RAIRO. INFORMATIQUE THÉORIQUE ET APPLICATIONS

MARC CHEMILLIER

# Monoïde libre et musique : deuxième partie

*RAIRO. Informatique théorique et applications*, tome 21, nº 4 (1987), p. 379-417

<http://www.numdam.org/item?id=ITA_1987__21_4_379_0>

Informatique théorique et Applications/Theoretical Informatics and Applications
(vol. 21, n° 4, 1987, p. 379 à 418)

# MONOÏDE LIBRE ET MUSIQUE : DEUXIÈME PARTIE *

par Marc CHEMILLIER [1]

Communiqué par J.-E. PIN

Résumé. – *Il s'agit de la deuxième partie d'un article consacré aux possibilités d'application à la combinatoire musicale, de méthodes issues de l'informatique théorique ou des mathématiques, faisant intervenir des automates, des arbres, des structures algébriques, etc. (Nous renvoyons à la première partie :* Les musiciens ont-ils besoin des mathématiques ? in *R.A.I.R.O., vol.* 21, *n°* 3). *On définit tout d'abord un cadre algébrique (partie A), puis on expose à l'intérieur de ce cadre, certains algorithmes dont l'un en particulier concerne la reconnaissabilité par automate fini de la superposition de séquences musicales (partie B, section* 4).

Abstract. – *This is the second part of an article which deals with the possible use of mathematical or informatical methods in musical combinatoric, using automatons, trees, algebraic structures,... (See first part:* Do musicians need mathematics? in *R.A.I.R.O., Vol.* 21, *No.* 3). *In this part, we propose first an algebraic structure (part A); then we expose some algorithms, one of them corresponding to the construction of an automaton recognizing a polyphony made of* regular *melodies (part B, section* 4).

## A. PROPRIÉTÉS ALGÉBRIQUES

### 1. LOIS DE COMPOSITION SUR $A^*$. TREILLIS. ALGÈBRE DE BOOLE. SOLFÈGE.

Soit $E$ un ensemble fini d'*événements*, $A=\mathscr{P}(E)$, ensemble des parties de $E$, sera considéré comme un *alphabet*.

On considère le monoïde libre $A^*$, muni de la concaténation usuelle, qui est l'ensemble des *séquences musicales*.

(*) Reçu octobre 1986, révisé janvier 1987.

[1] L.I.T.P., Université Paris-VII, 4, place Jussieu, 75005 Paris.


0296-1598 87/04 379 40/$ 6.00/© Gauthier-Villars

**Définition d'une loi de composition sur $A^*$**

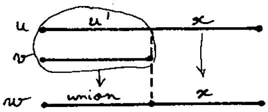
On définit sur $A^*$ une loi de composition, notée $\|$, correspondant à la *superposition* de séquences musicales.

Soit $u \| v = w$, pour $u,\ v \in A^*$, défini par :

(i) Si $u = v = 1$ (mot vide), alors $w = 1$

(ii) Si $u$ et $v$ sont de même longueur, alors chaque lettre de $w$ est l'union de la lettre correspondante dans $u$ et $v$

(iii) Si $|u| > |v|$, soit $u = u'x$, alors $w = u \| v = (u' \| v).x$, où $|u'| = |v|$



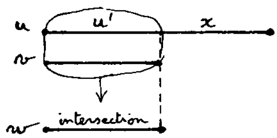

*Remarque.* — La loi $\|$ est, du point de vue formel, une sorte d'extension à $A^*$, de la loi $\cup$ sur $\mathcal{P}(E)$. Il est alors naturel, pour des raisons de symétrie de la construction, d'introduire de la même façon l'extension de la loi $\cap$ de $\mathcal{P}(E)$. On définit donc sur $A^*$ une seconde loi, notée $\perp$.

Soit $u \perp v = w$, pour $u,\ v \in A^*$, défini par :

(i) Si $u = v = 1$, alors $w = 1$

(ii) Si $|u| = |v|$, alors chaque lettre de $w$ est l'*intersection* des lettres correspondantes de $u$ et $v$

(iii) Si $|u| > |v|$, soit $u = u'\,x$, alors $w = u \perp v = u' \perp v$




*Exemple* : $E = \{a,\ b,\ c\}$.

Convenons de noter les éléments de $A = \mathcal{P}(E)$ sous forme verticale : $\{a,\ b\}$ s'écrit $\begin{smallmatrix}a\\b\end{smallmatrix}$

On a alors :

$$\begin{matrix}a&c&c\\b&&a\end{matrix} \| ab = \begin{matrix}a&b&c\\b&c&a\end{matrix}$$

et :

$$\begin{matrix}a&c&c\\b&&a\end{matrix} \perp ab = a\varnothing$$

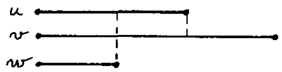
*Remarque.* — Si $|u|$ désigne la longueur d'un mot $u$, on a :

$$|u.v|=|u|+|v|$$
$$|u\|v|=\max(|u|, |v|)$$
$$|u \perp v|=\min(|u|, |v|).$$

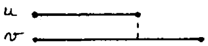
**Propriétés de calcul**

De nombreuses propriétés de $\cap$ et $\cup$ dans $\mathscr{P}(E)$ sont conservées pour $\|$ et $\perp$ dans $A^*$.

1. *idempotence* : $u\|u=u$, et $u \perp u=u$.
2. *associativité* : $u\|(v\|w)=(u\|v)\|\, w$, et $u \perp (v \perp w)=(u \perp v) \perp w$




3. *commutativité* : $u\|v=v\|u$, et $u \perp v=v \perp u$.
4. *absorption* : $u\|(u \perp v)=u$, et $u \perp (u\|v)=u$.




On démontre de plus (en étudiant tous les cas de longueurs relatives de $u$, $v$, $w$) :

5. *distributivité* : $u\|\,(v \perp w)=(u\|v) \perp (u\|w)$, et $u \perp (v\|w)=(u \perp v)\|(u \perp w)$.
6. *1 est neutre pour la superposition* : $1\|u=u\|1=u$.

*Démonstration* : On a : $1\|u=1\|\,(1.u)=(1\|1).u=1.u=u$. ■

*Remarque.* — Ce neutre est alors unique (si $\varepsilon$ était un autre neutre pour $\|$, alors on aurait : $\varepsilon=\varepsilon\|1=1$).

Certaines propriétés de $\cap$ et $\cup$ sur $\mathscr{P}(E)$ sont perdues dans $A^*$ :

PROPOSITION : *Il n'y a pas de neutre pour* $\perp$.

*Démonstration* : Supposons $e$ tel que $u \perp e=u$ pour tout $u$ appartenant à $A^*$. Pour les longueurs des séquences, il vient : $|u \perp e|=|u|$. Mais d'après une remarque faite précédemment, on a : $|u \perp e|=\min(|u|, |e|)$, ce qui donne : $|e| \geqq |u|$, pour tout $u$ de $A^*$. Donc $e$ serait un mot infini. ■

Les propriétés 1, 2, 3, 4, donnent à $A^*$ une structure de *treillis*; il est de surcroît *distributif* par 5, et admet un neutre pour $\|$ grâce à 6. Mais il n'y a pas de neutre pour $\perp$, donc ce treillis n'est pas *complémenté*.

Dans ce contexte, rappelons les propriétés de la concaténation, notée ., de $A^*$ :

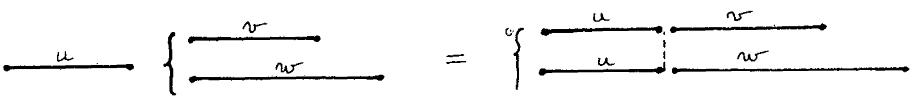
7. *associativité* de la concaténation

8. *1 est neutre pour la concaténation*

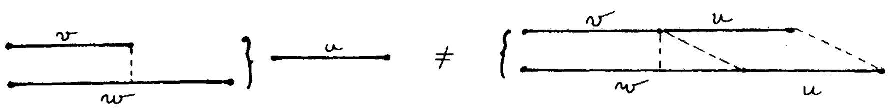
On a, de plus :

9. *distributivité* à gauche de . sur $\|$ et $\perp$ :

$$\begin{cases} u(v \| w) = uv \| uw \\ u(v \perp w) = uv \perp uw \end{cases}$$




*Attention.* – La concaténation n'est pas distributive *à droite* sur $\|$ et $\perp$ :




Signalons la propriété suivante :

PROPRIÉTÉ : *1 est* absorbant *pour* $\perp$, *c'est-à-dire :* $1 \perp u = u \perp 1 = 1$, *pour tout u.*

*Démonstration :* On a : $u \| 1 = u$, donc $u \perp 1 = (u \| 1) \perp 1 = 1$ (d'après l'*absorption,* propriété 4).

## Treillis. Algèbre de Boole. Solfège

Rappelons les définitions des structures suivantes :

DÉFINITION : Un *treillis* est un ensemble muni de deux lois $(T, \wedge, \vee)$ vérifiant :

1. *idempotence :*

$$\begin{cases} u \wedge u = u \\ u \vee u = u \end{cases}$$

2. *associativité :*

$$\begin{cases} u \wedge (v \wedge w) = (u \wedge v) \wedge w \\ u \vee (v \vee w) = (u \vee v) \vee w \end{cases}$$

3. *commutativité :*

$$\begin{cases} u \wedge v = v \wedge u \\ u \vee v = v \vee u \end{cases}$$

4. *absorption :*

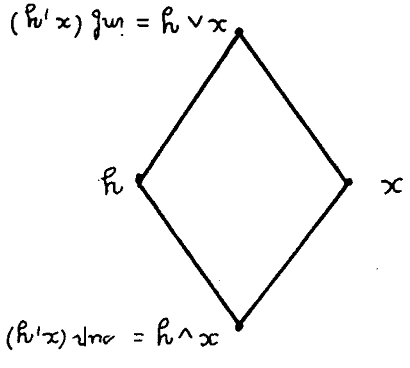
$$\begin{cases} u \wedge (u \vee v) = u \\ u \vee (u \wedge v) = u. \end{cases}$$

Un ensemble *réticulé* est un ensemble muni d'une relation d'ordre $(E, \leqq)$ tel que pour tout $x$ et $y$ de $E$, $\inf(x, y)$ et $\sup(x, y)$ existent.

THÉORÈME : *Tout ensemble réticulé est un treillis, et réciproquement. La correspondance entre les deux structures se fait canoniquement par :*

(i) *Si* $(T, \wedge, \vee)$ *est un treillis, on définit une relation d'ordre* $\leqq$ *sur* $T$ *par :* $x \leqq y$ *si et seulement si* $x \vee y = y$ *(ou bien, ce qui est équivalent :* $x \wedge y = x$)

(ii) *Si* $(E, \leqq)$ *est un ensemble réticulé, on définit* $\wedge$ *et* $\vee$ *par :* $x \wedge y = \inf(x, y)$ *et* $x \vee y = \sup(x, y)$



*Démonstration* : Treillis $\Rightarrow$ réticulé.

Posons $x \leqq y$ pour $x = x \wedge y$, on a alors : $x \vee y = (x \wedge y) \vee y = y$. La relation $\leqq$ est une relation d'ordre :

$$\text{(i)} \quad x \leqq x, \ \text{car} \ x \wedge x = x$$

$$\text{(ii)} \quad x \leqq y \ \text{et} \ y \leqq x \Rightarrow x = y, \ \text{car} \ x = x \wedge y = y \wedge x = y$$

$$\text{(iii)} \quad x \leqq y \ \text{et} \ y \leqq z \Rightarrow x \leqq z, \ \text{car} \ x \wedge z = (x \wedge y) \wedge z = x \wedge (y \wedge z) = x \wedge y = x$$

Par ailleurs, pour $x$, $y$ dans $T$, $\inf(x, y)$ existe$=x \wedge y$, car :

$$(x \wedge y) \wedge x = x \wedge y \text{ donc } x \wedge y \leqq x$$
$$(x \wedge y) \wedge y = x \wedge y \text{ donc } x \wedge y \leqq y$$

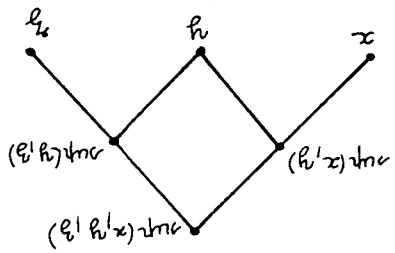
et de plus, si $m \wedge x = m \wedge y = m$ (c'est-à-dire $m \leqq x$ et $m \leqq y$), alors $m \wedge (x \wedge y) = m \wedge y = m$, donc $m \leqq x \wedge y$, donc on a bien $x \wedge y = \inf(x, y)$, et de même l'existence de $\sup(x, y) = x \vee y$.

Réticulé $\Rightarrow$ treillis.

Posons $x \wedge y = \inf(x, y)$ et $x \vee y = \sup(x, y)$

(i) *idempotence :* $\inf(x, x) = \sup(x, x) = x$

(ii) *associativité :* les majorants communs à $x$ et $\sup(y, z)$ sont ceux de $\{x, y, z\}$, qui sont ceux de $\sup(x, y)$ et $z$



(iii) *commutativité* évidente

(iv) *absorption :* $\inf(a, \sup(a, b)) = \sup(a, \inf(a, b))$, car $a \leqq \sup(a, b)$ et $a \leqq \inf(a, b)$. ■

Définition : Une *algèbre de Boole* est un treillis $(A, \vee, \wedge)$ qui est de surcroît :

5. *distributif :*

$$\begin{cases} a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c) \\ a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c) \end{cases}$$

et *complémenté :*

6. éléments *neutres* 0 et $u$ :

$$\begin{cases} a \vee 0 = a \\ a \wedge u = a \end{cases}$$

7. *compléments :*

$$\begin{cases} a \vee \bar{a} = u \\ a \wedge \bar{a} = 0. \end{cases}$$

PROPRIÉTÉ : *On démontre alors :*

— *unicité du complément pour tout a*

— *involution :* $\bar{\bar{a}} = a$

— *lois de Morgan :*

$$\begin{cases} \overline{a \wedge b} = \bar{a} \vee \bar{b} \\ \overline{a \vee b} = \bar{a} \wedge \bar{b} \end{cases}$$

AXIOMES FAIBLES : On peut même démontrer que les axiomes 1 à 7 sont redondants, et qu'il suffit pour une algèbre de Boole d'avoir :

3 (*commutativité*), 5 (*distributivité*), 6 (*éléments neutres*) et 7 (*compléments*).

DÉFINITION : Un *anneau de Boole* est obtenu à partir d'une algèbre de Boole en posant :

$$x \Delta y = (x \wedge \bar{y}) \vee (\bar{x} \wedge y)$$

On vérifie alors que $(A, \Delta, \wedge)$ a une structure d'anneau :

(i) $(A, \Delta)$ est un groupe commutatif, d'élément neutre 0, et tel que $x \Delta x = 0$, c'est-à-dire $x^{-1} = x$

(ii) $\Lambda$ est associative et distributive sur $\Delta$.

On a établi que dans $(A^*, \|, \perp)$ il était impossible que $\perp$ possède un élément neutre. Ainsi $(A^*, \|, \perp)$ n'est pas *complémenté*, et n'a donc pas une structure d'*algèbre de Boole*; *a fortiori*, ce n'est pas un *anneau de Boole*. La structure de $(A^*, \|, \perp)$ est donc intermédiaire entre celle de treillis et celle d'algèbre de Boole.

On peut maintenant renverser le problème, c'est-à-dire se donner *a priori*, sur un ensemble quelconque, trois lois de composition $\|$, $\perp$, ainsi que la concaténation ., vérifiant les propriétés 1 à 9 étudiées ci-dessus (cf. *Propriétés de calcul*). On appellera *solfège* la structure obtenue.

DÉFINITION : Un *solfège* est un ensemble muni de trois lois $(S, \|, \perp, .)$ vérifiant :

(i) $(S, \|, \perp)$ est un treillis distributif (c'est-à-dire 1, 2, 3, 4, avec 5)

(ii) $(S, .)$ est un monoïde d'élément neutre 1 (*cf.* 7 et 8)

(iii) 1 est aussi neutre pour $\|$ (*cf.* 6)

(iv) la concaténation . est distributive *à gauche* sur $\|$ et $\perp$ (*cf.* 9).

Donnons quelques propriétés immédiates à partir de la définition.

PROPRIÉTÉ : *Dans un solfège, la relation d'ordre $\leqq$ canoniquement associée à la structure de treillis, est compatible à gauche avec la concaténation :*

$$x \leqq y \quad \Rightarrow \quad zx \leqq zy$$

*Démonstration :* Cela vient de la distributivité à gauche de . sur $\|$.

$x \leqq y$ équivaut par définition à $x \| y = y$, donc $z(x \| y) = zy$, donc $zx \| zy = zy$, ce qui donne par définition $zx \leqq zy$. ■

*Remarque.* — Dans le monoïde $A^*$, où $A = \mathscr{P}(E)$, cette relation d'ordre $\leqq$ est une extension de la relation d'ordre *préfixielle* : $x$ est *préfixe* de $y$ si et seulement si : $\exists u \mid y = xu$.

On voit en effet que : $x \| y = x \| (xu) = (x.1) \| (xu) = x(1 \| u) = xu = y$, c'est-à-dire $x \leqq y$.

*Exemple* 1. **N** ensemble des entiers naturels 0, 1, 2, etc.

PROPRIÉTÉ : *(N, max, min, +) est un solfège.*

*Démonstration :* L'ordre sur **N** est total, c'est-à-dire : $\forall m, n \in \mathbf{N}$, $m \leqq n$ ou bien $n \leqq m$.

Ceci implique que $(\mathbf{N}, \leqq)$ est réticulé, car pour tout $m$, $n$ de **N**: $\inf(m, n) = m$ si $m \leqq n$, ou $\inf(m, n) = n$ si $n \leqq m$, donc existe; et de même $\sup(m, n)$ existe. Ainsi, **N** est muni canoniquement d'une structure de treillis, avec les opérations max et min associées à l'ordre $\leqq$

(i) ce treillis est distributif :

$$\begin{cases} \max(m, \min(n, p)) = \min(\max(m, n), \max(m, p)) \\ \min(m, \max(n, p)) = \max(\min(m, n), \min(m, p)) \end{cases}$$

(ii) 0 est neutre pour max : $\forall m \in \mathbf{N}$, $\max(0, m) = \max(m, 0) = m$

(iii) l'addition sur **N** est associative, et 0 est neutre pour + $(m + 0 = 0 + m = m)$

(iv) l'addition est distributive sur max et min :

$$\begin{cases} m + \max(n, p) = \max(m + n, m + p) \\ m + \min(n, p) + \min(m + n, m + p). \quad ■ \end{cases}$$

*Exemple* 2. Soit $E$ un ensemble, $\mathscr{P}(E)$ l'ensemble de ses parties. On sait que $(\mathscr{P}(E), \cup, \cap)$ est une algèbre de Boole pour l'union et l'intersection, donc c'est un treillis, qui est de plus :

(i) distributif

$$\begin{cases} X \cap (Y \cup Z) = X \cap Y) \cup (X \cap Z) \\ X \cup (Y \cap Z) = (X \cup Y) \cap (X \cup Z) \end{cases}$$

(ii) avec éléments neutres : $X \cup \varnothing = X$ (et aussi : $X \cap E = X$).

PROPRIÉTÉ : *Les lois $\cup$ et $\cap$ sont distributives* sur elles-mêmes.

*Démonstration :*

$$X \cup (Y \cup Z) = (X \cup X) \cup (Y \cup Z) = (X \cup Y) \cup (X \cup Z)$$

et de même :

$$X \cap (Y \cap Z) = (X \cap Y) \cap (X \cap Z). \quad \blacksquare$$

En prenant pour concaténation la loi $\cup$ elle-même, on obtient :

PROPRIÉTÉ : *$(\mathscr{P}(E), \cup, \cap, \cup)$ est un solfège*

THÉORÈME : *Tout treillis distributif, admettant un neutre pour l'une de ses lois, est un solfège si l'on prend cette loi comme concaténation.*

## 2. MORPHISMES DE SOLFÈGES.

Lorsque l'on introduit une structure, il est naturel d'étudier ensuite les applications qui « transportent » la structure. On peut étudier les morphismes :

– de monoïdes

– de treillis

– de solfège.

Rappelons les définitions et résultats suivants :

DÉFINITION : Un *morphisme* $\varphi$ du monoïde $M$ dans le monoïde $M'$ :

$$\begin{cases} \varphi(xy) = \varphi(x)\,\varphi(y) \\ \quad \varphi(1_M) = 1_{M'} \end{cases}$$

*Remarque.* – Dès qu'on a $\varphi(xy) = \varphi(x)\,\varphi(y)$, on voit que $\varphi(1)$ est *neutre dans* $\varphi(M)$ qui est inclus dans $M'$, mais cela n'implique pas que $\varphi(1)$ soit *neutre dans $M'$*. Par contre, si $\varphi$ est *surjective*, alors $\varphi(M)=M'$, donc $\varphi(1)$ est neutre dans $M'$, donc : $\varphi(1)=1_{M'}$.

THÉORÈME : *Pour toute application $\theta$ de $A$ alphabet dans un monoïde $M$, il existe un unique morphisme $\varphi$ du monoïde $A^*$ dans $M$, qui prolonge $\theta$.*

On a donc :

$$\theta=\varphi\,|_A \quad (\varphi \text{ restreint à } A)$$
$$\varphi(xy)=\varphi(x)\,\varphi(y)$$
$$\varphi(1)=1_M \quad (\text{où } 1 \text{ est le mot vide de } A^*)$$

$$\begin{tikzcd} A \arrow[r, "\text{injection}"] \arrow[dr, "\theta=\varphi|_A"'] & A^* \arrow[d, "\varphi"] \\ & M \end{tikzcd}$$

Soit $E$ un ensemble d'événements, $A=\mathscr{P}(E)$ l'alphabet, et soit $(S, \|, \perp, .)$ un solfège quelconque. On peut se demander si, à partir d'une application de $E$ dans $S$, il est possible de construire, de façon unique, un morphisme de $A^*$ dans $S$, tel que :

$$f(1)=1$$
$$f(xy)=f(x)\,f(y)$$
$$f(x\,\|\,y)=f(x)\,\|\,f(y)$$

La construction n'est en général pas possible; le problème vient de ce qu'un élément d'un solfège peut se décomposer de plusieurs façons par . et $\|$.

*Exemple :* $E=\{a, b\}$, et soit $(\mathbf{N}, \max, \min, +)$ le solfège déja étudié. On considère l'application de $E$ dans $\mathbf{N}$, définie par : $a\to 1$ et $b\to 2$.

Soit alors :

$$u=\begin{array}{cc} a & a \\ b & b \end{array}$$

On a : $u=aa\,\|\,bb=ab\,\|\,ba$.

Il vient :

$$f(u)=f(aa\,\|\,bb)=\max(f(aa), f(bb))=\max(f(a)+f(a), f(b)+f(b))=\max(2,4)=4$$

et, par ailleurs :

$$f(u)=f(ab\,\|\,ba)=\max(f(ab), f(ba))=\max(f(a)+f(b), f(b)+f(a))=f(a)+f(b)=$$

ce qui est absurde.

Restreignons le point de vue en considérant deux ensembles d'événements $E$ et $F$, et les solfèges associés $A^*$ et $B^*$. Étudions les prolongements éventuels d'une application de $E$ dans $B^*$, en morphisme tel que :

$$f(1)=1$$
$$f(xy)=f(x)f(y)$$
$$f(x\|y)=f(x)\|f(y).$$

Supposons qu'un tel prolongement existe. Remarquons alors que la situation : $|f(a)|\neq|f(b)|$ est impossible, avec $a,\ b\in E$. En effet, prenons par exemple $|f(a)|>|f(b)|$. On prend comme dans l'exemple ci-dessus

$$u=\begin{matrix}a & a\\ b & b\end{matrix},\ \text{ce qui donne : } u=aa\|bb=ab\|ba.$$

Il vient :

$$f(u)=f(aa\|bb)=f(aa)\|f(bb)=f(a)f(a)\|f(b)f(b),$$

ce qui conduit à :

$$|f(u)|=2\times|f(a)|,$$

alors que :

$$f(ab\|ba)=f(ab)\|f(ba)=f(a)f(b)\|f(b)f(a),$$

d'où :

$$|f(u)|=|f(a)|+|f(b)|<2\times|f(a)|,$$

ce qui est absurde.

On voit donc qu'on a nécessairement : $\forall a,\ b\in E,\ |f(a)|=|f(b)|$.

Il en résulte : $\forall u,\ v\in A^*,\ |u|=|v|\Rightarrow|f(u)|=|f(v)|$. Ceci se voit par récurrence sur $|u|$, compte tenu de :

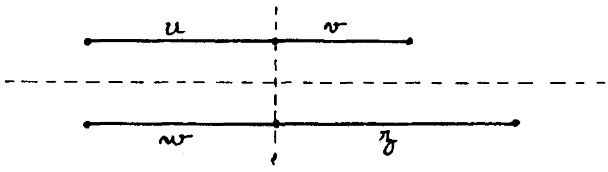
$$|f(u)|=|f(u'x)|=|f(u')|+|f(x)|=|f(v')|+|f(y)|=|f(v)|.$$

Remarquons enfin que $\varnothing\in A=\mathscr{P}(E)$, donc $|f(\varnothing)|$ est égale à $|f(a)|$ pour tout $a$ dans $E$. Appelons $n$ cette longueur commune. Prenons par exemple $E=\{a,\ b\}$ et $F=\{x,\ y\}$. Supposons $f(a)=x,\ f(b)=\begin{matrix}x\\y\end{matrix}$. On a alors $f(a)\cap f(b)=\{x\}\neq\varnothing$, et dès lors, il y a deux possibilités pour

$f(\varnothing)$ : $f(\varnothing)=x$ ou $f(x)=\varnothing$. Si l'on veut l'unicité de $f$, il faut : $\underset{b\in E}{\perp} f(b)=\varnothing\varnothing\ldots\varnothing$ (avec $n$ occurrences, si $n$ est la longueur des $f(a)$, $a\in E$). Dès lors : $f(\varnothing)=\varnothing\varnothing\ldots\varnothing$ ($n$ occurrences).

Avant de donner une réciproque des conditions obtenues, donnons une propriété de calcul :

PROPRIÉTÉ : *Si l'on a* $|u|=|w|$, *alors :* $uv\|wz=(u\|w)(v\|z)$




DÉFINITION : Un morphisme $f$, du solfège $S$ dans le solfège $S'$ est *additif* et *multiplicatif* si :

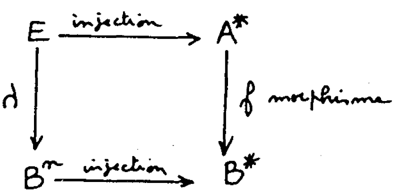
$$f(xy)=f(x)f(y)$$
$$f(x\|y)=f(x)\|f(y)$$
$$f(1)=1$$

THÉORÈME : *Pour toute application* $\lambda$ *de* $E$ *dans le solfège* $B^*$, *où* $B=\mathscr{P}(F)$, *il existe un unique morphisme additif et multiplicatif* $f$, *du solfège* $A^*$, *où* $A=\mathscr{P}(E)$, *dans le solfège* $B^*$, *prolongeant* $\lambda$, *si et seulement si :*

(i) $\forall a, b\in E, |\lambda(a)|=|\lambda(b)|$

(ii) $\underset{b\in E}{\perp} \lambda(b)=\varnothing\varnothing\ldots\varnothing$.

On a alors : $f(\varnothing)=\varnothing\varnothing\ldots\varnothing$




*Remarque.* — Dans le cas où $\lambda$ est une application *injective de E dans F*, les deux conditions (i) et (ii) sont vérifiées.

*Démonstration : C.N.* si $f$ est un morphisme de $A^*$ dans $B^*$, on a vu que nécessairement, sa restriction à $E$, $f|_E$, vérifie (i), et (ii) si l'on a unicité.

*C.S.* supposons (i) et (ii) vérifiées pour $\lambda$ de $E$ dans $B^*$ (soit $k=|\lambda(a)|$ pour $a\in E$).

On introduit une application $f$ de $A^*$ dans $B^*$ :

$$f(\varnothing)=\varnothing\varnothing\ldots\varnothing \ (k \text{ fois}),\ f(1)=1$$

$$f(x)=\mathop{\|}_{a\in x}\lambda(a) \text{ pour } x\in A=\mathscr{P}(E),\ x\neq\varnothing$$

$$f(u)=f(u_1)\,f(u_2)\ldots f(u_n) \text{ pour } u=u_1\,u_2\ldots u_n,\ u_i\in A$$

Alors $f$ est un morphisme additif et multiplicatif

*additif :* soient

$$u=vw,\ u=u_1\,u_2\ldots u_n,\ v=u_1\,u_2\ldots u_p,\ w=u_{p+1}\,u_{p+2}\ldots u_n$$

$$f(v)\,f(w)=f(u_1\ldots u_p)\,f(u_{p+1}\ldots u_n)=f(u_1)\ldots f(u_p)\,f(u_{p+1})\ldots f(u_n)=f(u)$$

*multiplicatif :* soient $u=v\|w$, $u=u_1\,u_2\ldots u_n$, $\max(|v|,|w|)=|u|$, donc supposons par exemple $|w|=|u|=n$, ce qui donne : $w=w_1\,w_2\ldots w_n$, et $v=v_1\,v_2\ldots v_p$, avec $u_i=w_i\cup v_i$ pour $i\leqq p$, et $u_i=w_i$ pour $i\geqq p+1$

$$f(v)\|f(w)=f(v_1)\ldots f(v_p)\|f(w_1)\ldots f(w_p)\,f(w_{p+1})\ldots f(w_n)$$

donc, comme d'après (i), on a : $|f(v_i)|=|f(w_i)|\,\forall i\leqq p$, on peut appliquer la propriété de calcul introduite plus haut :

$$\begin{aligned}f(v)\|f(w)&=(f(v_1)\ldots f(v_p)\|f(w_1)\ldots f(w_p))\cdot f(w_{p+1})\ldots f(w_n)\\&=[(f(v_1)\|f(w_1)\ldots(f(v_p)\|f(w_p)]\cdot f(w_{p+1})\ldots f(w_n)\end{aligned}$$

mais si $x$, $y\in A$, $f(x)\|f(y)=(\mathop{\|}_{a\in x}\lambda(a))\|(\mathop{\|}_{a\in y}\lambda(a))=\mathop{\|}_{a\in x\cup y}\lambda(a)=f(x\cup y)$, d'où :

$$f(v)\|f(w)=[f(v_1\cup w_1)\ldots f(v_p\cup w_p)]\cdot f(w_{p+1})\ldots f(w_n)=f(u)$$

*unicité :* si un autre morphisme $h$ additif et multiplicatif vérifie $h|_E=\lambda=f|_E$, alors $h=f$. En effet, d'après (ii), on a : $h(\varnothing)=\varnothing\varnothing\ldots\varnothing=f(\varnothing)$; on a par définition d'un morphisme : $h(1)=1=f(1)$; pour tout $a\in A$, on voit par récurrence sur card $a$, que si $a=a'\cup x$ où $x\notin a'$, $h(a)=h(a')\|h(x)=f(a')\|f(x)$; pour tout $u\in A^*$, on voit par récurrence sur $|u|$, que si $u=u'x$, $x\in A$, $h(u)=h(u')\,h(x)=f(u')\,f(x)=f(u)$. ■

Avant d'étudier le comportement de la loi $\perp$, donnons une propriété de calcul analogue à celle de $\|$.

Avant d'étudier le comportement de la loi $\perp$, donnons une propriété de calcul analogue à celle de $\|$.

PROPRIÉTÉ : *Si l'on a* $|u|=|w|$, *alors :* $uv \perp wz=(u \perp w)(v \perp z)$.

Supposons que le morphisme obtenu dans les conditions du théorème précédent soit *intersectif* c'est-à-dire : $f(x \perp y)=f(x) \perp f(y)$. Il est nécessaire que l'application $\lambda$ vérifie : pour $a$, $b \in E$, avec $a \neq b$, c'est-à-dire $a \cap b=\varnothing$, $\lambda(a) \perp \lambda(b)=f(a \perp b)=f(\varnothing)=\varnothing\varnothing \ldots \varnothing$, d'où la condition sur $\lambda$:

(iii) $\forall a, b \in E, a \neq b \Rightarrow \lambda(a) \perp \lambda(b)=\varnothing\varnothing \ldots \varnothing$.

Remarquons que cette condition (iii) implique la condition (ii) : $\underset{b \in E}{\perp} \lambda(b)=\underset{a \neq b}{\perp}[\lambda(a) \perp \lambda(b)]=\varnothing\varnothing \ldots \varnothing$.

DÉFINITION : Un *morphisme de solfège* $f$ de $S$ dans $S'$ est un morphisme additif, multiplicatif, et intersectif.

THÉORÈME : *Pour toute application* $\lambda$ *de* $E$ *dans le solfège* $B^*$, *où* $B=\mathscr{P}(F)$, *il existe un unique morphisme de solfège* $f$, *du solfège* $A^*$, *où* $A=\mathscr{P}(E)$, *dans* $B^*$, *qui prolonge* $f$, *si et seulement si :*

(i) $\forall a, b \in E, |\lambda(a)|=|\lambda(b)|$

(iii) $\forall a, b \in E, a \neq b \Rightarrow \lambda(a) \perp \lambda(b)=\varnothing\varnothing \ldots \varnothing$.

*Démonstration : C.N.* les conditions (i) et (iii) sont nécessaires, comme on l'a vu plus haut.

*C.S.* supposons (i) et (iii) vérifiées pour $\lambda$ de $E$ dans $B^*$, et soit $f$ le morphisme additif et multiplicatif construit précédemment. Soient $u=v \perp w$, $u=u_1 u_2 \ldots u_p$, $v=v_1 v_2 \ldots v_p$, $w=w_1 w_2 \ldots w_n$, et avec $u_i=v_i \cap w_i$, pour tout $i \leq p$.

$$f(v) \perp f(w)=f(v_1) \ldots f(v_p) \perp f(w_1) \ldots f(w_p) f(w_{p+1}) \ldots f(w_n)$$

donc, comme d'après (i), on a : $|f(v_i)|=|f(w_i)|$ pour tout $i \leq p$, on peut appliquer la propriété de calcul citée plus haut :

$$\begin{aligned} f(v) \perp f(w) &= f(v_1) \ldots f(v_p) \perp f(w_1) \ldots f(w_p) \\ &= [f(v_1) \perp f(w_1)] \ldots [f(v_p) \perp f(w_p)] \end{aligned}$$

Or : $\forall x, y \in A$, $f(x) \perp f(y)=(\underset{a \in x}{\|} \lambda(a)) \perp (\underset{b \in y}{\|} \lambda(b))=\underset{\substack{a \in x \\ b \in y}}{\|}[\lambda(a) \perp \lambda(b)]$; mais pour $a \neq b$, on a $\lambda(a) \perp \lambda(b)=\varnothing\varnothing \ldots \varnothing$,

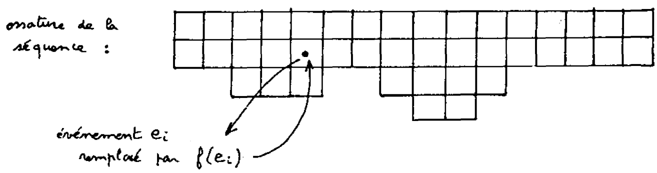
donc $f(x) \perp f(y)=\underset{a \in x \cap y}{\|}(\lambda(a) \perp \lambda(a))=f(x \cap y)$,

d'où : $f(v) \perp f(w)=f(v_1 \cap w_1) \ldots f(v_p \cap w_p)=f(u)$. ■

*Remarque.* — Si $\lambda$ est injective de $E$ dans $F$, alors (iii) est vérifiée.

**Exemples de morphismes**

*Remarque.* – Intuitivement, on voit qu'un morphisme de solfège est une application qui peut remplacer, dans une séquence musicale, des événements par d'autres événements, mais tout en conservant la forme générale de la séquence.

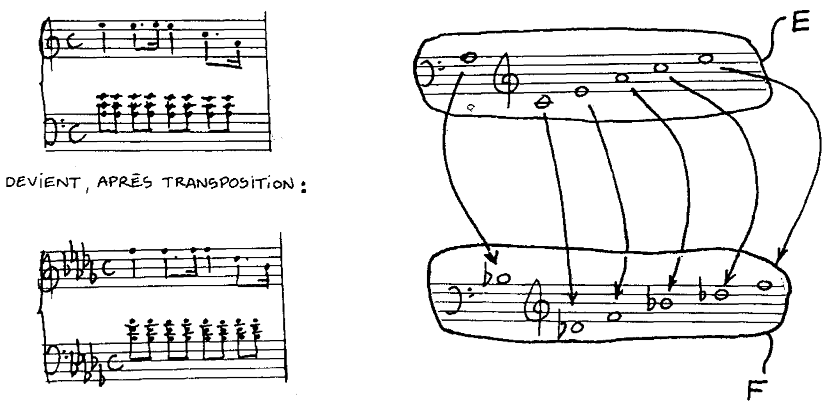



*Exemple* 1. Morphisme de transposition

La *transposition* d'une séquence musicale consiste à récrire cette séquence un intervalle plus haut ou plus bas.

PROPRIÉTÉ : *La transposition est un morphisme de solfège.*

Considérons par exemple, une transposition au demi-ton supérieur. Si $E = \{$la, do, mi$\}$, on a $F = \{$si , ré , fa$\}$. La transposition apparait alors comme le prolongement de l'application de $E$ dans $F$ définie par : la → si , do → ré , mi → fa. On est dans le cas d'une application injective de $E$ dans $F$.




*Exemple* 2. Augmentation, diminution rythmique

L'*augmentation* rythmique consiste à transformer une séquence en remplaçant chaque valeur par un multiple de cette valeur.

Propriété : *L'augmentation rythmique est un morphisme additif et multiplicatif.*

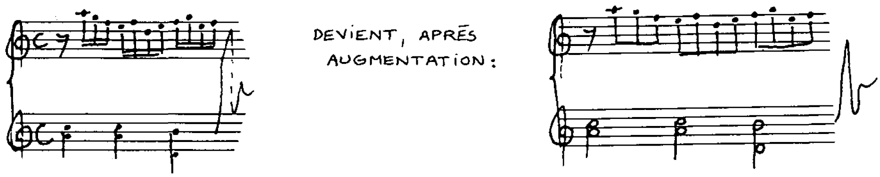
Considérons l'augmentation qui double les valeurs; voilà comment on peut la décrire algébriquement : $E$ ensemble d'événements, est constitué de deux parties disjointes de même cardinal, $E=E_0 \cup E'$; pour $a$ dans $E_0$ ($a$ dure une unité minimale de temps), $a'$ signifiera que *$a$ se prolonge d'une unité de temps*, c'est-à-dire que $a'$ fait durer $a$. Soit l'application $\lambda : E \to E^2$, définie par : $\lambda(a)=aa'$, et $\lambda(a')=a'a'$; $\lambda$ vérifie :

(i) $\forall x \in E, |\lambda(x)|=2$

(ii) $\underset{x \in E}{\perp} \lambda(x)=\varnothing\varnothing$

donc l'augmentation, qui prolonge $\lambda$, est un morphisme additif et multiplicatif.

*Remarque.* — Il n'est pas intersectif; on a, en effet : $a \neq a'$, mais $\lambda(a) \perp \lambda(a')=aa' \perp a'a'=\varnothing a' \neq \varnothing\varnothing$.




**Applications croissantes**

Définition : Un *morphisme de treillis*, de $T$ dans $T'$, est une application additive (loi $\vee$) et intersective (loi $\wedge$).

Propriété : *Dans un solfège, les* translations à gauche $\gamma_z : x \to zx$, *sont des morphismes de treillis.*

*Démonstration :* La concaténation est distributive à gauche sur $\|$ et $\perp$. ■

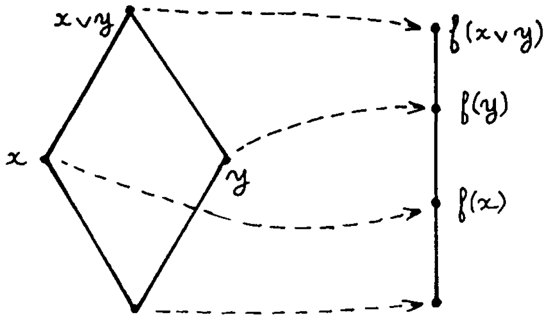
Définition : Une application *croissante*, entre ensembles ordonnés $E$ et $E'$ : $x \leqq y \Rightarrow f(x) \leqq f(y)$.

Propriété : *Tout morphisme de treillis, de $T$ dans $T'$, est croissant.*

*Démonstration :* Dans un treillis, $x \leqq y$ s'écrit : $y=x \vee y$ (ou bien, ce qui est équivalent : $x=x \wedge y$). Il vient :

$$f(y)=f(x \vee y)=f(x) \vee f(y), \text{ donc : } f(x) \leqq f(y). \blacksquare$$

*Remarque.* — Par contre, une application croissante n'est pas toujours un morphisme : dans l'exemple ci-dessous, l'application est croissante, mais $f(y) \vee f(x)=f(y) \neq f(y \vee x)$.




On a, en revanche le résultat suivant, dans un cas particulier :

DÉFINITION : Un ordre $\leqq$ sur $E$ est *total* si : $\forall\, x, y \in E,\ x \leqq y$ ou $y \leqq x$.

PROPRIÉTÉ : *Toute application croissante, de T dans T′, est un morphisme de treillis, si l'ordre est total sur T.*

*Démonstration :* On a : $\forall x, y \in T,\ y \leqq x$ ou $x \leqq y$, donc supposons par exemple : $x \leqq y$, c'est-à-dire $x \vee y = y$ et $x \wedge y = x$; $f$ est croissante, donc $f(x) \leqq f(y)$, d'où : $f(x) \vee f(y) = f(y) = f(x \vee y)$ et $f(x) \wedge f(y) = f(x) = f(x \wedge y)$, donc $f$ est un morphisme de treillis. ■

COROLLAIRE : *Un ensemble totalement ordonné, ayant un plus petit élément* 1, *muni d'une loi associative d'élément neutre* 1 *et dont les translations à gauche sont croissantes, est un solfège.*

*Exemple* 1. Il est fréquent, dans l'étude du monoïde libre sur un alphabet $A$, de faire l'hypothèse selon laquelle l'alphabet est totalement ordonné par un *ordre alphabétique.* On peut alors munir $A^*$ d'un ordre total, appelé *ordre lexicographique.* On voit alors facilement que la concaténation à gauche conserve cet ordre lexicographique. D'où :

PROPRIÉTÉ : $(A^*, \leqq, .)$ *muni de l'ordre lexicographique est un solfège.*

*Exemple* 2. Le dernier corollaire énoncé permet de retrouver facilement que (**N**, max, min, $+$) est un solfège :

- (**N**, $\leqq$) est totalement ordonné, 0 est le plus petit élément
- l'addition est associative, d'élément neutre 0
- pour tout $a$, l'application $x \to a + x$ est croissante.

D'une façon analogue, on montrerait que (**N**$-0$, max, min, $\times$) est un solfège d'élément neutre 1.

Remarquons que, si $(A^*, \|, \perp, .)$ est le solfège sur $A=\mathcal{P}(E)$, on a le résultat suivant :

PROPRIÉTÉ : *L'application de* $(A^*, \|, \perp, .)$ *dans* $(\mathbf{N}, max, min, +)$ *qui à un mot u, associe sa longueur* $|u|$, *est un morphisme de solfège.*

*Démonstration :* On a déjà vu que

$$|uv|=|u|+|v|$$
$$|u\|v|=\max(|u|, |v|)$$
$$|u\perp v|=\min(|u|, |v|)$$

et on a de plus : $|1|=0$ où 1 est le mot vide. ■

*Exemple* 3. Revenons maintenant au solfège $(\mathcal{P}(E), \cup, \cap, \cup)$.

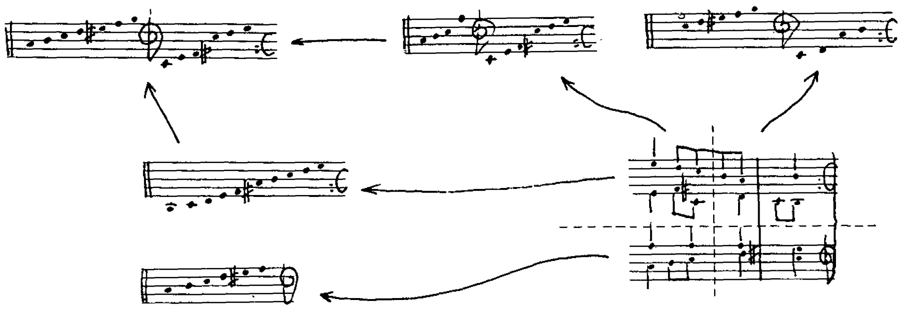
PROPRIÉTÉ : *L'application de* $(A^*, \|, \perp, .)$ *où* $A=\mathcal{P}(E)$, *dans* $(\mathcal{P}(E), \cup, \cap, \cup)$ *qui à un mot u, associe l'ensemble des événements intervenants dans u, est un morphisme additif et multiplicatif.*

*Démonstration :* Soit $\varepsilon : A^* \to \mathcal{P}(E)$ cette application. On voit facilement que $\varepsilon(uv)=\varepsilon(u\|v)=\varepsilon(u)\cup\varepsilon(v)$, et $\varepsilon(1)=\varnothing$. ■

*Remarque.* — Il n'est pas intersectif : pour $a, b\in E$, on a :

$$\varepsilon(ab)=\varepsilon(ba)=\{a, b\} \quad \text{donc} \quad \varepsilon(ab)\cap\varepsilon(ba)=\{a, b\},$$

mais $\varepsilon(ab\perp ba)=\varepsilon(\varnothing\varnothing)=\varnothing$. Pourtant, l'application étant additive, elle est croissante : $u\leqq v$ donne par définition $u\|v=v$, donc $\varepsilon(u)\cup\varepsilon(v)=\varepsilon(v)$, d'où $\varepsilon(u)\subset\varepsilon(v)$.



## Solfège fini

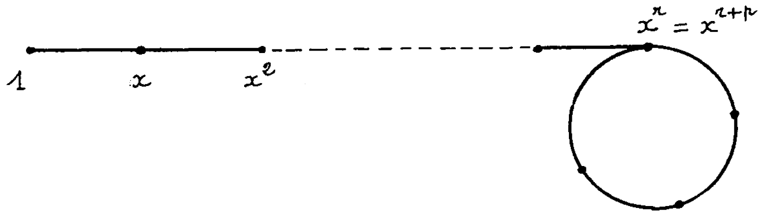
Rappelons la définition suivante :

DÉFINITION : Un monoïde fini $M$ est *apériodique* si : $\exists n\geqq 0, \forall x\in M, x^{n+1}=x^n$.

PROPRIÉTÉ : *Dans un solfège fini, le monoïde pour la concaténation est apériodique.*

*Démonstration :* $S$ un solfège fini.

Tout d'abord, $S$ fini implique : $\forall x \in S$, l'application : $n \to x^n$, de $N$ dans $S$, ne peut pas être injective, donc il existe deux entiers $p$ et $r$ tels que : $x^r = x^{r+p}$ ($r$ est dit *indice* de $x$, et $p$ sa *période*).




Mais, soit $\leqq$ l'ordre de $S$ : $1 \leqq x$, pour tout $x$ de $S$ (car $1 \| x = x$), donc, comme les translations sont croissantes : $x \leqq x^2$, puis par itération : $x^r \leqq x^{r+1} \leqq \ldots \leqq x^{r+p} = x^r$, d'où : $x^r = x^{r+1}$ et $p = 1$

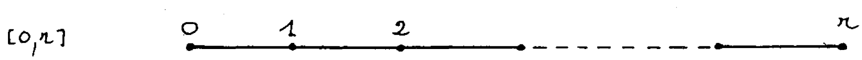
Soit alors $n = \max\{\text{indices de } x,\ x \in S\}$, il vient : $\forall x \in S$, $x^{n+1} = x^n$. ∎

*Remarque.* — On voit dans ce paragraphe, qu'il peut être commode de remplacer les lois $\|$ et $\perp$ par l'ordre $\leqq$ qui leur est associé. (Rappelons qu'on pose : $x \leqq y \Leftrightarrow x \| y = y$; ceci équivaut à : $x \perp y = x$).

*Exemples :* La propriété précédente permet de construire un modèle simple de solfège fini : revenons à (**N**, max, min, +), où l'on notera max par $\vee$, et min par $\wedge$; soit $[0, r]$ l'ensemble des entiers 0, 1, ... jusqu'à $r$. On définit une loi sur $[0, r]$ : $\forall p, q \in [0, r]$, $p \oplus q = \min(r, p+q) = r \wedge (p+q)$.

PROPRIÉTÉ : ($[0, r]$, *max*, *min*, $\oplus$) *est un solfège.*

*Démonstration :* On utilise un corollaire précédemment démontré.




(i) les applications $i \to p \oplus i$ sont *croissantes* :

$$i \leqq j \Rightarrow p+i \leqq p+j \Rightarrow r \wedge (p+i) \leqq r \wedge (p+j \Rightarrow p \oplus i \leqq p \oplus j$$

(ii) la loi $\oplus$ admet 0 pour *neutre* : $\forall\, i \in [0, r]$, $i \oplus 0 = r \wedge i = i$

(iii) la loi $\oplus$ est *associative* : rappelons que (**N**, $\vee$, $\wedge$, +) est un solfège, donc que + est distributive à gauche sur $\wedge$; il vient :

$$i \oplus (j \oplus k) = r \wedge [i + (r \wedge (j+k))] = r \wedge [(i+r) \wedge (i+j+k)]$$
$$= [r \wedge (i+r)] \wedge (i+j+k) = r \wedge (i+j+k);$$

donc $i\oplus(j\oplus k)=(i\oplus j)\oplus k$, donc ([0, r], $\vee$, $\wedge$, $\oplus$) est un solfège. ■

Soit alors $S$ un solfège fini, et $x\neq 1\in S$. On définit l'application suivante : $n\to x^n$, de $\mathbf{N}$ dans $S$. Si $r$ est l'indice de $x$, on sait qu'on peut restreindre l'application à $[0, r]$.

PROPRIÉTÉ : *Cette application est un morphisme de solfège.*

*Démonstration :* On a : $x^p.x^q=x^{p+q}$ si $p+q\leqq r$ et $x^p.x^q=x^r$ si $r\leqq p+q$, donc $x^p.x^q=x^{\min(r,\,p+q)}=x^{p\oplus q}$.

On a aussi : $x^p \,\|\, x^q=x^{\max(p,\,q)}$, et $x^p\perp x^q=x^{\min(p,\,q)}$, ainsi que $x^0=1$. ■

PROPRIÉTÉ : *L'ensemble des puissances de* $x$ *:* $\{1, x, x^2, \ldots, x^r\}$, *est un sous-solfège de* $S$.

On peut généraliser cette construction à $(A^*, \|, \perp, .)$ où $A=\mathscr{P}(E)$. Soit $r$ un entier, et $p$ la *projection* de $A^*$ sur $A^r$ :

(i) si $|u|\leqq r$, $p(u)=u$

(ii) si $|u|\leqq r$, soit $u=u'u''$ où $|u'|=r$, alors $p(u)=u'$.

On voit facilement que $p$ est un morphisme additif et intersectif :

$$\begin{cases} p(u\,\|\,v)=p(u)\,\|\,p(v)\\ p(u\perp v)=p(u)\perp p(v)\end{cases}$$

Par contre, la concaténation n'est pas une loi interne sur $A^r$.

La décomposition canonique de l'application $p$ définit une relation d'équivalence sur $A^*$ : $u\equiv v$ $[p]$ si et seulement si $p(u)=p(v)$. Cette relation d'équivalence est compatible bilatéralement avec la concaténation. En effet : $p(u)=p(v)\Rightarrow\forall w\in A^*$, $p(uw)=p(vw)$ et $p(wu)=p(wv)$. Dans ces ces conditions, il est possible de munir $A^r$ d'une loi que l'on notera $\otimes$, en « transportant » la concaténation de $A^*$ sur $A^r$, par la projection $p$. Formellement : pour $u$, $v$ dans $A^r$, soit $x$, $y$ de $A^*$ tels que $p(x)=u$ et $p(y)=v$; on pose :

$$u\otimes v=p(xy).$$

*Remarque.* — Cette définition est bien consistante, car si $x'$, $y'$ sont tels que $p(x')=u$ et $p(y')=v$, alors : $p(x'y')=p(xy')=p(xy)$ par compatibilité bilatère. Cette loi $\otimes$ sur $A^r$ est une sorte de « concaténation tronquée ».

L'application $p$ devient un morphisme multiplicatif. D'où :

PROPRIÉTÉ : $(A^r, \|, \perp, \otimes)$ *est un solfège fini, et la projection de* $A^*$ *sur* $A^r$ *est un morphisme de solfège.*

COROLLAIRE : *L'application qui, à une séquence u de $A^r$ associe sa longueur* $|u|$, *est un morphisme de solfège, de* $(A^r, \|, \perp, \otimes)$ *dans* $([0, r], \max, \min, \oplus)$.

Revenons à la structure de treillis.

PROPRIÉTÉ : *Un solfège* fini $(S, \|, \perp, .)$ *admet un élément neutre u pour* $\perp$ (*u est le plus grand élément pour l'ordre* $\leqq$, *associé au treillis S*), *et u est absorbant pour* $\|$.

*Démonstration :* Puisque $S$ fini, on peut calculer $u = \underset{s \in S}{\|} s$

Montrons d'abord que $u$ est absorbant pour $\|$:

$$\forall s \in S,\ s \| u = s \| (\underset{x \in s}{\|} x) = \underset{x \in s}{\|} x = u$$

Ensuite, $u$ est neutre pour $\perp$ : $\forall s \in S,\ s = s \perp (s \| u) = s \perp u$.

Pour terminer, donnons une propriété générale d'un solfège :

DÉFINITION : Un treillis *modulaire* $(T, \vee, \wedge)$ est tel que : $a \leqq c \Rightarrow a \vee (b \wedge c) = (a \vee b) \wedge c$, pour tout $b$ de $T$.

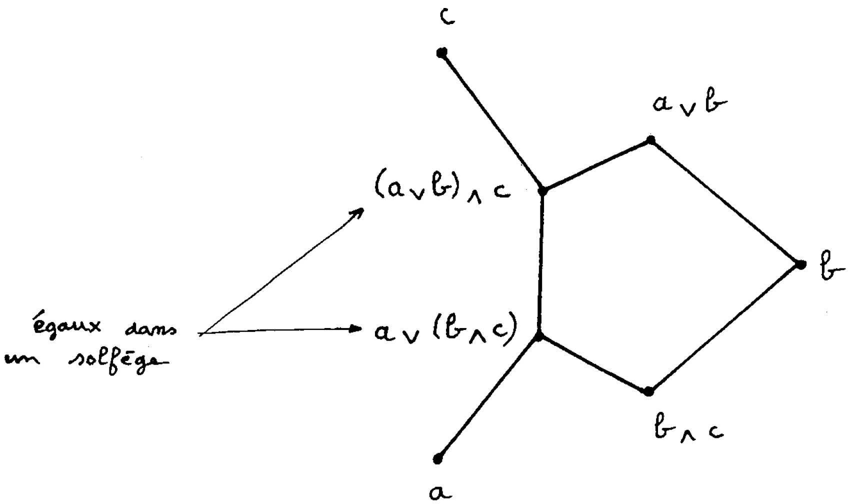
PROPRIÉTÉ : *Tout solfège* $(S, \|, \perp, .)$ *est un treillis modulaire.*

*Démonstration :*

Tout treillis distributif est modulaire : $a \leqq c \Rightarrow a \vee c = c$, donc $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c) = (a \vee b) \wedge c$

Par ailleurs, un solfège est un treillis distributif.

*Remarque.* — Dans un treillis $(T, \vee, \wedge)$ quelconque, on a toujours : $a \leqq c \Rightarrow a \vee (b \wedge c) \leqq (a \vee b) \wedge c$, pour tout $b$. En effet, $b \wedge c \leqq b$ donne $a \vee (b \wedge c) \leqq a \vee b$, et $a \leqq c$, avec $b \wedge c \leqq c$ donnent : $a \vee (b \wedge c) \leqq c$; d'où : $a \vee (b \wedge c) \leqq (a \vee b) \wedge c$.



## 3. RELATION D'ÉQUIVALENCE. COMPATIBILITÉ.

DÉFINITION : Une relation $\mathscr{R}$ est *compatible à gauche* avec une loi . sur $E$ :

$$\forall x, y, z \in E, x \mathscr{R} y \Rightarrow zx \mathscr{R} zy$$

et elle est compatible *bilatéralement* si elle l'est à gauche et à droite.

PROPRIÉTÉ : *Pour une relation d'équivalence $\mathscr{R}$ compatible bilatéralement, la surjection canonique $p : E \to E/\mathscr{R}$ est un morphisme. Réciproquement, tout morphisme $f$ de E dans F, définit par sa décomposition canonique, une relation d'équivalence : $x \equiv y \Leftrightarrow f(x) = f(y)$, compatible bilatéralement avec la loi de E.*

Rappelons les définitions suivantes concernant le monoïde $(A^*, .)$ sur lequel on s'est donné un *langage*, c'est-à-dire une partie $X$ de $A^*$ :

DÉFINITIONS :

(i) La relation d'équivalence sur $A^*$ par les *bonnes finales* :

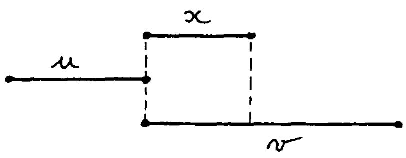
$x \sim y$ si et seulement si : $\forall z \in A^*$, $xz \in X \Leftrightarrow yz \in X$, c'est-à-dire, avec les translations à gauche : $\gamma_x^{-1}(X) = \gamma_y^{-1}(X)$.

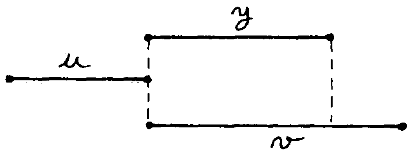
(ii) La relation d'équivalence sur $A^*$ par les *bons contextes* :

$$x \equiv y \text{ si et seulement si} : \forall u, v \in A^*, uxv \in X \Leftrightarrow uyv \in X.$$

On définit de même, sur le solfège $(A^*, \|, \perp, .)$ où $A = \mathscr{P}(E)$, la relation d'équivalence par les *bonnes harmonies* :

(iii) $x \mathscr{H} y$ si et seulement si : $\forall u, v \in A^*$, $u(x \| v) \in X \Leftrightarrow u(y \| v) \in X$




Avant d'étudier les propriétés de la relation $\mathscr{H}$, rappelons que, si $\mathscr{R}$ est une relation d'équivalence sur $E$ :

DÉFINITION : Une partie $X$ de $E$ est *saturée* pour $\mathscr{R}$ si : $\mathscr{R}(X) \subset X$, c'est-à-dire : $\forall x \in X, \forall y \in E, x \mathscr{R} y \Rightarrow y \in X$.

PROPRIÉTÉ : *Une condition nécessaire et suffisante pour que $X$ soit saturée pour $\mathscr{R}$ est : $p^{-1}(p(X)) = X$, où $p$ est la surjection canonique de E dans $E/\mathscr{R}$.*

Soit $(A^*, \|, \perp, .)$ où $A=\mathscr{P}(E)$, et $X$ un langage, $X$ partie de $A^*$, $\mathscr{H}$ l'équivalence harmonique définie par $X$.

PROPRIÉTÉ : *La partie $X$ est saturée pour $\mathscr{H}$.*

*Démonstration :* $x \in X$, soit $y \mathscr{H} x$; on a : $1(x \| 1) \in X$, donc : $1(y \| 1) = y \in X$. ■

*Remarque.* — Ainsi, $X$ est une union de classes d'équivalences pour $\mathscr{H}$.

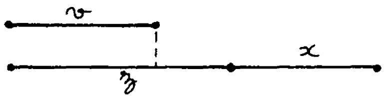
PROPRIÉTÉ : *La relation $\mathscr{H}$ est compatible avec $\|$.*

*Démonstration :* $x \mathscr{H} y$, soit $z \in A^*$; supposons $u((x \| z) \| v) \in X$, alors $u(x \| (z \| v)) \in X$, donc $u(x \| v') \in X$, donc : $u(y \| v') \in X$, car $y \mathscr{H} x$, donc $u((y \| z) \| v) \in X$, donc : $x \| z \mathscr{H} y \| z$. ■

PROPRIÉTÉ : *la relation $\mathscr{H}$ est compatible* à gauche *avec*

*Démonstration :* $x \mathscr{H} y$, et $z \in A^*$; supposons $u(zx \| v) \in X$;

(i) si $|v| \leqq |z|$, alors $u(zx \| v) = u((z \| v)x) = u(z \| v)(x \| 1) \in X$, car $y \mathscr{H} x$, donc : $u(zy \| v) \in X$




(ii) si $|v| \geqq |z|$, soit $v = v'v''$, avec $|v'| = |z|$;

$$u(zx \| v) = u(zx \| v'v'') = u((z \| v')(x \| v'')) = u(z \| v')(x \| v'') \in X,$$

donc $u(z \| v')(y \| v'') \in X$, car $y \mathscr{H} x$, donc $u(zy \| v) \in X$; d'où : $zx \mathscr{H} zy$. ■

PROPRIÉTÉ : *Les classes d'équivalences pour $\mathscr{H}$ sont stables par $\|$.*

*Démonstration :* $x \mathscr{H} y$; montrons $x \| y \mathscr{H} y$; $\forall u, v \in A^*$, soit $u((x \| y) \| v) \in X$, donc $u(x \| (y \| v)) \in X$, donc $u(y \| (y \| v)) \in X$, c'est-à-dire $u(y \| v) \in X$, donc $x \| y \mathscr{H} y$. ■

PROPRIÉTÉ : *Toute relation d'équivalence $\mathscr{R}$, compatible avec $\|$ et à gauche avec ., et saturant $X$, est incluse dans $\mathscr{H}$.*

*Démonstration :* Montrons $x \mathscr{R} y \Rightarrow x \mathscr{H} y$; soient $u, v \in A^*$ tels que $u(x \| v) \in X$; $\mathscr{R}$ est compatible; donc $x \| v \mathscr{R} y \| v$; puis à gauche : $u(x \| v) \mathscr{R} u(y \| v)$, et comme $\mathscr{R}$ sature $X$, et que $u(x \| v) \in X$, il vient : $u(y \| v) \in X$; d'où : $x \mathscr{H} y$. ■

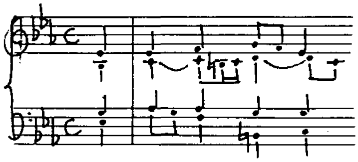
*Remarque.* — On dit que $\mathscr{H}$ est plus *grossière* que $\mathscr{R}$, pour exprimer qu'elle rend équivalents entre eux plus d'éléments que ne le fait $\mathscr{R}$.

Propriété : *Si les ensembles quotients sont finis, card* $A^*/\mathscr{H} \leqq$ *card* $A^*/\mathscr{R}$.

*Remarque.* — Plus une relation d'équivalence est grossière, plus elle a de flèches et moins elle a de classes.

**Exemple musical**

Soit le langage $X$ contenant les deux séquences :



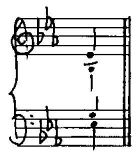
et

*Remarque.* — Cela revient à considérer comme facultative la note de passage « sol » entre « la » et « fa ».

Alors les deux séquences suivantes sont équivalentes :

Leurs *contextes harmoniques* communs sont :

u est nécessairement:



v peut être:

etc...

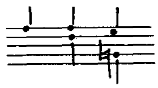
*Remarque.* — Les séquences suivantes  et  ne sont pourtant pas équivalentes. Voici un contexte harmonique convenant à la seconde, mais pas à la première :

u=

et v=

Par contre, les séquences suivantes sont équivalentes :

et



ou encore:

et

**B. RECONNAISSABILITÉ. AMBIGUITÉ**

**4. SUPERPOSITION DE PARTIES RECONNAISSABLES.**

Soit le solfège $(A^*, \|, \perp, .)$ où $A=\mathscr{P}(E)$. Pour $X$, $Y \subset A^*$, on définit la *superposition* de $X$ et $Y$ : $X \| Y=\{x \| y,\ x \in X,\ y \in Y\}$.

## Algorithme de Dan Timis

Pour $X$, $Y \subset A^*$, on définit aussi la *superposition à longueurs égales* de $X$ et de $Y$ : $X \mathbin{\overline{\|}} Y=\{x \| y,\ x \in X,\ y \in Y,\ |x|=|y|\}$.

THÉORÈME : *Si $X$ et $Y$ sont des parties reconnaissables de $A^*$, alors $X \mathbin{\overline{\|}} Y$ est reconnaissable.*

*Démonstration :* Soient $\mathscr{A}=(Q, i, T)$ et $\mathscr{A}'=(Q', i', T')$, deux automates déterministes reconnaissant $X$ et $Y$. On forme l'automate $\mathscr{A}''=(Q \times Q', (i, i'), T \times T')$, avec les flèches suivantes :

$$x \cup y : (p, q) \rightarrow (z, s) \text{ si l'on a } x : p \rightarrow r \text{ dans } \mathscr{A}, \text{ et } y : q \rightarrow s \text{ dans } \mathscr{A}'$$

Si $|\mathscr{A}''|$ est le comportement de l'automate $\mathscr{A}''$, on a : $|\mathscr{A}''| \subset X \mathbin{\overline{\|}} Y$; en effet, soit $w$ l'étiquette d'un chemin réussi de $\mathscr{A}''$, de $(i, i')$ à $(t, t')$; on peut lui associer deux chemins de $\mathscr{A}$ et de $\mathscr{A}'$ de la façon suivante : si $z : (p, q) \rightarrow (r, s)$ est une flèche du chemin d'étiquette $w$ dans $\mathscr{A}''$, il lui correspond, par construction, deux flèches dans $\mathscr{A}$ et $\mathscr{A}'$ : $p \rightarrow r$, et $q \rightarrow s$, qui peuvent être étiquetées par : $x$ et $y$, si $z=x \cup y$; d'où deux chemins $i \rightarrow t$ et $i' \rightarrow t'$ de $\mathscr{A}$ et $\mathscr{A}'$, d'étiquettes $u$ et $v$ tels que : $|u|=|v|=|w|$, et $w=u \| v$

Réciproquement : $X \mathbin{\overline{\|}} Y \subset |\mathscr{A}''|$; soit $w=u \| v$, où $|u|=|v|=|w|$; $u$ et $v$ sont étiquettes de chemins réussis de $\mathscr{A}$ et $\mathscr{A}'$ : $u : i \rightarrow t$ et $v : i' \rightarrow t'$, et alors, le chemin $(i, i') \rightarrow (t, t')$ d'étiquette $u \| v=w$, est réussi dans $\mathscr{A}''$. ■

*Remarque.* — Si $X$ et $Y$ sont des parties non vides de $A^*$, alors $X \| Y$ n'est pas vide. Il n'en est pas de même de $X \overline{\|} Y$, qui peut être vide, s'il n'existe pas de mots de $X$ et $Y$ ayant même longueur. Rappelons en effet, le résultat suivant :

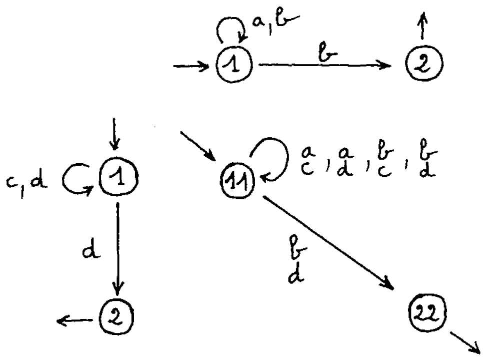
PROPRIÉTÉ : *Si X est une partie reconnaissable* infinie *de $A^*$, il existe deux entiers k et p, tels que pour tout entier n, il existe un mot de X de longueur $k+np$.*

*Démonstration :* Soit $\mathscr{A}$ un automate reconnaissant $X$; X infini, donc il existe un circuit fermé dans $\mathscr{A}$ : $q \to q$; soit $p$ la longueur du circuit fermé, et $k$ la longueur d'un chemin réussi passant par $q$ : $i \to q \to t$; alors le chemin de $i$ à $t$, passant par $q$, et parcourant $n$ fois le circuit fermé est réussi, et sa longueur est : $k+np$. ■

Mais réciproquement, cette construction montre que pour tous entiers $k$ et $p$, il est toujours possible de fabriquer un automate ne reconnaissant que des mots de longueur $k+np$, $n \in \mathbb{N}$. Mais, par ailleurs, il est possible de trouver quatre entiers $k$, $p$, $k'$, $p'$ tels que : $\forall m, n \in \mathbb{N}\, k+mp \neq k'+np'$. Il suffit de prendre par exemple $k=1$ et $k'=p=p'=2$. Alors $k+mp=2m+1$ est impair, donc différent de $k'+np'=2.(n+1)$ qui est pair. Si $\mathscr{A}$ et $\mathscr{A}'$ sont deux automates ne reconnaissant que des mots de longueur $2m+1$ et $2(n+1)$ respectivement, alors la *superposition à longueurs égales* de leurs comportements est vide : $|\mathscr{A}| \overline{\|} |\mathscr{A}'| = \varnothing$.

*Exemples* 1. Donnons tout d'abord un exemple pour lequel la superposition des comportements n'est pas vide :

$$\begin{cases} X=(a+b)^* b \\ Y=(c+d)^* d \end{cases}$$



On a :

$$X \overline{\|} Y = \left(\genfrac{}{}{0pt}{}{a}{c} + \genfrac{}{}{0pt}{}{a}{d} + \genfrac{}{}{0pt}{}{b}{c} + \genfrac{}{}{0pt}{}{b}{d}\right)^* \genfrac{}{}{0pt}{}{b}{d}$$

Remarquons que la séquence :

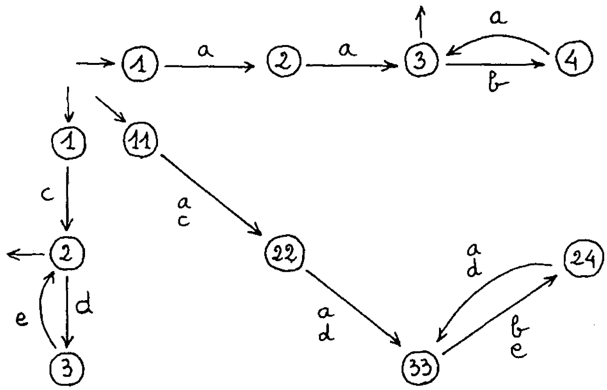
$$\begin{matrix} a & a & b & a & b & a & b & b \\ c & d & c & c & d & & & \end{matrix}$$

appartient à $X \| Y$, mais elle n'est pas reconnue par l'automate, et n'appartient pas à $X \top\!\!\!\top Y$.

*Exemple* 2. Voici maintenant un exemple pour lequel $X \top\!\!\!\top Y$ est vide :

$$\begin{cases} X = aa(ba)^* \\ Y = c(de)^* \end{cases}$$

L'algorithme donne :




Mais *il n'y a pas d'état terminal.* Une séquence :

$$\begin{matrix} a & a & b & a & b & a & b & a \\ c & d & e & d & e & d & e & d \end{matrix}$$

ne peut pas à la fois se terminer par $a$ et par $e$.

**Généralisation. Reconnaissabilité d'une superposition**

En fait, l'algorithme précédent peut très facilement se généraliser, et on a le résultat suivant :

THÉORÈME : *Si $X$ et $Y$ sont des parties reconnaissables de $A^*$, alors $X \| Y$ est reconnaissable.*

*Démonstration :* Soient $\mathscr{A} = (Q, i, T)$ et $\mathscr{A}' = (Q', i', T')$ deux automates déterministes reconnaissant $X$ et $Y$, et on forme, comme précédemment, l'automate : $\mathscr{A}'' = (Q \times Q', (i, i'), T \times T')$;

On va compléter $\mathscr{A}''$ en $\mathscr{A}''_0$ de la façon suivante :

$$\mathscr{A}''_0 = [Q \times Q' \cup Q \cup Q', (i, i'), T \times T' \cup T \cup T']$$

avec les flèches :

$$(p, q) \to (r, s) \text{ si elle est dans } \mathscr{A}''$$

$$p \to r \text{ si } p, r \in Q, \text{ et la flèche dans } \mathscr{A}$$

$$q \to s \text{ si } q, s \in Q', \text{ et la flèche dans } \mathscr{A}'$$

$$(p, t') \to r \text{ si } t' \in T', r \in Q, \text{ et } p \to r \text{ dans } \mathscr{A}$$

$$(t, q) \to s \text{ si } t \in T, s \in Q', \text{ et } q \to s \text{ dans } \mathscr{A}'$$

On a : $X \| Y \subset |\mathscr{A}''_0|$; soit en effet, $w = u \| v$, $u \in X$, $v \in Y$; on a : $u : i \to t$ dans $\mathscr{A}$, et $v : i' \to t'$ dans $\mathscr{A}'$; si $|u| = |v|$, alors $w : (i, i') \to (t, t')$ est dans $\mathscr{A}''$; si $|u| \neq |v|$, supposons $|u| > |v|$, et $u = u' u''$; soit $u' : i \to p$ et $u'' : p \to t$, alors $u' \| v : (i, i') \to (p, t')$ est dans $\mathscr{A}''$ et $u'' : p \to t$ est dans $\mathscr{A}$; mais si l'on écrit $u'' = xu'''$, on a un chemin $p \to r \to t$ d'étiquette $xu'''$ dans $\mathscr{A}$, or la flèche $x : (p, t') \to r$ est dans $\mathscr{A}''_0$ puisque $x : p \to r$ est dans $\mathscr{A}$, d'où le chemin : $(i, i') \to (p, t') \to r \to t$ dans $\mathscr{A}''_0$ d'étiquette

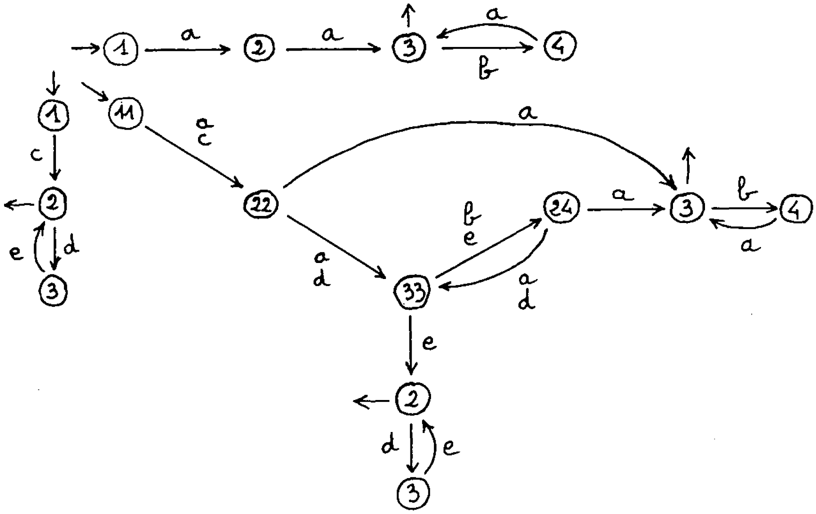
$$(u' \| v)\, xu''' = (u' \| v)\, u'' = u' u'' \| v = u \| v = w.$$

Réciproquement : $|\mathscr{A}''_0| \subset X \| Y$; soit $w$ l'étiquette d'un chemin réussi dans $\mathscr{A}''_0$; si le chemin est de la forme $w : (i, i') \to (t, t')$, alors il existe deux chemins $u : i \to t$ et $v : i' \to t'$ de $\mathscr{A}$ et $\mathscr{A}'$, et on a : $w = u \| v$, avec $u \in X$, $v \in Y$; si le chemin est de la forme $w : (i, i') \to t$, où $t \in T$ par exemple, alors soit $p$ le premier état de $Q$ figurant dans le chemin $w : (i, i') \to t$; il vient : $w' xw'' : (i, i') \to (q, t') \to p \to t$; le chemin $w' : (i, i') \to (q, t')$ peut se décomposer en $u : i \to q$ et $v : i' \to t'$, avec $w' = u \| v$, et $v \in Y$; mais alors : $uxw'' : i \to q \to p \to t$ est un chemin réussi de $\mathscr{A}$, d'où : $uxw'' \in X$; mais : $w = w' xw'' = (u \| v)\, xw''$, avec $|u| = |v|$, donc $w = uxw'' \| v$, où $uxw'' \in X$ et $v \in Y$. ■

*Exemple :* Reprenons l'exemple précédent :

$$\begin{cases} X = aa\,(ba)^* \\ Y = c\,(de)^* \end{cases}$$

On reconnaît $X \| Y$ par :




La séquence :

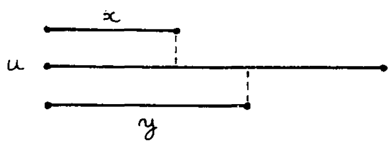
$$\begin{matrix} a & a & b & a & & & \\ c & d & e & d & e & d & e \end{matrix}$$

est reconnue par l'automate.

**Formules**

On a : $X \| Y \subset (X \| A^*) \cap (Y \| A^*)$.

*Attention :* Il n'y a pas égalité; on peut s'en convaincre avec le schéma suivant : la séquence $x \| u \| y$ est dans $X \| A^*$ et dans $Y \| A^*$, mais pas dans $X \| Y$ :




En revanche :

$$(X \| A^*) \cap (Y \| A^*) = (X \| Y) \| A^*.$$

De même :

$$(X \mathbin{\overline{\|}} A^*) \cap (Y \mathbin{\overline{\|}} A^*) = (X \mathbin{\overline{\|}} Y) \mathbin{\overline{\|}} A^*.$$

On a aussi :

PROPRIÉTÉ : *Si $X$ est une partie reconnaissable de $A^*$, alors l'ensemble des séquences contenues (au sens de l'ordre $\leqq$ sur le solfège $A^*$) dans une séquence de $X$ est aussi reconnaissable.*

*Démonstration :* Soit $\mathscr{A} = (Q, i, T)$ un automate déterministe reconnaissant $X$; on rend tous les états terminaux, ce qui permet de reconnaître tous les préfixes des mots de $X$; ensuite, pour toute flèche $a : p \to q$, on ajoute toutes les flèches $x : p \to q$ telles que $x \subset a$; il y en a un nombre fini. ■

Soient maintenant $E$ et $F$ deux ensembles d'événements disjoints, $A$ et $B$ les alphabets correspondants, c'est-à-dire $A = \mathscr{P}(E)$, et $B = \mathscr{P}(F)$; et soit $X \subset A^*$, et $Y \subset B^*$. Alors on peut écrire :

**Formule**

$$X \mathbin{\overline{\|}} Y = (X \mathbin{\overline{\|}} B^*) \cap (Y \mathbin{\overline{\|}} A^*).$$

*Démonstration :* Soit $w$ une séquence appartenant à $X \mathbin{\overline{\|}} B^*$ et à $Y \mathbin{\overline{\|}} A^*$; pour chaque lettre $w_i$ de $w$, on forme les lettres $u_i = w_i \cap E$ et $v_i = w_i \cap F$, puis les mots $u$ et $v$ formés avec les lettres $u_i$ et $v_i$; on a : $w_i = w_i \cap (E \cup F) = (w_i \cap E) \cup (w_i \cap F) = u_i \cup v_i$, donc $w = u \| v$; alors, nécessairement : $u \in X$ et $v \in Y$; en effet, pour $u$ par exemple, on écrit : $w \in X \mathbin{\overline{\|}} B^*$, donc $w = u' \| v'$ où $u' \in X$; alors, $u = u'$, car pour la $i$-ème lettre, $u'_i = w_i \cap E = u_i$. ■

*Remarque.* — La formule ci-dessus permet de retrouver la démonstration du théorème concernant la reconnaissabilité de $X \mathbin{\overline{\|}} Y$. On sait, en effet, que l'intersection $|\mathscr{A}| \cap |\mathscr{A}'|$ des comportements de deux automates est reconnues par l'automate du produit cartésien des états de $\mathscr{A}$ et $\mathscr{A}'$. D'autre part, si $X$ est une partie reconnaissable, on voit facilement que $X \mathbin{\overline{\|}} B^*$ est aussi reconnaissable.

Pour la superposition, on a :

$$(X \| Y) . \emptyset^* = (X \| B^*) \cap (Y \| A^*)$$

**Perspectives de recherche. Commentaires musicaux**

On peut se poser des questions analogues concernant la superposition de langages de Chomsky, c'est-à-dire, en anglais : « context-free ». Il serait peut-être intéressant aussi de revenir aux machines de Turing, et d'envisager d'autres types d'automates qui permettraient de cerner la réalité musicale de plus près.

Musicalement, il est évident qu'il est inutile de chercher à construire un automate pour reconnaître *une séquence musicale* (une œuvre par exemple). L'automate n'est intéressant que lorsqu'il reconnaît *une famille de séquences*, toutes également acceptables au regard de certaines contraintes. C'est le problème cher à Chomsky : être capable d'engendrer une infinité de phrases, à partir d'un nombre fini de règles de grammaires. En musique, il s'agirait de traduire ce fait d'expérience qu'on est capable de reconnaître une œuvre qu'on n'aurait jamais entendue, comme étant par exemple de Mozart. Précisons cependant que si les phrases grammaticales d'une langue sont potentiellement en quantité infinie, il n'en est pas de même malheureusement des œuvres de Mozart, et on ne pourrait guère envisager que des *pastiches de Mozart*, pour constituer le comportement d'un automate. Cela évoque à nouveau la distinction entre catégories générales et catégories particulières envisagées par P. Greussay (*cf.* première partie : *Les musiciens ont-ils besoin des mathématiques?* in *R.A.I.R.O.*, *vol.* 21, *n°* 3). Pourtant, ces questions de reconnaissabilité de familles de séquences ne sont pas dénuées de sens en musique. Les langages classiques et romantiques sont difficiles à ramener à quelques règles simples, car ils sont le résultat de la stratification d'une longue tradition. Lerdahl et Potard écrivent dans [Lerdahl, 1986], p. 28 : « Dans la musique romantique tardive, la tonique est rarement présentée mais plutôt inférée par des fonctions dominantes et sous-dominantes. Pour admirable que puisse être un art aussi allusif, il ne représente pas un modèle réaliste de construction d'une nouvelle grammaire. Le pouvoir de l'allusion dépend d'une grammaire préalablement bien comprise. » Mais cette situation n'est pas générale dans l'histoire de la musique. Après des périodes de rupture, il arrive que la tradition soit moins profondément ancrée, et que les compositeurs envisagent alors des procédés de création plus systématiques et plus facilement réductibles à quelques principes simples, que dans le cas d'une longue tradition très développée. (Précisons que la simplicité des principes qui guident une création ne saurait en aucun cas suffire comme critère de jugement esthétique.) On trouve cette situation de rupture par rapport à la tradition, dans la production musicale du début du siècle. Certaines pièces de ce répertoire sont construites, en partie, sur des procédés purement algorithmiques : Riotte en donne un exemple avec une pièce de Stravinsky, construite sur plusieurs formules périodiques, de périodes premières entre elles, qui se superposent et donnent ainsi naissance à des rencontres inattendues (*cf.* [Riotte, 1979]).

## 5. PROBLÈMES D'AMBIGUÏTÉ.

### Mélodie et polyphonie

DÉFINITION : Une séquence $w$ de $A^*$ est une *mélodie*, si pour toute lettre $w_i$ de $w$ : $\operatorname{card} w_i \leqq 1$.

Une séquence $w$ de $A^*$ est une *polyphonie stricte à n voix*, si :

(i) il existe une partition de $E = E_1 \cup E_2 \cup \ldots \cup E_n$

(ii) il existe $n$ mélodies $w_1$ sur $E_1$, $w_2$ sur $E_2$, ... $w_n$ sur $E_n$, telles que :

$$w = w_1 \| w_2 \| \ldots \| w_n$$

*Remarque.* — La famille des $w_i$ est alors unique, pour une partition donnée de $E$, car pour chaque lettre $\omega$ de $w$, on a : $\operatorname{card}(\omega \cap E_i) \leqq 1$, quelques soit $i \leqq n$.

Il peut être intéressant de caractériser les séquences de $A^*$ qui sont des polyphonies strictes à $n$ voix. Rappelons pour cela la remarque suivante :

*Remarque.* — Il y a bijection entre les *partitions* de $E$, et les *relations d'équivalences* sur $E$ :

— si $\mathcal{R}$ est une relation d'équivalence sur $E$, $E/\mathcal{R}$ définit une partition de $E$.

— si les $E_i$ forment une partition de $E$, alors $\mathcal{R} = (E_1 \times E_1) \cup (E_2 \times E_2) \cup \ldots (E_n \times E_n)$ est une relation d'équivalence sur $E$.

On parlera donc de *classe* de $x$ pour désigner la composante $E_i$ de la partition, qui contient $x$.

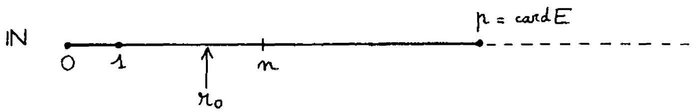
Soit $w$ un mot de $A^*$. On cherche une condition nécessaire et suffisante pour que ce mot soit une polyphonie stricte à $n$ voix. Il faut remarquer ceci : dès qu'un événement $a$ figure avec un autre $b$, dans une lettre $\omega$ de $w$, il l'exclut de sa classe; en effet, si $\omega$ contenait deux éléments $a$ et $b$ d'une même classe $E_i$, la lettre $\omega_i$ de la mélodie correspondant à $E_i$ serait telle que : $\operatorname{card} \omega_i > 1$, ce qui est faux. Ainsi la classe $\mathring{x}$ de $x$ vérifie :

$$\mathring{x} \subset \overline{\bigcup_{\substack{\text{lettres } \omega \\ \text{contenant} x}} (\omega - x)}$$

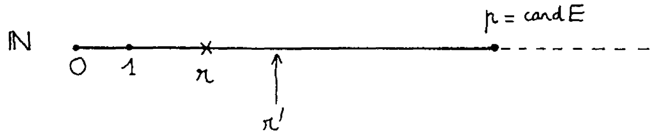
Posons alors : $C(x) = \overline{\bigcup_{\substack{\text{lettres } \omega \\ \text{contenant} x}} (\omega - x)}$. On définit une relation binaire sur $E$ : $x \gamma y$ si et seulement si $x \in C(y)$ et $y \in C(x)$. La relation $\gamma$ est *reflexive*, donc non vide, et *symétrique*. On pose : $R = \{\theta \subset E \times E / \theta$ relation d'équivalence, $\theta \subset \gamma\}$; on voit que $R$ est non vide, car la diagonale $\Delta$ de $E$ appartient à $R$, puisque $\gamma$ est reflexive; pour $\rho \in R$, on définit l'*index* de $\rho$ par : $r(\rho) = \operatorname{card} E/\rho$; on voit que, pour tout $\rho$ de $R$, on a : $r(\rho) \leqq p$, où $p = \operatorname{card} E$. On a remarqué que $\mathring{x} \subset C(x)$, où $\mathring{x}$ est la composante $E_i$ de la partition de $E$ (que l'on cherche), qui contient $x$; donc : $x \sim y$ implique $x \in \mathring{y} \subset C(y)$ et $y \in \mathring{x} \subset C(x)$, ce qui donne : $\sim \in R$, car $\sim \subset \gamma$. Posons alors : $r_0 = \min_{\rho \in R} r(\rho)$.

On peut énoncer :

PROPOSITION : *Une condition* nécessaire *pour qu'un mot w soit une polyphonie stricte à n voix est :* $r_0 \leqq n$. *Cette condition est* suffisante.




*Démonstration :* Pour démontrer que la condition est suffisante, il suffit de remarquer ceci : soit un entier $r$ compris entre 1 et $p$ ($p = \operatorname{card} E$), s'il existe $\rho \in R$ tel que $r(\rho) = r$, alors pour tout $r' > r$, il existe $\rho' \in R$ tel que $r(\rho') = r'$; en effet, soit $x$ de $E$ tel qu'il existe un $y \neq x$ dans $E$ vérifiant $y \rho x$, alors $\rho' = \rho - \{(x, y), (y, x) / y \rho x\}$ convient.



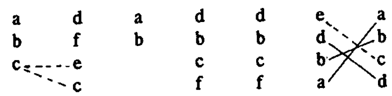

Dès lors, si $r_0 \leqq n$, il existe $\rho$ dans $R$ telle que $r(\rho) = n$; l'ensemble quotient $E/\rho$ définit une partition de $E$ en $n$ classes qui fait de $w$ une polyphonie stricte à $n$ voix. ■

COROLLAIRE : *Une condition suffisante pour que w ne soit pas une polyphonie stricte à n voix, est que le nombre de composantes connexes du graphe de* $\gamma$ *soit* $\geqq n$.

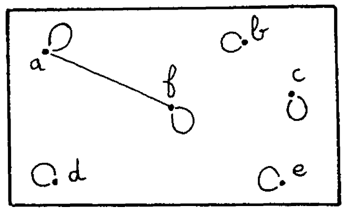
*Remarque.* — Le nombre $r_0$ correspond à la plus grande relation d'équivalence incluse dans une relation binaire quelconque $\gamma$. Il existe des moyens simples de construire cette relation d'équivalence : on peut par exemple utiliser les matrices booléenes; c'est un problème classique de la théorie des graphes (*cf.* par exemple [Kauffman, 1968]). Pour les exemples ci-dessous, on étudie directement un schéma.

*Exemples :* On considère $E = \{a, b, c, d, e, f\}$.

Soit la séquence :

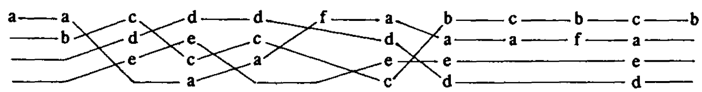



| | a | b | c | d | e | f | |
|---|---|---|---|---|---|---|---|
| C(a) | o | / | / | / | / | o | {a,f} |
| C(b) | / | o | / | / | / | / | {b} |
| C(c) | / | / | o | / | / | / | {c} |
| C(d) | / | / | / | o | / | / | {d} |
| C(e) | / | / | / | / | o | / | {e} |
| C(f) | o | / | / | / | / | o | {a,f} |

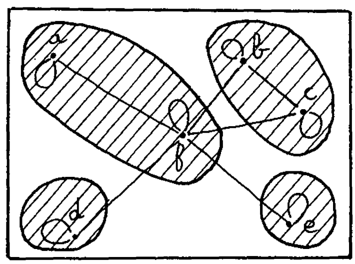



Cette séquence n'est pas une polyphonie stricte à 4 voix.

Soit maintenant la séquence :



| | a | b | c | d | e | f | |
|---|---|---|---|---|---|---|---|
| C(a) | o | / | / | / | / | o | {a,f} |
| C(b) | / | o | o | / | / | o | {b,c,f} |
| C(c) | / | o | o | / | / | o | {b,c,f} |
| C(d) | / | / | / | o | / | o | {d,f} |
| C(e) | / | / | / | / | o | o | {e,f} |
| C(f) | o | o | o | o | o | o | {f,a,b,c,d,e,} |



Cette séquence est bien une polyphonie stricte à 4 voix.

**Point de vue des automates**

La famille des mélodies sur un ensemble d'événements donné est bien entendu reconnaissable par automate fini. Par ailleurs, on a vu au chapitre précédent que la superposition d'un nombre fini de parties reconnaissables est encore reconnaissable. Enfin, pour un ensemble fini, le nombre de partitions qu'on peut constituer sur cet ensemble est fini. Ainsi, un moyen de reconnaître les polyphonies à $n$ voix sur un ensemble d'événements $E$ donné serait de construire un automate de la façon suivante :

– on cherche toutes les partitions possibles de l'ensemble $E$ à $n$ classes

– pour une partition donnée, on détermine les automates reconnaissants les mélodies qu'on peut former sur chaque classe

– on construit l'automate reconnaissant la superposition de ces $n$ familles de mélodies

– l'automate reconnaissant une polyphonie stricte à $n$ voix sur $E$ sera obtenu en regroupant tous les automates ainsi calculés, pour toutes les partitions possibles de $E$ en $n$ classes.

Pour un ensemble à $p$ éléments, le nombre de partitions en $n$ classes qu'on peut former sur $E$, noté $P_p^n$, est lié au nombre de surjections d'un ensemble à $p$ éléments dans un ensemble à $n$ éléments, noté $S_p^n$, par :

$$n!\,P_p^n = S_p^n$$

Le nombre de surjections d'un ensemble à $p$ éléments dans un ensemble à $n$ éléments se calcule au moyen des nombres de Stirling (*cf.* [Kaufmann, 1968]). Sans donner le détail de ce calcul, on peut cependant donner un ordre de grandeur des $P_p^n$, en faisant la remarque suivante :

**Formule**

$$P_{p+1}^{n+1} = P_p^n + (n+1)\,P_p^{n+1}.$$

*Démonstration :* Pour passer d'un ensemble à $p$ éléments à un ensemble à $p+1$ éléments, on rajoute un élément noté $x$; une partition de $E$ en $n+1$ classes s'obtient : soit en isolant $x$ dans une classe, et il y a $P_p^n$ façons de former les $n$ autres classes avec les éléments restants; soit en ajoutant $x$ à l'une des classes d'une partition en $n+1$ classes des autres éléments, ce qui donne $(n+1)\,P_p^{n+1}$ nouvelles partitions; d'où la formule ci-dessus. ■

Ainsi, les nombres $P_p^n$ se prêtent à une construction de type triangle de Pascal :

| $n$ ......... <br> $p$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1 ............ | 1 | | | | | | | |
| 2 ............ | 1 | 1 | | | | | | |
| 3 ............ | 1 | 3 | 1 | | | | | |
| 4 ............ | 1 | 7 | 6 | 1 | | | | |
| 5 ............ | 1 | 15 | 25 | 10 | 1 | | | |
| 6 ............ | 1 | 31 | 90 | 65 | 15 | 1 | | |
| 7 ............ | 1 | 63 | 301 | 350 | 140 | 21 | 1 | |
| 8 ............ | 1 | 127 | 966 | 1 701 | 1 050 | 266 | 28 | 1 |

Dans les exemples étudiés ci-dessus, on avait : $p=6$ et $n=4$, ce qui donne déjà 65 partitions possibles. On voit que ce nombre croît très vite avec $p$ et

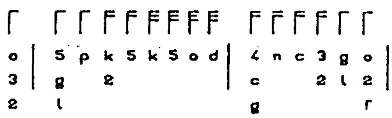
$n$. Malgré cela, l'automate garde cet avantage qu'il est valable pour étudier n'importe quelle séquence, alors que la méthode proposée ci-dessus oblige à réitérer toutes les opérations pour chaque nouveau mot.

**Généralisations. Exemple musical : les tablatures de luth**

Si $E_1, E_2, \ldots, E_n$ est une partition de $E$, soient $M_1, M_2, \ldots, M_n$ les familles des mélodies sur ces ensembles d'événements; on a dit ci-dessus que ces familles sont reconnaissables par automate fini. L'étude précédente portait sur les ensembles de type $E_1 \| E_2 \ldots \| E_n$, et on a vu entre autre, que les mots de cette superposition se décomposent de manière unique : $m_1 \| m_2 \| \ldots \| m_n$. Ce problème pourrait être généralisé : étant donnés $n$ langages $L_1, L_2, \ldots, L_n$ de $A^*$, quels sont les mots de $L_1 \| L_2 \| \ldots \| L_n$ qui se décomposent de manière unique?

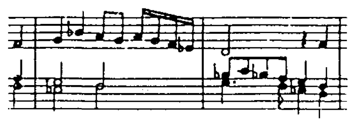
Musicalement, ces questions appellent un commentaire : on a vu, dans les hypothèses théoriques qui justifiaient le point de vue adopté ici (*cf.* première partie : *Les musiciens ont-ils besoin des mathématiques?* in *R.A.I.R.O.*, *vol.* 21, *n*° 3), que l'on considérait une séquence musicale, en quelque sorte comme une séquence de « paquets d'événements simultanés ». Cela implique que, par rapport à la notation traditionnelle, on perd l'information de la répartition des notes entre les différentes voix de la polyphonie. Bien sûr, dans le cas où les voix sont nettement différenciées par les timbres, on se trouve en présence d'une partition (au sens mathématique . . .) divisant l'ensemble global de tous les événements possibles, selon les différentes voix, et il n'y a pas d'ambiguïté. Mais on ne peut pas toujours considérer qu'il en est ainsi : par exemple dans le cas où la polyphonie est réalisée par plusieurs instruments identiques. Cependant, il faut bien préciser ceci : les problèmes d'ambiguïtés évoqués ici ne viennent pas du formalisme adopté, ils sont *inhérents au phénomène musical.* Tous les pianistes savent bien que dans une polyphonie jouée au piano, il est difficile de « faire ressortir les différentes voix »!

Il est amusant de rapprocher ces questions, des recherches d'une équipe de musicologues du C.N.R.S. qui travaillent à la transcription par ordinateur de tablatures de luth du XVI$^{e}$ siècle. Le luth est un instrument polyphonique, qui peut jouer plusieurs parties en même temps, mais sa notation traditionnelle en tablature, qui indique en fait l'emplacement où l'on met les doigts sur le manche, ne donne pas le détail des différentes voix. Une tablature de luth ressemble d'ailleurs beaucoup aux séquences que l'on a écrites dans le cours de cet article, puisque les événements (événement = « mettre tel doigt sur tel case du manche »)ȷ y sont notés par des lettres :



A partir de la notation ci-dessus, l'équipe ERATTO (Équipe de Recherche sur l'Analyse et la Transcription des Tablatures par Ordinateur) est parvenue sans difficultés, à obtenir automatiquement, par un programme de décodage, la notation sur portées ci-dessous; on voit qu'il y manque les queues des notes, pour attribuer les différentes notes aux différentes voix :

Dans leurs publications de 1975, consacrées à des tablatures de Hans Gerle, ERATTO terminait la transcription manuellement, par un travail d'analyse musicale qui permettait finalement d'attribuer des queues aux notes. C'est Raymond Meylan qui a effectué la transcription définitive ci-dessous ([Charnassé, 1975], p. 81) :



Au cours des journées Informatique-Musique qu'elle avait organisées en 1977 ([Charnassé, 1977]), Hélène Charnassé annonçait pour les années à venir une extension de ces recherches, pour parvenir à une transcription complète, uniquement par ordinateur. Bien sûr, les programmes qui sont mis au point sont essentiellement fondés sur des hypothèses de nature musicale.

**CONCLUSION**

Dans l'exemple des transcriptions de tablatures que l'on vient d'évoquer, la recherche qui est faite par le groupe ERATTO est surtout guidée par des considérations musicales : c'est parce qu'on *connaît a priori* les caractéristiques du répertoire étudié que l'on peut faire des choix. Mais l'utilisation de l'ordinateur a cet avantage qu'il oblige à formuler cette « connaissance *a priori* » en termes rigoureux, acceptables par la machine. Par rapport à l'analyse traditionnelle, qui commente sans comprendre, on fait ici un bond dans la voie dans l'explicitation, c'est-à-dire de la compréhension. Cependant, un bond supplémentaire semble possible : les caractéristiques que la recherche en transcription automatique parvient à expliciter, sont malgré tout tributaires du répertoire étudié; n'est-il pas possible de s'intéresser au problème d'une manière plus générale, en considérant toutes les musiques possibles (au moins celles de la tradition occidentale polyphonique), pour essayer de dégager une *spécificité de la combinatoire musicale?*

Les chapitres qu'on a abordés ici ne sont en fait que des « têtes de chapitre », qui demanderaient des approfondissements. De plus, d'autres questions sont envisageables : algorithmes de détermination des facteurs, des sous-mots d'une séquence donnée, qui conduiraient musicalement à des analyses thématiques. Par ailleurs, le formalisme adopté ici — monoïde libre sur un alphabet $A=\mathcal{P}(E)$ — ne couvre que l'un des cas délimités dans la première partie de cet article : *Les musiciens ont-ils besoin des mathématiques?* (in, *R.A.I.R.O.*, *vol.* 21, *n*° 3). Il se pourrait que d'autres se révèlent plus féconds; en particulier, ainsi que l'a souligné Greussay, les formalismes issus de l'étude du paraléllisme en informatique pourraient enrichir les perspectives. Enfin, puisqu'on essaie de se placer dans le cadre le plus général possible, il serait intéressant de relier l'étude faite ici aux théories envisagées dans la première partie. (Par exemple : étudier mathématiquement les caractéristiques de l'application construite par la théorie de Lerdahl, qui à une séquence musicale associe un arbre abstrait; *cf.* première partie, § 2.) Les musiciens auraient tout à gagner de la constitution d'une véritable *algorithmique de la combinatoire musicale.*

## BIBLIOGRAPHIE

### DEUXIÈME PARTIE :

1. J. Berstel, *Transductions and context-free languages*, Teubner, 1979.
2. [Charnassé, 1975] H. Charnasse, *Tablatures de luth de Hans Gerle*, vol. II, transcriptions automatiques par le groupe ERATTO, 1975.
3. [Charnassé, 1977] H. Charnasse, *Informatique musicale*, textes des conférences, équipe ERATTO, 1977.
4. M. Chemillier, *Monoïde libre et musique, première partie : les musiciens ont-ils besoin des mathématiques?*, in R.A.I.R.O., vol. 21, n° 3, 1987.
5. S. Eilenberg, *Automata, languages, and machines*, vol. A, Academic Press, 1974.

6. M. GROSS et A. LENTIN, *Notions sur les grammaires formelles*, Gauthier-Villars, 1967.
7. M. JOSEPH, *Théorie des groupes*, polycopié du cours de Maîtrise, Paris-VI, 1981.
8. [Kaufmann 1968] A. KAUFMANN, *Introduction à la combinatorique en vue des applications*, Dunod, 1968.
9. [Lerdahl 1986] F. LERDAHL et Y. POTARD, *La composition assistée par ordinateur*, rapport I.R.C.A.M., 1986.
10. M. LOTHAIRE, *Combinatorics on words*, Addison-Wesley, 1983.
11. [Riotte 1979] A. RIOTTE, *Formalisation des structures musicales*, Université de Paris-VIII, 1979.