. Le principe et qu'on veut reecrire chaque fois un seul symbole pour que celui-ci puisse etre la racine d'un sous-arbre de l'indicateur syntagmatique. Alors les regles seront de la forme:

A -> p ou m A n -> m p n

cet a dire soit dependantes soit independentes du contexte.

Pour pouvoir obtenir des arbres binaires on poura introduire la restriction suivante: un symbole se reecrit dans deux autres symboles ou dans un terminal. On aura alors des regles comme:

A -> B C ou m A n -> m B C n ou

A -> a ou m A n -> m a n

Dans l'etude des langues naturelles a l'aide des grammaires generatives on peut distinguer plusieurs directions. L'une consiste par exemple a trouver une methode pour deduire a partir d'un certain nombre de phrases consideres comme corectes dans un certain langage, une gramaire generative capable d'engendrer le langage respectif (bottom-up). Une autre consisterai a formuler un ensemble de regles et puis de verifier si la grammaire ainsi definie engendre un certain langage (top-down).

Dans le domain des applications musicales des grammaires generatives on a essaye d'appliquer les duex methodes. Nicholas Ruwet apres avoir utiliser des procedees d'analyse basees sur la segmentation prefere (Theorie et Methodes dans les Etudes Musicales, *Musique en jeu*, 1975) a commencer par un model theorique et puis a verifier sa validite. Par contre, J. -J. Nattiez *(Fondements d'une Semiologie de la musique*, 1975) adopte la methode bottom-up.

Nous considerons que la methode top-down est beaucoup plus interesante. On n'a pas encore des dethodes pour decouvrir des thories; par contre on a des methodes pour les verifier.

On poura signaler quelques aspects de cette methode. Une possibilite sera la verification de la capacite d'une grammaire donne d'engendrer une oeuvre ou un ensemble d'oeuvres.

Decider si une grammaire est bonne est une chose delicate. Il ne suffit pas que la grammaire soit capable de generer l'oeuvre ou l'ensemble d'oeuvres. On peut concevoir des grammaires assez generales pour etre capable d'engendrer presque n'importe quoi. Le model propose doit etre en meme temps le plus restrictifs possible. On doit montrer non seulement que notre grammaire engendre le langage mais aussi que les phrases qui n'appartient pas au langage, ne peuvent pas etre generer par la grammaire.

On poura alors admettre meme des grammaires qui ne peuvent pas generer exactement notre oeuvre mais quelque chose qui rasemble beaucoup a celle-ci. Chomsky parle de gramaticalite comme mesure de la capacite d'une phrase a rassembler aux productions ou aux non-productions d'une grammaire.

Une autre direction a suivre c'est l'etude, a partir d'une grammaire donne, des divers derivations possibles pour generer une oeuvre. On parle de grammaire ambigue alors qu'on a plusieurs derivations, voir plusieurs indicateurs syntagmatiques, conduisant a la meme derivation.

On pourra aussi etudier d'autres variantes de derivation pour essayer de comprendre les raisons qui ont mener le compositeur a choisir une derivation et pas une autre. Avant de passer a l'application d'un prcedee ou d'un autre il est necessaire de definir une classe de grammaires capables d'engendrer mieux les productions musicales.

Les grammaires relationalles

Comancons par un exemple. On essaye de definir les regles gramaticales capable de generer des suites de sons ascendentes et descendantes chromatiques. Dans l'ensemble des hauteurs on va etablir des classe d'equivalence pour les sons entre lesqueles il y a un intervalle d'octave ou plusiers octaves. Si on atribue a chaque note un numero comme sui

do = 0

do# = 1

re = 2

et ainsi de suite jusqu'a

si = 11

pour toutes les notes indifferament de l'octave

on est alors dans l'ensemble des classes de restes modulo 12.

Notre vocabulaire terminal sera alors:

{do, do#, re, ... si} ou bien

{0, 1, 2, ... 11}

On va choisir le vocabulaire non-terminal suivant:

{s0, A}

avec s0 = le symbole initial ou l'axiome et A le symbole de ranlongement

Voila maintenant les regles:

s0 -> do A

s0 -> do# A

s0 -> re A

et ainsi de suite jusqu'a

s0 -> si A

pour pouvoir commencer avec n'importe quel son

do A -> do do# A

do A -> do si A

do# A -> do# re A

do# A -> do# do A

et ainsi de suite jusqu'a

si A -> si do A si A -> si la# A

pour pouvoir monter ou descendre d'un demi-ton

et finalement

A -> e

pour pouvoir s'arreter.

de demi-ton ascendant ou descendant et pas les notes. Dans le groupe des classe de restes modulo 12 l'interval est la difference modulo 12 entre deux hauteurs. Un demi-ton ascendant sera 1 est un demi-ton descendent sera -1 ou bien 11. L'operation d'addition dans le groupe modulo 12 peut s'ecrire sous forme d'un tableau:

+	0	1	2	3	4	5	6	7	8	9	10	11
0	0	1	2	3	4	5	6	7	8	9	10	11
1	1	2	3	4	5	6	7	8	9	10	11	0
2	2	3	4	5	6	7	8	9	10	11	0	1
3	3	4	5	6	7	8	9	10	11	0	1	2
4	4	5	6	7	8	9	10	11	0	1	2	3
5	5	6	7	8	9	10	11	0	1	2	3	4
6	6	7	8	9	10	11	0	1	2	3	4	5
7	7	8	9	10	11	0	1	2	3	4	5	6
8	8	9	10	11	0	1	2	3	4	5	6	7
9	9	10	11	0	1	2	3	4	5	6	7	8
10	10	11	0	1	2	3	4	5	6	7	8	9
11	11	0	1	2	3	4	5	6	7	8	9	10

La colonne 1 represente le demi-ton ascendent et la colonne 11 le demi-ton descendant. Dans le cas des noms des notes on poura noter le demi-ton ascentant avec +1 ou simplement + et le demi-ton descendant avec -1 ou simplement - . On aura alors le tableau suivant:

	+	-
do	do#	si
do#	re	do
re	re#	do#
re#	mi	re
mi	fa	re#
fa	fa#	mi

	+	-
fa#	sol	fa
sol	sol#	fa#
sol#	la	sol
la	la#	sol#
la#	si	la
si	do	la#

L'utilisation du tableau se fera de la maniere suivante, une suite comme: "do +" sera remplace par "do do#" et la suite "mi -" sera remplace par "mi re#". Pour pouvoir reduire le nombre de regles dans notre grammaire il faut introduire un symbole nouveau. On va l'appeler symbole de reference et va le noter sr. Notre grammaire va avoir le meme vocabulaire terminal. Le vocabulaire non-terminal sera:

{s0, sr, +, -, A}

et les regles seront:

s0 -> sr A

A -> + A

A -> - A

A -> e

apres un certain nombre de derivations on peut arriver a une suite qui commence par sr suivi d'un certain nombre de + et de - . Il suffit maintenant de remplacer sr par n'importe quel element du vocabulaire terminal:

sr -> do

sr -> do#

sr -> re

et ainsi de suite jusqu'a

sr -> si

Maintenant on aplique les remplacement definis plus haut, a savoir : un plus ou un moins preceder par un terminal sera remplace par le terminal correspondant. Voici un exemple de derivation:

s0 -> sr A -> sr + A -> sr + + A -> sr + + - A ->

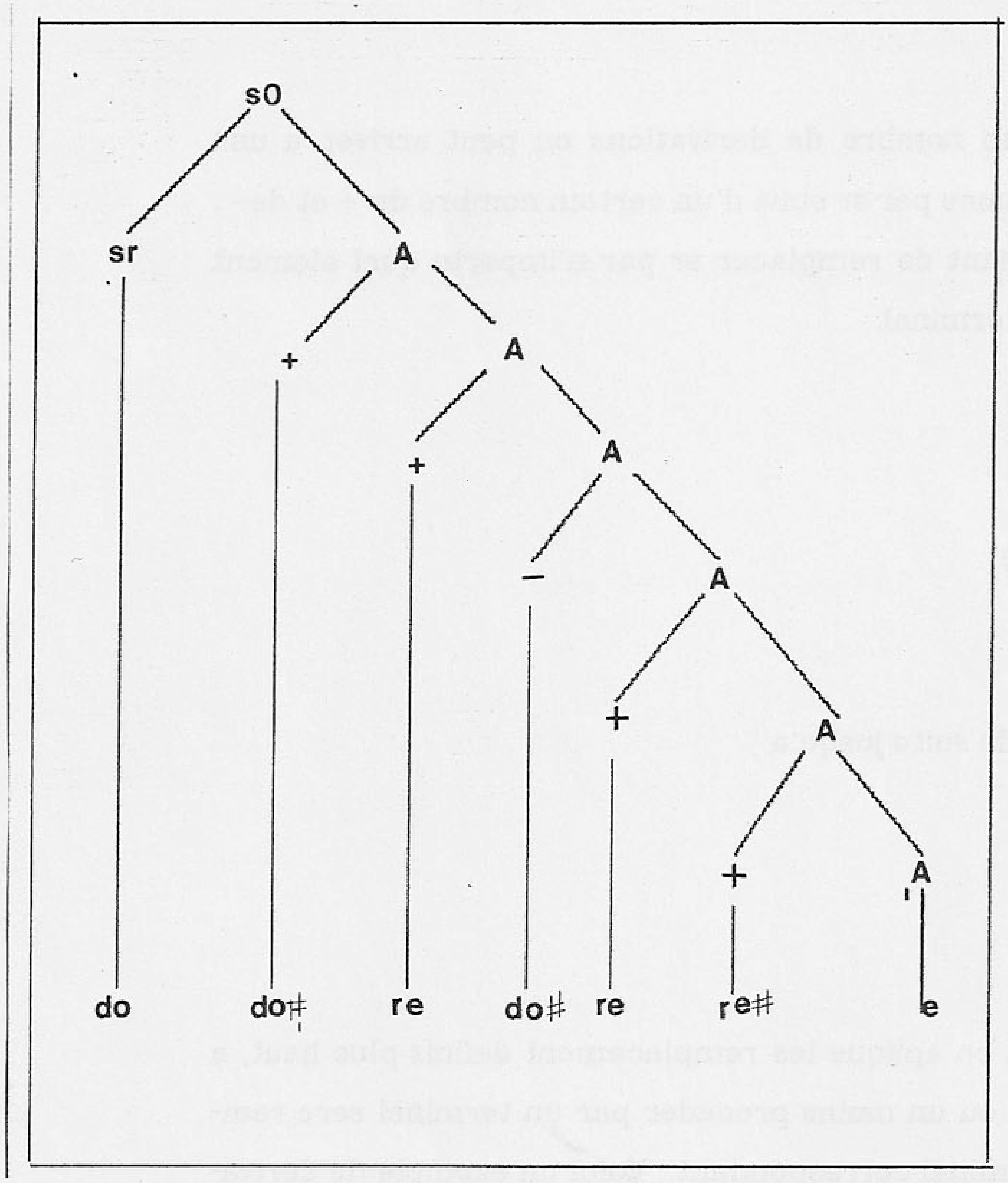
sr + + - + A -> sr + + - + + A -> sr + + - + + ->

do + + - + + -> do do# + - + + -> do do# re - + + ->

do do# re do# + + -> do do# re do# re + ->

do do# re do# re re#

L'arbre syntagmatique sera :



Les grammaires qui ont des regles qui s'applique toujours a droite s'appele une grammaire reguliere droite. On constate que notre grammaire se comporte dans un premier temps comme une grammaire reguliere droite puis comme une grammaire reguliere gauche. On poura separer deux etape. Dans la premiere on genere une suite contenant les symboles sr, + et -, dans la deuxieme on remplace ces symboles par des terminaux.

Alors on peut diviser nos symboles en deux. D'une part on aura le symbole initial et des symboles qui nous permetent de ralonger la chaine, d'autre part on aura le symbole de reference et des symboles qu'on va appeler symboles relationels.

Toute derivation aura lieu dans deux etapes. La premiere consiste a la generation d'une suite de symboles relationels, symbole de reference inclus, la seconde consiste a remplacer le symbole de reference et ensuite les symboles relationels en terminaux.

Bien sur on poura melanger les etapes. On arrive au meme resultat final avec la derivation suivante:

s0 -> sr A -> sr + A -> do + A -> do + + A ->
do do# + A -> do do# + - A -> do do# + - + A ->
do do# re - + A -> do do# re do# + A ->
do do# re do# + + A -> do do# re do# + + ->
do do# re do# re + -> do do# re do# re re#

On poura faire notre grammaire totalement reguliere droite en changeant la premiere regle:

s0 -> A sr

Alors on va arriver a une suite de symboles relationelles avec le symbole de reference a la fin:

+ + - + + sr
dans notre exemple

On est amener a bien definir alors si la suite "+ do" se traduit en "si do" ou en "do# do". C'est une question de convention. De toute facon notre grammaire va engendrer le meme langage.

On peut imaginer aussi une grammaire reguliere gauche en employant les regles suivantes:

s0 -> sr A

A -> A +

A -> A -

A -> e

L'important est qu'on s'assure de ne pas avoir plus d'un symbole de reference. Cului ci peut se trouver soit en debut de notre derivation, soit a la fin, comme on a montrer jusqu'ici mais aussi il est possible qu'il se trouve au milieu comme dans l'exemple suivant:

s0 -> A sr A

A -> A +

et

A -> A -

ou

A -> + A

et

A -> - A

et finalement

A -> e

Caracteristiques des grammaires relationelles

Une grammaires gemerative relationelle a trois vocabulaires:

Vn ou Vocabulaires non-terminal ave . un element special le symbole initial note s0 . Toutes les derivation commence par ce symbole. On va noter les elements de Vn: A, B, C etc.

Vr ou Vocabulaire relationel avec un element special le symbole de reference note sr. On va noter dans le cas d'une grammaire qui genere des suites d'hauteur dans l'ensemble des restes modulo 12: 0, +1, -1, +2, -2 etc. ou plus generalement +I. Dans le cas de durees on poura avoir: 1, x2, x1/2 etc. ou plus generalement xI. Le signe * va designer une operation (addition, multiplication etc.), alors on dva noter dans le cas general un symbole relationel par *i, *j etc. et une suites de symboles relationels par *I, *J.

Vt le Vocabulaire terminal. Dans notre cas on aura des element comme: do do# etc. ou
On va noter les termineaux a, b, c etc.

Les regles de production seront diviser en deux categorie:

1) **regles generatives** qui seront de la forme:

n p m -> n q m

avec n, q, et m des suites contenant des element non-termineaux a l'exception de s0 et/ou des elements relationels a l'exception de sr (elles peuvent etre aussi vides). On s'assure ainsi de l'unicite du symbole de reference.

n p m -> n q m

avec n, q, et m des suites contenant des elements non-termineaux et des symboles relationels et avec p une suite contenant au moins un element non-terminal et eventuellement des elements relationels (a l'exception, bien sur de s0 et de sr). Une suite d'elements relationels issu d'une suite de derivations a partir du symbole initial sera appele une production intermediaire ou la structure de la phrase.

2) **remplacements relationels** qui seront de la forme:

sr -> a

ou

a *i -> a b

ou

*i a -> b a

Touts les remplacements relationels vont commencer par le symbole de reference. Si celui-ci se trouve au debut ou a la fin de la production intermediaire les remplacements vont se produire dans un seul sens (vers la droite si il au debut, vers la gauche si il est a la fin). Si le symbole de reference se trouve au milieu de la structure alors on poura faire les remplacements dans les deux sens indiferemant.

Une fois le symbole de reference remplace la production est totalement determine. Les productions intermediaires determinent des classes d'equivalence sur l'ensemble des productions terminalles.

Les regles generatives peuvent determiner le type de grammaire dans le meme sens que pour les gramaires generatives. Il suffit de penser que le vocabulaire relationel est le vocabulaire terminal.

Les relations doivent etre definis sur tout l'ensemble des elements termineaux. Autrement, on risque de ne pas pouvoir aboutir a une production finale.

Une notation condense

Dans notre exemple on n'arrive pas a reduire beaucoup le nombre de regles. Si au lieu d'une enumeration explicite des remplacements relationels on utilise les tableaux ou on emploie une operation connue, on arrive quand meme a une simplification de notre grammaire. Important est le fait qu'on exprime d'une facon tres claire la structure de la production. La concretisation de la structure n'est alors qu'une chose subsidiaire.

On poura exprimer la premiere variante de grammaire donnee comme exemple, d'une facon plus condense. Pour ca on represente les elements termineaux sous forme d'un symbole indice. Par exemple dans le cas des hauteurs temperes on poura dire "ai" (a indice i) avec i un numero entier de 0 a 11. Les regles seront:

s0 -> ai A

ai A -> ai ai+1 A

ai A -> ai ai-1 A

A -> e

avec i+1 et i-1 l'incrementation et respectivement la decrementation modulo 12.

Voici le meme exemple de production qu'avant:

s0 -> do A -> do do# A -> do do# re A ->

do do# re do# A -> do do# re do# re A ->

do do# re do# re re# A -> do do# re do# re re#

On a absolument besoin dans ce cas-la du symbole A, autrement avec la grammaire:

s0 -> ai

ai -> ai ai+1

ai -> ai ai-1

on arrive a des productions comme:

s0 -> do -> do do# -> do do# re ->

do si do# re -> do si la# do# re etc.

Bien sur ce dernier exemple est correct, mais il engendre un autre langage. Le langage de notre premier exemple est un sous-ensemble du langage qu'engendre la derniere grammaire donnee comme exemple. On peut remarquer le fait que ce dernier exemple est assez restrictif, il ne peut pas produire des suites comme "do mi sol" ou "do re mi fa#".

Bien sur cette classe de grammaires peut s'appliquer a beaucoup de parametres musicaux (hauteur des sons, duree, fonction harmonique, profile melodique, densite de sons, dynamique etc.). On poura imaginer qu'a la base d'une oeuvre musicale il y a plusieurs grammaires correspondant aux divers parametres musicaux, independement l'une de l'autre. L'acte de la composition sera alors une recherche a l'interieur de la matiere sonore du lieu de rencontre des divers productions relatives aux divers parametres dans le concret des sons musicaux.

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.

Lerdahl, F. et Jackendoff R. - *A Generative Theory of Tonal Music*, The MIT Press, 1983.

Nattiez, J.-J. - *Fondements d'une Semiologie de la Musique*, Union Generale d'Editions, 1975.

Ruwet, N. - *Langage*, *Musique*, *Poesie*, Seuil, 1972.

Ruwet, N. - *Theorie et Methodes dans les Etudes Musicales*, dans Musique en Jeu, No. 17, 1975.